Logarithmic-Size Ring Signatures With Tight Security from the DDH Assumption

Benoît Libert, Thomas Peters, Chen Qian

Logarithmic-Size Ring Signatures With Tight Security from the DDH Assumption

Benoît Libert[1,2], Thomas Peters[3], and Chen Qian[4]

[1] CNRS, Laboratoire LIP, France
[2] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), France
[3] FNRS & Université catholique de Louvain (Belgium)
[4] IRISA Rennes (France)

Abstract. Ring signatures make it possible for a signer to anonymously and, yet, convincingly leak a secret by signing a message while concealing his identity within a flexibly chosen *ring* of users. Unlike group signatures, they do not involve any setup phase or tracing authority. Despite a lot of research efforts in more than 15 years, most of their realizations require linear-size signatures in the cardinality of the ring. In the random oracle model, two recent constructions decreased the signature length to be only logarithmic in the number N of ring members. On the downside, their suffer from rather loose reductions incurred by the use of the Forking Lemma. In this paper, we consider the problem of proving them tightly secure without affecting their space efficiency. Surprisingly, existing techniques for proving tight security in ordinary signature schemes do not trivially extend to the ring signature setting. We overcome these difficulties by combining the Groth-Kohlweiss Σ-protocol (Eurocrypt'15) with dual-mode encryption schemes. Our main result is a fully tight construction based on the Decision Diffie-Hellman assumption in the random oracle model. By full tightness, we mean that the reduction's advantage is as large as the adversary's, up to a constant factor.

Keywords. Ring signatures, anonymity, tight security, random oracles.

1 Introduction

As introduced by Rivest, Shamir and Tauman [34], ring signatures make it possible for a signer to sign messages while hiding his identity within an *ad hoc* set of users, called a *ring*, that includes himself. To this end, the signer only needs to know the public keys of all ring members (besides his own secret key) in order to generate an anonymous signature on behalf of the entire ring. Unlike group signatures [16], ring signatures do not require any setup, coordination or registration phase and neither do they involve a tracing authority to de-anonymize signatures. Whoever has a public key can be appointed as a ring member without being asked for his agreement or even being aware of it. Moreover, signatures should ideally provide everlasting anonymity and carry no information as to which ring member created them. The main motivation of ring signatures is to enable the anonymous leakage of secrets, by concealing the identity of a source (e.g., a whistleblower in a political scandal) while simultaneously providing guarantees of its reliability.

In this paper, we consider the exact security of ring signatures in the random oracle model [4]. So far, the only known solutions with logarithmic signature length [24,28] suffered from loose reductions: the underlying hard problem could only be solved with a probability smaller than the adversary's advantage by a linear factor in the number of hash queries. Our main result is to give the first construction that simultaneously provides tight security – meaning that there is essentially no gap between the adversary's probability of success and the reduction's advantage in solving a hard problem – *and* logarithmic signature size in the number of ring members. In particular, the advantage of our reduction is not multiplicatively affected by the number Q_H of random oracle queries nor the number of Q_V of public verification keys in a ring.

Our Contribution. We describe the first logarithmic-size ring signatures with tight security proofs in the random oracle model. The unforgeability of our construction is proved under the standard Decision Diffie-Hellman (DDH) assumption in groups without a bilinear map while anonymity is achieved against unbounded adversaries. Our security proof eliminates *both* the linear gap in the number of random oracle queries *and* the $\Theta(Q_V)$ security loss. It thus features a *fully tight* reduction, meaning that – up to statistically negligible terms – the reduction's advantage as a DDH distinguisher is only smaller than the adversary's forging probability by a factor 2. To our knowledge, our scheme is the first ring signature for which such a fully tight reduction is reported. It is obtained by tweaking a construction due to Groth and Kohlweiss [24] and achieves tight security at the expense of increasing the number of scalars and group elements per signature by a small constant factor. For the same exact security, our reduction allows smaller key sizes which essentially decrease the signature length of [24] by a logarithmic factor n in the cardinality N of the ring and the time complexity by a factor $\omega(n^2)$. For rings of cardinality $N = 2^6$, for example, our signatures can be 36 times faster to compute and 6 times shorter than [24].

Our Techniques. Our scheme builds on the Groth-Kohlweiss proof system [24] that allows proving that one-out-of-N commitments opens to 0 with a communication complexity $O(\log N)$. This proof system was shown to imply logarithmic-size ring signatures with perfect anonymity assuming that the underlying commitment scheme is perfectly hiding. At the heart of the protocol of [24] is a clever use of a Σ-protocol showing that a committed value ℓ is 0 or 1, which proceeds in the following way. In order to prove that a commitment $\boldsymbol{C}_\ell \in \{\boldsymbol{C}_i\}_{i=0}^{N-1}$ opens to 0 without revealing the index $\ell \in \{0, \ldots, N-1\}$, the n-bit indexes ℓ_j of the binary representation $\ell_1 \ldots \ell_n \in \{0,1\}^n$ of $\ell \in \{0, \ldots, N-1\}$ are committed to and, for each of them, the prover uses the aforementioned Σ-protocol to prove that $\ell_j \in \{0,1\}$. The response $f_j = a_j + \ell_j x$ of the Σ-protocol is then viewed as a degree-one polynomial in the challenge $x \in \mathbb{Z}_q$ and used to define polynomials

$$P_i[Z] = \prod_{j=1}^{n} f_{j,i_j} = \delta_{i,\ell} \cdot Z^n + \sum_{k=0}^{n-1} p_{i,k} \cdot Z^k \qquad \forall i \in [N],$$

where $f_{j,0} = f_j$ and $f_{f,1} = x - f_j$, which have degree $n = \log N$ if $i = \ell$ and degree $n-1$ otherwise. In order to prove that one of the polynomials $\{P_i[Z]\}_{i=0}^{N-1}$ has degree n without revealing which one, Groth and Kohlweiss [24] homomorphically compute the commitment $\prod_{i=0}^{N-1} \boldsymbol{C}_i^{P_i(x)}$ and multiply it with $\prod_{k=0}^{n-1} \boldsymbol{C}_{d_k}^{-x^k}$, for auxiliary homomorphic commitments $\{\boldsymbol{C}_{d_k} = \prod_{i=0}^{N-1} \boldsymbol{C}_i^{p_{i,k}}\}_{k=0}^{n-1}$, in order to cancel out the terms of degree 0 to $n-1$ in the exponent. Then, they prove that the product $\prod_{i=0}^{N-1} \boldsymbol{C}_i^{P_i(x)} \cdot \prod_{k=0}^{n-1} \boldsymbol{C}_{d_k}^{-x^k}$ is indeed a commitment of 0. The soundness of the proof relies on the Schwartz-Zippel lemma, which ensures that $\prod_{i=0}^{N-1} \boldsymbol{C}_i^{P_i(x)} \cdot \prod_{k=0}^{n-1} \boldsymbol{C}_{d_k}^{-x^k}$ is unlikely to be a commitment to 0 if $\boldsymbol{C}_\ell$ is not.

As an application of their proof system, Groth and Kohlweiss [24] obtained logarithmic-size ring signatures from the discrete logarithm assumption in the random oracle model. While efficient and based on a standard assumption, their scheme suffers from a loose security reduction incurred by the use of the Forking Lemma [33]. In order to extract a discrete logarithm from a ring signature forger, the adversary has to be run $n = \log N$ times with the same random tape (where N is the ring cardinality), leading to a reduction with advantage $\varepsilon' \approx \frac{\varepsilon^n}{Q_V \cdot Q_{\mathcal{H}}}$, where $Q_{\mathcal{H}}$ is the number of hash queries and Q_V is the number of public keys. This means that, if we want to increase the key size so as to compensate for the concrete security gap, we need to multiply the security parameter by a factor $n = \log N$, even without taking into account the factors $Q_{\mathcal{H}}$ and Q_V.

In our pursuit of a tight reduction, a first idea is to apply the lossy identification paradigm [27,2] where the security proofs proceed by replacing a well-formed public key by a so-called lossy public key, with respect to which forging a signature becomes statistically impossible. In particular, the DDH-based instantiation of Katz and Wang [27] appears as an ideal candidate since, somewhat analogously to [24], well-formed public keys can be seen as homomorphic Elgamal encryptions of 0. However, several difficulties arise when we try to adapt the techniques of [27,2] to the ring signature setting.

The first one is that the Groth-Kohlweiss ring signatures [24] rely on perfectly *hiding* commitments in order to achieve unconditional anonymity whereas the Elgamal encryption scheme is a perfectly binding commitment. This fortunately leaves us the hope for computational anonymity if we trade the perfectly hiding commitments for Elgamal encryptions. A second difficulty is to determine which public keys should be replaced by lossy keys in the reduction. At each public key generation query, the reduction has to decide if the newly generated key will be lossy or injective. Replacing all public keys by lossy keys is not possible because of corruptions (indeed, lossy public keys have no underlying secret key) and the reduction does not know in advance which public keys will end up in the target ring $\mathcal{R}^\star$ of the forgery. Only replacing a randomly chosen key by a lossy key does not work either: indeed, in the ring signature setting, having one lossy public key $PK^\dagger$ in the target ring $\mathcal{R}^\star$ does not prevent an unbounded adversary from using the secret key of a well-formed key $PK^\star \in \mathcal{R}^\star \setminus \{PK^\dagger\}$ to create a forgery. Moreover, as long as the reduction can only embed the challenge (injective or lossy) key in one output of the key generation oracle, it remains stuck with an advantage $\Theta(\varepsilon/Q_V)$ if the forger has advantage ε. Arguably, this bound is the

best we can hope for by directly applying the lossy identification technique.

To obtain a fully tight reduction, we depart from the lossy identification paradigm [2] in that, instead of tampering with one user's public keys at some step, our security proof embeds a DDH instance in the public parameters pp of the scheme. This allows the reduction to have all users' private keys at disposal and reveal them to the adversary upon request. In the real system, the set pp contains uniformly random group elements $(g, h, \tilde{g}, \tilde{h}, U, V) \in \mathbb{G}^6$ and each user's public key consists of a pair $(X, Y) = (g^{\alpha} \cdot h^{\beta}, \tilde{g}^{\alpha} \cdot \tilde{h}^{\beta})$, where $(\alpha, \beta) \in \mathbb{Z}_q^2$ is the secret key. The idea of the security proof is that, if $(g, h, \tilde{g}, \tilde{h}) \in \mathbb{G}^4$ is not a Diffie-Hellman tuple, the public key $PK = (X, Y)$ uniquely determines $(\alpha, \beta) \in \mathbb{Z}_q^2$. In the case $\tilde{h} = \tilde{g}^{\log_g(h)}$, the public key (X, Y) is compatible with q equally likely pairs (α, β) since it only reveals the information $\log_g(X) = \alpha + \log_g(h) \cdot \beta$.

The reduction thus builds a DDH distinguisher by forcing the adversary's forgery to contain a committed encoding $\Gamma = U^{\alpha} \cdot V^{\beta}$ of the signer's secret key $(\alpha, \beta) \in \mathbb{Z}_q^2$, which can be extracted using some trapdoor information. So long as (U, V) is linearly independent of (g, h), the encoding $\Gamma = U^{\alpha} \cdot V^{\beta}$ is independent of the adversary's view if $(g, h, \tilde{g}, \tilde{h})$ is a Diffie-Hellman tuple. In contrast, this encoding is uniquely determined by the public key if $\tilde{h} \neq \tilde{g}^{\log_g(h)}$. This allows the reduction to infer that $(g, h, \tilde{g}, \tilde{h})$ is a Diffie-Hellman tuple whenever it extracts $\Gamma = U^{\alpha} \cdot V^{\beta}$ from the adversary's forgery. To apply this argument, however, we need to make sure that signing queries do not leak any more information about (α, β) than the public key $PK = (X, Y)$ does. For this purpose, we resort to *lossy* encryption schemes [3] (a.k.a. dual-mode encryption/commitments [25,32]), which can either behave as perfectly hiding or perfectly binding commitments depending on the distribution of the public key. In each signature, we embed a lossy encryption $(T_0, T_1) = (g^{\theta_1} \cdot h^{\theta_2}, U^{\alpha} \cdot V^{\beta} \cdot H_1^{\theta_1} \cdot H_2^{\theta_2})$ of $\Gamma = U^{\alpha} \cdot V^{\beta}$, which is computed using the DDH-based lossy encryption scheme of [3]. If $(H_1, H_2) \in \mathbb{G}^2$ is linearly independent of (g, h), then (T_0, T_1) perfectly hides Γ. At the same time, the reduction should be able to extract Γ from (T_0, T_1) in the forgery. To combine these seemingly conflicting requirements, we derive (H_1, H_2) from a (pseudo-)random oracle which is programmed to have $(H_1, H_2) = (g^{\gamma}, h^{\gamma})$, for some $\gamma \in_R \mathbb{Z}_q$, in the adversary's forgery and maintain the uniformity of all pairs $(H_1, H_2) \in \mathbb{G}^2$ in all signing queries. By doing so, the witness indistinguishability of the Groth-Kohlweiss Σ-protocol [24] implies that the adversary only obtains a limited amount of information from uncorrupted users' private keys. While the above information theoretic argument is reminiscent of the security proof of Okamoto's identification scheme [30], our proof departs from [30] in that we do not rewind the adversary as it would not enable a tight reduction.

