Monero's Building Blocks
Part 4 of 10 – *Herranz & Saèz Generic Ring Signature Scheme [1]*

Bassam El Khoury Seguias

BTC: 3FcVvBZwTUkUrcqJd16RcjR42qT2tDWHWn

ETH: 0xb79Fb9194C8Cc6221368bb70976e18609Ab9AcA8

March 8, 2018

1 Introduction

In the next 4 parts of this series, we look at various ring signature schemes and prove their security in the RO model. This part is dedicated to the analysis of a generic class of ring signatures introduced in [1] and inspired by Pointcheval & Stern [2]. We also introduce a specific instance of the generic scheme which is itself a generalization of the non-interactive *Schnorr* signature.

2 Herranz & Saèz generic scheme

The scheme is built on a security parameter k, which by design corresponds to the length in bits of the output of the random oracle $\mathcal{H}$. Given a message m and a ring $L \equiv \{A_1, ..., A_n\}$ of n members, the signing algorithm Σ outputs a signature $\sigma(m, L) \equiv (r_1, ..., r_n, h_1, ...h_n, \delta)$ where:

- The r_i's are pairwise-different random elements chosen from a pre-defined large set. The term *pairwise-different* means that $\forall i, j \in \{1, ...n\}$, $(i \neq j) \Rightarrow (r_i \neq r_j)$.
- $\forall i \in \{1, ...n\}, h_i = \mathcal{H}(m, r_i)$. That means that h_i is the RO's output on query (m, r_i).
- δ is fully determined by m, r_i, and h_i, for all $i \in \{1, ...n\}$.

By design, we require that the probability of selecting any particular r_i be upper-bounded by $\frac{1}{2^{k-1}}$. For example, consider the finite field $\mathbb{Z}_q$ over a large prime $q \geq 2^k$. The probability of choosing a particular value for r_i in the mutiplicative cyclic group $\mathbb{Z}_q^*$ is equal to $\frac{1}{q-1}$ (assuming a uniform distribution over $\mathbb{Z}_q^*$). Clearly, this is less than or equal to $\frac{1}{2^k-1} < \frac{1}{2^{k-1}}$.

3 Security analysis - Unforgeability vis-a-vis EFACM

For unforgeability proofs, we follow the 5-step approach mentioned in part 1 of this series.

Step 1 : To prove that this generic 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:

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

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 respective random tapes r' and r).
- Has indistinguishable probability distribution from that of Σ over this range .

$\mathcal{S}(r')$ is specific to the particular instance of the generic scheme being used. In what follows, we build a simulator for the case of a *Schnorr ring signature*.

The *Schnorr ring signature* scheme is built on the finite field $\mathbb{Z}_q$. Here $q \geq 2^k$ is a large prime number and k is the security parameter as described earlier. We let g be a generator of the multiplicative cyclic group $\mathbb{Z}_q^*$. We also let $L \equiv \{A_1, ..., A_n\}$ be a ring of n members where A_i has an associated key-pair given by ($x_i \in \mathbb{Z}_q^*$, $y_i \equiv g^{x_i} \pmod q$). The *Schnorr ring signature* scheme is defined as a set of 3 algorithms:

- **The key generation algorithm** $\mathcal{G}$. On input 1^k, it produces a pair $(sk, pk) \equiv (x, y)$ of matching secret and public keys. The algorithm 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 $L \equiv \{A_1, ..., A_n\} \ni A_\pi$. Σ proceeds as follows:

  1. $\forall i \in \{1, ...n\}, i \neq \pi$, choose pairwise different a_i's at random in $\mathbb{Z}_q^*$. Assign $r_i \equiv g^{a_i} \pmod q$. Set $h_i \equiv \mathcal{H}(m, r_i)$.
  2. Choose a random $a \in \mathbb{Z}_q^*$. Assign $r_\pi \equiv g^a \Pi_{i \neq \pi} y_i^{-h_i} \pmod q$. If $\exists i \in \{1, ...n\}$ s.t. $i \neq \pi$ and $r_\pi = r_i$, then pick a different a. Set $h_\pi \equiv \mathcal{H}(m, r_\pi)$.
  3. Compute $\delta = a + \sum_{i \neq \pi} a_i + x_\pi h_\pi \pmod q$

  Σ finally outputs a signature $\sigma_\pi(m, L) \equiv (r_1, ..., r_n, h_1, ...h_n, \delta)$. The algorithm is modeled as a PPT Turing machine.
- **The ring verification algorithm** $\mathcal{V}$. Given a ring signature σ, a message m, the set $\{y_1, ...y_n\}$ of public keys of the ring members, $\mathcal{V}$ verifies the validity of $\sigma(m, L)$ by checking the following:
  - *(Verification equations #1 to #n)*: $h_i = \mathcal{H}(m, r_i)$, for $i \in \{1, ...n\}$

– *(Verification equation #$(n+1)$)*: $g^{\delta} = r_1 ... r_n y_1^{h_1} ... y_n^{h_n} \pmod q$

$\mathcal{V}$ is a deterministic algorithm as opposed to probabilistic.

