Tighter Security for Schnorr Identification and Signatures: A High-Moment Forking Lemma for Σ-Protocols

Lior Rotem*† Gil Segev*

Abstract

The Schnorr identification and signature schemes have been amongst the most influential cryptographic protocols of the past three decades. Unfortunately, although the best-known attacks on these two schemes are via discrete-logarithm computation, the known approaches for basing their security on the hardness of the discrete logarithm problem encounter the "square-root barrier". In particular, in any group of order p where Shoup's generic hardness result for the discrete logarithm problem is believed to hold (and is thus used for setting concrete security parameters), the best-known t-time attacks on the Schnorr identification and signature schemes have success probability t^2/p, whereas existing proofs of security only rule out attacks with success probabilities $(t^2/p)^{1/2}$ and $(q_{\mathsf{H}} \cdot t^2/p)^{1/2}$, respectively, where q_{H} denotes the number of random-oracle queries issued by the attacker.

We establish tighter security guarantees for identification and signature schemes which result from Σ-protocols with special soundness based on the hardness of their underlying relation, and in particular for Schnorr's schemes based on the hardness of the discrete logarithm problem. We circumvent the square-root barrier by introducing a high-moment generalization of the classic forking lemma, relying on the assumption that the underlying relation is "d-moment hard": The success probability of any algorithm in the task of producing a witness for a random instance is dominated by the d-th moment of the algorithm's running time.

In the concrete context of the discrete logarithm problem, already Shoup's original proof shows that the discrete logarithm problem is 2-moment hard in the generic-group model, and thus our assumption can be viewed as a highly-plausible strengthening of the discrete logarithm assumption in any group where no better-than-generic algorithms are currently known. Applying our high-moment forking lemma in this context shows that, assuming the 2-moment hardness of the discrete logarithm problem, any t-time attacker breaks the security of the Schnorr identification and signature schemes with probabilities at most $(t^2/p)^{2/3}$ and $(q_{\mathsf{H}} \cdot t^2/p)^{2/3}$, respectively.

*School of Computer Science and Engineering, Hebrew University of Jerusalem, Jerusalem 91904, Israel. Email: {lior.rotem,segev}@cs.huji.ac.il. Supported by the European Union's Horizon 2020 Framework Program (H2020) via an ERC Grant (Grant No. 714253).

†Supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

Contents

1 Introduction

The Schnorr identification and signature schemes [Sch89, Sch91] have been amongst the most influential cryptographic protocols of the past three decades, due to their conceptual simplicity and practical efficiency. Accordingly, the analysis of their security guarantees has attracted much attention over the years. Though from the onset, it was observed that their asymptotic security can be tied to that of the discrete logarithm problem, characterizing their concrete security has remained an elusive feat. On the one hand, to this day there are no known attacks on these schemes that improve upon the existing algorithms for computing discrete logarithms. On the other hand, essentially all known security reductions to the discrete logarithm problem are non-tight, which may lead to significant blowups when setting concrete security parameters (i.e., the group size), and hence to degraded efficiency.[1] Concretely, the known approaches for basing the security of the Schnorr identification and signature schemes on the hardness of the discrete logarithm problem encounter the "square-root barrier".

The square-root barrier. In order to base the security of the Schnorr identification scheme and signature scheme on the hardness of the discrete logarithm problem, one has to transform any malicious impersonator and any malicious forger, respectively, into a discrete-logarithm algorithm. The existing approaches are based on the classic "forking lemma" of Pointcheval and Stern [PS00] (see also [AAB+02, BN06, BCC+16, KMP16] and the references therein). The difference between the various approaches is reflected by the different trade-offs between the success probability and the running time of their discrete-logarithm algorithms.

For the Schnorr identification scheme, any malicious impersonator that runs in time t and breaks the security of the scheme with probability ϵ, can be transformed for example into a discrete-logarithm algorithm that has success probability roughly ϵ^2 and runs in time roughly t. Similarly, for the Schnorr signature scheme, any malicious forger that runs in time t, issues q_{H} random-oracle queries and breaks the security of the scheme with probability ϵ, can be transformed into a discrete-logarithm algorithm that has success probability roughly $\epsilon^2/q_{\mathsf{H}}$ and runs in time roughly t.

Thus, in any group of order p where Shoup's generic hardness result for computing discrete logarithms is believed to hold [Sho97], this leads to the bound $\epsilon \leq (t^2/p)^{1/2}$ on the security of the Schnorr identification scheme, and to the bound $\epsilon \leq (q_{\mathsf{H}} \cdot t^2/p)^{1/2}$ on the security of the Schnorr signature scheme (we refer the reader to Section 3 for a variety of other trade-offs that were established over the years, all of which lead to the same square-root bounds, as recently observed by Bellare and Dai [BD20] and by Jaeger and Tessaro [JT20]).

However, the best-known attack on the security the Schnorr identification and signature schemes is via discrete-logarithm computation, which has success probability t^2/p in such groups. For example, for a 256-bit prime p, the success probability of the best-known 2^{80}-time attack on the Schnorr identification scheme is roughly 2^{-96}, whereas the square-root bound only rules out attacks with success probability greater than 2^{-48} (for the Schnorr signature scheme this gap only increases due to the additional dependency on q_{H}).

A wider perspective: Identification and signatures from Σ-protocols. The square-root barrier is encountered not only when proving the security of the Schnorr identification and signatures schemes, but also when proving the security of additional ones, such as the Okamoto identification and signature schemes [Oka92] (see [AAB+02, KMP16] for various other examples). The Schnorr and

[1] These exclude reductions in the generic-group model [Sho97] and algebraic-group model [FKL18], as discussed below.

Okamoto schemes are prime examples of the more general approach of constructing identification schemes based on Σ-protocols with special soundness, and of constructing signature schemes based on such identification schemes via the Fiat-Shamir paradigm [FS86, AAB+02]. In such schemes, the square-root barrier arises due to the rewinding-based methodology underlying their security proofs, as we further discuss in Section 3.

It should be noted that additional approaches were suggested as alternatives to basing the security of the Schnorr identification and signature schemes on the hardness of the discrete logarithm problem. Shoup [Sho97] and Fuchsbauer, Plouviez and Seurin [FPS20] provided tight security proofs in the generic-group model and in the algebraic-group model, respectively, and Bellare and Dai [BD20] provided tight security proofs based on the hardness of their multi-base discrete logarithm problem. These approaches do not encounter the square-root barrier, at the cost of considering either idealized models that considerably restrict attackers, or a newly-introduced interactive problem instead of the long-studied discrete logarithm problem.

1.1 Our Contributions

We establish tighter security guarantees for identification and signature schemes by circumventing the square-root barrier. Our approach applies to schemes that result from Σ-protocols with special soundness based on the hardness of their underlying relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$, and in particular to the Schnorr and Okamoto identification and signature schemes based on the hardness of the discrete logarithm problem.

We prove our results by introducing a high-moment generalization of the classic forking lemma, relying on the assumption that the success probability of any algorithm in the task of producing a witness $w \in \mathcal{W}$ given a random instance $x \in \mathcal{X}$ is dominated by the d-th moment of the algorithm's running time. In what follows we provide a high-level description of our assumption, and then state our bounds on the security of identification and signature schemes.

Our assumption: *d*-moment hardness. Given a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$ underlying a Σ-protocol, and a distribution $\mathcal{D}$ over pairs $(x, w) \in \mathcal{R}$, we put forward the *d-moment assumption* that considers the task of producing a witness w given an instance x that is sampled via $\mathcal{D}$. Informally, in its most simplistic form, our assumption asks that the success probability of any algorithm A in this task is at most $\mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}})^d\right]/|\mathcal{W}|$, where $\mathsf{T}_{A,\mathcal{D}}$ denotes the random variable corresponding to A's running time.[2] We refer the reader to Section 3 for a formal statement.

In the specific context of the discrete logarithm problem, instances are of the form $x = (\mathbb{G}, p, g, h)$ where $\mathbb{G}$ is a cyclic group of order p that is generated by g, and h is a group element. The relation $\mathcal{R}$ consists of all pairs $((\mathbb{G}, p, g, h), w)$ for which $h = g^w$, and the distribution $\mathcal{D}$ consists of a group-generation algorithm that produces the description $(\mathbb{G}, p, g)$ of the group, together with a uniformly-distributed group element h.

As recently observed by Jaeger and Tessaro [JT20], already Shoup's original proof shows that the discrete logarithm problem is 2-moment hard in the generic-group model [Sho97].[3] Thus, our assumption can be viewed a highly-plausible strengthening of the discrete logarithm assumption in any group where no better-than-generic algorithms are currently known for the discrete logarithm

[2] More generally, our assumption asks that the latter probability is at most $\Delta \cdot \mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}})^d\right]/|\mathcal{W}|^\omega$ for functions Δ and ω of the security parameter. Looking ahead, the Schnorr identification and signature schemes will correspond to $\Delta = \omega = 1$, whereas the Okamoto identification and signature scheme will correspond to $\Delta = 1$ and $\omega = 1/2$.

[3] In fact, Shoup proved the following stronger statement: For any $t \geq 0$, the success probability of any algorithm in computing the discrete logarithm of a uniformly-distributed group element, conditioned on running in time at most t, is at most t^2/p. This implies, in particular, 2-moment hardness (with $\Delta = \omega = 1$).

problem. In such groups, the generic hardness of the problem is used for setting concrete security parameters, and thus the assumption that the discrete logarithm problem is 2-moment hard can be viewed as identifying some of the core essence of the problem's generic hardness in the form of a standard-model assumption.

Tighter security for identification schemes. Given an identification scheme resulting from a Σ-protocol for a relation $\mathcal{R}$, we follow the approach underlying the classic "forking lemma" of Pointcheval and Stern [PS00], and show that any attacker can be transformed into an algorithm A that takes as input an instance $x \in \mathcal{X}$ and produces (with a certain probability) a witness $w \in \mathcal{W}$ such that $(x, w) \in \mathcal{R}$. However, unlike existing variants of the forking lemma (see, for example, [AAB+02, BN06, KMP16, BCC+16, JT20]), we design our algorithm A with the goal of optimizing the trade-off between its success probability and the dth moment of its running time. Assuming the d-moment hardness of the relation $\mathcal{R}$, this trade-off leads to the following tighter bound on the success probability of the attacker when considering the standard notion of security against passive impersonation attacks (in Section 3 we demonstrate that the existing variants of the forking lemma do not circumvent the square-root barrier when relying on our assumption):

Theorem 1.1 (informal)**.** *Let $\mathcal{ID}$ be an identification scheme with special soundness for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$. If $\mathcal{R}$ is d-moment hard, then any attacker that runs in time t breaks the security of $\mathcal{ID}$ with probability at most $(t^d/|\mathcal{W}|)^{d/(2d-1)}$.*

In particular, our theorem yields the following corollary for the Schnorr and Okamoto identification schemes (Table 1 exemplifies our concrete improvement over the square-root bound for a few typical choices of parameters):

Corollary 1.2 (informal)**.** *Assuming that the discrete logarithm problem is 2-moment hard, then any attacker that runs in time t breaks the security of the Schnorr and Okamoto identification schemes with probability at most $(t^2/p)^{2/3}$, where p is the order of the underlying group.*

Attacker's running time t	**Security parameter λ**	**Square-root bound $(t^2/p)^{1/2}$**	**Our bound $(t^2/p)^{2/3}$**
2^{64}	256	2^{-64}	$2^{-85.34}$
2^{80}	256	2^{-48}	2^{-64}
2^{100}	512	2^{-156}	2^{-208}

Table 1: A comparison of the security guarantees for the Schnorr and Okamoto identification schemes provided by the square-root bound and by our bound.