Related Work. The concept of ring signatures was coined by Rivest, Shamir and Tauman [34] who gave constructions based on trapdoor functions and proved their security in the ideal cipher model. They also mentioned different realizations based on proofs of partial knowledge [19]. The latter approach was extended by Abe *et al.* [1] to support rings containing keys from different underlying signatures and assumptions. Bresson, Stern and Szydlo [11] modified the scheme of Rivest *et al.* [34] so as to prove it secure in the random oracle model.

In 2006, Bender, Katz and Morselli [7] provided rigorous security definitions and theoretical constructions without random oracles. In the standard model, the first efficient instantiations were put forth by Shacham and Waters [36] in groups with a bilinear map. Brakerski and Tauman-Kalai [10] gave alternative constructions based on lattice assumptions. Meanwhile, Boyen [9] suggested a generalization of the primitive with standard-model instantiations.

The early realizations [34,11] had linear size in the cardinality of the ring. Dodis *et al.* [20] mentioned constant-size ring signatures as an application of their anonymous *ad hoc* identification protocols. However, their approach requires a setup phase where an RSA modulus is generated by some trusted entity. Chase and Lysyanskaya [15] suggested a similar construction of constant-size ring signatures from cryptographic accumulators [6]. However, efficiently instantiating their construction requires setup-free accumulators which are compatible with zero-knowledge proofs. The hash-based accumulators of [12,13] would not provide efficient solutions as they would incur proofs of knowledge of hash function pre-images. While the lattice-based construction of [28] relies on hash-based accumulators, its security proof is not tight and its efficiency is not competitive with discrete-logarithm-based techniques. Sander's number-theoretic accumulator [35] is an alternative candidate to instantiate [15] without a setup phase. However, it is not known to provide practical protocols: as observed in [24], it would involve much larger composite integers than standard RSA moduli (besides zero-knowledge proofs for double discrete logarithms). Moreover, it is not clear how it would be compatible with tight security proofs.

Chandran, Groth and Sahai [14] gave sub-linear-size signatures in the standard model, which were recently improved in [23]. Assuming a common reference string, Malavolta and Schröder [29] suggested a scheme with constant-size signatures. Their construction, however, relies on SNARKs (and thus non-falsifiable assumptions) to obtain a signature length independent of the number of ring members. In the random oracle model, Groth and Kohlweiss [24] described an elegant construction of logarithmic-size ring signatures based on the discrete logarithm assumption. Libert *et al.* [28] obtained logarithmic-size lattice-based ring signatures in the random oracle model.

The logarithmic-size ring signatures of [24,8,28] are obtained by applying the Fiat-Shamir heuristic [21] to interactive Σ-protocols. While these solutions admit security proofs under well-established assumptions in the random oracle model, their security reductions are pretty loose. In terms of exact security, they are doomed [31] to lose a linear factor in the number $Q_{\mathcal{H}}$ of random oracle queries as long as they rely on the Forking Lemma [33].

The exact security of digital signatures was first considered by Bellare and Rogaway [5] and drew a lot of attention [17,22,27,2,26] since then.

2 Background

2.1 Syntax and Security Definitions for Ring Signatures

Definition 1. *A ring signature scheme consists of a tuple of efficient algorithms* $(\mathsf{Par\text{-}Gen}, \mathsf{Keygen}, \mathsf{Sign}, \mathsf{Verify})$ *with the following specifications:*

Par-Gen(1^λ)**:** *Given a security parameter* λ*, outputs the public parameters* pp*.*
Keygen(pp)**:** *Given* pp*, outputs a key pair* (PK, SK) *for the user.*
Sign$(\mathsf{pp}, SK, \mathcal{R}, M)$**:** *Given the user's secret key* SK*, a ring* $\mathcal{R}$ *and a message* M*, outputs the signature* σ *of the message* M *on behalf of the ring* $\mathcal{R}$*.*
Verify$(\mathsf{pp}, M, \mathcal{R}, \sigma)$**:** *Given the message* M*, a ring* $\mathcal{R}$ *and a candidate signature* σ*, the verification algorithm outputs* 0 *or* 1*.*

These algorithms must also verify the correctness, meaning that for all $\mathsf{pp} \leftarrow \mathsf{Par\text{-}Gen}(1^\lambda)$, $(PK, SK) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$, for all M, and for all $\mathcal{R}$ such that $PK \in \mathcal{R}$, we have w.h.p $\mathsf{Verify}(\mathsf{pp}, M, \mathcal{R}, \mathsf{Sign}(\mathsf{pp}, SK, \mathcal{R}, M)) = 1$.

From a security point of view, Bender *et al.* [7] suggested the following stringent definitions of anonymity and unforgeability.

Definition 2. *A ring signature* $(\mathsf{Par\text{-}Gen}, \mathsf{Keygen}, \mathsf{Sign}, \mathsf{Verify})$ *provides statistical* **anonymity** *under* **full key exposure** *if, for any computationally unbounded adversary* $\mathcal{A}$*, there exists a negligible function* $\varepsilon(\lambda)$ *such that*

$$\begin{aligned}|\Pr[\mathsf{pp} \leftarrow \mathsf{Par\text{-}Gen}(1^\lambda); (M^\star, i_0, i_1, \mathcal{R}^\star) \leftarrow \mathcal{A}^{\mathsf{Keygen}(\cdot)}; b \xleftarrow{R} \{0,1\};\\ \sigma^\star \leftarrow \mathsf{Sign}(pp, SK_{i_b}, \mathcal{R}^\star, M^\star) : \mathcal{A}(\sigma^\star) = b] - \frac{1}{2}| < \varepsilon(\lambda),\end{aligned}$$

where $PK_{i_0}, PK_{i_1} \in \mathcal{R}^\star$ *and* Keygen *is an oracle that generates a fresh key pair* $(PK, SK) \leftarrow \mathsf{Keygen}(\mathsf{pp})$ *at each query and returns both* PK *and* SK *to* $\mathcal{A}$*.*

Definition 3. *A ring signature* $(\mathsf{Par\text{-}Gen}, \mathsf{Keygen}, \mathsf{Sign}, \mathsf{Verify})$ *provides* **unforgeability** *w.r.t* **insider corruption** *if, for any PPT adversary* $\mathcal{A}$*, there exists a negligible function* $\varepsilon(\lambda)$ *such that, for any* $\mathsf{pp} \leftarrow \mathsf{Par\text{-}Gen}(1^\lambda)$*, we have*

$$\Pr[(M, \mathcal{R}, \sigma) \leftarrow \mathcal{A}^{\mathsf{Keygen}(\cdot), \mathsf{Sign}(\cdot), \mathsf{Corrupt}(\cdot)}(\mathsf{pp}) : \mathsf{Verify}(\mathsf{pp}, M, \mathcal{R}, \sigma) = 1] < \varepsilon(\lambda),$$

- $\mathsf{Keygen}()$*: is an oracle that maintains a counter* j *initialized to* 0*. At each query, it increments* j*, generates* $(PK_j, SK_j) \leftarrow \mathsf{KeyGen}(\mathsf{pp})$ *and outputs* PK_j*.*
- $\mathsf{Sign}(i, M, \mathcal{R})$ *is an oracle that returns* $\sigma \leftarrow \mathsf{Sign}(\mathsf{pp}, SK_i, \mathcal{R}, M)$ *if* $PK_i \in \mathcal{R}$ *and* (PK_i, SK_i) *has been generated by* Keygen*. Otherwise, it returns* $\perp$*.*
- $\mathsf{Corrupt}(i)$ *returns* SK_i *if* (PK_i, SK_i) *was output by* Keygen *and* $\perp$ *otherwise.*
- $\mathcal{A}$ *is restricted to output a triple* $(M, \mathcal{R}, \sigma)$ *such that: (i) No query of the form* $(\star, M, \mathcal{R})$ *has been made to* $\mathsf{Sign}(\cdot,\cdot,\cdot)$*; (ii)* $\mathcal{R}$ *only contains public keys* PK_i *produced by* Keygen *and for which* i *was never queried to* $\mathsf{Corrupt}(\cdot)$*.*

2.2 Hardness Assumptions

Definition 4. *The* **Decision Diffie-Hellman** *(DDH) problem in $\mathbb{G}$, is to distinguish the distributions (g^a, g^b, g^{ab}) and (g^a, g^b, g^c), with $a, b, c \xleftarrow{R} \mathbb{Z}_q$. The DDH assumption is the intractability of the problem for any PPT distinguisher.*

2.3 Reminders on Σ-Protocols

Definition 5 ([18]). *Let a prover P and a verifier V, which are PPT algorithms, and a binary relation $\mathcal{R}$. A protocol (P, V) is a Σ-protocol w.r.t. $\mathcal{R}$, the challenge set $\mathcal{C}$, the public input u and the private input w, if it satisfies the following:*

- **3-move form:** *The protocol is of the following form:*
  - *P compute commitments $\{c_i\}_{i=0}^{j}$, where $j \in \mathbb{N}$, and sends $\{c_i\}_{i=0}^{j}$ to V.*
  - *The verifier V generates a random challenge $x \xleftarrow{R} \mathcal{C}$ and sends c to P.*
  - *The prover P sends a response s to V.*
  - *On input of a transcript $(\{c_i\}_{i=0}^{j}, x, s)$, V outputs 1 or 0.*
- **Completeness:** *If $(u, w) \in \mathcal{R}$ and the prover P honestly generates the transcript $(\{c_i\}_{i=0}^{j}, x, s)$ for a random challenge $x \xleftarrow{R} \mathcal{C}$ sent by V, there is a negligible function $\varepsilon(\lambda)$ such that V accepts with probability $1 - \varepsilon(\lambda)$.*
- **2-Special soundness:** *There exists a PPT knowledge extractor $\mathcal{E}$ that, for any public input u, on input of two accepting transcripts $(\{c_i\}_{i=0}^{j}, x, s)$ and $(\{c_i\}_{i=0}^{j}, x', s')$ with $x \neq x'$, outputs a witness w' such that $(u, w') \in \mathcal{R}$.*
- **Special Honest Verifier Zero-Knowledge (SHVZK):** *There is a PPT simulator $\mathcal{S}$ that, given u and a random $x \in \mathcal{C}$, outputs a simulated transcript $(\{c'_i\}_{i=0}^{j}, x, s')$ which is computationally indistinguishable from a real one.*

2.4 Σ-protocol Showing that a Commitment Opens to 0 or 1

We recall the Σ-protocol used in [24] to prove that a commitment opens to 0 or 1. Let $\mathcal{R} = \{(ck, c, (m, r)) \mid c = \mathsf{Com}_{ck}(m, r) \ \wedge \ (m, r) \in \{0, 1\} \times \mathbb{Z}_q\}$ the binary relation, where ck is the commitment key generated for the underlying commitment scheme, $u = c$ is the public input and $w = (m, r)$ is the private input. Figure 1 gives us a Σ-protocol (P, V) for $\mathcal{R}$.