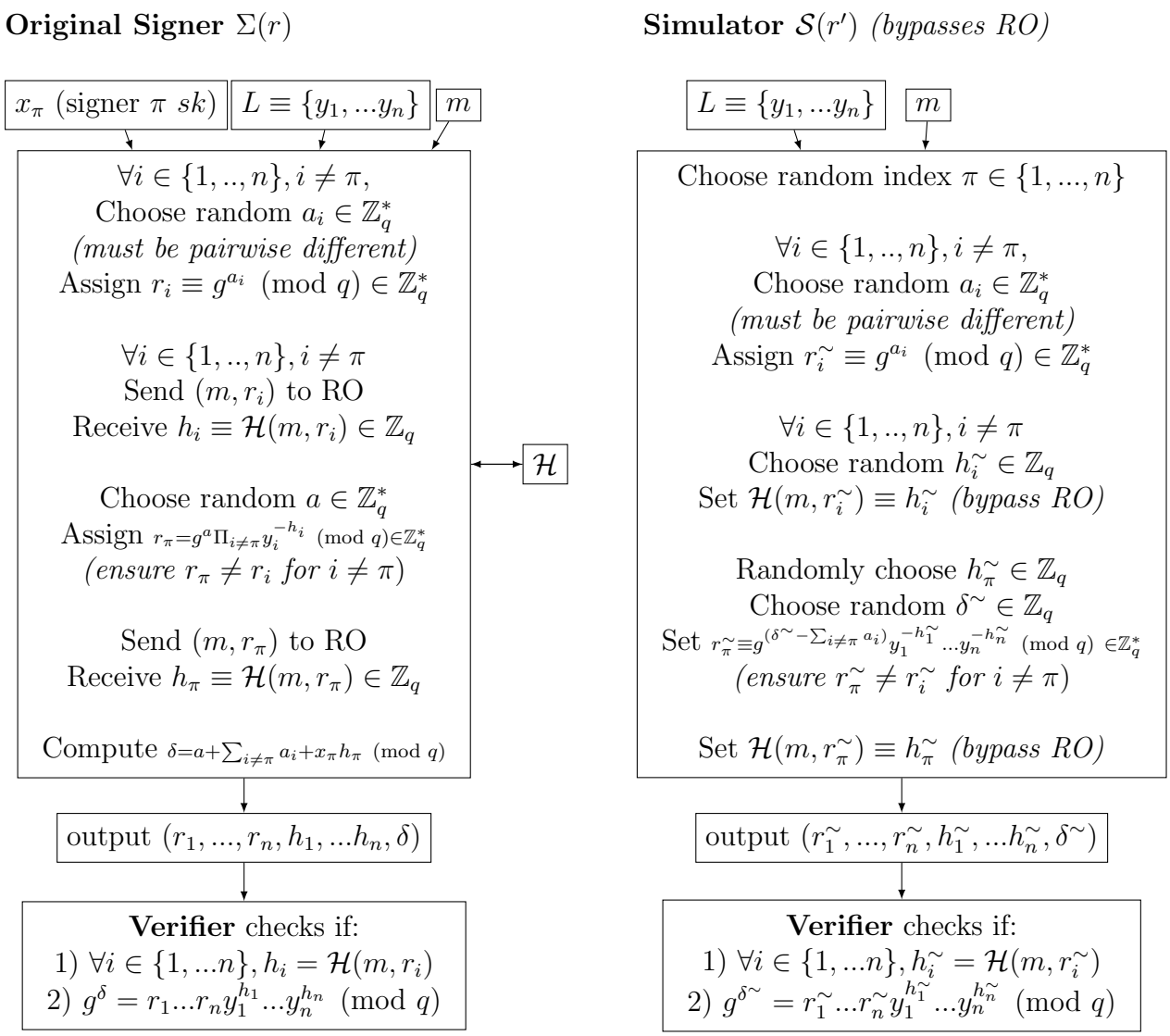
Note that this scheme satisifies the correctness property. That means that any signature generated by Σ will satisfy the verification equations with overwhelming probability. To see why, let $\sigma_\pi(m, L) \equiv (r_1, ..., r_n, h_1, ...h_n, \delta)$ be a signature issued by user π on message m and ring L of size n. By construction, we automatically have $\forall i \in \{1, ...n\}, h_i = \mathcal{H}(m, r_i)$. The first n verification equations are thus met. Moreover,

$$g^{\delta} = g^{a + \sum_{i \neq \pi} a_i + x_\pi h_\pi} \pmod q, \text{ (by definition of } \delta \text{ in } \Sigma)$$

$$= g^a (\Pi_{i \neq \pi} r_i) y_\pi^{h_\pi} \pmod q, \text{ (since by construction, } r_i = g^{a_i} \text{ for } i \neq \pi, \text{ and } y_\pi = g^{x_\pi}).$$

Finally, note that the construction of Σ also mandates that $r_\pi = g^a \Pi_{i \neq \pi} y_i^{-h_i} \pmod q$ and so $g^a = r_\pi \Pi_{i \neq \pi} y_i^{h_i} \pmod q$. Hence, $g^{\delta} = r_1 ... r_n y_1^{h_1} ... y_n^{h_n} \pmod q$. The last verification equation is thus met.