Tighter security for signature schemes. We show that our approach extends to establishing tighter security guarantees for signature schemes that are obtained from identification schemes via the Fiat-Shamir paradigm [FS86]. The generic analysis of the Fiat-Shamir transform in this context [AAB+02], when combined with Theorem 1.1, yields the bound $\epsilon \leq q_{\mathsf{H}} \cdot (t^d/|\mathcal{W}|)^{d/(2d-1)}$ on the success probability of any malicious forger that runs in time t and issues q_{H} random-oracle queries assuming the d-moment hardness of the underlying relation. Although this bound may already be useful on its own, we nevertheless show that it can be further improved by applying our proof technique directly for reducing the dependence on q_{H}:

Theorem 1.3 (informal)**.** *Let $\mathcal{ID}$ be an identification protocol with special soundness for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$, and let $\mathcal{SIG}_{\mathcal{ID},\mathsf{H}}$ be its corresponding signature schemes obtained via the Fiat-Shamir transform using the hash function H. If $\mathcal{R}$ is d-moment hard and H is modeled as a random oracle, then any attacker that runs in time t and issues q_{H} random-oracle queries breaks the security of $\mathcal{SIG}_{\mathcal{ID},\mathsf{H}}$ with probability at most $(q_{\mathsf{H}} \cdot t^d/|\mathcal{W}|)^{d/(2d-1)}$.*

As above, our theorem yields the following corollary for the Schnorr and Okamoto signature schemes (Table 2 exemplifies our concrete improvement over the square-root bound for a few typical choices of parameters):

Corollary 1.4 (informal)**.** *Assuming that the discrete logarithm problem is 2-moment hard, then any attacker that runs in time t and issues q_{H} random-oracle queries breaks the security of the Schnorr and Okamoto signature schemes with probability at most $(q_{\mathsf{H}} \cdot t^2/p)^{2/3}$, where p is the order of the underlying group.*

Attacker's running time t	**Attacker's oracle queries q_{H}**	**Security parameter λ**	**Square-root bound $(q_{\mathsf{H}} \cdot t^2/p)^{1/2}$**	**Our bound $(q_{\mathsf{H}} \cdot t^2/p)^{2/3}$**
2^{64}	2^{50}	256	2^{-39}	2^{-52}
2^{80}	2^{60}	256	2^{-18}	2^{-24}
2^{80}	2^{60}	512	2^{-146}	$2^{-194.67}$
2^{100}	2^{80}	512	2^{-116}	$2^{-142.67}$

Table 2: A comparison of the security guarantees for the Schnorr Okamoto signature schemes provided by the square-root bound and by our bound.

1.2 Paper Organization

The remainder of this paper is organized as follows. First, in Section 2 we present the basic notation and standard cryptographic primitives that are used throughout the paper. In Section 3 we formally define our d-moment assumption, and demonstrate that the existing variants of the forking lemma do not circumvent the square-root barrier when relying on our assumption. In Sections 4 and 5 we present and prove our bounds on the security of identification and signature schemes, respectively, from which in Section 6 we derive concrete security bounds for the Schnorr and Okamoto identification and signature schemes.

2 Preliminaries

In this section we present the basic notions and standard cryptographic primitives that are used in this work. For an integer $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, \ldots, n\}$. For a distribution X we denote by $x \leftarrow X$ the process of sampling a value x from the distribution X. Similarly, for a set $\mathcal{X}$ we denote by $x \leftarrow \mathcal{X}$ the process of sampling a value x from the uniform distribution over $\mathcal{X}$.

Σ-protocols. Let $\mathcal{R} = \{\mathcal{R}_\lambda\}_{\lambda \in \mathbb{N}}$ be a relation, where $\mathcal{R}_\lambda \subseteq \mathcal{X}_\lambda \times \mathcal{W}_\lambda$ for any $\lambda \in \mathbb{N}$, for sets $\mathcal{X} = \{\mathcal{X}_\lambda\}_{\lambda \in \mathbb{N}}$ and $\mathcal{W} = \{\mathcal{W}_\lambda\}_{\lambda \in \mathbb{N}}$. A Σ-protocol Π for the relation $\mathcal{R}$ is a 4-tuple $(\mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$, where P_1 is a probabilistic polynomial-time algorithm, P_2 and V are deterministic polynomial-time algorithms, and $\mathcal{C} = \{\mathcal{C}_x\}_{x \in \mathcal{X}}$ is an ensemble of efficiently sampleable sets. The protocol π is defined as follows:

1. The algorithm P_1 on input (x, w), where $x \in \mathcal{X}_\lambda$ and $w \in \mathcal{W}_\lambda$, produces a message α and a state st.
2. A challenge β is sampled uniformly at random from the challenge set $\mathcal{C}_x$.
3. The algorithm P_2 on input (st, β) produces a message γ.
4. The algorithm V on input $(x, \alpha, \beta, \gamma)$ determines the output of the protocol by outputting either 0 or 1.

In terms of completeness, we ask that for every $\lambda \in \mathbb{N}$ and for every $(x, w) \in \mathcal{R}_\lambda$, it holds that $\mathsf{V}(x, \alpha, \beta, \mathsf{P}_2(\mathsf{st}, \beta)) = 1$ with an overwhelming probability over the choice of $(\alpha, \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$ and $\beta \leftarrow \mathcal{C}_x$. In terms of soundness, we consider the following standard special soundness property for Σ-protocols. Roughly, the property requires that given an instance $x \in \mathcal{X}$ and two accepting transcripts for x which share the same first message α but differ on their second message β, one can efficiently compute a witness $w \in \mathcal{W}$ such that $(x, w) \in \mathcal{R}$.

Definition 2.1. Let $\Pi = (\mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ be a Σ-protocol for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$, and let $t = t(\lambda)$ be a function of the security parameter $\lambda \in \mathbb{N}$. Then, Π has *t-time special soundness* if there exists a deterministic t-time algorithm $\mathsf{WitnessExt}$ for which the following holds: For every $\lambda \in \mathbb{N}$, for every instance $x \in \mathcal{X}_\lambda$, and for every $(\alpha, (\beta, \gamma), (\beta', \gamma'))$ such that $\mathsf{V}(x, \alpha, \beta, \gamma) = \mathsf{V}(x, \alpha, \beta', \gamma') = 1$ and $\beta \neq \beta'$ it holds that $(x, \mathsf{WitnessExt}(x, \alpha, (\beta, \gamma), (\beta', \gamma')) \in \mathcal{R}$.

Identification schemes. An identification scheme consists of a Σ-protocol for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$ and of an algorithm Gen that produces a distribution over instances $x \in \mathcal{X}$ together with a corresponding witness $w \in \mathcal{W}$ such that $(x, w) \in \mathcal{R}$. We say that an identification protocol has t-time special soundness if its underlying Σ-protocol has t-time special soundness.

Additionally, we consider the standard notion of security against passive impersonation attacks, asking that a malicious prover on input an instance x produced by Gen should not be able to convince the verifier to accept even when given access to an oracle that produces honestly-generated transcripts for the instance x. In what follows, given an identification protocol, we let $\mathsf{Trans}_{x,w}$ denote an oracle that (when queried without any input) runs an honest execution of the protocol on input (x, w) and returns the resulting transcript (α, β, γ).

Definition 2.2. Let $t = t(\lambda)$ and $\epsilon = \epsilon(\lambda)$ be function of the security parameter $\lambda \in \mathbb{N}$. An identification scheme $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ is (t, ϵ)-*secure against passive impersonation attacks* if for any t-time probabilistic prover $\bar{\mathsf{P}} = (\bar{\mathsf{P}}_1, \bar{\mathsf{P}}_2)$ it holds that

$$\mathbf{Adv}^{\mathsf{PA\text{-}IMP}}_{\mathcal{ID}, \bar{\mathsf{P}}}(\lambda) \stackrel{\mathsf{def}}{=} \Pr\left[\mathsf{PA\text{-}IMP}_{\mathcal{ID}, \bar{\mathsf{P}}}(\lambda) = 1\right] \leq \epsilon(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the experiment $\mathsf{PA\text{-}IMP}_{\mathcal{ID}, \bar{\mathsf{P}}}(\lambda)$ is defined as follows:

1. $(x, w) \leftarrow \mathsf{Gen}(1^\lambda)$.
2. $(\alpha, \mathsf{st}) \leftarrow \bar{\mathsf{P}}_1^{\mathsf{Trans}_{x,w}}(1^\lambda, x)$.
3. $\gamma \leftarrow \bar{\mathsf{P}}_2^{\mathsf{Trans}_{x,w}}(\mathsf{st}, \beta)$ for $\beta \leftarrow \mathcal{C}_x$.
4. If $\mathsf{V}(x, \alpha, \beta, \gamma) = 1$ then output 1 and otherwise output 0.

In this work we consider identification schemes that are *simulatable*: There exists an efficient algorithm that on input $x \in \mathcal{X}$, for $(x, w) \leftarrow \mathsf{Gen}(1^\lambda)$, samples a transcript (α, β, γ) from the distribution of honest executions of the protocol on input (x, w).

Definition 2.3. Let $t = t(\lambda)$ be function of the security parameter $\lambda \in \mathbb{N}$. An identification scheme $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ is *t-time simulatable* if there exists a t-time algorithm Sim such that the distributions $\{(x, (\alpha, \beta, \gamma))\}_{\lambda\in\mathbb{N}}$ and $\{(x, \mathsf{Sim}(1^\lambda, x))\}_{\lambda\in\mathbb{N}}$ are identical, where $(x, w) \leftarrow \mathsf{Gen}(1^\lambda)$, $(\alpha, \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$, $\beta \leftarrow \mathcal{C}_x$ and $\gamma \leftarrow \mathsf{P}_2(\mathsf{st}, \beta)$.

Note that for any simulatable identification scheme $\mathcal{ID}$ we can thus assume that malicious provers do not query the transcript-generation oracle $\mathsf{Trans}_{x,w}$ as such queries can be internally simulated given the instance x. Specifically, if $\mathcal{ID}$ is t_{Sim}-time simulatable then any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ and issues $q_{\bar{\mathsf{P}}}$ queries to the transcript-generation oracle can be simulated by a malicious prover that runs in time $t_{\bar{\mathsf{P}}} + q_{\bar{\mathsf{P}}} \cdot t_{\mathsf{Sim}}$ and does not issue any queries. Such a malicious prover is in fact attacking the Σ-protocol underlying $\mathcal{ID}$ with respect to the distribution over instances that is determined by Gen.

Finally, for considering the standard transformation of identification schemes to signature schemes via the Fiat-Shamir paradigm, we rely on the following notion of first-message unpredictability (originally referred to as "min-entropy of commitments" by Abdalla et al. [AAB+02]):

Definition 2.4. Let $\delta = \delta(\lambda)$ be function of the security parameter $\lambda \in \mathbb{N}$. An identification scheme $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ is *δ-first-message unpredictable* if for any $\lambda \in \mathbb{N}$, for any (x, w) produced by $\mathsf{Gen}(1^\lambda)$ and for any α^* it holds that $\Pr[\alpha = \alpha^*] \leq \delta(\lambda)$, where $(\alpha, \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$.

Signature schemes. A signature scheme is a tuple $\mathcal{SIG} = (\mathsf{KG}, \mathsf{Sign}, \mathsf{Verify})$ of algorithms defined as follows:

- The algorithm KG is a probabilistic algorithm that receives as input the security parameter $\lambda \in \mathbb{N}$ and outputs a pair $(\mathsf{sk}, \mathsf{vk})$ of a signing key and a verification key.
- The algorithm Sign is a (possibly) probabilistic algorithm that receives as input a signing key sk and a message m and outputs a signature σ.
- The algorithm Verify is a deterministic algorithm that receives as input a verification key vk, a message m and a signature σ, and outputs a bit $b \in \{0, 1\}$.