Theorem 1 ([24, Theorem 2]). *Let $(\mathsf{Setup}, \mathsf{Com})$ be a perfectly binding, computationally hiding, strongly binding and additively homomorphic commitment scheme. The Σ-protocol presented in figure 1 for the commitment to 0 or to 1 is perfectly complete, perfectly 2-special sound and perfectly SHVZK.*

2.5 Σ-protocol for One-out-of-N Commitments Containing 0

Groth and Kohlweiss [24] used the Σ-protocol of Section 2.4 to build an efficient Σ-protocol allowing to prove knowledge of an opening of one-out-of-N commitments $\{c_i\}_{i=0}^{N-1}$ to $m = 0$. Their protocol outperforms the standard OR-proof approach [19] in that its communication complexity is only $O(\log N)$, instead of $O(N)$. The idea is to see the responses $f = mx + a$ of the basic Σ protocol as degree-1 polynomials in $x \in \mathbb{Z}_q$ and exploit the homomorphism of the commitment.

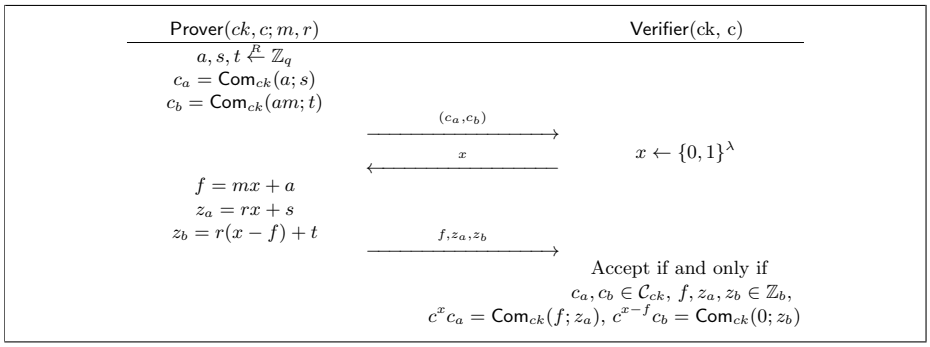

Fig. 1: Σ-protocol for commitment to $m \in \{0,1\}$

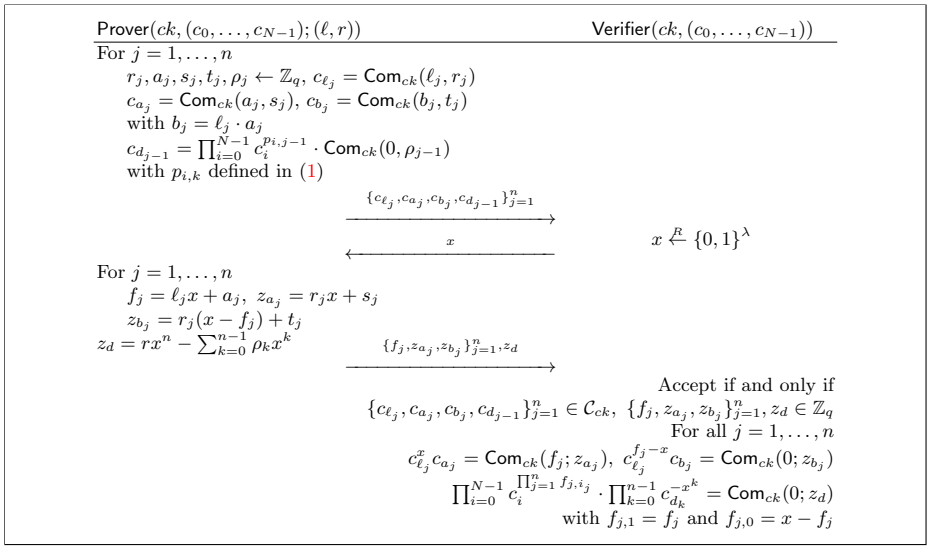

Fig. 2: Σ-protocol for one of $(c_0, \ldots, c_{N-1})$ commits to 0

Theorem 2 ([24, Theorem 3]). *The Σ-protocol of figure 2 is perfectly complete. It is (perfectly) $(n+1)$-special sound if the commitment is (perfectly) binding. It is (perfectly) SHVZK if the commitment scheme is (perfectly) hiding.*

In Figure 2, for each i, $p_{i,0}, \ldots, p_{i,n-1} \in \mathbb{Z}_q$ are the coefficients of the polynomial

$$P_i[Z] = \prod_{j=1}^{n} F_{j,i_j}[Z] = \delta_{i,\ell} \cdot Z^n + \sum_{k=0}^{n-1} p_{i,k} \cdot Z^k \qquad \forall i \in \{0, \ldots, N-1\} \tag{1}$$

obtained by defining $F_{j,1}[Z] = \ell_j \cdot Z + a_j$ and $F_{j,0}[Z] = Z - F_{j,1}[Z]$ for all $j \in [n]$. Note that the equality (1) stems from the fact that, for each index $i = i_1 \ldots i_n \in \{0, \ldots, N-1\}$, we have $F_{j,i_j}[Z] = \delta_{i_j,\ell_j} \cdot Z + (-1)^{\delta_{0,i_j}} \cdot a_j$ for all $j \in [n]$, so that the coefficient of Z^n in (1) is non-zero if and only if $i = \ell$.

2.6 A Note on the Application to Ring Signatures

In [24], Groth and Kohlweiss obtained a ring signature scheme by applying the Fiat-Shamir paradigm [21] to the above Σ-protocol. In short, key pairs are of the form (c, r) such that $c = \mathsf{Com}(0; r)$ and a ring signature associated with $\mathcal{R} = \{c_0, \ldots, c_N\}$ is simply a proof that the signer knows how to open to 0 one of the N commitments in that ring. In [24], the following theorem states about the security of the resulting construction, denoted $(\mathsf{Setup}, \mathsf{KGen}, \mathsf{Sign}, \mathsf{Vfy})$.

Theorem 3 ([24, Theorem 4]). *The scheme* $(\mathsf{Setup}, \mathsf{KGen}, \mathsf{Sign}, \mathsf{Vfy})$ *is a ring signature scheme with perfect correctness. It has perfect anonymity if the commitment scheme is perfectly hiding. It is unforgeable in the random oracle model if the commitment scheme is perfectly hiding and computationally binding.*

As the security of the ring signature relies on that of the Σ-protocol, it is interesting to take a closer look at the computation of commitments $\{C_{d_{j-1}}\}_{j=1}^{n}$ in Figure 2. This part of the Σ-protocol is the only point where the ring signature generation may involve adversarially-generated values. In the anonymity game, the signer's public key may be one of the only two honestly-generated public keys in the ring $\mathcal{R}$. The security proof of [24] argues that, as long as the commitment is perfectly hiding, the fact that each $C_{d_{j-1}}$ contains a (randomizing) factor $\mathsf{Com}(0; \rho_{j-1})$, for some uniformly random ρ_{j-1}, is sufficient to guarantee perfect anonymity. We point out an issue that arises when $\mathcal{R} = \{c_0, \ldots, c_N\}$ contains maliciously generated keys outside the space of honestly generated commitments (even if they are perfectly hiding). In short, multiplying a maliciously generated commitment by a fresh commitment may not fully "clean-up" its distribution.

The following example is a perfectly hiding commitment where re-randomizing does not wipe out maliciously generated commitments components: the setup algorithm outputs generators $ck = (g, h)$ cyclic group $\mathbb{G}$ of prime order q; committing to $m \in \mathbb{Z}_q$ using randomness $\rho = (r, s) \xleftarrow{R} \mathbb{Z}_q^2$ is achieved by computing $\mathsf{Com}_{ck}(m; \rho) = (c_1, c_2, c_3) = (g^m h^r, g^s, h^s) \in \mathbb{G}^3$, which is a perfectly hiding commitment since c_1 is a Pedersen commitment and the Elgamal encryption (c_2, c_3) of 0 is independent of c_1. If we consider the maliciously generated commitment $(c_1^\star, c_2^\star, c_3^\star) = (h^u, g^v, g \cdot h^v)$, multiplying it by any $\mathsf{Com}_{ck}(0; \rho)$ does not bring it back in the range of Com. Therefore, in an instantiation with the above commitment, an unbounded adversary can defeat the anonymity property.

The only missing requirement on behalf of the underlying perfectly hiding commitment is that it should be possible to efficiently recognize elements in the range of the commitment algorithm. This assumption is quite natural and satisfied by schemes like Pedersen's commitment. Hence, this observation does not affect the perfect anonymity of the discrete-log-based instantiation of [24].

3 A Fully Tight Construction from the DDH Assumption

We modify the scheme of [24] so as to prove its unforgeability via a fully tight reduction from the DDH assumption. The advantage of the DDH distinguisher is only smaller than the adversary's advantage by a (small) constant factor.

The price to pay for this fully tight reduction is relatively small since signatures are only longer than in [24] by roughly $2n$ group elements. Moreover, as in [24], our signing algorithm requires $\Theta(N)$ exponentiations if N is the size of the ring.

3.1 Description

We exploit the fact that, in the Σ-protocol of [24], not all first-round messages should be computed using the same commitment scheme as the one used to compute the public key. The second step of the signing algorithm computes perfectly hiding commitments $\{\boldsymbol{C}_{d_k}\}_{k=0}^{n-1}$ which are vectors of dimension 4. They live in a different space than public keys $(X, Y) = (g^{\alpha} \cdot h^{\beta}, \tilde{g}^{\alpha} \cdot \tilde{h}^{\beta})$, which are DDH-based lossy encryptions of (and thus perfectly hiding commitments to) 0.

The signer generates a commitment $(T_0, T_1) = (g^{\theta_1} \cdot h^{\theta_2}, \Gamma \cdot H_1^{\theta_1} \cdot H_2^{\theta_2})$ to $\Gamma = U^{\alpha_\ell} \cdot V^{\beta_\ell}$, which encodes his secret key $(\alpha_\ell, \beta_\ell) \in \mathbb{Z}_q^2$. This defines a vector $\boldsymbol{V}_\ell = (X_\ell, Y_\ell, T_0, T_1) \in \mathbb{G}^4$ in the column space of a matrix $\mathbf{M}_H \in \mathbb{G}^{4\times 4}$, which has full rank in the scheme but not in the proof of unforgeability. Then, for each key $\boldsymbol{X}_i = (X_i, Y_i)$ in the ring $\mathcal{R}$, the signer defines $\boldsymbol{V}_i = (X_i, Y_i, T_0, T_1)^{\top} \in \mathbb{G}^4$ and, by extending the technique of [24], generates a NIZK proof that one of the vectors $\{\boldsymbol{V}_i\}_{i=0}^{N-1}$ is in the column span of $\mathbf{M}_H$. To prove this without revealing which $\boldsymbol{V}_\ell \in \mathbb{G}^4$ is used, the commitments $\{\boldsymbol{C}_{d_{j-1}}\}_{j=1}^{n}$ are re-randomized by multiplying them with a random vector in the column space of $\mathbf{M}_H$.

Par-Gen(1^λ)**:** Given a security parameter λ, choose a cyclic group $\mathbb{G}$ of prime order q with generators $g, h, \tilde{g}, \tilde{h} \xleftarrow{R} \mathbb{G}$ and $U, V \xleftarrow{R} \mathbb{G}$. Choose hash functions $\mathcal{H}_{\mathsf{FS}} : \{0,1\}^* \to \mathbb{Z}_q$ and $\mathcal{H} : \{0,1\}^* \to \mathbb{G}^2$ which will be modeled as random oracles. Output the common public parameters $\mathsf{pp} = \big(\lambda, \mathbb{G}, g, h, \tilde{g}, \tilde{h}, U, V\big)$.

Keygen(pp)**:** Given pp, choose a secret key is $SK = (\alpha, \beta) \xleftarrow{R} \mathbb{Z}_q^2$ and compute the public key $PK = \boldsymbol{X} = (X, Y) = (g^{\alpha} \cdot h^{\beta}, \tilde{g}^{\alpha} \cdot \tilde{h}^{\beta})$.