We can now build a simulator specific to the *Schnorr ring signature* scheme:



By construction, the output of $\mathcal{S}$ will satisfy the verification equations. Moreover, it assigns a random value for each $h_i, i \in \{1, ..., n\}$ and bypasses the RO in doing so. Next, note the following:

1. $\mathcal{S}$ does not use any private key.
2. Σ and $\mathcal{S}$ both have a range $R \equiv \{(\epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n, \gamma) \in (\mathbb{Z}_q^*)^n \times (\mathbb{Z}_q)^{n+1} \ s.t. \ g^\gamma = \epsilon_1 ... \epsilon_n y_1^{\beta_1} ... y_n^{\beta_n} \pmod q\}$.
3. Σ and $\mathcal{S}$ have the same probability distribution over R. Indeed, $\forall(\epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n, \gamma) \in R$ we have:

   - For Σ :

   $$P[(r_1, ..., r_n, h_1, ..., h_n, \delta) = (\epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n, \gamma)] =$$

   $P_{a_i \in \mathbb{Z}_q^*, \ h_i \in \mathbb{Z}_q, \ a \in \mathbb{Z}_q^*}[(\epsilon_i = g^{a_i}, \ \forall i \in \{1, ..., n\}, i \neq \pi) \cap (\beta_i = h_i, \ \forall i \in \{1, ..., n\}) \cap (\epsilon_\pi = g^a \Pi_{i \neq \pi} y_i^{-h_i}) \ \cap \ (\gamma = a + \sum_{i \neq \pi} a_i + x_\pi h_\pi) \ \cap \ (\forall j, k \in \{1, ..., n\}, \ \epsilon_j \neq \epsilon_k)]$.

   $$= \frac{1}{(q-1)...(q-n)} \times (\frac{1}{q})^n$$

   The first factor is the probability of choosing the exact n values given by the ϵ_i's $\in \mathbb{Z}_q^*$ that are pairwise different. The second factor is the probability of choosing the exact n values given by the β_i's $\in \mathbb{Z}_q$.

   - For $\mathcal{S}$:

   $$P[(r_1^\sim, ..., r_n^\sim, h_1^\sim, ..., h_n^\sim, \delta^\sim) = (\epsilon_1, ..., \epsilon_n, \beta_1, ..., \beta_n, \gamma)] =$$

   $$P_{a_i \in \mathbb{Z}_q^*, \ h_i^\sim \in \mathbb{Z}_q, \ \delta^\sim \in \mathbb{Z}_q}[(\epsilon_i = g^{a_i}, \ \forall i \in \{1, ..., n\}, i \neq \pi) \ \cap \ (\beta_i = h_i, \ \forall i \in \{1, ..., n\}) \ \cap \ (\epsilon_\pi = g^{\delta^\sim - \sum_{i \neq \pi} a_i} \Pi_{j \in \{1, ..., n\}} y_j^{-h_j^\sim}) \ \cap \ (\gamma = \delta^\sim) \ \cap \ (\forall j, k \in \{1, ..., n\}, \ \epsilon_j \neq \epsilon_k)].$$

   $$= \frac{1}{(q-1)...(q-n)} \times (\frac{1}{q})^n$$

   The first factor is the probability of choosing the exact n values given by the ϵ_i's $\in \mathbb{Z}_q^*$ that are pairwise different. Note that in the above, ϵ_π is also an element of $\mathbb{Z}_q^*$ that is different than all the other ϵ_i's. The second factor is the probability of choosing the exact n values given by the β_i's $\in \mathbb{Z}_q$.

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

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

Step 3 : We now show that the probability of faulty collisions is negligible (refer to section 6 of part 1 for a description of collision types). The 2 tyes of collisions fo the generic scheme are:

- $Col_{Type\ 1}$: A tuple (m, r) that $\mathcal{S}$ encounters – recall that $\mathcal{S}$ makes its own random assignment to $\mathcal{H}(m, r)$ and bypasses RO – also appears in the list of queries that $\mathcal{A}(\omega)$ sends to RO. A conflict in the 2 values will happen with overwhelming probability and the execution will halt.
- $Col_{Type\ 2}$: A tuple (m, r) that $\mathcal{S}$ encounters – recall that $\mathcal{S}$ makes its own random assignment to $\mathcal{H}(m, r)$ – is the same as another tuple (m', r') that $\mathcal{S}$ encountered at an earlier time instance – here too, $\mathcal{S}$ would have made its own random assignment to $\mathcal{H}(m', r')$. Since the 2 tuples are identical (i.e., $(m, r) = (m', r')$), it must be that the 2 random assignments match (i.e., $\mathcal{H}(m, r) = \mathcal{H}(m', r')$). However, the 2 values 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 queries to RO and a maximum of Q_S queries to $\mathcal{S}(r')$. Q and Q_S are both assumed to be polynomial in the security parameter k, since the adversary is modeled as a PPT Turing machine.