In terms of correctness, the standard requirement for signature schemes asks that

$$\Pr[\mathsf{Verify}_{\mathsf{vk}}(m, \mathsf{Sign}_{\mathsf{sk}}(m)) = 1] = 1$$

for every $\lambda \in \mathbb{N}$ and for every message m, where the probability is taken over the choice of $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$ and over the internal randomness of Sign and Verify. In terms of security, we rely on the following standard notion of existential unforgeability under adaptive chosen-message attack (see, for example, [Gol04]) which naturally generalizes to the random-oracle model by providing all algorithm access to the oracle.

Definition 2.5. Let $t = t(\lambda)$ and $\epsilon = \epsilon(\lambda)$ be function of the security parameter $\lambda \in \mathbb{N}$. A signature scheme $\mathcal{SIG} = (\mathsf{KG}, \mathsf{Sign}, \mathsf{Verify})$ is *(t, ϵ)-existentially unforgeable under adaptive chosen-message attacks* if for t-time probabilistic algorithm F it holds that

$$\mathbf{Adv}^{\mathsf{Forge}}_{\mathcal{SIG},F}(\lambda) \stackrel{\mathsf{def}}{=} \Pr\left[\mathsf{Forge}_{\mathcal{SIG},F}(\lambda) = 1\right] \leq \epsilon(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the experiment $\mathsf{Forge}_{\mathcal{SIG},F}(\lambda)$ is defined as follows:

1. $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$.
2. $(m^*, \sigma^*) \leftarrow F^{\mathsf{Sign}_{\mathsf{sk}}(\cdot)}(1^\lambda, \mathsf{vk})$. Let $\mathcal{Q}$ denote the set of all messages with which F queried its oracle.
3. If $\mathsf{Verify}_{\mathsf{vk}}(m^*, \sigma^*) = 1$ and $m^* \notin \mathcal{Q}$ then output 1, and otherwise output 0.

3 Our Assumption: *d*-Moment Hardness

In this section we first formally define the computational assumption on which our approach is based. Then, we demonstrate that the existing approaches for proving the security of identification schemes and signature schemes that are based on Σ-protocols with special soundness do not yield improved results when relying on our assumption.

The assumption. In what follows, we consider relations $\mathcal{R} = \{\mathcal{R}_\lambda\}_{\lambda\in\mathbb{N}}$, where $\mathcal{R}_\lambda \subseteq \mathcal{X}_\lambda \times \mathcal{W}_\lambda$ for any $\lambda \in \mathbb{N}$, and distributions $\mathcal{D} = \{\mathcal{D}_\lambda\}_{\lambda\in\mathbb{N}}$ where each $\mathcal{D}_\lambda$ produces pairs $(x, w) \in \mathcal{R}_\lambda$. For any such distribution $\mathcal{D}$ and for any probabilistic algorithm A, we denote by $\mathsf{T}_{A,\mathcal{D}_\lambda}$ the random variable corresponding to the running time of A on input x where $(x, w) \leftarrow \mathcal{D}_\lambda$.

Definition 3.1. Let $d = d(\lambda)$, $\Delta = \Delta(\lambda)$ and $\omega = \omega(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\mathcal{R} = \{\mathcal{R}_\lambda\}_{\lambda\in\mathbb{N}}$ be a relation, where $\mathcal{R}_\lambda \subseteq \mathcal{X}_\lambda \times \mathcal{W}_\lambda$ for any $\lambda \in \mathbb{N}$. We say that $\mathcal{R}$ is *d-moment (Δ, ω)-hard with respect to a distribution* $\mathcal{D} = \{\mathcal{D}_\lambda\}_{\lambda\in\mathbb{N}}$ if for every algorithm A it holds that

$$\Pr\left[(x, A(x)) \in \mathcal{R}_\lambda\right] \leq \frac{\Delta \cdot \mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}_\lambda})^d\right]}{|\mathcal{W}_\lambda|^\omega},$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability is taken over the choice of $(x, w) \leftarrow \mathcal{D}_\lambda$ and over the internal randomness of A.

When $\Delta(\lambda) = 1$ and $\omega(\lambda) = 1$ for all $\lambda \in \mathbb{N}$, we will simply say that the relation $\mathcal{R}$ is d-moment hard. As discussed in Section 1.1, in the specific context of the discrete logarithm problem the relation $\mathcal{R}$ consists of all pairs $((\mathbb{G}, p, g, h), w)$ for which $h = g^w$, and the distribution $\mathcal{D}$ consists of a group-generation algorithm that produces the description $(\mathbb{G}, p, g)$ of the group, together with a uniformly-distributed group element h. Given that the discrete logarithm problem is 2-moment hard in the generic-group model [Sho97, JT20], the assumption that the discrete logarithm problem is 2-moment hard (in the standard model) can be viewed as identifying the core essence of the problem's generic hardness in the form of a standard-model assumption.

Existing approaches. Extensive research has been devoted over the years for analyzing the security of identification schemes and signature schemes that are based on Σ-protocols with special soundness. For concreteness, we focus in this discussion on identification schemes as they already capture the main difficulties (the reader is referred to Section 5 for a discussion on transforming such schemes into signature schemes via the Fiat-Shamir paradigm [FS86]).

Given an identification scheme that is based on a Σ-protocol for a relation $\mathcal{R}$, the security of the scheme is proved by showing that any malicious prover $\bar{\mathsf{P}}$ can be transformed into an algorithm A that takes as input an instance $x \in \mathcal{X}$ and produces two accepting transcripts (α, β, γ) and $(\alpha, \beta', \gamma')$ with $\beta' \neq \beta$. The special soundness of the Σ-protocol guarantees that these two transcripts can then be used to retrieve a witness $w \in \mathcal{W}$ such that $(x, w) \in \mathcal{R}$. To the best of our knowledge, all known approaches for the construction of such an algorithm A are based on the following fundamental idea: The algorithm A uses the malicious prover $\bar{\mathsf{P}}$ to obtain an accepting transcript (α, β, γ), and then rewinds it to the same first message α and feeds it with fresh challenges β' with the hope of obtaining an additional accepting transcript $(\alpha, \beta', \gamma')$ with $\beta' \neq \beta$.

This fundamental idea traces back to the classic "forking lemma" of Pointcheval and Stern [PS00], later generalized and refined by Bellare and Neven [BN06], and by Kiltz, Masny and Pan [KMP16]. The difference between the existing approaches is reflected by the different trade-offs between the success probability of the algorithm A and its running time.

Given a malicious prover $\bar{\mathsf{P}}$ that runs in time t and breaks the security of the identification scheme with probability ϵ, then on one end of the spectrum $\bar{\mathsf{P}}$ is invoked roughly $1/\epsilon$ times, leading to an algorithm A with constant success probability and running time t/ϵ [KMP16]. On the other end of the spectrum, $\bar{\mathsf{P}}$ is invoked only twice, leading to an algorithm A with success probability roughly ϵ^2 and running time $2t$ [BN06]. When the relation $\mathcal{R}$ corresponds to the discrete logarithm problem in a group of order p where Shoup's generic hardness result is believed to hold, in both cases one obtains the bound $\epsilon \leq (t^2/p)^{1/2}$ (which is inferior to our bound $\epsilon \leq (t^2/p)^{2/3}$). More generally, if the discrete logarithm problem is d-moment (Δ, ω)-hard for some $d \geq 2$, $\Delta \geq 1$ and $\omega \leq 1$, one obtains the bound $\epsilon \leq (\Delta \cdot t^d/p^\omega)^{1/d}$ in the first case and the bound $\epsilon \leq (\Delta \cdot t^d/p^\omega)^{1/2}$ in the second case (both of which are inferior to our bound $\epsilon \leq (\Delta \cdot t^d/p^\omega)^{d/(2d-1)}$).

An approach that is closer to ours is to optimize the trade-off between the success probability of the algorithm A and its expected running time [PS00, BCC$^+$16, JT20]. In their recent work, Jaeger and Tessaro [JT20] showed that in the generic-group model any algorithm A with an expected running time $\mathbb{E}[\mathsf{T}]$ computes the discrete logarithm of a random group element with probability at most $(\mathbb{E}[\mathsf{T}]^2/p)^{1/2}$ (omitting small constants for simplicity), and this can be used for establishing concrete bounds for algorithms that do not have a strict running time.[4]

In this setting, given a malicious prover $\bar{\mathsf{P}}$ that runs in time t and breaks the security of the identification scheme with probability ϵ, Bootle et al. [BCC$^+$16] suggested the following algorithm A: It invokes $\bar{\mathsf{P}}$ once, and only if successful then it repeatedly rewinds A to the same first message and feeds it with a fresh challenge until it succeeds again.[5] A simple argument shows that A's success probability is roughly ϵ, and its expected running time is t. A similar algorithm A suggested by Pointcheval and Stern [PS00] has constant success probability and expected running time t/ϵ. In both cases, using the work of Jaeger and Tessaro one again obtains the bound $\epsilon \leq (t^2/p)^{1/2}$ as above (which is inferior to our bound $\epsilon \leq (t^2/p)^{2/3}$).[6]

4 Tighter Security for Σ-Protocols and Identification Schemes

In this section we introduce our high-moment forking lemma for establishing tighter security guarantee for Σ-protocols and identification schemes. We first focus on our result for Σ-protocols, and then extend it to identification schemes.

Given a Σ-protocol for a relation $\mathcal{R}$, we follow the approach underlying the forking lemma [PS00], and show that any malicious prover $\bar{\mathsf{P}}$ can be transformed into an algorithm A that takes as input an instance $x \in \mathcal{X}$ and produces (with a certain probability) two accepting transcripts (α, β, γ) and $(\alpha, \beta', \gamma')$ for x such that $\beta' \neq \beta$. Assuming that Π has special soundness, these two transcripts can then be used to retrieve a witness $w \in \mathcal{W}$ such that $(x, w) \in \mathcal{R}$.

However, unlike existing variants of the forking lemma, we design our algorithm A with the goal of optimizing the trade-off between its success probability and the dth moment of its running time. Assuming that $\mathcal{R}$ is a d-moment (Δ, ω)-hard relation (recall Definition 3.1), this trade-off leads to an upper bound on the success probability of the malicious prover $\bar{\mathsf{P}}$.

At a high level, given a malicious prover that runs in time t and convinces the verifier with

[4] More generally, if the discrete logarithm problem is d-moment hard for some $d \geq 2$, their approach shows that any algorithm A with an expected running time $\mathbb{E}[\mathsf{T}]$ computes the discrete logarithm of a random group element with probability at most $(\mathbb{E}[\mathsf{T}]^d/p)^{1/d}$.

[5] The rewinding technique of Bootle et al. is actually a more general one that is motivated by recent protocols with a generalized special soundness property (for which the classic forking lemma is insufficient).

[6] More generally, if the discrete logarithm problem is d-moment (Δ, ω)-hard, then using the expected-time rewinding techniques of Bootle et al. and of Pointcheval and Stern one obtains the bound $\epsilon \leq (\Delta \cdot t^d/p^\omega)^{1/d}$ (which is inferior to our bound $\epsilon \leq (\Delta \cdot t^d/p^\omega)^{d/(2d-1)}$).

probability ϵ, the description of our algorithm A is quite intuitive. First, it invokes the malicious prover to obtain a transcript (α, β, γ) of the protocol. Then, if this transcript is accepted by the verifier, it rewinds the malicious prover $B \approx 1/\epsilon^{1/d}$ times, providing it with randomly sampled challenges $\beta_1, \ldots, \beta_B$ and obtaining respective responses $\gamma_1, \ldots, \gamma_B$. If any one of these additional transcripts $(\alpha, \beta_i, \gamma_i)$ is accepted by the verifier and $\beta_i \neq \beta'$, then the algorithm A successfully retrieves a witness.