Sign$(\mathsf{pp}, SK, \mathcal{R}, M)$**:** To sign $M \in \{0,1\}^*$ on behalf of $\mathcal{R} = \{\boldsymbol{X}_0, \ldots, \boldsymbol{X}_{N-1}\}$ such that $\boldsymbol{X}_i = (X_i, Y_i) \in \mathbb{G}^2$ for each $i \in [N]$, the signer uses $SK = (\alpha, \beta)$ and $PK = \boldsymbol{X} = (X, Y) = (g^{\alpha} \cdot h^{\beta}, \tilde{g}^{\alpha} \cdot \tilde{h}^{\beta}) \in \mathcal{R}$ as follows. We assume that $N = 2^n$ for some n. Let $\ell \in \{0, \ldots, N-1\}$ the index of $PK = \boldsymbol{X}$ in $\mathcal{R}$ when $\mathcal{R}$ is arranged in lexicographical order and write it as $\ell = \ell_1 \ldots \ell_n \in \{0,1\}^n$.

1. Choose $\theta_1, \theta_2 \xleftarrow{R} \mathbb{Z}_q$. For all $j \in [n]$, choose $a_j, r_j, s_j, t_j, u_j, v_j, w_j, \rho_{j-1} \xleftarrow{R} \mathbb{Z}_q$ and compute $(T_0, T_1) = \big(g^{\theta_1} \cdot h^{\theta_2}, U^{\alpha} \cdot V^{\beta} \cdot H_1^{\theta_1} \cdot H_2^{\theta_2}\big)$, as well as

$$\begin{aligned}
\boldsymbol{C}_{\ell_j} &= (C_{\ell_j,0}, C_{\ell_j,1}) = \big(g^{r_j} \cdot h^{s_j},\ g^{\ell_j} \cdot H_1^{r_j} \cdot H_2^{s_j}\big) \\
\boldsymbol{C}_{a_j} &= (C_{a_j,0}, C_{a_j,1}) = \big(g^{t_j} \cdot h^{u_j},\ g^{a_j} \cdot H_1^{t_j} \cdot H_2^{u_j}\big) \\
\boldsymbol{C}_{b_j} &= (C_{b_j,0}, C_{b_j,1}) = \big(g^{v_j} \cdot h^{w_j},\ g^{\ell_j \cdot a_j} \cdot H_1^{v_j} \cdot H_2^{w_j}\big),
\end{aligned} \tag{2}$$

   where $(H_1, H_2) = \mathcal{H}\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^{n}\big) \in \mathbb{G}^2$. Define

$$\mathbf{M}_H = \begin{bmatrix} g & h & 1 & 1 \\ \tilde{g} & \tilde{h} & 1 & 1 \\ 1 & 1 & g & h \\ U & V & H_1 & H_2 \end{bmatrix} \in \mathbb{G}^{4\times 4} \tag{3}$$

and its corresponding discrete logarithms $\mathbf{L}_h = \log_g(\mathbf{M}_H)$ matrix

$$\mathbf{L}_h = \begin{bmatrix} 1 & \log_g(h) & 0 & 0 \\ \log_g(\tilde{g}) & \log_g(\tilde{h}) & 0 & 0 \\ 0 & 0 & 1 & \log_g(h) \\ \log_g(U) & \log_g(V) & \log_g(H_1) & \log_g(H_2) \end{bmatrix} \in \mathbb{Z}_q^{4\times 4}. \quad (4)$$

Note that the signer's witnesses $(\alpha, \beta, \theta_1, \theta_2) \in \mathbb{Z}_q^4$ satisfy

$$\log_g \left[X \mid Y \mid T_0 \mid T_1\right]^\top = \mathbf{L}_h \cdot \left[\alpha \mid \beta \mid \theta_1 \mid \theta_2\right]^\top. \quad (5)$$

In the following, we will sometimes re-write relation (5) as

$$\begin{bmatrix} X \\ Y \\ T_0 \\ T_1 \end{bmatrix} = \begin{bmatrix} g & h & 1 & 1 \\ \tilde{g} & \tilde{h} & 1 & 1 \\ 1 & 1 & g & h \\ U & V & H_1 & H_2 \end{bmatrix} \odot \begin{bmatrix} \alpha \\ \beta \\ \theta_1 \\ \theta_2 \end{bmatrix}. \quad (6)$$

For each $i \in [N]$, define the vector $\boldsymbol{V}_i = (X_i, Y_i, T_0, T_1)^\top \in \mathbb{G}^4$. The next step is to prove knowledge of witnesses $(\alpha_\ell, \beta_\ell, \theta_1, \theta_2) \in \mathbb{Z}_q^4$ such that $\boldsymbol{V}_\ell = (X_\ell, Y_\ell, T_0, T_1)^\top = g^{\mathbf{L}_h \cdot (\alpha_\ell, \beta_\ell, \theta_1, \theta_2)^\top}$, for some $\ell \in [N]$.

2. For each $j \in [n]$, pick $\rho_{j-1,\alpha}, \rho_{j-1,\beta}, \rho_{j-1,\theta_1}, \rho_{j-1,\theta_2} \xleftarrow{R} \mathbb{Z}_q$ and compute

$$\boldsymbol{C}_{d_{j-1}} = \prod_{i=0}^{N-1} \boldsymbol{V}_i^{p_{i,j-1}} \cdot g^{\mathbf{L}_h \cdot (\rho_{j-1,\alpha}, \rho_{j-1,\beta}, \rho_{j-1,\theta_1}, \rho_{j-1,\theta_2})^\top} \in \mathbb{G}^4, \quad (7)$$

where, for each $i \in \{0, \ldots, N-1\}$, $p_{i,0}, \ldots, p_{i,n-1}$ are the coefficients of

$$P_i[Z] = \prod_{j=1}^{n} F_{j,i_j}[Z] = \delta_{i,\ell} \cdot Z^n + \sum_{k=0}^{n-1} p_{i,k} \cdot Z^k \in \mathbb{Z}_q[Z], \quad (8)$$

where $F_{j,1}[Z] = \ell_j \cdot Z + a_j$ and $F_{j,0}[Z] = Z - F_{j,1}[Z]$ for all $j \in [n]$. Note that the coefficient of Z^n in (8) is non-zero if and only if $i = \ell$.

3. Compute $x = \mathcal{H}_{\mathsf{FS}}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^n) \in \mathbb{Z}_q$.
4. For each $j \in [n]$, compute (modulo q) $f_j = \ell_j \cdot x + a_j = F_{j,1}(x)$ and

$$z_{r_j} = r_j \cdot x + t_j, \qquad \bar{z}_{r_j} = r_j \cdot (x - f_j) + v_j$$
$$z_{s_j} = s_j \cdot x + u_j, \qquad \bar{z}_{s_j} = s_j \cdot (x - f_j) + w_j$$

and

$$z_{d,\alpha} = \alpha \cdot x^n - \sum_{k=0}^{n-1} \rho_{k,\alpha} \cdot x^k, \qquad z_{d,\beta} = \beta \cdot x^n - \sum_{k=0}^{n-1} \rho_{k,\beta} \cdot x^k$$
$$z_{d,\theta_1} = \theta_1 \cdot x^n - \sum_{k=0}^{n-1} \rho_{k,\theta_1} \cdot x^k, \qquad z_{d,\theta_2} = \theta_2 \cdot x^n - \sum_{k=0}^{n-1} \rho_{k,\theta_2} \cdot x^k$$

Let $\boldsymbol{\Sigma}_j = \big(\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}, f_j, z_{r_j}, z_{s_j}, \bar{z}_{r_j}, \bar{z}_{s_j}\big)$ for all $j \in [n]$ and output

$$\sigma = \big(\{\boldsymbol{\Sigma}_j\}_{j=1}^n, T_0, T_1, z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2}\big). \tag{9}$$

Verify$(\mathsf{pp}, M, \mathcal{R}, \sigma)$**:** Given a ring $\mathcal{R} = \{\boldsymbol{X}_0, \ldots, \boldsymbol{X}_{N-1}\}$ and a pair (M, σ), parse σ as in (9) and define $f_{j,1} = f_j$ and $f_{j,0} = x - f_j$ for each $j \in [n]$.

1. Compute $(H_1, H_2) = \mathcal{H}\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big) \in \mathbb{G}^2$ and, for each public key $\boldsymbol{X}_i = (X_i, Y_i) \in \mathbb{G}^2$ in $\mathcal{R}$, set $\boldsymbol{V}_i = (X_i, Y_i, T_0, T_1)^\top \in \mathbb{G}^4$.
2. Let $x = \mathcal{H}_{\mathsf{FS}}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^n)$. If the equalities

$$\begin{aligned} \boldsymbol{C}_{a_j} \cdot \boldsymbol{C}_{\ell_j}^{x} &= \big(g^{z_{r_j}} \cdot h^{z_{s_j}},\ g^{f_j} \cdot H_1^{z_{r_j}} \cdot H_2^{z_{s_j}}\big), \\ \boldsymbol{C}_{b_j} \cdot \boldsymbol{C}_{\ell_j}^{x-f_j} &= \big(g^{\bar{z}_{r_j}} \cdot h^{\bar{z}_{s_j}}, H_1^{\bar{z}_{r_j}} \cdot H_2^{\bar{z}_{s_j}}\big), \qquad \forall j \in [n] \end{aligned} \tag{10}$$

   are not satisfied, return 0. Then, return 1 if and only if

$$\begin{aligned} &\prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^n f_{j,i_j}} \cdot \prod_{j=1}^{n} \boldsymbol{C}_{d_{j-1}}^{-(x^{j-1})} \\ &\qquad = \begin{bmatrix} g & h & 1 & 1 \\ \tilde{g} & \tilde{h} & 1 & 1 \\ 1 & 1 & g & h \\ U & V & H_1 & H_2 \end{bmatrix} \odot \begin{bmatrix} z_{d,\alpha} \\ z_{d,\beta} \\ z_{d,\theta_1} \\ z_{d,\theta_2} \end{bmatrix}. \end{aligned} \tag{11}$$

Correctness is shown by observing from (8) that $\prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^n f_{j,i_j}}$ equals

$$\begin{aligned} \prod_{i=0}^{N-1} \boldsymbol{V}_i^{P_i(x)} &= \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\delta_{i,\ell} \cdot x^n + \sum_{k=0}^{n-1} p_{i,k} \cdot x^k} = \boldsymbol{V}_\ell^{x^n} \cdot \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\sum_{k=0}^{n-1} p_{i,k} \cdot x^k} \\ &= \boldsymbol{V}_\ell^{x^n} \cdot \prod_{k=0}^{n-1} \cdot \Big(\prod_{i=0}^{N-1} \boldsymbol{V}_i^{p_{i,k}}\Big)^{x^k} = \boldsymbol{V}_\ell^{x^n} \cdot \prod_{k=0}^{n-1} \cdot \big(\boldsymbol{C}_{d_k} \cdot g^{-\mathbf{L}_h \cdot (\rho_{k,\alpha}, \rho_{k,\beta}, \rho_{k,\theta_1}, \rho_{k,\theta_2})^\top}\big)^{x^k}, \end{aligned}$$

where the last equality follows from (7). Since $\boldsymbol{V}_\ell = g^{\mathbf{L}_h \cdot (\alpha_\ell, \beta_\ell, \theta_1, \theta_2)^\top}$, we obtain

$$\begin{aligned} \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^n f_{j,i_j}} \cdot \prod_{k=0}^{n-1} \cdot \boldsymbol{C}_{d_k}^{-x^k} &= \boldsymbol{V}_\ell^{x^n} \cdot \prod_{k=0}^{n-1} g^{-\mathbf{L}_h \cdot (\rho_{k,\alpha}, \rho_{k,\beta}, \rho_{k,\theta_1}, \rho_{k,\theta_2})^\top (x^k)}, \\ &= g^{\mathbf{L}_h \cdot (z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2})^\top}. \end{aligned}$$

3.2 Security Proofs

Statistical anonymity is achieved because $\{\boldsymbol{C}_{d_{j-1}}\}_{j=1}^n$ are uniformly distributed. The reason is that the matrices (4) have full rank in the scheme (but not in the proof of unforgeability), so that computing $\boldsymbol{C}_{d_{j-1}}$ as per (7) makes its distribution uniform over $\mathbb{G}^4$.