$$P[Col_{Type\ 1}] = P[\cup_{all\ (m,r)}\{(m,r)\ appeared\ in\ at\ least\ one\ of\ the\ Q_S\ queries\ to\ \mathcal{S}\ and\ Q\ queries\ to\ RO\}]$$

$$\leq P[\cup_{all\ r}\{r\ was\ part\ of\ at\ least\ one\ of\ the\ Q_S\ queries\ to\ \mathcal{S}\ and\ Q\ queries\ to\ RO\}]$$

$$\leq \sum_{all\ r\in\mathbb{Z}_q^*} P[\cup_{(j=1,..,Q_S),\ (k=1,...,Q)}\{r\ was\ part\ of\ at\ least\ the\ j^{th}\ query\ to\ \mathcal{S}\ and\ k^{th}\ queries\ to\ RO\}]$$

$$\leq \sum_{all\ r\in\mathbb{Z}_q^*} \sum_{j=1}^{Q_S} \sum_{k=1}^{Q} P[r\ was\ part\ of\ at\ least\ the\ j^{th}\ query\ to\ \mathcal{S}\ and\ k^{th}\ queries\ to\ RO]$$

Note that the j^{th} query (and any query in general) to $\mathcal{S}(r')$ includes an assignment of n random values of the form $h_i \equiv \mathcal{H}(m, r_i)$ for $i \in \{1, ..., n\}$. This is in contrast to the *Schnorr* signature scheme that we encountered in part 2 of the series, and where the j^{th} query to $\mathcal{S}(r')$ consisted of a single assignment of the form $h \equiv \mathcal{H}(m, r)$. So we get:

$$P[Col_{Type\ 1}] \leq \sum_{all\ r\in\mathbb{Z}_q^*} \sum_{j=1}^{Q_S} \sum_{k=1}^{Q} \frac{n}{(q-1)^2} = (q-1) \times \frac{nQ_SQ}{(q-1)^2} = \frac{nQ_SQ}{(q-1)} \leq \frac{nQ_SQ}{2^{k-1}}$$

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

Next, we compute:

$$P[Col_{Type\ 2}] = P[\cup_{all\ (m,r)}\{(m, r)\ appeared\ at\ least\ twice\ during\ queries\ to\ \mathcal{S}\}]$$

$$\leq P[\cup_{r\in\mathbb{Z}_q^*}\{r\ was\ part\ of\ at\ least\ 2\ queries\ to\ \mathcal{S}\}]$$

Recall that the j^{th} query (and any query in general) to $\mathcal{S}(r')$ includes an assignment of n random values of the form $h_i \equiv \mathcal{H}(m, r_i)$ for $i \in \{1, ..., n\}$. Note that by construction of $\mathcal{S}(r')$, all the r_i's corresponding to the n random assignments are pairwise-different and hence distinct from each-other. So in order for a certain r value to appear twice, it must be part of 2 different queries to $\mathcal{S}(r')$. We can choose the 2 queries in $\binom{Q_S}{2}$ ways.

And for each one of these 2 queries, the r value can appear in any one of the n assignments. So we get:

$$P[Col_{Type\ 2}] \leq \sum_{r\in\mathbb{Z}_q^*} n\binom{Q_S}{2} \times \frac{1}{(q-1)^2} = n\binom{Q_S}{2} \times \frac{q-1}{(q-1)^2} = \frac{nQ_S(Q_S-1)}{2(q-1)} \leq \frac{nQ_S^2}{2\times 2^{k-1}}$$

And so $P[Col_{Type\ 2}]$ is also negligible in k.

Putting it altogether, we find:

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

which is negligible in k. We can finally conclude (as was shown in section 6 of part 1), that:

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

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

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

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

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

Any successful forgery must satisfy the $(n+1)$ verification equations. The first n verification equations check if $h_i = \mathcal{H}(m, r_i)$ for all $i \in \{1, ..., n\}$. And so we distinguish between 2 scenarios (*w.l.o.g. we assume that all $\mathcal{A}$-queries sent to RO are distinct from each-other since $\mathcal{A}$ can keep 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 $\exists i \in \{1, ..., n\}$ such that it never queried RO on input (m, r_i).
- Scenario 2: $\mathcal{A}$ was successful in its forgery, and no collisions occured, and $\forall i \in \{1, ..., n\}$ it queried RO on input (m, r_i) during its execution.