Ignoring various approximations and other technical challenges, we prove that the algorithm A has success probability roughly $B \cdot \epsilon^2 \approx \epsilon^{2-1/d}$, and the d-th moment of its running time is at most $\epsilon \cdot t^d / B^d \approx T^d$. Thus, assuming that $\mathcal{R}$ is a d-moment (Δ, ω)-hard relation leads to the bound $\epsilon \leq (\Delta \cdot t^d / |\mathcal{W}|^\omega)^{d/(2d-1)}$ on the probability of a t-time malicious prover to convince the verifier. This should be compared with the approaches discussed in Section 3, leading roughly either to success probability ϵ^2 and dth moment t^d, or to success probability ϵ and dth moment at least t^d/ϵ^{d-1}, or to constant success probability and dth moment at least t^d/ϵ^d – all of which lead to inferior bounds. Formally, we prove the following theorem:

Theorem 4.1. *Let $d = d(\lambda)$, $\Delta = \Delta(\lambda)$, $\omega = \omega(\lambda)$, $t_{\mathsf{W}} = t_{\mathsf{W}}(\lambda)$ and $t_{\bar{\mathsf{P}}} = t_{\bar{\mathsf{P}}}(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\Pi = (\mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ be a Σ-protocol with t_{W}-time special soundness for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$. If $\mathcal{R}$ is d-moment (Δ, ω)-hard with respect to a distribution $\mathcal{D}$ then for any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ it holds that*

$$\Pr\left[\mathsf{V}(x, \alpha, \beta, \gamma) = 1\right] \leq \left(\frac{\Delta \cdot (16(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}} + \frac{2}{|\mathcal{C}_\lambda|},$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability is taken over $(x, w) \leftarrow \mathcal{D}_\lambda$, $(\alpha, \mathsf{st}) \leftarrow \bar{\mathsf{P}}_1(x)$, $\beta \leftarrow \mathcal{C}_x$ and $\gamma \leftarrow \bar{\mathsf{P}}_2(\mathsf{st}, \beta)$, and where $t_{\mathsf{V}} = t_{\mathsf{V}}(\lambda)$ denotes the running time of the algorithm V, $|\mathcal{C}_\lambda|$ denotes the size of the challenge set $\mathcal{C}_x$ for any $x \in \mathcal{X}_\lambda$.

Recall that the notion of security against passive impersonations attacks for an identification scheme $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ is obtained from the experiment considered in Theorem 4.1 for its underlying Σ-protocol, by additionally providing the malicious prover with access to a transcript-generation oracle (recall Definition 2.2). As discussed in Section 2, if $\mathcal{ID}$ is t_{Sim}-time simulatable (recall Definition 2.3), then any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ and issues $q_{\bar{\mathsf{P}}}$ queries to the transcript-generation oracle can be simulated by a malicious prover that runs in time $t_{\bar{\mathsf{P}}} + q_{\bar{\mathsf{P}}} \cdot t_{\mathsf{Sim}}$ and does not issue any queries. Thus, Theorem 4.1 immediately yields the following corollary:

Corollary 4.2. *Let $d = d(\lambda)$, $\Delta = \Delta(\lambda)$, $\omega = \omega(\lambda)$, $t_{\mathsf{Sim}} = t_{\mathsf{Sim}}(\lambda)$, $t_{\mathsf{W}} = t_{\mathsf{W}}(\lambda)$, $t_{\bar{\mathsf{P}}} = t_{\bar{\mathsf{P}}}(\lambda)$ and $q_{\bar{\mathsf{P}}} = q_{\bar{\mathsf{P}}}(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ be a t_{Sim}-time simulatable identification protocol with t_{W}-time special soundness for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$. If $\mathcal{R}$ is d-moment (Δ, ω)-hard with respect to Gen, then for any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ and issues $q_{\bar{\mathsf{P}}}$ transcript-generation queries it holds that*

$$\mathbf{Adv}^{\mathsf{PA\text{-}IMP}}_{\mathcal{ID}, \bar{\mathsf{P}}}(\lambda) \leq \left(\frac{\Delta \cdot (16(t_{\bar{\mathsf{P}}} + q_{\bar{\mathsf{P}}} \cdot t_{\mathsf{Sim}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}} + \frac{2}{|\mathcal{C}_\lambda|},$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $t_{\mathsf{V}} = t_{\mathsf{V}}(\lambda)$ denotes the running time of the algorithm V, and $|\mathcal{C}_\lambda|$ denotes the size of the challenge set $\mathcal{C}_x$ for any $x \in \mathcal{X}_\lambda$.

In the remainder of this section we prove Theorem 4.1.

Proof of Theorem 4.1. Let $\bar{\mathsf{P}} = (\bar{\mathsf{P}}_1, \bar{\mathsf{P}}_2)$, and for any $\lambda \in \mathbb{N}$ let $\epsilon = \epsilon(\lambda) = \Pr\left[\mathsf{V}(x, \alpha, \beta, \gamma) = 1\right]$, where $(x, w) \leftarrow \mathcal{D}_\lambda$, $(\alpha, \mathsf{st}) \leftarrow \bar{\mathsf{P}}_1(x)$, $\beta \leftarrow \mathcal{C}_x$ and $\gamma = \bar{\mathsf{P}}_2(\mathsf{st}, \beta)$ (without loss of generality we assume that $\bar{\mathsf{P}}_2$ is deterministic given st). Let $B = \lceil 1/\epsilon^{1/d} - 1 \rceil$, and consider the following algorithm A:

The algorithm A

Input: An instance $x \in \mathcal{X}_\lambda$.

1. Sample $(\alpha, \mathsf{st}) \leftarrow \bar{\mathsf{P}}_1(x)$, $\beta_0 \leftarrow \mathcal{C}_x$ and compute $\gamma_0 = \bar{\mathsf{P}}_2(\mathsf{st}, \beta_0)$. If $\mathsf{V}(x, \alpha, \beta_0, \gamma_0) = 0$ then output $\perp$ and terminate.
2. For every $j \in [B]$ sample $\beta_j \leftarrow \mathcal{C}_x$ and compute $\gamma_j = \bar{\mathsf{P}}_2(\mathsf{st}, \beta_j)$. If for every $j \in [B]$ it holds that either $\mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 0$ or $\beta_j = \beta_0$, then output $\perp$ and terminate.
3. Output $w = \mathsf{WitnessExt}(\alpha, (\beta_0, \gamma_0), (\beta_{j^*}, \gamma_{j^*}))$, where j^* is the minimal index for which $\mathsf{V}(x, \alpha, \beta_{j^*}, \gamma_{j^*}) = 1$ and $\beta_{j^*} \neq \beta_0$.

The following lemma establishes a lower bound on the success probability of the algorithm A:

Lemma 4.3. *For any $\lambda \in \mathbb{N}$ it holds that either $\Pr[(x, A(x)) \in \mathcal{R}] \geq B \cdot \epsilon^2/8$ or $\epsilon < 2/|\mathcal{C}_\lambda|$.*

Proof of Lemma 4.3. Whenever the algorithm A reaches Step 3 the witness extraction algorithm $\mathsf{WitnessExt}$ guarantees that $(x, A(x)) \in \mathcal{R}$. Therefore,

$$\begin{aligned}
&\Pr[(x, A(x)) \in \mathcal{R}] \\
&\quad = \Pr\left[\mathsf{V}(x, \alpha, \beta_0, \gamma_0) = 1 \ \wedge \ \left(\bigvee_{j=1}^{B} \left\{ \begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 1 \\ \wedge\ \beta_j \neq \beta_0 \end{array} \right\}\right)\right] \\
&\quad = \sum_{\mathsf{st}} \left(\Pr[\mathsf{st}] \cdot \Pr\left[\mathsf{V}(x, \alpha, \beta_0, \gamma_0) = 1 \ \wedge \ \left(\bigvee_{j=1}^{B} \left\{ \begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 1 \\ \wedge\ \beta_j \neq \beta_0 \end{array} \right\}\right)\right]\right)
\end{aligned}$$

where $(x, w) \leftarrow \mathcal{D}_\lambda$, $(\alpha, \mathsf{st}) \leftarrow \bar{\mathsf{P}}_1(x)$, $\beta_0, \ldots, \beta_B \leftarrow \mathcal{C}_x$ and $\gamma_j = \bar{\mathsf{P}}_2(\mathsf{st}, \beta_j)$ for every $j \in \{0, \ldots, B\}$; and we assume without loss of generality that for any $\lambda \in \mathbb{N}$, $x \in \mathcal{X}_\lambda$ and for any (α, st) produced by $\mathsf{P}_1^*(x)$ it holds that the state st consists of λ, x and α (in addition to any other information determined by P_1^*). In what follows, for every state st, let β^*_{st} denote the lexicographically first $\beta \in \mathcal{C}_x$ for which $\mathsf{V}(x, \alpha, \beta, \bar{\mathsf{P}}_2(\mathsf{st}, \beta)) = 1$. If no such β exists, let $\beta^*_{\mathsf{st}} = \perp$. It thus holds that

$$\begin{aligned}
&\Pr[(x, A(x)) \in \mathcal{R}] \\
&\quad = \sum_{\mathsf{st}} \left(\Pr[\mathsf{st}] \cdot \Pr\left[\mathsf{V}(x, \alpha, \beta_0, \gamma_0) = 1 \ \wedge \ \left(\bigvee_{j=1}^{B} \left\{ \begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 1 \\ \wedge\ \beta_j \neq \beta^*_{\mathsf{st}} \end{array} \right\}\right)\right]\right)
\end{aligned}$$

where for every state st, the probability is taken only over the choice of $\beta_0, \ldots, \beta_B \leftarrow \mathcal{C}_x$. Then, for every fixed state st, the events $\mathsf{V}(x, \alpha, \beta_0, \gamma_0) = 1$ and $\{\mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 1 \wedge \beta_j \neq \beta^*_{\mathsf{st}}\}_j$ are independent, and therefore

$$\begin{aligned}
\Pr\left[\bigvee_{j=1}^{B} \left\{ \begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 1 \\ \wedge\ \beta_j \neq \beta^*_{\mathsf{st}} \end{array} \right\}\right] &= 1 - \Pr\left[\bigwedge_{j=1}^{B} \left\{ \begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 0 \\ \vee\ \beta_j = \beta^*_{\mathsf{st}} \end{array} \right\}\right] \\
&= 1 - \prod_{j=1}^{B} \Pr\left[\begin{array}{c} \mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 0 \\ \vee\ \beta_j = \beta^*_{\mathsf{st}} \end{array} \right] \\
&\geq 1 - \prod_{j=1}^{B} \min\Big\{1, \Pr[\mathsf{V}(x, \alpha, \beta_j, \gamma_j) = 0] + \Pr[\beta_j = \beta^*_{\mathsf{st}}]\Big\} \\
&\geq 1 - \left(1 - \max\left\{0, \epsilon(\mathsf{st}) - \frac{1}{|\mathcal{C}_\lambda|}\right\}\right)^B,
\end{aligned}$$

where $\epsilon(\mathsf{st}) = \Pr_\beta\left[\mathsf{V}(x,\alpha,\beta,\bar{\mathsf{P}}_2(\mathsf{st},\beta)) = 1\right]$ for each st. Denoting $\widetilde{\epsilon}(\mathsf{st}) = \max\{0, \epsilon(\mathsf{st}) - 1/|\mathcal{C}_\lambda|\}$ for every st, we obtain

$$\begin{aligned}\Pr\left[(x,A(x)) \in \mathcal{R}\right] &\geq \sum_{\mathsf{st}} \left(\Pr\left[\mathsf{st}\right] \cdot \epsilon(\mathsf{st}) \cdot \left(1 - (1-\widetilde{\epsilon}(\mathsf{st}))^B\right)\right) \\ &= \mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st}) \cdot \left(1 - (1-\widetilde{\epsilon}(\mathsf{st}))^B\right)\right].\end{aligned}$$

The following claim (which is proved in Appendix A.1) provides a lower bound on the above term $\mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st}) \cdot \left(1 - (1-\widetilde{\epsilon}(\mathsf{st}))^B\right)\right]$. Note that this term is the expectation of a non-convex function of $\widetilde{\epsilon}(\mathsf{st})$ over the interval $[0,1]$, and therefore such a lower bound is not directly implied by Jensen's inequality.