Theorem 4. *Any unbounded* **anonymity** *adversary $\mathcal{A}$ has advantage at most* $\mathbf{Adv}^{\mathrm{anon}}_{\mathcal{A}}(\lambda) \leq \frac{2}{q} + \frac{Q_{\mathcal{H}_{\mathsf{FS}}}}{q^2}$, *where $Q_{\mathcal{H}_{\mathsf{FS}}}$ is the number of hash queries to $\mathcal{H}_{\mathsf{FS}}$.*

The proof of Theorem 4 is given in Appendix B.1.

Theorem 5. *The scheme is* **unforgeable** *under the DDH assumption in the random oracle model. For any adversary $\mathcal{A}$ with running time t and making Q_V queries to the key generation oracle, Q_S signing queries as well as $Q_{\mathcal{H}}$ and $Q_{\mathcal{H}_{\mathsf{FS}}}$ queries to the random oracles $\mathcal{H}$ and $\mathcal{H}_{\mathsf{FS}}$, respectively, there is a DDH distinguisher $\mathcal{B}$ with running time $t' \leq t + \mathsf{poly}(\lambda, Q_S, Q_V, Q_{\mathcal{H}})$ and such that*

$$\mathbf{Adv}^{\mathrm{euf-cma}}_{\mathcal{A}}(\lambda) \leq \; 2 \cdot \mathbf{Adv}^{\mathrm{DDH}}_{\mathcal{B}}(\lambda) + \frac{Q_S + Q_{\mathcal{H}_{\mathsf{FS}}} \cdot (1 + \log Q_V) + 5}{q} + \frac{Q_S \cdot (Q_{\mathcal{H}_{\mathsf{FS}}} + 2Q_{\mathcal{H}} + 2Q_S)}{q^2}. \tag{12}$$

Proof. We use a sequence of games where, for each i, W_i stands for the event that the challenger outputs 1 in Game i.

Game 0**:** This is the real game. At each query $i \in [Q_V]$ to the key generation oracle $\mathsf{Keygen}(\cdot)$, the challenger $\mathcal{B}$ honestly chooses $\alpha_i, \beta_i \xleftarrow{R} \mathbb{Z}_q$ and returns the public key $PK_i = \boldsymbol{X}_i = (X_i, Y_i) = (g^{\alpha_i} \cdot h^{\beta_i}, \tilde{g}^{\alpha_i} \cdot \tilde{h}^{\beta_i})$ and retains $SK_i = (\alpha_i, \beta_i)$ for later use. If $\mathcal{A}$ subsequently submits $\boldsymbol{X}_i = (X_i, Y_i)$ to the corruption oracle, $\mathcal{B}$ reveals $SK_i = (\alpha_i, \beta_i)$. Moreover, all signing queries are answered by faithfully running the signing algorithm. At the end of the game, $\mathcal{A}$ outputs a forgery $(M^\star, \sigma^\star, \mathcal{R}^\star)$, where $\mathcal{R}^\star = \{\boldsymbol{X}^\star_0, \ldots, \boldsymbol{X}^\star_{N^\star - 1}\}$,

$$\sigma^\star = \big(\{\boldsymbol{\Sigma}^\star_j\}_{j=1}^n, T^\star_0, T^\star_1, z^\star_{d,\alpha}, z^\star_{d,\beta}, z^\star_{d,\theta_1}, z^\star_{d,\theta_2}\big), \tag{13}$$

with $\boldsymbol{\Sigma}^\star_j = \big(\boldsymbol{C}^\star_{\ell_j}, \boldsymbol{C}^\star_{a_j}, \boldsymbol{C}^\star_{b_j}, \boldsymbol{C}^\star_{d_{j-1}}, f^\star_j, z^\star_{r_j}, z^\star_{s_j}, \bar{z}^\star_{r_j}, \bar{z}^\star_{s_j}\big)$. At this point, $\mathcal{B}$ outputs 1 if and only if $\mathcal{A}$ wins, meaning that: (i) $\sigma^\star$ correctly verifies; (ii) $\mathcal{R}^\star$ only contains uncorrupted public keys; (iii) No signing query involved a tuple of the form $(\cdot, M^\star, \mathcal{R}^\star)$. By definition, we have $\mathbf{Adv}^{\mathrm{euf-cma}}_{\mathcal{A}}(\lambda) = \Pr[W_0]$.

Game 1**:** This game is like Game 0 but we modify the signing oracle. Note that each signing query triggers a query to the random oracle $\mathcal{H}(.)$ since the challenger $\mathcal{B}$ has to faithfully compute T_0 and $\{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n$ before obtaining $\mathcal{H}\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big)$. In Game 1, at each signing query, $\mathcal{B}$ aborts in the event that $\mathcal{H}(\cdot)$ was already defined for the input $\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big)$. Since such an input contains uniformly random elements, the probability to abort during the entire game is at most $Q_S \cdot (Q_S + Q_{\mathcal{H}})/q^2$ and we have $|\Pr[W_1] - \Pr[W_0]| \leq Q_S \cdot (Q_S + Q_{\mathcal{H}})/q^2$.

Game 2**:** We modify the random oracle $\mathcal{H}$ when it is directly invoked by $\mathcal{A}$ (i.e., $\mathcal{H}$-queries triggered by signing queries are treated as in Game 0). At each $\mathcal{H}$-query $\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big)$, the challenger $\mathcal{B}$ returns the previously defined value if it exists. Otherwise, it picks $\gamma \xleftarrow{R} \mathbb{Z}_q$ and defines the hash value as $(H_1, H_2) = \mathcal{H}\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big) = (g^\gamma, h^\gamma)$.

Note that $\mathcal{H}(\cdot)$ is no longer a truly random oracle since (g, h, H_1, H_2) is a Diffie-Hellman tuple. Still, under the DDH assumption, this modification has no noticeable effect on $\mathcal{A}$'s winning probability. Lemma 1 describes a DDH distinguisher such that $|\Pr[W_2] - \Pr[W_1]| \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda) + 1/q$.

Since (g, h, H_1, H_2) is a Diffie-Hellman tuple in Game 2, $\gamma \in \mathbb{Z}_q$ can be used as a decryption key for the DDH-based dual-mode encryption scheme. Another consequence of the last transition is that the matrix $\mathbf{L}_h$ of (3) has no longer full rank since its last row is linearly dependent with the first three rows.

Game 3: We introduce a failure event F_3 which causes the challenger $\mathcal{B}$ to output 0. When $\mathcal{A}$ outputs its forgery $\sigma^\star$, $\mathcal{B}$ parses $\sigma^\star$ as in (13) and computes $(H_1^\star, H_2^\star) = \mathcal{H}\big(M^\star, \mathcal{R}^\star, T_0^\star, \{C_{\ell_j,0}^\star, C_{a_j,0}^\star, C_{b_j,0}^\star\}_{j=1}^n\big)$. Event F_3 is defined to be the event that either: (1) The hash value $(H_1^\star, H_2^\star)$ was not defined at any time; (2) It was defined but collides with a pair $(H_1, H_2) = \mathcal{H}\big(M, \mathcal{R}, T_0, \{C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0}\}_{j=1}^n\big)$ defined in response to a signing query $(\ell, M, \mathcal{R})$ for some index $\ell \in \{0, \ldots, |\mathcal{R}| - 1\}$, when $\mathcal{R}$ is arranged in lexicographic order. Note that the probability of case (1) cannot exceed $1/q$ because $\mathcal{H}(\cdot)$ is unpredictable as a random oracle. Moreover, since a winning adversary must forge a signature on some $(M^\star, \mathcal{R}^\star)$ that has never been queried for signature, the probability of case (2) is bounded by Q_S/q^2 multiplied by $Q_{\mathcal{H}}$ since we must consider the probability that a tuple (g, h, H_1, H_2) defined in a signing query is accidentally a Diffie-Hellman tuple and collides with the response of a hash query. We find $|\Pr[W_3] - \Pr[W_2]| \leq \Pr[F_3] \leq 1/q + Q_S \cdot Q_{\mathcal{H}}/q^2$.

Game 4: This game is identical to Game 3 with one modification. When the adversary $\mathcal{A}$ outputs its forgery $\sigma^\star$, $\mathcal{B}$ parses $\sigma^\star$ as in (13) and computes $(H_1^\star, H_2^\star) = \mathcal{H}\big(M^\star, \mathcal{R}^\star, T_0^\star, \{C_{\ell_j,0}^\star, C_{a_j,0}^\star, C_{b_j,0}^\star\}_{j=1}^n\big)$. Then, $\mathcal{B}$ recalls the previously defined exponent $\gamma^\star \in \mathbb{Z}_q$ such that $(H_1^\star, H_2^\star) = (g^{\gamma^\star}, h^{\gamma^\star})$ and uses it to decrypt the dual-mode ciphertexts $\{\boldsymbol{C}_{\ell_j}^\star\}_{j=1}^n$. It aborts and outputs 0 if one of these ciphertexts turns out not to encrypt a bit $\ell_j^\star \in \{0,1\}$. Note that, if $\mathcal{B}$ does not abort, it decodes an n-bit string $\ell^\star = \ell_1^\star \ldots \ell_n^\star \in \{0,1\}^n$ from $\{\boldsymbol{C}_{\ell_j}^\star\}_{j=1}^n$. We claim that we have $|\Pr[W_4] - \Pr[W_3]| \leq (1 + Q_{\mathcal{H}_{\mathsf{FS}}})/q$.

The only situation where Game 4 deviates from Game 3 is the event F_4 that either: (i) $\mathcal{A}$ did not query $\mathcal{H}_{\mathsf{FS}}(\cdot)$ on the input that the forgery relates to; (ii) $\mathcal{A}$ manages to break the soundness of the proof system showing that each of the ciphertexts $\{\boldsymbol{C}_{\ell_j}^\star\}_{j=1}^n$ encrypts a bit. Lemma 2 shows that $\Pr[F_4] \leq (1 + Q_{\mathcal{H}_{\mathsf{FS}}})/q$.

Game 5: In this game, we modify the challenger's behavior when $\mathcal{A}$ outputs a forgery $\sigma^\star$. Having decoded the n-bit string $\ell^\star = \ell_1^\star \ldots \ell_n^\star \in \{0,1\}^n$ from the dual-mode ciphertexts $\{\boldsymbol{C}_{\ell_j}^\star\}_{j=1}^n$, $\mathcal{B}$ also runs the decryption algorithm for $(T_0^\star, T_1^\star)$ to compute $\Gamma^\star = T_1^\star / {T_0^\star}^{\gamma^\star}$. At this point, $\mathcal{B}$ recalls the secret key $SK = (\alpha_{\ell^\star}, \beta_{\ell^\star})$ of the $\ell^\star$-th member of the ring $\mathcal{R}^\star = \{\boldsymbol{X}_0^\star, \ldots \boldsymbol{X}_{N^\star-1}^\star\}$ in lexicographical order. If $\Gamma^\star = U^{\alpha_{\ell^\star}} \cdot V^{\beta_{\ell^\star}}$, $\mathcal{B}$ outputs 1. Otherwise, it outputs 0. Lemma 3 shows that $|\Pr[W_5] - \Pr[W_4]| \leq Q_{\mathcal{H}_{\mathsf{FS}}} \cdot \log(Q_V)/q$.

Game 6: This game is identical to Game 5 except that we change the distribution of $\mathsf{pp} = \big(\lambda, \mathbb{G}, g, h, \tilde{g}, \tilde{h}, U, V\big)$. Here, instead of choosing $g, h, \tilde{g}, \tilde{h} \xleftarrow{R} \mathbb{G}$ uniformly, we set $(g, h, \tilde{g}, \tilde{h}) = (g, h, g^{\rho}, h^{\rho})$ for a randomly chosen $\rho \xleftarrow{R} \mathbb{Z}_q$. Clearly, the two distributions of pp are indistinguishable under the DDH assumption and $\mathcal{B}$ can immediately be turned into an efficient DDH distinguisher (the proof is straightforward) such that $|\Pr[W_6] - \Pr[W_5]| \leq \mathbf{Adv}^{\mathrm{DDH}}_{\mathcal{B}}(\lambda)$.