Given a certain $i \in \{1, ..., n\}$, the probability of scenario 1 is upperbounded by the probability that $\mathcal{A}$ picks a value for h_i that matches the value of $\mathcal{H}(m, r_i)$. Here, $\mathcal{H}(m, r_i)$ is the value that RO returns to $\mathcal{V}$ (the verification algorithm) when verifying the validity of the forged signature. (It is upper-bounded because at the very least, the constraint $h_i = \mathcal{H}(m, r_i)$ must be observed for a valid signature). And since h_i can be any value in $\mathbb{Z}_q$, we get:

$$P[Scenario\ 1] \leq \sum_{i=1}^{n} \frac{1}{q} \leq \frac{n}{2^k}, \text{ which is negligible in } k.$$

So we assume that a successful forgery will likely be of the Scenario 2 type. We have:

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

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

By definition of scenario 2, we know for a fact that $\forall i \in \{1, ..., n\}$, there exists an integer $l_i \in \{1, ...Q\}$ such that l_i is the index of the query (m, r_i) to RO. (Recall that Q represents the total number of queries that $\mathcal{A}(\omega)$ sends to RO). We define $Ind(\omega, r', \mathcal{H})$ to be the vector of indices $(l_1, ..., l_n)$ corresponding to the queries $(m, r_i), i \in \{1, ..., n\}$ that $\mathcal{A}(\omega)$ sends to RO during execution. Note that since we requested by definition that all the r_i's be distinct, then so will the l_i's. By convention, if a certain (m, r_i) is not queried to RO, we let its corresponding $l_i = \infty$. This definition allows us to build the following sets:

- $S =$
  $\{(\omega, r', \mathcal{H}) \mid \mathcal{A}(\omega)^{\mathcal{H},\mathcal{S}(r')} succeeds\ in\ EFACM\ \cap \overline{Col} \cap\ \max_{i=1}^{n}[Ind(\omega, r', \mathcal{H})] \neq \infty\}$

  In other terms, S is the set of tuples $(\omega, r', \mathcal{H})$ that yield a successful EFACM forgery when no collisions occur, and when $\mathcal{A}(\omega)$ queried RO on all inputs $(m, r_i)\ \forall i \in \{1, ..., n\}$ (i.e., scenario 2).

- $S_{\vec{l}} = \{(\omega, r', \mathcal{H}) \mid \mathcal{A}(\omega)^{\mathcal{H},\mathcal{S}(r')} succeeds\ in\ EFACM\ \cap \overline{Col} \cap\ Ind(\omega, r', \mathcal{H}) = \vec{l}\}$

  where
  $\vec{l} \in L_n \equiv \{(l_1, ..., l_n) \mid (1 \leq l_i \leq Q),\ and\ (\forall i, j \in \{1, ..., n\},\ (i \neq j)\ \Rightarrow (l_i \neq l_j)\}$.
  We let $V_{Q,n}$ denote that the cardinality of L_n. We have:

  $$V_{Q,n} = Q.(Q-1)...(Q-n+1)$$

  We can see that $S_{\vec{l}}$ represents the set of tuples $(\omega, r', \mathcal{H})$ that yield a successful EFACM forgery when no collisions occur, and when $\mathcal{A}(\omega)$ queried RO on all inputs $(m, r_i)\ \forall i \in \{1, ..., n\}$, such that the index of the input query (m, r_i) is equal to $(\vec{l})_i$ (i.e., the i^{th} component of $\vec{l}$).

Recall that, $P_{\omega,r',\mathcal{H}}[(\omega, r', \mathcal{H}) \in S] = P[Scenario2] \geq \nu(k)$, which is non-negligible in k.

And clearly, $\{\cup_{\vec{l} \in L_n} S_{\vec{l}}\}$ partitions S. So $\sum_{\vec{l} \in L_n} P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S] = 1$.

This implies that $\exists \vec{l} \in L_n\ s.t.\ P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S] \geq \frac{1}{2V_{Q,n}}$.

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

$$1 = \sum_{\vec{l} \in L_n} P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S] < V_{Q,n} \times \frac{1}{2V_{Q,n}} = \frac{1}{2} < 1.$$

So we introduce the set I consisting of all vectors $\vec{l}$ that meet the $\frac{1}{2V_{Q,n}}$ threshold, i.e.

$$I = \{\vec{l} \in L_n \mid P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S] \geq \frac{1}{2V_{Q,n}}\}$$

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

Proof By definition of the sets $S_{\vec{l}}$ we have:

$$P[Ind(\omega, r', \mathcal{H}) \in I \mid (\omega, r', \mathcal{H}) \in S] = \sum_{\vec{l} \in I} P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S]$$

$$= 1 - \sum_{\vec{u} \notin I} P[(\omega, r', \mathcal{H}) \in S_{\vec{u}} \mid (\omega, r', \mathcal{H}) \in S] > 1 - \sum_{\vec{u} \notin I} \frac{1}{2V_{Q,n}} > 1 - \frac{V_{Q,n}}{2V_{Q,n}} = \frac{1}{2}$$