Claim 4.4. *It holds that* $\mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st}) \cdot \left(1 - (1-\widetilde{\epsilon}(\mathsf{st}))^B\right)\right] \geq \frac{1}{2} \cdot B \cdot \left(\epsilon - \frac{1}{|\mathcal{C}_\lambda|}\right)^2$.

Given Claim 4.4, it holds that either $\epsilon < 2/|\mathcal{C}_\lambda|$ or $\Pr\left[(x,A(x)) \in \mathcal{R}\right] \geq \frac{1}{2} \cdot B \cdot (\epsilon/2)^2$, and this concludes the proof of Lemma 4.3. ∎

The following lemma establishes an upper bound on the dth moment of the running time of the algorithm A (recall that $\mathsf{T}_{A,\mathcal{D}_\lambda}$ denotes the random variable corresponding to the running time of A on input x where $(x,w) \leftarrow \mathcal{D}_\lambda$):

Lemma 4.5. *For any* $\lambda \in \mathbb{N}$ *it holds that* $\mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}_\lambda})^d\right] \leq 2(1+B)^d \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}})^d \cdot \epsilon$.

Proof of Lemma 4.5. The description of A yields that with probability $1-\epsilon$ it runs in time at most $t_{\bar{\mathsf{P}}} + t_{\mathsf{V}}$, and with probability ϵ it runs in time at most $(1+B) \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}}) + t_{\mathsf{W}}$ (for simplicity we assume that the time required for sampling a uniform $\beta \in \mathcal{C}_x$ is subsumed by $t_{\bar{\mathsf{P}}} + t_{\mathsf{V}}$). Therefore,

$$\begin{aligned}\mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}_\lambda})^d\right] &\leq (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}})^d \cdot (1-\epsilon) + ((1+B) \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d \cdot \epsilon \\ &\leq (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}})^d + ((1+B) \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d \cdot \epsilon \\ &\leq 2(1+B)^d \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}})^d \cdot \epsilon. \qquad (4.1)\end{aligned}$$

where Eq. (4.1) follows from the fact that $B \geq 1/\epsilon^{1/d} - 1$ (and thus $1 \leq (1+B)^d \cdot \epsilon$). ∎

Equipped with Lemma 4.3 and Lemma 4.5, the assumption that $\mathcal{R}$ is a d-moment (Δ,ω)-hard relation with respect to the distribution $\mathcal{D}$ implies that either $\epsilon < 2/|\mathcal{C}_\lambda|$ or

$$\begin{aligned}\frac{B \cdot \epsilon^2}{8} &\leq \Pr\left[(x,\mathsf{A}(x)) \in \mathcal{R}\right] \\ &\leq \frac{\Delta \cdot \mathbb{E}\left[(\mathsf{T}_{A,\mathcal{D}_\lambda})^d\right]}{|\mathcal{W}_\lambda|^\omega} \\ &\leq \frac{\Delta \cdot 2(1+B)^d \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}})^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega} \\ &\leq \frac{\Delta \cdot 2^{d+1} B^d \cdot (t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}})^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega} \\ &\leq \frac{\Delta \cdot B^d \cdot (2(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega}\end{aligned}$$

Our choice of $B = \lceil 1/\epsilon^{1/d} - 1 \rceil$ guarantees that $B^{d-1} \leq \epsilon^{1-1/d}$, and therefore

$$\epsilon^{2-\frac{1}{d}} \leq \frac{\epsilon}{B^{d-1}} \leq \frac{\Delta \cdot 8 \cdot (2(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega}$$

leading to

$$\epsilon \leq \left(\frac{\Delta \cdot 8 \cdot (2(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega} \right)^{\frac{d}{2d-1}}.$$

Therefore, overall we obtain

$$\begin{aligned} \epsilon &\leq \max \left\{ \left(\frac{\Delta \cdot 8 \cdot (2(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega} \right)^{\frac{d}{2d-1}}, \frac{2}{|\mathcal{C}_\lambda|} \right\} \\ &\leq \left(\frac{\Delta \cdot (16(t_{\bar{\mathsf{P}}} + t_{\mathsf{V}} + t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega} \right)^{\frac{d}{2d-1}} + \frac{2}{|\mathcal{C}_\lambda|}. \end{aligned}$$

■

5 Tighter Security for Signature Schemes

In this section we show that our approach extends to establishing tighter security guarantees for signature schemes that are obtained from identification schemes via the Fiat-Shamir paradigm [FS86]. The generic analysis of the Fiat-Shamir transform in this context [AAB+02] shows that if any malicious prover that runs in time t breaks the security of the identification scheme with probability at most ϵ, then any malicious forger that runs in time roughly t and issues q_{H} random-oracle queries breaks the security of the signature scheme with probability at most roughly $q_{\mathsf{H}} \cdot \epsilon$. Therefore, given our result from Section 4, if the relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$ underlying the identification scheme is a d-moment (Δ, ω)-hard relation, then any such forger breaks the security of the signature scheme with probability at most roughly $q_{\mathsf{H}} \cdot (\Delta \cdot t^d / |\mathcal{W}|^\omega)^{d/(2d-1)}$.

Here, we show that the latter bound can be further improved by applying our proof technique directly, showing that any forger as above breaks the security of the signature scheme with probability at most roughly $(q_{\mathsf{H}} \cdot \Delta \cdot t^d / |\mathcal{W}|^\omega)^{d/(2d-1)}$. Note that some dependency on q_{H} seems to be unavoidable, at least for a very large class of reductions which includes in particular all reductions based on the underlying paradigm of the forking lemma [PV05, GBL08, Seu12, FJS14]. In what follows, we first recall the standard transformation from identification schemes to signature schemes via the Fiat-Shamir paradigm [FS86, AAB+02], and then state and prove our result.

Let $\mathcal{ID} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ be an identification scheme for a relation $\mathcal{R} \subseteq \mathcal{X} \times \mathcal{W}$, and let H be a hash function mapping triplets of the form (x, m, α) to challenges in $\mathcal{C}_x$. The Fiat-Shamir paradigm then defines the following signature scheme $\mathcal{SIG}_{\mathcal{ID},\mathsf{H}} = (\mathsf{KG}, \mathsf{Sign}, \mathsf{Verify})$:

- $\mathsf{KG}(1^\lambda)$ samples $(x, w) \leftarrow \mathsf{Gen}(1^\lambda)$ and outputs $\mathsf{sk} = (x, w)$ and $\mathsf{vk} = x$.
- $\mathsf{Sign}(\mathsf{sk}, m)$ parses $\mathsf{sk} = (x, w)$ and outputs $\sigma = (\alpha, \beta, \gamma)$, where $(\alpha, \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$, $\beta = \mathsf{H}(\mathsf{vk}, m, \alpha)$ and $\gamma \leftarrow \mathsf{P}_2(\mathsf{st}, \beta)$.
- $\mathsf{Verify}(\mathsf{vk}, m, \sigma)$ parses $\sigma = (\alpha, \beta, \gamma)$, and outputs 1 if and only $\mathsf{V}(\mathsf{vk}, \alpha, \beta, \gamma) = 1$ and $\beta = \mathsf{H}(\mathsf{vk}, m, \alpha)$.

Note that the value β in fact does not have to be included in the signature $\sigma = (\alpha, \beta, \gamma)$ as it can be computed given vk, m and α. Alternatively, in some identification protocols, for any x, β and

γ there is a unique and efficiently computable α for which $\mathsf{V}(x,\alpha,\beta,\gamma)=1$, and in such cases the value α does not have to be included in the signature $\sigma=(\alpha,\beta,\gamma)$.

We prove the following theorem (the reader is referred to Section 2 for the standard notions of t_{Sim}-time simulatability, t_{W}-time special soundness, and δ-first-message unpredictability for identification protocols):

Theorem 5.1. *Let $d=d(\lambda)$, $\Delta=\Delta(\lambda)$, $\omega=\omega(\lambda)$, $t_{\mathsf{Sim}}=t_{\mathsf{Sim}}(\lambda)$, $t_{\mathsf{W}}=t_{\mathsf{W}}(\lambda)$, $\delta=\delta(\lambda)$, $t_F=t_F(\lambda)$, $q_{\mathsf{H}}=q_{\mathsf{H}}(\lambda)$ and $q_{\mathsf{Sign}}=q_{\mathsf{Sign}}(\lambda)$ be functions of the security parameter $\lambda\in\mathbb{N}$, and let $\mathcal{ID}=(\mathsf{Gen},\mathsf{P}_1,\mathsf{P}_2,\mathsf{V},\mathcal{C})$ be a t_{Sim}-time simulatable identification protocol with t_{W}-time special soundness and δ-first-message unpredictability for a relation $\mathcal{R}\subseteq\mathcal{X}\times\mathcal{W}$. If $\mathcal{R}$ is d-moment (Δ,ω)-hard with respect to Gen, and the hash function H is modeled as a random oracle, then for every t_F-time algorithm F that issues q_{H} oracle queries and q_{Sign} signing queries it holds that*

$$\mathbf{Adv}^{\mathsf{Forge}}_{\mathcal{SIG}_{\mathcal{ID},\mathsf{H},F}}(\lambda)\leq\left(\frac{q_{\mathsf{H}}\cdot\Delta\cdot(16(t_F+q_{\mathsf{Sign}}\cdot t_{\mathsf{Sim}}+t_{\mathsf{V}}+t_{\mathsf{W}}))^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}}+2\cdot\left(\frac{q_{\mathsf{H}}^2+1}{|\mathcal{C}_\lambda|}+q_{\mathsf{Sign}}\cdot q_{\mathsf{H}}^2\cdot\delta\right)$$

for all sufficiently large $\lambda\in\mathbb{N}$, where $t_{\mathsf{V}}=t_{\mathsf{V}}(\lambda)$ denotes the running time of the algorithm V and $|\mathcal{C}_\lambda|$ denotes the size of the challenge set $\mathcal{C}_x$ for any $x\in\mathcal{X}_\lambda$.

At a high level, the proof of Theorem 5.1 follows a similar outline to that Theorem 4.1, while carefully handling additional technical challenges that arise when considering the unforgeability of signatures schemes in the random oracle model, as to minimize the increase in the adversary's success probability. Concretely, let F be a forger that runs in time t, issues at most q_{H} random-oracle queries and produces a successful forgery with probability ϵ. Our algorithm A invokes the forger to obtain a message-signature pair $(m,\sigma=(\alpha,\beta,\gamma))$, while simulating the random oracle and the signing oracle using the simulatability of the underlying Σ-protocol. Then, it checks that this pair is a valid one and that the forger queried the random oracle for the hash value of (x,m,α). If so, it rewinds the forger $B\approx 1/\epsilon^{1/d}$ times to the point just before (x,m,α) was queried, simulating a fresh random oracle from that point on each time, and obtaining respective message-signature pairs $(m_1,\sigma_1=(\alpha_1,\beta_1,\gamma_1)),\ldots,(m_B,\sigma_B=(\alpha_B,\beta_B,\gamma_B))$. If any one of these additional pairs (m_i,σ_i) is a valid one, and in addition $\alpha_i=\alpha$ and $\beta_i\neq\beta$, then the algorithm A successfully retrieves a witness.

Technical challenges and approximations omitted, we prove that the algorithm A has success probability roughly $B\cdot\epsilon^2/q_{\mathsf{H}}\approx\epsilon^{2-1/d}/q_{\mathsf{H}}$, and the d-th moment of its running time is at most $\epsilon\cdot t^d/B^d\approx T^d$. Thus, assuming that $\mathcal{R}$ is a d-moment (Δ,ω)-hard relation leads to the bound $\epsilon\leq(q_{\mathsf{H}}\cdot\Delta\cdot t^d/|\mathcal{W}|^\omega)^{d/(2d-1)}$ on the advantage of a t-time forger which issues q_{H} random oracle queries in breaking the existential unforgeability of the signature schemes via an adaptive-chosen message attack.