Game 7: This game is like Game 6 except that we now simulate the proof of knowledge of secret keys in all outputs of the signing oracle. On a signing query $(M, \mathcal{R}, \ell)$, where $(0 \leq \ell \leq |\mathcal{R}| - 1)$, the challenger parses $\mathcal{R}$ as $\{\boldsymbol{X}_0, \ldots, \boldsymbol{X}_{N-1}\}$ and returns $\perp$ if $\boldsymbol{X}_{\ell}$ is not public keys produced by the $\mathsf{Keygen}(.)$ oracle. Otherwise, the challenger chooses $x \xleftarrow{R} \mathbb{Z}_q$ as well as $z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2} \xleftarrow{R} \mathbb{Z}_q$ and $f_j, z_{r_j}, z_{s_j}, \bar{z}_{r_j}, \bar{z}_{s_j} \xleftarrow{R} \mathbb{Z}_q$, for all $j \in [n]$. Then, it picks $T_0 \xleftarrow{R} \mathbb{G}$ as well as $r_j, s_j \xleftarrow{R} \mathbb{Z}_q$ for all $j \in [n]$, and honestly computes $C_{\ell_j,0} = g^{r_j} \cdot h^{s_j}$ for all $j \in [n]$. It can now compute for all $j \in [n]$,

$$C_{a_j,0} = g^{z_{r_j}} \cdot h^{z_{s_j}} \cdot C_{\ell_j,0}^{-x}, \qquad C_{b_j,0} = g^{\bar{z}_{r_j}} \cdot h^{\bar{z}_{s_j}} \cdot C_{\ell_j,0}^{f_j - x},$$

and define $(H_1, H_2) = \mathcal{H}(M, \mathcal{R}, T_0, \{(C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0})\}_{j=1}^{n})$. Then, it completes the computation of dual-mode commitments as follows. First, it chooses $T_1 \xleftarrow{R} \mathbb{G}$ and computes $C_{\ell_j,1} = g^{\ell_j} \cdot H_1^{r_j} \cdot H_2^{s_j}$ for all $j \in [n]$. Then, it computes

$$C_{a_j,1} = \big(g^{f_j} \cdot H_1^{z_{r_j}} \cdot H_2^{z_{s_j}}\big) \cdot C_{\ell_j,1}^{-x}, \qquad C_{b_j,1} = \big(H_1^{\bar{z}_{r_j}} \cdot H_2^{\bar{z}_{s_j}}\big) \cdot C_{\ell_j,1}^{f_j - x},$$

for all $j \in [n]$. Then, for each $j \in \{2, \ldots, n\}$, the challenger faithfully computes $\boldsymbol{C}_{d_{j-1}}$ as per (7) but, for index $j = 1$, it computes

$$\boldsymbol{C}_{d_0} = \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^{n} f_{j,i_j}} \prod_{j=2}^{n} \boldsymbol{C}_{d_{j-1}}^{-(x^{j-1})} \Big(\mathbf{M}_{\mathcal{H}} \odot (-z_{d,\alpha}, -z_{d,\beta}, -z_{d,\theta_1}, -z_{d,\theta_2})^{\top}\Big),$$

where $\boldsymbol{V}_i = (X_i, Y_i, T_0, T_1)^{\top}$, $f_{j,1} = f_j$ and $f_{j,0} = x - f_j$ for each $j \in [n]$. Finally, the challenger $\mathcal{B}$ programs the random oracle $\mathcal{H}_{\mathsf{FS}}$ to have the equality $x = \mathcal{H}_{\mathsf{FS}}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^{n})$. If $\mathcal{H}_{\mathsf{FS}}$ was already defined for this input, $\mathcal{B}$ aborts and outputs 0. If the simulation does not fail, the oracle sets $\boldsymbol{\Sigma}_j = \big(\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}, f_j, z_{r_j}, z_{s_j}, \bar{z}_{r_j}, \bar{z}_{s_j}\big)$ for all $j \in [n]$ and outputs the signature $\sigma = \big(\{\boldsymbol{\Sigma}_j\}_{j=1}^{n}, T_0, T_1, z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2}\big)$, which is distributed exactly as in Game 6 unless (g, h, H_1, H_2) happens to form a Diffie-Hellman tuple. Indeed, although the adversary's signing queries may involve rings $\mathcal{R}$ that contain maliciously generated keys of the form $\boldsymbol{X}_i = (X_i, Y_i) = (X_i, \Omega_i \cdot X_i^{\log_g(\tilde{g})})$, with $\Omega_i \neq 1_{\mathbb{G}}$, this does not prevent the simulated commitments $\{\boldsymbol{C}_{d_{j-1}}\}_{j=1}^{n}$ from having the same distribution as in Game 6. In simulated signatures, we indeed have

$$\boldsymbol{C}_{d_{j-1}} = \prod_{i=0}^{N-1} \boldsymbol{V}_i^{p_{i,j-1}} \cdot g^{\mathbf{L}_h \cdot \boldsymbol{\rho}_j} \qquad \forall j \in \{2, \ldots, n-1\}$$

for random $\boldsymbol{\rho}_2, \ldots, \boldsymbol{\rho}_{n-1} \in_R \mathbb{Z}_q^4$, where $p_{i,0}, \ldots, p_{i,n-1}$ are the coefficients of $\prod_{j=1}^n f_{j,i_j} = \delta_{i,\ell} x^n + \sum_{j=1}^n p_{i,j-1} x^{j-1}$. Since $\boldsymbol{V}_\ell = g^{\mathbf{L}_h \cdot (\alpha_\ell, \beta_\ell, \theta_1, \theta_2)^\top}$ and defining $\boldsymbol{\rho}_1 = -(z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_1})^\top - \sum_{j=2}^n \boldsymbol{\rho}_j x^{j-1} + (\alpha_\ell, \beta_\ell, \theta_1, \theta_2) \cdot x^n$, we have

$$\boldsymbol{C}_{d_0} = \boldsymbol{V}_\ell^{x^n} \cdot \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\sum_{j=1}^n p_{i,j-1} x^{j-1}} \cdot \prod_{i=0}^{N-1} \boldsymbol{V}_i^{-\sum_{j=2}^n p_{i,j-1} x^{j-1}}$$
$$\cdot g^{-\mathbf{L}_h \cdot (z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_1})^\top - \mathbf{L}_h \cdot \sum_{j=2}^n \boldsymbol{\rho}_j x^{j-1}} = \prod_{i=0}^{N-1} \boldsymbol{V}_i^{p_{i,0}} \cdot g^{\mathbf{L}_h \cdot \boldsymbol{\rho}_1}$$

Note that the Fiat-Shamir proof does not hide which index $\ell \in \{0,1\}^n$ the signing oracle uses (and it does not have to since $\mathcal{A}$ knows ℓ): indeed, for any signing query, the matrix $\mathbf{L}_h$ has only rank 3 and $\boldsymbol{X}_\ell$ may be the only key of the ring $\mathcal{R}$ to be in the column span of $\mathbf{M}_H$. However, the same holds in Game 6. As long as the simulation does not fail because of a collision on $\mathcal{H}_{\mathsf{FS}}$ or because (H_1, H_2) accidentally lands in the span of (g, h) at some signing query, the simulated proof is perfectly indistinguishable from a real proof that would be generated as in Game 6. Taking into account the probability that the signing oracle fails at some query, we obtain the inequality $|\Pr[W_7] - \Pr[W_6]| \leq Q_S/q + Q_S \cdot (Q_{\mathcal{H}_{\mathsf{FS}}} + Q_S)/q^2$.

In Game 7, we claim that $\Pr[W_7] = 2/q$. To prove this claim, we recall that $\mathcal{B}$ only outputs 1 if $(T_0^\star, T_1^\star)$ decrypts to $\Gamma^\star = U^{\alpha_{\ell^\star}} \cdot V^{\beta_{\ell^\star}}$. We next argue that, except with probability $1/q$, $\Gamma^\star$ is independent of $\mathcal{A}$'s view in Game 7.

Indeed, since $(g, h, \tilde{g}, \tilde{h})$ is a Diffie-Hellman tuple, the only information that $\boldsymbol{X}_{\ell^\star} = (X_{\ell^\star}, Y_{\ell^\star}) = (g^{\alpha_{\ell^\star}} \cdot h^{\beta_{\ell^\star}}, \tilde{g}^{\alpha_{\ell^\star}} \cdot \tilde{h}^{\beta_{\ell^\star}})$ reveals about $(\alpha_{\ell^\star}, \beta_{\ell^\star}) \in \mathbb{Z}_q^2$ is $\log_g(X_{\ell^\star}) = \alpha_{\ell^\star} + \log_g(h) \cdot \beta_{\ell^\star}$ since $\log_g(Y_{\ell^\star})$ only provides redundant information. Also, in all outputs of the signing oracle, the pair $(T_0, T_1) \xleftarrow{R} \mathbb{G}^2$ is chosen independently of $U^{\alpha_{\ell^\star}} \cdot V^{\alpha_{\ell^\star}}$. Finally, in Game 7, all signing queries are answered by simulating a NIZK proof without using the witnesses $SK_{\ell^\star} = (\alpha_{\ell^\star}, \beta_{\ell^\star}) \in \mathbb{Z}_q^2$ at any time. This ensures that no information is leaked about $(\alpha_{\ell^\star}, \beta_{\ell^\star})$ whatsoever.

Taking into account the event that (U, V) accidentally falls in the span of (g, h), we find that $\Gamma^\star$ remains independent of $\mathcal{A}$'s view until the forgery stage. In this case, $(T_0^\star, T_1^\star)$ only decrypts to $U^{\alpha_{\ell^\star}} \cdot V^{\beta_{\ell^\star}}$ with probability $1/q$, which implies $\Pr[W_7] = 2/q$. When counting probabilities, we obtain the bound (12). □

Lemma 1. *There exists an efficient DDH distinguisher $\mathcal{B}$ that bridges between Game 1 and Game 2 and such that $|\Pr[W_2] - \Pr[W_1]| \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda) + 1/q$.*

Proof. We consider a DDH instance (g, g^a, g^b, g^{ab+c}) for which $\mathcal{B}$ has to decide if $c = 0$ or $c \in_R \mathbb{Z}_q$. To do this, $\mathcal{B}$ initially defines $h = g^b$ and emulates the random oracle $\mathcal{H}(\cdot)$ at each (direct) query by randomly choosing $\delta_1, \delta_2 \xleftarrow{R} \mathbb{Z}_q$ and setting $(H_1, H_2) = \big((g^a)^{\delta_1} \cdot g^{\delta_2}, (g^{ab+c})^{\delta_1} \cdot (g^b)^{\delta_2}\big) = \big(g^{a\delta_1+\delta_2}, g^{(a\delta_1+\delta_2)b+c\delta_1}\big)$. If $c = 0$, (H_1, H_2) is distributed as in Game 2 for $\gamma = a\delta_1 + \delta_2$. If $c \in_R \mathbb{Z}_q$, we have $c \neq 0$ with probability $1 - 1/q$, so that (H_1, H_2) are uniform over $\mathbb{G}^2$ and

independently distributed across distinct queries, exactly as in Game 1. When $\mathcal{A}$ halts, $\mathcal{B}$ outputs 1 if $\mathcal{A}$ creates a valid forgery and 0 otherwise. □

Lemma 2. *From Game* 3 *to Game* 4*, the adversary's winning probabilities differ by at most* $|\Pr[W_4] - \Pr[W_3]| \leq (1 + Q_{\mathcal{H}_{\mathsf{FS}}})/q$.