The next step is to apply the splitting lemma to each $S_{\vec{l}},\ \vec{l} \in I$. First note that:

$$P_{\omega,r',\mathcal{H}}[(\omega, r', \mathcal{H}) \in S_{\vec{l}}] = P_{\omega,r',\mathcal{H}}[(\omega, r', \mathcal{H}) \in (S_{\vec{l}} \cap\ S)]$$

$$= P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S] \times P_{\omega,r',\mathcal{H}}[(\omega, r', \mathcal{H}) \in S]$$

$$\geq \frac{1}{2V_{Q,n}} \times \nu(k)$$

Let $\mu_{\vec{l}} \equiv max\{(\vec{l})_1, ..., (\vec{l})_n)\}$. Referring to the notation used in the splitting lemma (section 7 of part 1), we let:

$$A \equiv S_{\vec{l}},\ X \equiv (\omega, r', \rho_1, ..., \rho_{\mu_{\vec{l}}-1}),\ Y \equiv (\rho_{\mu_{\vec{l}}}, ..., \rho_Q),\ \epsilon \equiv \frac{\nu(k)}{2V_{Q,n}},\ \text{and}\ \alpha \equiv \frac{\nu(k)}{4V_{Q,n}} = \frac{\epsilon}{2}$$

X is defined as the space of tuples of all random tapes ω, all random tapes r', and all possibe RO answers to the first $\mu_{\vec{l}} - 1$ queries sent by $\mathcal{A}(\omega)$. Y is defined as the space of all possible RO answers to the last $(Q - \mu_{\vec{l}} + 1)$ queries sent by $\mathcal{A}(\omega)$. (Recall that $\rho_i \equiv \mathcal{H}(q_i)$). The splitting lemma guarantees the existence of a subset $\Omega_{\vec{l}}$ of tuples $(\omega, r', \mathcal{H})$ such that:

- $P_{\omega,r',\mathcal{H}}[(\omega, r', \mathcal{H}) \in \Omega_{\vec{l}}] \geq \frac{\nu(k)}{4V_{Q,n}}$
- $\forall[(\omega^\sim, r'^\sim, \mathcal{H}^\sim) \equiv (\omega^\sim, r'^\sim, \rho_1^\sim, ..., \rho_{\mu_{\vec{l}}-1}^\sim, \rho_{\mu_{\vec{l}}}^\sim ... \rho_Q^\sim]) \in \Omega_{\vec{l}}$, we have

  $$P_{\mathcal{H}}[(\omega^\sim, r'^\sim, \rho_1^\sim, ..., \rho_{\mu_{\vec{l}}-1}^\sim, \rho_{\mu_{\vec{l}}} ... \rho_Q) \in S_{\vec{l}} \mid (\omega^\sim, r'^\sim, \mathcal{H}^\sim) \in \Omega_{\vec{l}}] \geq \frac{\nu(k)}{4V_{Q,n}},\ \text{and so}$$

  $$P_{\mathcal{H}}[(\omega^\sim, r'^\sim, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^\sim, r'^\sim, \mathcal{H}^\sim) \in \Omega_{\vec{l}},\ \rho_1 = \rho_1^\sim, ..., \rho_{\mu_{\vec{l}}-1} = \rho_{\mu_{\vec{l}}-1}^\sim)] \geq \frac{\nu(k)}{4V_{Q,n}}$$

- $P[(\omega, r', \mathcal{H}) \in \Omega_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S_{\vec{l}}] \geq (\frac{\nu(k)}{4V_{Q,n}})/(\frac{\nu(k)}{2V_{Q,n}}) = \frac{1}{2}$

We would like to compute the probability of finding a 2^{nd} successful tuple $(\omega^*, r'^*, \mathcal{H}^\sim)$ given that $(\omega^*, r'^*, \mathcal{H}^*)$ was a successful 1^{st} tuple and s.t. $\rho_j^\sim = \rho_j^*,\ j \in \{1, ..., \mu_{\vec{l}} - 1\}$. That means finding the following probability:

$$P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in S_{\vec{l}},\ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{l}}-1} = \rho_{\mu_{\vec{l}}-1}^*].$$

From the splitting lemma results, we have a (non-negligible in k) lower-bound on $P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in \Omega_{\vec{l}},\ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{l}}-1} = \rho^*_{\mu_{\vec{l}}-1}]$.

Note however, that $\Omega_{\vec{l}}$ and $S_{\vec{l}}$ are generally distinct sets. And so we **cannot** conclude that

$$P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in S_{\vec{l}},\ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{l}}-1} = \rho^*_{\mu_{\vec{l}}-1}]$$

$$= P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in \Omega_{\vec{l}},\ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{l}}-1} = \rho^*_{\mu_{\vec{l}}-1}]$$

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

$$P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{l}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in S_{\vec{l}},\ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{l}}-1} = \rho^*_{\mu_{\vec{l}}-1}]$$

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:

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

This would imply that with non-negligible probability, we can find a tuple that belongs to $S_{\vec{\beta}}$ (and hence corresponds to a successful forgery) and at the same time belongs to $\Omega_{\vec{\beta}}$. 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:

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

$$= P[\cup_{\vec{l} \in I} \{(\omega, r', \mathcal{H}) \in (\Omega_{\vec{l}} \cap S_{\vec{l}}) \mid (\omega, r', \mathcal{H}) \in S\}]$$

$$= \textstyle\sum_{\vec{l} \in I} P[(\omega, r', \mathcal{H}) \in (\Omega_{\vec{l}} \cap S_{\vec{l}}) \mid (\omega, r', \mathcal{H}) \in S]$$, since the $S_{\vec{l}}$'s are disjoint.

$$=$$

$$\textstyle\sum_{\vec{l} \in I} \{P[(\omega, r', \mathcal{H}) \in \Omega_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in (S_{\vec{l}} \cap S)] \times P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S]\}$$

$$\textstyle\sum_{\vec{l} \in I} \{P[(\omega, r', \mathcal{H}) \in \Omega_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S_{\vec{l}}] \times P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S]\}$$

$$\geq \tfrac{1}{2} \textstyle\sum_{\vec{l} \in I} P[(\omega, r', \mathcal{H}) \in S_{\vec{l}} \mid (\omega, r', \mathcal{H}) \in S]$, *($3^{rd}$ result of splitting lemma above)*

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

And so we conclude that:

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

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

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

$$\geq \frac{\nu(k)}{4}, \text{ which is non-negligible in } k.$$

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

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

$$P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{\beta}} \mid (\omega^*, r'^*, \mathcal{H}^*) \in \Omega_{\vec{\beta}}, \ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{\beta}}-1} = \rho^*_{\mu_{\vec{\beta}}-1})] \geq \frac{\nu(k)}{4V_{Q,n}}$$

We still have one last constraint to impose and that is that $\rho^*_{\mu_{\vec{\beta}}} \neq \rho^{\sim}_{\mu_{\vec{\beta}}}$. We show that the following quantity is non-negligible:

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

To prove this, 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 results allows us to write:

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

$$\geq P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{\beta}}| \ (\omega^*, r'^*, \mathcal{H}^*) \in S_{\vec{\beta}}, \ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{\beta}}-1} = \rho^*_{\mu_{\vec{\beta}}-1})] - P_{\mathcal{H}}[\rho_{\mu_{\vec{\beta}}} = \rho^*_{\mu_{\vec{\beta}}}]$$

$$= P_{\mathcal{H}}[(\omega^*, r'^*, \mathcal{H}) \in S_{\vec{\beta}}| \ (\omega^*, r'^*, \mathcal{H}^*) \in \Omega_{\vec{\beta}}, \ \rho_1 = \rho_1^*, ..., \rho_{\mu_{\vec{\beta}}-1} = \rho^*_{\mu_{\vec{\beta}}-1})] - P_{\mathcal{H}}[\rho_{\mu_{\vec{\beta}}} = \rho^*_{\mu_{\vec{\beta}}}]$$

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

$$\geq \frac{\nu(k)}{4V_{Q,n}} - \frac{1}{2^k}, \text{ which is non-negligible in } k.$$

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 $(\omega^*, r'^*, \mathcal{H}^*)$, s.t. $(\omega^*, r'^*, \mathcal{H}^*) \in (\Omega_{\vec{\beta}} \cap S_{\vec{\beta}})$ for some vector of indices $\vec{\beta} \in I$. So by running $\mathcal{A}$ a number of times polynomial in k, we can confidently find such a tuple.
- Once we find such a tuple, we've also shown that with non-negligible probability of at least $\frac{\nu(k)}{4V_{Q,n}} - \frac{1}{2^k}$, we can find another successful tuple $(\omega^*, r'^*, \mathcal{H}^{\sim})$ such that $(\omega^*, r'^*, \mathcal{H}^{\sim}) \in S_{\vec{\beta}}$ and $(\rho_1^{\sim} = \rho_1^*), .., (\rho^{\sim}_{\mu_{\vec{\beta}}-1} = \rho^*_{\mu_{\vec{\beta}}-1})$, but $(\rho^{\sim}_{\mu_{\vec{\beta}}} \neq \rho^*_{\mu_{\vec{\beta}}})$.

W.l.o.g, let $(\omega^*, r'^*, \mathcal{H}^*)$ correspond to $\sigma_{forge}(m, L) \equiv (r_1, ..., r_n, h_1, ..., h_n, \delta)$, and $(\omega^*, r'^*, \mathcal{H}^{\sim})$ correspond to $\sigma_{forge}(m', L) \equiv (r'_1, ..., r'_n, h'_1, ..., h'_n, \delta')$.