Proof of Theorem 5.1. For any $\lambda\in\mathbb{N}$ let $\epsilon=\epsilon(\lambda)=\mathbf{Adv}^{\mathsf{Forge}}_{\mathcal{SIG}_{\mathcal{ID},\mathsf{H},F}}(\lambda)$, and $B=\lceil 1/\epsilon^{1/d}-1\rceil$. We make the following assumptions about the forger F without loss of generality:

- F does not issue the same query twice to H, as F can always store the answers received from the oracle.
- After querying the signing oracle $\mathsf{Sign}(\mathsf{sk},\cdot)$ on a message m and receiving a signature $\sigma=(\alpha,\beta,\gamma)$, F does not query H on (vk,m,α). This is without loss of generality, since in the real experiment $\mathsf{Forge}_{\mathcal{SIG}_{\mathcal{ID},\mathsf{H},F}}(\lambda)$, it is always the case $\mathsf{H}(\mathsf{vk},m,\alpha)=\beta$, and hence F can just store this value.
- If $F^{\mathsf{H},\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk})$ outputs a pair $(m,\sigma=(\alpha,\beta,\gamma))$ and F queried H for $y=\mathsf{H}(\mathsf{vk},m,\alpha)$, then $\beta=y$. If this is not the case, then it necessarily holds that $\mathsf{Verify}(\mathsf{vk},m,\sigma)=0$ and thus $\mathsf{Forge}_{\mathcal{SIG}_{\mathcal{ID},\mathsf{H},F}}(\lambda)=0$.

- F never outputs a message m on which it has queried $\mathsf{Sign}(\mathsf{sk},\cdot)$.

Consider the following algorithm A (which uses the algorithms Sim and $\mathsf{WitnessExt}$ provided by the simulatability and special soundness of $\mathcal{ID}$, respectively):

The Algorithm A

Input: An instance $x \in \mathcal{X}_\lambda$.

1. Set $\mathsf{vk} = x$, sample randomness $r \leftarrow \{0,1\}^*$ for F, sample q_H hash values $\vec{y_0} = (y_{0,1}, \ldots, y_{0,q_\mathsf{H}}) \leftarrow \mathcal{C}_x^q$, and sample q_Sign transcripts $(\alpha'_0, \beta'_0, \gamma'_0), \ldots, (\alpha'_{q_\mathsf{Sign}}, \beta'_{q_\mathsf{Sign}}, \gamma'_{q_\mathsf{Sign}}) \leftarrow \mathsf{Sim}(x)$.
2. Invoke $(m_0, \alpha_0, \beta_0, \gamma_0) \leftarrow F^{\mathsf{H},\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk}; r)$ while simulating the oracles to F as follows:
   - H-queries: For each $i \in [q_\mathsf{H}]$ respond to the ith query with $y_{0,i}$.
   - Sign-queries: For each $i \in [q_\mathsf{Sign}]$ let m denote the ith query and responds as follows. If $\mathsf{H}(\mathsf{vk}, m, \alpha'_i)$ was already queried and the response was different than β'_i, then output $\perp$ and terminate. Otherwise, respond with the signature $\sigma = (\alpha'_i, \beta'_i, \gamma'_i)$.
3. If $\mathsf{V}(x, m_0, \alpha_0, \beta_0, \gamma_0) = 0$ or if F did not query for $\mathsf{H}(\mathsf{vk}, m_0, \alpha_0)$ then output $\perp$ and terminate. Otherwise, let $i^* \in [q_\mathsf{H}]$ denote the index of query in which F queried for $\mathsf{H}(\mathsf{vk}, m_0, \alpha_0)$.
4. For every $j \in [B]$:
   (a) Sample $y_{j,i^*}, \ldots, y_{j,q} \leftarrow \mathcal{C}_x$. If $y_{j,i^*} = y_{0,i^*}$ then skip to the next iteration.
   (b) Invoke $(m_j, \alpha_j, \beta_j, \gamma_j) \leftarrow F^{\mathsf{H},\mathsf{Sign}(\mathsf{sk},\cdot)}(\mathsf{vk}; r)$ while simulating the oracles as in Step 2 with the following modification: For each $\ell \in \{i^*, \ldots, q\}$ respond to F's ℓth H-query with $y_{j,\ell}$.
   (c) If $m_j = m_0$, $\alpha_j = \alpha_0$, $\beta_j = y_{j,i^*}$ and $\mathsf{V}(x, \alpha_j, \beta_j, \gamma_j) = 1$ then output $w = \mathsf{WitnessExt}(\alpha_0, (\beta_0, \gamma_0), (\beta_j, \gamma_j))$ and terminate.
5. Output $\perp$.

The following lemma establishes a lower bound on the success probability of the algorithm A:

Lemma 5.2. *For any $\lambda \in \mathbb{N}$ it holds that either*

$$\Pr\left[(x, A(x)) \in \mathcal{R}\right] \geq \frac{B \cdot \epsilon^2}{8 \cdot q_\mathsf{H}}$$

or

$$\epsilon < 2 \cdot \left(\frac{q_\mathsf{H}^2 + 1}{|\mathcal{C}_\lambda|} + q_\mathsf{Sign} \cdot q_\mathsf{H}^2 \cdot \delta \right).$$

Proof of Lemma 5.2. Denote by I_0 the random variable corresponding to the index of the H-query in which F queries H with $(\mathsf{vk}, m_0, \alpha_0)$ in its invocation in Step 2. If in this invocation F does not query H with $(\mathsf{vk}, m_0, \alpha_0)$ or if $\beta_0 \neq y_{0,I_0}$, then we set $I_0 = 0$. Similarly, for each $j \in [B]$ denote by I_j the random variable corresponding to the index of the H-query in which F queries H with $(\mathsf{vk}, m_j, \alpha_j)$ in its invocation in the jth iteration of Step 4. If in this invocation F does not query $(\mathsf{vk}, m_j, \alpha_j)$ or if $\beta_j \neq y_{j,I_j}$, then we set $I_j = 0$.

For every $i \in [q_\mathsf{Sign}]$ let $\mathsf{Bad}_{0,i}$ denote the event in which A aborts in the ith Sign-query of F in its invocation in Step 2. That is, if we denote by m the ith Sign-query of F in its invocation in Step 2, then $\mathsf{Bad}_{0,i}$ is the event in which F already queried H with $(\mathsf{vk}, m, \alpha'_i)$ in an earlier stage of this invocation, and the response was different than β'_i. For every $j \in [B]$ and $i \in [q_\mathsf{Sign}]$, let $\mathsf{Bad}_{j,i}$ be defined analogously with respect to the jth invocation of F in Step 4, and let $\mathsf{Bad}_\ell = \bigvee_{i \in [q_\mathsf{Sign}]} \mathsf{Bad}_{\ell,i}$

for every $\ell \in \{0,\ldots,B\}$. Since transcripts sampled using Sim are distributed identically as honestly-generated transcripts, then by the δ-first-message unpredictability of the identification scheme $\mathcal{ID}$, it holds that

$$\begin{aligned}\Pr\left[\mathsf{Bad}_\ell\right] &\leq \sum_{i=1}^{q_{\mathsf{Sign}}} \mathsf{Bad}_{\ell,i} \\ &\leq \sum_{i=1}^{q_{\mathsf{Sign}}} q_{\mathsf{H}} \cdot \delta \\ &\leq q_{\mathsf{Sign}} \cdot q_{\mathsf{H}} \cdot \delta.\end{aligned}$$

Whenever A reaches Step 4c, it is guaranteed that it invokes the witness extraction algorithm on two accepting transcripts with distinct challenges. Therefore,

$$\begin{aligned}&\Pr\left[(x,A(x)) \in \mathcal{R}\right] \\ &\quad= \Pr\left[\begin{pmatrix}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0>0\ \wedge\ \overline{\mathsf{Bad}_0}\end{pmatrix}\ \wedge\ \left(\bigvee_{j=1}^{B}\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\\ \wedge\ I_0=I_j\ \wedge\ y_{j,I_j}\neq y_{0,I_0}\\ \wedge\ \overline{\mathsf{Bad}_j}\end{array}\right\}\right)\right] \\ &\quad= \sum_{i=1}^{q_{\mathsf{H}}}\Pr\left[\begin{pmatrix}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0=i\ \wedge\ \overline{\mathsf{Bad}_0}\end{pmatrix}\ \wedge\ \left(\bigvee_{j=1}^{B}\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\\ \wedge\ I_j=i\ \wedge\ y_{j,i}\neq y_{0,i}\\ \wedge\ \overline{\mathsf{Bad}_j}\end{array}\right\}\right)\right] \\ &\quad= \sum_{i=1}^{q_{\mathsf{H}}}\sum_{\substack{x,r,\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}\\ y_{0,1},\ldots,y_{0,i-1}}}\left(\Pr\left[\begin{array}{c}x\ \wedge\ r\ \wedge\ \{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}\\ \wedge\ y_{0,1},\ldots,y_{0,i-1}\end{array}\right]\right. \\ &\qquad\left.\times\Pr\left[\begin{array}{c}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0=i\ \wedge\ \overline{\mathsf{Bad}_0}\\ \wedge\ \left(\bigvee_{j=1}^{B}\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\\ \wedge\ I_j=i\ \wedge\ y_{j,i}\neq y_{0,i}\\ \wedge\ \overline{\mathsf{Bad}_j}\end{array}\right\}\right)\end{array}\right]\right) \\ &\quad\geq \sum_{i=1}^{q_{\mathsf{H}}}\sum_{\substack{x,r,\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}\\ y_{0,1},\ldots,y_{0,i-1}}}\left(\Pr\left[\begin{array}{c}x\ \wedge\ r\ \wedge\ \{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}\\ \wedge\ y_{0,1},\ldots,y_{0,i-1}\end{array}\right]\right. \\ &\qquad\left.\times\Pr\left[\begin{array}{c}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0=i\ \wedge\overline{\mathsf{Bad}_0}\\ \wedge\ \left(\bigvee_{j=1}^{B}\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\\ \wedge\ I_j=i\ \wedge\ \overline{\mathsf{Bad}_j}\\ \wedge\ \forall\ell\in\{i,\ldots,q_{\mathsf{H}}\}\ :\ y_{j,\ell}\neq y_{0,\ell}\end{array}\right\}\right)\end{array}\right]\right)\end{aligned}$$

where $(x,w)\leftarrow\mathsf{Gen}(1^\lambda)$, and the values $r,\{\{y_{j,\ell}\}_{\ell\in[q_{\mathsf{H}}]},m_j,\alpha_j,\beta_j,\gamma_j\}_{j\in\{0,\ldots,B\}}$ and $\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}$ are distributed as in the description of A.

For every $y_{0,1},\ldots,y_{0,i-1}$ let us denote $\vec{y}[i-1]=(y_{0,1},\ldots,y_{0,i-1})$ and $\vec{\tau}=\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}$. For every i, x, r, $\vec{\tau}$ and $\vec{y}[i-1]$, denote by $(y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1]),\ldots,y_{q_{\mathsf{H}}}^*(i,x,r,\vec{\tau},\vec{y}[i-1]))$ the lexicographically first tuple of $q_{\mathsf{H}}-i+1$ values in $\mathcal{C}_x$ for which the following holds: In the simulation

$F^{\mathsf{H},\mathsf{Sign}(\mathsf{sk},\cdot)}(x;r)$ (where the the oracles are simulated to F as in the description of A using the values $\vec{\tau}$ and $\vec{y}[i-1], y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1]),\ldots,y_{q_\mathsf{H}}^*(i,x,r,\vec{\tau},\vec{y}[i-1]))$, F outputs (m,α,β,γ) such that:

- $\mathsf{V}(x,m,\alpha,\beta,\gamma)=1$;
- F's ith query to H is (x,m,α);
- For every $\ell\in[q_\mathsf{Sign}]$: If m_ℓ is the ℓth query of F to $\mathsf{Sign}(\mathsf{sk},\cdot)$, then F does not query H on (x,m_ℓ,α'_ℓ) before its ℓth query to $\mathsf{Sign}(\mathsf{sk},\cdot)$.