Proof. We bound the probability $\Pr[F_4]$. Recall that F_4 occurs if $\mathcal{A}$ breaks the soundness of the proof that a dual-mode ciphertext encrypts a bit. This implies that $\sigma^\star = \big(\{\boldsymbol{\Sigma}_j^\star\}_{j=1}^n, T_0^\star, T_1^\star, z_{d,\alpha}^\star, z_{d,\beta}^\star, z_{d,\theta_1}^\star, z_{d,\theta_2}^\star\big)$ verifies and there exists $k \in [n]$ such that $\boldsymbol{\Sigma}_k^\star = (\boldsymbol{C}_{\ell_k}^\star, \boldsymbol{C}_{a_k}^\star, \boldsymbol{C}_{b_k}^\star, \boldsymbol{C}_{d_{k-1}^\star}, f_k^\star, z_{r_k}^\star, z_{s_k}^\star, \bar{z}_{r_k}^\star, \bar{z}_{s_k}^\star)$ contains a ciphertext $\boldsymbol{C}_{\ell_k}^\star$ that decrypts to $\ell_k \notin \{0,1\}$. For this index k, $\sigma^\star$ contains a NIZK proof

$$\big((\boldsymbol{C}_{a_k}^\star, \boldsymbol{C}_{b_k}^\star), x, (f_k^\star, z_{r_k}^\star, z_{s_k}^\star, \bar{z}_{r_k}^\star, \bar{z}_{s_k}^\star)\big) \tag{14}$$

that $\boldsymbol{C}_{\ell_k}^\star$ encrypts $\ell_k^\star \in \{0,1\}$. This proof, which is obtained from the Σ-protocol of [24, Figure 1], is known [24, Theorem 2] to provide special soundness with soundness error $1/q$. Hence, if the statement is false and $\boldsymbol{C}_{\ell_k}^\star$ does not encrypt a bit, for any given pair $(\boldsymbol{C}_{a_k}^\star, \boldsymbol{C}_{b_k}^\star)$, only one challenge value $x \in \mathbb{Z}_q$ admits a response $(f_k^\star, z_{r_k}^\star, z_{s_k}^\star, \bar{z}_{r_k}^\star, \bar{z}_{s_k}^\star)$ that makes (14) into an accepting transcript.

At each query $\mathcal{H}_{\mathsf{FS}}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^n)$ such that one of the $\{\boldsymbol{C}_{\ell_j}\}_{j=1}$ does not encrypt a binary value, the probability that oracle $\mathcal{H}_{\mathsf{FS}}(\cdot)$ returns the unique "bad" $x \in \mathbb{Z}_q$ for which a correct response exists is exactly $1/q$. Finally, since $\mathcal{H}_{\mathsf{FS}}$ is simulated by the challenger $\mathcal{B}$, we may assume that $\mathcal{B}$ makes the query $\mathcal{H}_{\mathsf{FS}}(M^\star, \mathcal{R}^\star, T_0^\star, T_1^\star, \{\boldsymbol{C}_{\ell_j}^\star, \boldsymbol{C}_{a_j}^\star, \boldsymbol{C}_{b_j}^\star, \boldsymbol{C}_{d_{j-1}}^\star\}_{j=1}^n)$ for itself in case it was not explicitly made by the time $\mathcal{A}$ terminates. Taking a union bound over all $\mathcal{H}_{\mathsf{FS}}$-queries, we obtain $|\Pr[W_4] - \Pr[W_3]| \leq \Pr[F_4] \leq (1 + Q_{\mathcal{H}_{\mathsf{FS}}})/q$. □

Lemma 3. *From Game* 4 *to Game* 5*, the adversary's winning probabilities differ by at most* $|\Pr[W_5] - \Pr[W_4]| \leq Q_{\mathcal{H}_{\mathsf{FS}}} \cdot \log(Q_V)/q$. (The proof is Appendix B.2.)

Acknowledgements

This work was funded in part by the French ANR ALAMBIC project (ANR-16-CE39-0006).

References

1. M. Abe, M. Ohkubo, K. Suzuki. 1-out-of-n Signatures from a Variety of Keys. In *Asiacrypt*, 2002.
2. M. Abdalla, P.-A. Fouque, V. Lyubashevsky, M. Tibouchi. Tightly-Secure Signatures from Lossy Identification Schemes. In *Eurocrypt*, 2012.
3. M. Bellare, D. Hofheinz, S. Yilek. Possibility and Impossibility Results for Encryption and Commitment Secure under Selective Opening. In *Eurocrypt*, 2009.
4. M. Bellare, P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM CCS*, 1993.
5. M. Bellare, P. Rogaway. The Exact Security of Digital Signatures - How to Sign with RSA and Rabin. In *Eurocrypt*, 1996.

6. J. Benaloh, M. de Mare. One-Way Accumulators: A Decentralized Alternative to Digital Sinatures. In *Eurocrypt*, 1993.
7. A. Bender, J. Katz, R. Morselli. Ring Signatures: Stronger Definitions, and Constructions Without Random Oracles. In *TCC*, 2006.
8. J. Bootle, A. Cerulli, P. Chaidos, E. Ghadafi, J. Groth, C. Petit. Short Accountable Ring Signatures Based on DDH. In *ESORICS*, 2015.
9. X. Boyen. Mesh Signatures. In *Eurocrypt*, 2007.
10. Z. Brakerski, Y. Tauman-Kalai. A Framework for Efficient Signatures, Ring Signatures and Identity Based Encryption in the Standard Model. Cryptology ePrint Archive: Report 2010/086, 2010.
11. E. Bresson, J. Stern, M. Szydlo. Threshold Ring Signatures and Applications to Ad-hoc Groups. In *Crypto*, 2002.
12. A. Buldas, P. Laud, H. Lipmaa. Accountable Certificate Management Using Undeniable Attestations. In *ACM-CCS*, 2000.
13. P. Camacho, A. Hevia, M. Kiwi, R. Opazo. Strong Accumulators from Collision-Resistant Hashing. In *ISC*, 2008.
14. N. Chandran, J. Groth, A. Sahai. Ring Signatures of Sub-linear Size Without Random Oracles. In *ICALP*, 2007.
15. M. Chase, A. Lysyanskaya. On Signatures of Knowledge. In *Crypto*, 2006.
16. D. Chaum, E. van Heyst. Group Signatures. In *Eurocrypt*, 1991.
17. J.-S. Coron. On the Exact Security of Full Domain Hash. In *Crypto*, 2000.
18. R. Cramer. Modular Design of Secure, yet Practical Cryptographic Protocols. PhD Thesis, University of Amsterdam, 1996.
19. R. Cramer, I. Damgård, B. Schoenmaekers. Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols. In *Crypto*, 1994.
20. Y. Dodis, A. Kiayias, A. Nicolosi, V. Shoup. Anonymous Identification in Ad Hoc Groups. In *Eurocrypt*, 2004.
21. A. Fiat, A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Crypto*, 1986.
22. E.-J. Goh, S. Jarecki. A Signature Scheme as Secure as the Diffie-Hellman Problem. In *Eurocrypt*, 2003.
23. A. González. A Ring Signature of size $O(\sqrt[3]{n})$ without Random Oracles. Cryptology ePrint Archive: Report 2017/905, 2017.
24. J. Groth, M. Kohlweiss. One-Out-of-Many Proofs: Or How to Leak a Secret and Spend a Coin. In *Eurocrypt*, 2015.
25. J. Groth, A. Sahai. Efficient Non-interactive Proof Systems for Bilinear Groups. In *Eurocrypt*, 2008.
26. S. Kakvi, E. Kiltz Optimal Security Proofs for Full Domain Hash, Revisited. In *Eurocrypt*, 2012.
27. J. Katz, N. Wang. Efficiency improvements for signature schemes with tight security reductions. In *ACM-CCS*, 2003.
28. B. Libert, S. Ling, K. Nguyen, H. Wang. Zero-Knowledge Arguments for Lattice-Based Accumulators: Logarithmic-Size Ring Signatures and Group Signatures Without Trapdoors. In *Eurocrypt*, 2016.
29. G. Malavolta, D. Schröder. Efficient Ring Signatures in the Standard Model. In *Asiacrypt*, 2017.
30. T. Okamoto. Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In *Crypto*, 1992.
31. P. Paillier, D. Vergnaud. Discrete-Log-Based Signatures May Not Be Equivalent to Discrete Log. In *Asiacrypt*, 2005.

32. C. Peikert, V. Vaikuntanathan, B. Waters. A Framework for Efficient and Composable Oblivious Transfer. In *Crypto*, 2008.
33. D. Pointcheval, J. Stern. Security Proofs for Signature Schemes. *Eurocrypt*'96.
34. R. Rivest, A. Shamir. Y. Tauman. How To Leak a Secret. In *Asiacrypt*, 2001.
35. T. Sander. Efficient Accumulators Without Trapdoor. *ICICS*, 1999.
36. H. Shacham, B. Waters. Efficient Ring Signatures Without Random Oracles. In *PKC*, 2007.

A Reminders on Commitment Schemes

A non-interactive commitment scheme allows a sender to commit to a message m by sending a commitment string to the receiver. Later on the sender can convince the receiver that the committed value was really m. A commitment scheme must satisfy two security properties called *hiding* and *binding*. The former captures that the commitment hides any partial information about the message. The latter requires that the sender be unable to open the commitment to two distinct messages. Formally, a non-interactive commitment scheme is a pair of PPT algorithms $(\mathsf{Setup}, \mathsf{Com})$. The setup algorithm $ck \leftarrow \mathsf{Setup}(1^\lambda)$ generates a commitment key ck, which specifies a message space $\mathcal{M}_{ck}$, a randomness space $\mathcal{R}_{ck}$ and a commitment space $\mathcal{C}_{ck}$. The commitment algorithm Com defines a function $\mathsf{Com}_{ck} : \mathcal{M}_{ck} \times \mathcal{R}_{ck} \to \mathcal{C}_{ck}$. On input of $m \in \mathcal{M}_{ck}$, the sender randomly chooses $r \xleftarrow{R} \mathcal{R}_{ck}$ and computes a commitment string $c = \mathsf{Com}_{ck}(m, r) \in \mathcal{C}_{ck}$.

A commitment is *perfectly hiding* if, for any $m \in \mathcal{M}_{ck}$, the distribution $\{\mathsf{Com}_{ck}(m, r) \mid r \xleftarrow{R} \mathcal{R}_{ck}\}$ is statistically independent of m. It is perfectly binding if any element of the commitment space $\mathcal{C}_{ck}$ uniquely determines the message. Groth and Kohlweiss [24] use the following additional properties.

Definition 6. *A commitment scheme* $(\mathsf{Setup}, \mathsf{Com})$ *is* **strongly binding** *if, for any PPT adversary* $\mathcal{A}$*, there exists a negligible function* $\varepsilon(\lambda)$ *such that*

$$\begin{aligned}|\Pr[&ck \leftarrow \mathsf{Setup}(1^\lambda); (c, m_1, r_1, m_2, r_2) \leftarrow \mathcal{A}(PK) : \\ &\mathsf{Com}_{ck}(m_1; r_1) = c \ \wedge \ \mathsf{Com}_{ck}(m_2; r_2) = c \ \wedge \ (m_1, r_1) \neq (m_2, r_2)]| < \varepsilon(\lambda).\end{aligned}$$

We consider a prime $q > 2^\lambda$ specified in the commitment key ck. The message space and the randomness space are both $\mathcal{M}_{ck} = \mathcal{R}_{ck} = \mathbb{Z}_q$.

Definition 7. *A commitment scheme* $(\mathsf{Setup}, \mathsf{Com})$ *is* **additively homomorphic** *if for all messages* $m_1, m_2 \in \mathcal{M}_{ck}$ *and all random coins* $r_1, r_2 \in \mathcal{R}_{ck}$*, we have* $\mathsf{Com}_{ck}(m_1; r_1) \cdot \mathsf{Com}_{ck}(m_2; r_2) = \mathsf{Com}_{ck}(m_1 + m_2; r_1 + r_2)$.

B Deferred Proofs for the Fully Tight Construction

B.1 Proof of Theorem 4

Proof. We consider a sequence of games and, for each i, we call W_i the event that the challenger outputs 1 in Game i, meaning that the adversary successfully guesses the challenger's bit and outputs $b' = b$. In each game, we also consider the event E_i by which the tuple $(g, h, \tilde{g}, \tilde{h})$ of the public parameter or the tuple (g, h, H_1, H_2) defined in the challenge signature forms a Diffie-Hellman tuple.

Game 0**:** This is the real game where the challenger outputs 1 if and only if $\mathcal{A}$ wins. By definition, $\mathcal{A}$'s advantage is $\mathbf{Adv}^{\text{anon}}_{\mathcal{A}}(\lambda) = |\Pr[W_0] - 1/2|$. We assume that, for all public keys generated by the $\mathsf{Keygen}(.)$ oracle, the adversary immediately obtains the secret keys. Since $(\tilde{g}, \tilde{h})$ is uniformly distributed in pp and since (H_1, H_2) is an independent random output of the random oracle $\mathcal{H}$, we find $\Pr[E_0] = 2/q - 1/q^2$ and then $\Pr[W_0] \leq \Pr[W_0|\neg E_0] + (2/q - 1/q^2)$. We are left with bounding $\Pr[W_0|\neg E_0]$.