Recall that $\vec{\beta}$ is the vector $((\vec{\beta})_1, ..., (\vec{\beta})_n)$ where $(\vec{\beta})_i$ denotes the index of query (m, r_i) that $\mathcal{A}$ sends to the RO. Since the 2 experiments corresponding to the 2 successful tuples have the same random tapes ω^* and r'^*, and since the 2 corresponding ROs $\mathcal{H}^*$ and $\mathcal{H}^{\sim}$ behave the same way on the first $\mu_{\vec{\beta}} - 1$ queries (*recall that* $\mu_{\vec{\beta}} = max_{i=1}^{n}(\vec{\beta})_i$), we can be confident that:

- The first $\mu_{\vec{\beta}}$ queries sent to the 2 ROs are identical. In particular, $\forall i \in \{1, ..., n\}$ we have $(m, r_i) = (m', r'_i)$.
- The first $(\mu_{\vec{\beta}} - 1)$ replies of the 2 oracles $\mathcal{H}^*$ and $\mathcal{H}^{\sim}$ are the same. Suppose w.l.o.g. that (m, r_ζ), (where $\zeta \in \{1, ..., n\}$), corresponds to the last query of this type that is sent to the ROs. (m, r_ζ) is actually the $\mu_{\vec{\beta}}^{th}$ query sent to RO (by definition of $\mu_{\vec{\beta}}$). We then have $\mathcal{H}^*(m, r_i) = \mathcal{H}^{\sim}(m, r_i),\ \forall i \in \{1, ..., n\},\ i \neq \zeta$.
- $h_\zeta = \mathcal{H}^*(m, r_\zeta) = \mathcal{H}^*(q_{\mu_{\vec{\beta}}}) = \rho^*_{\mu_{\vec{\beta}}} \neq \rho^{\sim}_{\mu_{\vec{\beta}}} = \mathcal{H}^{\sim}(q_{\mu_{\vec{\beta}}}) = \mathcal{H}^{\sim}(m, r_\zeta) = h'_\zeta$.

So we have 2 successful forgeries $\sigma_{forge}(m) \equiv (r_1, ..., r_n, h_1, ...h_\zeta, ..., h_n, \delta)$ and $\sigma_{forge}(m) \equiv (r_1, ..., r_n, h_1, ..., h'_\zeta, ..., h_n, \delta')$, with $h_\zeta \neq h'_\zeta$. Since both are valid signatures, they must satisfy the verification equations. For the particular case of a Schnorr ring signature, they must satisfy the following 2 equations (1 equation per signature):

- $g^\delta = r_1...r_n y_1^{h_1}...y_\zeta^{h_\zeta}...y_n^{h_n}$, where $\{y_1, ..., y_n\}$ is the set of public keys of the n ring members associated with the signature.
- $g^{\delta'} = r_1...r_n y_1^{h_1}...y_\zeta^{h'_\zeta}...y_n^{h_n}$, where $\{y_1, ..., y_n\}$ is the set of public keys of the n ring members associated with the signature.

Writing $y_\zeta = g^{x_\zeta}$ (x_ζ is the secret key corresponding to y_ζ), we get:

$$g^{\delta - \delta'} = y_\zeta^{h_\zeta - h\zeta'} \Rightarrow x_\zeta = \tfrac{\delta - \delta'}{h_\zeta - h_{\zeta'}} \pmod q.$$

Since, $h_\zeta \neq h_{\zeta'}$, we can solve for x_ζ (the DL of y_ζ) in polynomial time. This contradicts the intractability of DL on multiplicative cyclic groups and we conclude that our signature scheme (in this case the Schnorr ring signature scheme) is secure against EFACM in the RO model.

4 Security analysis - Anonymity

In this section, we show that our generic scheme satisfies the anonymity definition #1 introduced in part 3 of this series. Recall that roughly speaking, this definition mandates that the probability of guessing the real signer be $\approx \frac{1}{n}$ (in an n-ring setting). This probability is independent of any knowledge about any member's private key. In other terms, even if a signer is coerced or subpoenaed to release her private key, nothing can be done to prove that she is the real signer (with probability better than random guessing).

To prove anonymity in our case, we show that any signature could have been created with equal probability by any of the n members of the ring. We show that releasing information about the secret key of any ring member does not modify this probability. That automatically implies that even when a subset of private keys gets compromised, there is still an equiprobable likelihood that the signature was created by any member.

Proof: Let $\sigma(m, L) \equiv (r_1, ..., r_n, h_1, ...h_n, \delta)$ be a valid signature on message m and ring L. That means that all $(n + 1)$ verification equations are satisfied. Let α be any member of the ring (with compromised or non-compromised secret key x_α). The probability that $\sigma(m, L)$ was issued by α is given by:

$$P[\alpha \; issued \; \sigma(m, L) \mid \sigma(m, L) \equiv (r_1, ..., r_n, h_1, ...h_n, \delta); \; and \; given \; a \; hash \; function \; \mathcal{H}] =$$

$$P[\alpha \; guesses \; the \; correct \; pairwise \; different \; r_i \; values \; in \; \mathbb{Z}_q^*] = \Pi_{i=1}^{n}(\tfrac{1}{q-i})$$

Note that once the r_i's are calculated, the h_i's will be automatically determined since we are using a specific hash function. Clearly, the above probability does not depend on any specific information about member α. It is the same for all ring members.

References

[1] J. Herranz and G. Saez. Forking lemmas in the ring signatures' scenario. *Proceedings of INDOCRYPT'03*, Lecture Notes in Computer Science(2904):266–279, 2003.

[2] D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 2000.