Then, it holds that

$$\begin{aligned}&\Pr\left[(x,A(x))\in\mathcal{R}\right]\\ &\quad\geq\sum_{i=1}^{q_\mathsf{H}}\sum_{\substack{x,r,\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}\\ y_{0,1},\cdots,y_{0,i-1}}}\left(\Pr\left[\begin{array}{c}x\ \wedge\ r\ \wedge\ \{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}\\ \wedge\ y_{0,1},\ldots,y_{0,i-1}\end{array}\right]\right.\\ &\qquad\left.\times\Pr\left[\begin{array}{c}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0=i\ \wedge\overline{\mathsf{Bad}_0}\\ \wedge\ \left(\bigvee_{j=1}^{B}\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\\ \wedge\ I_j=i\ \wedge\ \overline{\mathsf{Bad}_j}\\ \wedge\,\forall\ell\in\{i,\ldots,q\}\ :\ y_{j,\ell}\neq y_\ell^*(i,x,r,\vec{\tau},\vec{y}[i-1])\end{array}\right\}\right)\end{array}\right]\right)\end{aligned}$$

For every fixing of i, x, r, $\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}$ and $y_{0,1},\ldots,y_{0,i-1}$, the event $\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0))=1\ \wedge\ I_0=i\ \wedge\ \overline{\mathsf{Bad}_0}$ and the events

$$\left\{\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\wedge\ I_j=i\ \wedge\ \overline{\mathsf{Bad}_j}\\ \wedge\,\forall\ell\in\{i,\ldots,q\}\ :\ y_{j,i}\neq y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1])\end{array}\right\}_{j\in[B]}$$

are independent. Therefore,

$$\begin{aligned}&\Pr\left[(x,A(x))\in\mathcal{R}\right]\\ &\quad\geq\sum_{i=1}^{q_\mathsf{H}}\sum_{\substack{x,r,\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}\\ y_{0,1},\cdots,y_{0,i-1}}}\left(\Pr\left[\begin{array}{c}x\ \wedge\ r\ \wedge\ \{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}\\ \wedge\ y_{0,1},\ldots,y_{0,i-1}\end{array}\right]\cdot\Pr\left[\begin{array}{c}\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1\\ \wedge\ I_0=i\ \wedge\ \overline{\mathsf{Bad}_0}\end{array}\right]\right.\\ &\qquad\left.\times\left(1-\prod_{j=1}^{B}\Pr\left[\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=0\ \vee\ I_j\neq i\ \vee\ \mathsf{Bad}_j\\ \vee\,\exists\ell\in\{i,\ldots,q\}\ :\ y_{j,i}=y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1])\end{array}\right]\right)\right),\end{aligned}$$

and for every $j\in[B]$ the union bound implies that

$$\begin{aligned}&\Pr\left[\begin{array}{c}\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=0\ \vee\ I_j\neq i\ \vee\ \mathsf{Bad}_j\\ \vee\,\exists\ell\in\{i,\ldots,q\}\ :\ y_{j,i}=y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1])\end{array}\right]\\ &\qquad\leq\min\left\{1,\Pr\left[\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=0\ \vee\ I_j\neq i\right]\right.\\ &\qquad\quad\left.+\Pr\left[\exists\ell\in\{i,\ldots,q\}\ :\ y_{j,i}=y_i^*(i,x,r,\vec{\tau},\vec{y}[i-1])\right]+\Pr\left[\mathsf{Bad}_j\right]\right\}\\ &\qquad\leq\min\left\{1,1-\Pr\left[\mathsf{V}(x,m_j,\alpha_j,\beta_j,\gamma_j)=1\ \wedge\ I_j=i\right]+\frac{q_\mathsf{H}}{|\mathcal{C}_\lambda|}+q_\mathsf{Sign}\cdot q_\mathsf{H}\cdot\delta\right\}.\end{aligned}$$

For every i, x, r, $\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_\mathsf{Sign}]}$ and $y_{0,1},\ldots,y_{0,i-1}$ denote

$$\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1])=\max\left\{0,\Pr\left[\mathsf{V}(\mathsf{vk},m_0,\alpha_0,\beta_0,\gamma_0)=1\ \wedge\ I_0=i\right]-\frac{q_\mathsf{H}}{|\mathcal{C}_\lambda|}-q_\mathsf{Sign}\cdot q_\mathsf{H}\cdot\delta\right\}.$$

Then, we obtain that

$$\Pr\left[(x,A(x))\in\mathcal{R}\right] \geq \sum_{i=1}^{q_{\mathsf{H}}} \sum_{\substack{x,r,\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]} \\ y_{0,1},\ldots,y_{0,i-1}}} \left(\Pr\left[\begin{matrix} x \ \wedge \ r \ \wedge \ \{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]} \\ \wedge\ y_{0,1},\ldots,y_{0,i-1}\end{matrix}\right] \cdot \widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]) \right.$$

$$\left. \times \left(1-(1-\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]))^B\right)\right)$$

$$= \sum_{i=1}^{q_{\mathsf{H}}} \mathbb{E}\left[\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]) \cdot \left(1-(1-\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]))^B\right)\right],$$

where the expectation is taken over the choice of $x, r, y_{0,1}, \ldots, y_{0,i-1}$ and of $\{(\alpha'_\ell,\beta'_\ell,\gamma'_\ell)\}_{\ell\in[q_{\mathsf{Sign}}]}$.

For each $i \in [q_{\mathsf{H}}]$, denote $\epsilon_i = \Pr\left[\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1 \ \wedge \ I_0=i\right]$ and $\widetilde{\epsilon}_i = \mathbb{E}\left[\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1])\right]$. The following claim (which is proved in Appendix A.2) provides a lower bound on each of the terms in the above sum (note that each term is the expectation of a non-convex function, and therefore such a lower bound is not directly implied by Jensen's inequality)

Claim 5.3. *For every $i \in [q_{\mathsf{H}}]$ it holds that*

$$\mathbb{E}\left[\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]) \cdot \left(1-(1-\widetilde{\epsilon}_i(x,r,\vec{\tau},\vec{y}[i-1]))^B\right)\right] \geq \frac{1}{2}\cdot B\cdot \widetilde{\epsilon}_i^2.$$

Claim 5.3 together with Jensen's inequality imply that

$$\begin{aligned}
\Pr\left[(x,A(x))\in\mathcal{R}\right] &\geq \frac{1}{2}\cdot B\cdot \sum_{i=1}^{q_{\mathsf{H}}} \widetilde{\epsilon}_i^2. \\
&\geq \frac{1}{2\cdot q_{\mathsf{H}}}\cdot B\cdot \left(\sum_{i=1}^{q_{\mathsf{H}}} \widetilde{\epsilon}_i\right)^2 \\
&\geq \frac{1}{2\cdot q_{\mathsf{H}}}\cdot B\cdot \left(\sum_{i=1}^{q_{\mathsf{H}}} \left(\epsilon_i - \frac{q_{\mathsf{H}}}{|\mathcal{C}_\lambda|} - q_{\mathsf{Sign}}\cdot q_{\mathsf{H}}\cdot\delta\right)\right)^2 \\
&= \frac{1}{2\cdot q_{\mathsf{H}}}\cdot B\cdot \left(\Pr\left[\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1 \ \wedge \ I_0>0\right] - \frac{q_{\mathsf{H}}^2}{|\mathcal{C}_\lambda|} - q_{\mathsf{Sign}}\cdot q_{\mathsf{H}}^2\cdot\delta\right)^2.
\end{aligned}$$

Observe that when F outputs a pair $(m,\sigma=(\alpha,\beta,\gamma))$ without querying H on (vk,m,α), the view of F at termination is independent of the value $\mathsf{H}(\mathsf{vk},m,\alpha)$. Hence, the probability that it outputs a value β such that $\mathsf{H}(\mathsf{vk},m,\alpha)=\beta$ (which is a necessary condition for F to win the experiment) is at most $1/|\mathcal{C}_\lambda|$. Therefore,

$$\Pr\left[\mathsf{V}(x,m_0,\alpha_0,\beta_0,\gamma_0)=1 \ \wedge \ I_0>0\right] \geq \epsilon - \frac{1}{|\mathcal{C}_\lambda|},$$

which implies that

$$\Pr\left[(x,A(x))\in\mathcal{R}\right] \geq \frac{1}{2\cdot q_{\mathsf{H}}}\cdot B\cdot \left(\epsilon - \frac{q_{\mathsf{H}}^2+1}{|\mathcal{C}_\lambda|} - q_{\mathsf{Sign}}\cdot q_{\mathsf{H}}^2\cdot\delta\right)^2.$$

Then, either $\epsilon < 2\cdot\left(\frac{q_{\mathsf{H}}^2+1}{|\mathcal{C}_\lambda|} + q_{\mathsf{Sign}}\cdot q_{\mathsf{H}}^2\cdot\delta\right)$, or

$$\Pr\left[(x,A(x))\in\mathcal{R}\right] \geq \frac{1}{8\cdot q_{\mathsf{H}}}\cdot B\cdot\epsilon^2.$$

■

The following lemma establishes an upper bound on the dth moment of the running time of the algorithm A (recall that $\mathsf{T}_{A,\mathsf{KG}(1^\lambda)}$ denotes the random variable corresponding to the running time of A on input x where $(x,w) \leftarrow \mathsf{KG}(1^\lambda)$):

Lemma 5.4. *For any $\lambda \in \mathbb{N}$ it holds that*

$$\mathbb{E}\left[\left(\mathsf{T}_{A,\mathsf{KG}(1^\lambda)}\right)^d\right] \leq 2(1+B)^d \cdot \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}}\right)^d \cdot \epsilon.$$

Proof of Lemma 5.4. The description of A yields that with probability $1-\epsilon$ it runs in time at most $q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}}$, and with probability ϵ it runs in time at most $q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + (1+B) \cdot (t_F + t_{\mathsf{V}}) + t_{\mathsf{W}}$ (for simplicity we assume that the time required for sampling a uniform $\beta \in \mathcal{C}_x$ is subsumed by $t_F + t_{\mathsf{V}}$). Therefore,

$$\begin{aligned}
\mathbb{E}\left[\left(\mathsf{T}_{A,\mathsf{KG}(1^\lambda)}\right)^d\right] &\leq \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}}\right)^d \cdot (1-\epsilon) + \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + (1+B) \cdot (t_F + t_{\mathsf{V}}) + t_{\mathsf{W}}\right)^d \cdot \epsilon \\
&\leq \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}}\right)^d + \left((1+B) \cdot \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}}\right)\right)^d \cdot \epsilon \\
&\leq 2(1+B)^d \cdot \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}}\right)^d \cdot \epsilon. \qquad (5.1)
\end{aligned}$$

where Eq. (5.1) follows from the fact that $B \geq 1/\epsilon^{1/d} - 1$ (and thus $1 \leq (1+B)^d \cdot \epsilon$). ∎

Lemma 5.2 and Lemma 5.4, together with the assumption that $\mathcal{R}$ is a d-moment (Δ,ω)-hard relation imply that either $\epsilon < 2 \cdot \left((q_{\mathsf{H}}^2+1)/|\mathcal{C}_\lambda| + q_{\mathsf{Sign}} \cdot q_{\mathsf{H}}^2 \cdot \delta\right)$ or

$$\begin{aligned}
\frac{B \cdot \epsilon^2}{8 \cdot q_{\mathsf{H}}} &\leq \Pr\left[(x, \mathsf{A}(x)) \in \mathcal{R}\right] \\
&\leq \frac{\Delta \cdot \mathbb{E}\left[\left(\mathsf{T}_{A,\mathsf{KG}(1^\lambda)}\right)^d\right]}{|\mathcal{W}_\lambda|^\omega} \\
&\leq \frac{\Delta \cdot 2(1+B)^d \cdot \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}}\right)^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega} \\
&\leq \frac{\Delta \cdot 2^{d+1} B^d \cdot \left(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}}\right)^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega} \\
&\leq \frac{\Delta \cdot B^d \cdot \left(2(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}})\right)^d \cdot \epsilon}{|\mathcal{W}_\lambda|^\omega}
\end{aligned}$$