Game 1**:** We modify the generation of the challenge signature. On a challenge query $(M, \mathcal{R}, \ell^{(0)}, \ell^{(1)})$, where $(0 \leq \ell^{(0)}, \ell^{(1)} \leq |\mathcal{R}|-1)$, the challenger $\mathcal{B}$ parses $\mathcal{R}$ as $\{\boldsymbol{X}_0, \ldots, \boldsymbol{X}_{N-1}\}$ and returns $\perp$ if $\boldsymbol{X}_{\ell^{(0)}}$ and $\boldsymbol{X}_{\ell^{(1)}}$ are not public keys produced by the $\mathsf{Keygen}(.)$ oracle. Otherwise, it flips a coin $b \xleftarrow{R} \{0,1\}$ and sets $(\ell_1, \ldots, \ell_n)$ as the bit representation of $\ell^{(b)}$. Then, it chooses $x \xleftarrow{R} \mathbb{Z}_q$ as well as $z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2} \xleftarrow{R} \mathbb{Z}_q$ and $f_j, z_{r_j}, z_{s_j}, \bar{z}_{r_j}, \bar{z}_{s_j} \xleftarrow{R} \mathbb{Z}_q$ for all $j \in [n]$. Then, it picks $T_0 \xleftarrow{R} \mathbb{G}$ as well $C_{\ell_j,0} \xleftarrow{R} \mathbb{Z}_q$ for all $j \in [n]$. It can now compute

$$C_{a_j,0} = g^{z_{r_j}} \cdot h^{z_{s_j}} \cdot C_{\ell_j,0}^{-x}, \quad C_{b_j,0} = g^{\bar{z}_{r_j}} \cdot h^{\bar{z}_{s_j}} \cdot C_{\ell_j,0}^{f_j - x} \qquad \forall j \in [n],$$

so as to define $(H_1, H_2) = \mathcal{H}(M, \mathcal{R}, T_0, \{(C_{\ell_j,0}, C_{a_j,0}, C_{b_j,0})\}_{j=1}^n)$. Then, $\mathcal{B}$ completes the computation of the dual-mode commitments as follows. First, it picks $T_1 \xleftarrow{R} \mathbb{G}$ as well as $C_{\ell_j,1} \xleftarrow{R} \mathbb{G}$ for all $j \in [n]$. Then, it computes

$$C_{a_j,1} = \big(g^{f_j} \cdot H_1^{z_{r_j}} \cdot H_2^{z_{s_j}}\big) \cdot C_{\ell_j,1}^{-x}, \qquad C_{b_j,1} = \big(H_1^{\bar{z}_{r_j}} \cdot H_2^{\bar{z}_{s_j}}\big) \cdot C_{\ell_j,1}^{f_j - x}.$$

It draws $\boldsymbol{C}_{d_{j-1}} \xleftarrow{R} \mathbb{G}$ for each $j \in \{2, \ldots, n\}$ while, for $j = 1$, it computes

$$\boldsymbol{C}_{d_0} = \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^n f_{j,i_j}} \prod_{j=2}^{n} \tilde{\boldsymbol{C}}_{d_{j-1}}^{-(x^{j-1})} \Big(\mathbf{M}_{\mathcal{H}} \odot (-z_{d,\alpha}, -z_{d,\beta}, -z_{d,\theta_1}, -z_{d,\theta_2})^\top\Big),$$

where $\boldsymbol{V}_i = (X_i, Y_i, T_0, T_1)^\top$, $f_{j,1} = f_j$ and $f_{j,0} = x - f_j$ for each $j \in [n]$. Finally, the challenger programs the random oracle $\mathcal{H}_{\mathsf{FS}}$ to have the equality $x = \mathcal{H}_{\mathsf{FS}}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^n)$. If $\mathcal{H}_{\mathsf{FS}}$ was already defined for this input, the challenger aborts and picks b' as a random bit. If the simulation does not fail, the oracle outputs the challenge signature $\sigma = \big(\{\boldsymbol{\Sigma}_j\}_{j=1}^n, T_0, T_1, z_{d,\alpha}, z_{d,\beta}, z_{d,\theta_1}, z_{d,\theta_2}\big)$, which is distributed exactly as in $W_0|\neg E_0$, assuming that E_1 does not occur. Indeed, if (g, h, H_1, H_2) is not a Diffie-Hellman tuple in both games, all the dual-mode commitments are perfectly hiding and if $(g, h, \tilde{g}, \tilde{h})$ is not a Diffie-Hellman tuple as well, the matrix $\mathbf{M}_H$ has full rank, meaning that $\{\boldsymbol{C}_{d_{j-1}}\}_{j=1}^n$ are uniformly distributed over $\mathbb{G}^4$. Therefore, as long as no collision occurs in the simulation of the challenge, $\mathcal{A}$'s view in $W_1|\neg E_1$ is the same as in $W_0|\neg E_0$. If we call F_1 the event that a hash collision prevents the correct generation of the challenge signature, we obtain the inequality $|\Pr[W_1|\neg(E_1 \cup F_1)] - \Pr[W_0|\neg E_0]| \leq \Pr[F_1] \leq Q_{\mathcal{H}_{\mathsf{FS}}}/q^2$.

In Game 1, when neither E_1 nor F_1 occurs, the signature is perfectly independent of $b \in_R \{0,1\}$, so that $\Pr[W_1|\neg(E_1 \cup F_1)] = 1/2$. All the above observations together thus implies $\mathbf{Adv}^{\text{anon}}_{\mathcal{A}}(\lambda) \leq 2/q + (Q_{\mathcal{H}_{\mathsf{FS}}} - 1)/q^2$. □

B.2 Proof of Lemma 3

Proof. The only situation where Game 5 differs from Game 4 is the event F_5 that extracting $\{\boldsymbol{C}^\star_{\ell_j}\}_{j=1}^n$ leads to a string $\ell^\star \in \{0,1\}^n$ but $(T_0^\star, T_1^\star)$ does not decrypt to an encoding $U^{\alpha_{\ell^\star}} \cdot V^{\beta_{\ell^\star}}$ of the $\ell^\star$-th ring member's secret key. This implies that $\boldsymbol{V}_{\ell^\star} = (X_{\ell^\star}, Y_{\ell^\star}, T_0^\star, T_1^\star)$ is not in the column space of $\mathbf{M}_H$ (as defined in (3)) and we show that this event can only happen with probability $Q_{\mathcal{H}_{\mathsf{FS}}} \cdot n/q \leq Q_{\mathcal{H}_{\mathsf{FS}}} \cdot \log(Q_V)/q$, where $n = \log N^\star$.

Note that (10) implies that $f_j^\star$ equals $f_j^\star = a_j^\star + \ell_j^\star \cdot x^\star$ for all $j \in [n]$, where $a_j^\star \in \mathbb{Z}_q$ is encrypted by $\boldsymbol{C}^\star_{a_j}$. Defining $f^\star_{j,1} = f_j^\star$ and $f^\star_{j,0} = x - f_j^\star$, we know that

$$\prod_{j=1}^{n} f^\star_{j,i_j} = \delta_{i,\ell^\star} \cdot x^{\star n} + \sum_{k=0}^{n-1} p_{i,k} \cdot x^{\star k} \qquad \forall i \in [N^\star],$$

for some $p^\star_{i,0}, \ldots, p^\star_{i,n-1} \in \mathbb{Z}_q$. This implies

$$\begin{aligned}\prod_{i=0}^{N-1} \boldsymbol{V}_i^{\prod_{j=1}^n f^\star_{j,i_j}} &= \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\delta_{i,\ell^\star}\cdot x^n + \sum_{k=0}^{n-1} p^\star_{i,k}\cdot x^k} \\ &= \boldsymbol{V}_{\ell^\star}^{x^n} \cdot \prod_{i=0}^{N-1} \boldsymbol{V}_i^{\sum_{k=0}^{n-1} p^\star_{i,k}\cdot x^k} = \boldsymbol{V}_{\ell^\star}^{x^n} \cdot \prod_{k=0}^{n-1} \cdot \Big(\prod_{i=0}^{N-1} \boldsymbol{V}_i^{p^\star_{i,k}}\Big)^{x^k}.\end{aligned}$$

Moreover, the last verification equation (11) implies

$$\boldsymbol{V}_{\ell^\star}^{x^n} \cdot \prod_{k=0}^{n-1} \cdot \Big(\prod_{i=0}^{N-1} \boldsymbol{V}_i^{p^\star_{i,k}}\Big)^{x^k} \cdot \prod_{k=0}^{n-1} \boldsymbol{C}_{d_k}^{-(x^k)} = g^{\mathbf{L}_h \cdot (z^\star_{d,\alpha}, z^\star_{d,\beta}, z^\star_{d,\theta_1}, z^\star_{d,\theta_2})^\top}. \tag{15}$$

By taking the discrete logarithms $\log_g(\cdot)$ of both members of (15), we get

$$x^n \cdot \boldsymbol{v}_{\ell^\star} + \sum_{i=0}^{N-1} \sum_{k=0}^{n-1} (p^\star_{i,k}\, x^k) \cdot \boldsymbol{v}_i - \sum_{k=0}^{n-1} x^k \cdot \boldsymbol{c}_{d_k} = \mathbf{L}_h \cdot (z^\star_{d,\alpha}, z^\star_{d,\beta}, z^\star_{d,\theta_1}, z^\star_{d,\theta_2})^\top. \tag{16}$$

Since $\mathbf{L}_h$ has rank at most 3 due to the modification introduced in Game 2 and Game 3, assuming that $\boldsymbol{v}_{\ell^\star} = \log_g(\boldsymbol{V}_{\ell^\star}) \in \mathbb{Z}_q^4$ is not in the column space of $\mathbf{L}_h$, there exists a non-zero vector $\boldsymbol{t} \in \mathbb{Z}_q^4$ such that $\boldsymbol{t}^\top \cdot \mathbf{L}_h = \mathbf{0}^{1\times 4}$ and $\boldsymbol{t}^\top \cdot \boldsymbol{v}_{\ell^\star} \neq 0$. If we multiply both members of (16) on the left by $\boldsymbol{t}^\top$, we obtain

$$x^n \cdot (\boldsymbol{t}^\top \cdot \boldsymbol{v}_{\ell^\star}) + \sum_{i=0}^{N-1} \sum_{k=0}^{n-1} (p^\star_{i,k} \cdot x^k) \cdot (\boldsymbol{t}^\top \cdot \boldsymbol{v}_i) - \sum_{k=0}^{n-1} x^k \cdot (\boldsymbol{t}^\top \cdot \boldsymbol{c}_{d_k}) = 0. \tag{17}$$

If $\boldsymbol{t}^\top \cdot \boldsymbol{v}_{\ell^\star} \neq 0$, equality (17) implies that x is a root of a non-zero polynomial of degree n. However, x is uniformly distributed over $\mathbb{Z}_q$ and the Schwartz-Zippel Lemma implies that (17) can only hold with probability $n/q < \log(Q_V)/q$.

In order to bound the probability $\Pr[F_5]$, we have to consider all hash queries $\mathcal{H}_{FS}(M, \mathcal{R}, T_0, T_1, \{\boldsymbol{C}_{\ell_j}, \boldsymbol{C}_{a_j}, \boldsymbol{C}_{b_j}, \boldsymbol{C}_{d_{j-1}}\}_{j=1}^n)$ for which $\mathcal{R}$ only contains honestly generated keys and (T_0, T_1) does not decrypt to an encoding $U^{\alpha_\ell} \cdot V^{\beta_\ell}$ of the ℓ-th key of $\mathcal{R}$, where $\ell \in \{0, \ldots, |\mathcal{R}| - 1\}$ is determined by $\{\boldsymbol{C}_{\ell_j}\}_{j=1}^n$. Taking a union bound over all hash queries, we obtain $\Pr[F_5] \leq Q_{\mathcal{H}_{\mathsf{FS}}} \cdot \log(Q_V)/q$. □