Our choice of $B = \lceil 1/\epsilon^{1/d} - 1 \rceil$ guarantees that $B^{d-1} \leq \epsilon^{1-1/d}$, and therefore

$$\epsilon^{2-\frac{1}{d}} \leq \frac{\epsilon}{B^{d-1}} \leq \frac{8 \cdot q_{\mathsf{H}} \cdot \Delta \cdot \left(2(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}})\right)^d}{|\mathcal{W}_\lambda|^\omega}$$

which yields

$$\epsilon \leq \left(\frac{8 \cdot q_{\mathsf{H}} \cdot \Delta \cdot \left(2(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}})\right)^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}}.$$

Therefore, overall we obtain

$$\begin{aligned}
\epsilon &\leq \max\left\{\left(\frac{8 \cdot q_{\mathsf{H}} \cdot \Delta \cdot \left(2(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}})\right)^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}}, 2 \cdot \left(\frac{q_{\mathsf{H}}^2+1}{|\mathcal{C}_\lambda|} + q_{\mathsf{Sign}} \cdot q_{\mathsf{H}}^2 \cdot \delta\right)\right\} \\
&\leq \left(\frac{q_{\mathsf{H}} \cdot \Delta \cdot \left(16(q_{\mathsf{Sign}} \cdot t_{\mathsf{Sim}} + t_F + t_{\mathsf{V}} + t_{\mathsf{W}})\right)^d}{|\mathcal{W}_\lambda|^\omega}\right)^{\frac{d}{2d-1}} + 2 \cdot \left(\frac{q_{\mathsf{H}}^2+1}{|\mathcal{C}_\lambda|} + q_{\mathsf{Sign}} \cdot q_{\mathsf{H}}^2 \cdot \delta\right).
\end{aligned}$$

∎

6 Implications to the Schnorr and Okamoto Schemes

In this section we derive concrete security bounds for the Schnorr identification and signature schemes and for the Okamoto identification and signature schemes based on Corollary 4.2 and Theorem 5.1, assuming the 2-moment hardness of the discrete logarithm problem. In the description of the schemes, we rely on the existence of a group generation algorithm $\mathsf{GroupGen}$, which takes as input the security parameter 1^λ and outputs a description $(\mathbb{G}, p, g)$ of a cyclic group $\mathbb{G}$ of prime order p, where g is a generator of the group. We focus on the typical case where the security parameter $\lambda \in \mathbb{N}$ determines a lower bound on the size of the group and thus $p \geq 2^\lambda$, and we denote by $t_{\mathsf{exp}} = t_{\mathsf{exp}}(\lambda)$ the time required for a single exponentiation in the group $\mathbb{G}$, where $(\mathbb{G}, p, g) \leftarrow \mathsf{GroupGen}(1^\lambda)$. Moreover, we assume for simplicity that the time required for multiplication in $\mathbb{G}$, for sampling elements in $\mathbb{Z}_p$, and for arithmetic computations in $\mathbb{Z}_p$ is subsumed by t_{exp}.

6.1 The Schnorr Identification and Signature Schemes

We start by recalling the definition of the Schnorr identification scheme $\mathcal{ID}_{\mathsf{Schnorr}} = (\mathsf{Gen}, \mathsf{P}_1, \mathsf{P}_2, \mathsf{V}, \mathcal{C})$ which is defined as follows:

$\underline{\mathsf{Gen}(1^\lambda)}$:	$\underline{\mathsf{P}_1(x, w)}$:
1. $(\mathbb{G}, p, g) \leftarrow \mathsf{GroupGen}(1^\lambda)$	1. Parse x as $((\mathbb{G}, p, g), h)$
2. $w \leftarrow \mathbb{Z}_p$	2. $r \leftarrow \mathbb{Z}_p$
3. $x = ((\mathbb{G}, p, g), g^w)$	3. $\alpha = g^r$
4. Output (x, w)	4. $\mathsf{st} = (w, r)$
	5. Output (α, st)
$\underline{\mathsf{V}(x, \alpha, \beta, \gamma)}$:	
1. Parse x as $((\mathbb{G}, p, g), h)$	$\underline{\mathsf{P}_2(\mathsf{st}, \beta)}$:
2. If $\alpha = g^\gamma \cdot h^{-\beta}$ then output 1 and otherwise output 0	1. Parse st as (w, r)
	2. Output $\gamma = w \cdot \beta + r \mod p$

Note that the scheme's challenge space $\mathcal{C} = \mathcal{C}_x$ is $\mathbb{Z}_p$ for any $x = ((\mathbb{G}, p, g), g^w)$ produced by Gen, and that $\mathcal{ID}_{\mathsf{Schnorr}}$ has a challenge space of size $|\mathcal{C}_\lambda| \geq 2^\lambda$ and δ-first message unpredictability for $\delta = \delta(\lambda) = 2^{-\lambda}$. Additionally, the verifier V preforms two exponentiations in the group $\mathbb{G}$ which yields a total running time of $t_{\mathsf{V}} = t_{\mathsf{V}}(\lambda) = 2t_{\mathsf{exp}}(\lambda)$. The following well-known claim establishes the special soundness and simulatability of $\mathcal{ID}_{\mathsf{Schnorr}}$.

Claim 6.1. *$\mathcal{ID}_{\mathsf{Schnorr}}$ is simulatable and has special soundness.*

For completeness, in Appendix A.3 we present the simulator Sim establishing the simulatability of the scheme, and the extractor $\mathsf{WitnessExt}$ which establishes its special soundness. The simulator Sim runs in time $t_{\mathsf{Sim}} = 2t_{\mathsf{exp}}$, and the extractor $\mathsf{WitnessExt}$ performs only arithmetic operations in the ring $\mathbb{Z}_p$, and hence for our purposes its running time is dominated by that of the other algorithms under consideration. Given Claim 6.1 and the above observations, we obtain the following theorem, establishing concrete security bounds for the Schnorr identification scheme, as an immediate implication of Corollary 4.2.

Theorem 6.2. *Let $t_{\bar{\mathsf{P}}} = t_{\bar{\mathsf{P}}}(\lambda)$ and $q_{\bar{\mathsf{P}}} = q_{\bar{\mathsf{P}}}(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. If the discrete logarithm problem is 2-moment hard with respect to Gen, then for any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ and issues $q_{\bar{\mathsf{P}}}$ transcript-generation queries it holds that*

$$\mathbf{Adv}^{\mathsf{PA\text{-}IMP}}_{\mathcal{ID}_{\mathsf{Schnorr}}, \bar{\mathsf{P}}}(\lambda) \leq \left(\frac{(16(t_{\bar{\mathsf{P}}} + 2(q_{\bar{\mathsf{P}}} + 1) \cdot t_{\mathrm{exp}})^2}{2^\lambda} \right)^{\frac{2}{3}} + \frac{2}{2^\lambda},$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Recall that Schnorr signatures are obtained from $\mathcal{ID}_{\mathsf{Schnorr}}$ via the Fiat-Shamir transform relative to hash function H, as described in Section 5. Hence, we obtain the following theorem, establishing concrete security bounds for the Schnorr signature scheme, as a corollary of Theorem 5.1.

Theorem 6.3. *Let* $t_F = t_F(\lambda)$, $q_{\mathsf{H}} = q_{\mathsf{H}}(\lambda)$ *and* $q_{\mathsf{Sign}} = q_{\mathsf{Sign}}(\lambda)$ *be functions of the security parameter* $\lambda \in \mathbb{N}$. *If the discrete logarithm problem is* 2*-moment hard with respect to* Gen*, and the hash function* H *is modeled as a random oracle, then for every* t_F*-time algorithm* F *that issues* q_{H} *oracle queries and* q_{Sign} *signing queries it holds that*

$$\mathbf{Adv}^{\mathsf{Forge}}_{\mathcal{SIG}_{\mathcal{ID}_{\mathsf{Schnorr}},\mathsf{H},F}}(\lambda) \leq \left(\frac{q_{\mathsf{H}} \cdot (16(t_F + 2(q_{\mathsf{Sign}}+1)\cdot t_{\mathsf{exp}}))^2}{2^\lambda}\right)^{\frac{2}{3}} + 2 \cdot \left(\frac{(q_{\mathsf{Sign}}+1)\cdot q_{\mathsf{H}}^2 + 1}{2^\lambda}\right)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

6.2 The Okamoto Identification and Signature Schemes

The Okamoto identification scheme $\mathcal{ID}_{\mathsf{Okamoto}}$ is defined as follows:

$\underline{\mathsf{Gen}(1^\lambda)}$:	$\underline{\mathsf{P}_1(x,w)}$:
1. $(\mathbb{G},p,g) \leftarrow \mathsf{GroupGen}(1^\lambda)$	1. Parse x as $((\mathbb{G},p,g),g_2,h)$
2. $g_2 \leftarrow \mathbb{G}$	2. $r_1, r_2 \leftarrow \mathbb{Z}_p$
3. $w_1, w_2 \leftarrow \mathbb{Z}_p$	3. $\alpha = g^r \cdot g_2^{r_2}$
4. $w = (w_1, w_2)$	4. $\mathsf{st} = (w, r_1, r_2)$
5. $x = ((\mathbb{G},p,g), g_2, g^{w_1}\cdot g_2^{w_2})$	5. Output (α, st)
6. Output (x,w)	
	$\underline{\mathsf{P}_2(\mathsf{st},\beta)}$:
$\underline{\mathsf{V}(x,\alpha,\beta,\gamma)}$:	1. Parse st as (w_1,w_2,r_1,r_2)
1. Parse x as $((\mathbb{G},p,g),g_2,h)$ and γ as (γ_1,γ_2)	2. $\gamma_i = w_i \cdot \beta + r_i \mod p$ for $i \in \{1,2\}$
2. If $\alpha = g^{\gamma_1}\cdot g_2^{\gamma_2}\cdot h^{-\beta}$ then output 1 and otherwise output 0	3. Output $\gamma = (\gamma_1,\gamma_2)$

Observe that the scheme's challenge space $\mathcal{C} = \mathcal{C}_x$ is $\mathbb{Z}_p$ for any $x = ((\mathbb{G},p,g),g^w)$ produced by Gen, and that $\mathcal{ID}_{\mathsf{Okamoto}}$ has a challenge space of size $|\mathcal{C}_\lambda| \geq 2^\lambda$ and δ-first message unpredictability for $\delta = \delta(\lambda) = 2^{-\lambda}$. Moreover, the verifier V preforms three exponentiations in the group $\mathbb{G}$ which yields a total running time of $t_{\mathsf{V}} = t_{\mathsf{V}}(\lambda) = 3t_{\mathsf{exp}}(\lambda)$.

Note that the instance-witness relation induced by Gen consists of all pairs of the form $((\mathbb{G},p,g_1,g_2,h),(w_1,w_2))$ for which $h = g_1^{w_1}\cdot g_2^{w_2}$. We denote this relation by $\mathcal{R}_{\mathsf{2DLog}}$. The following claim establishes the special soundness (with respect to the relation $\mathcal{R}_{\mathsf{2Dlog}}$) and simulatability of $\mathcal{ID}_{\mathsf{Okamoto}}$.

Claim 6.4. $\mathcal{ID}_{\mathsf{Okamoto}}$ *is simulatable and has special soundness.*

For completeness, in Appendix A.4 we present the simulator Sim establishing the simulatability of the scheme, and the extractor $\mathsf{WitnessExt}$ which establishes its special soundness. The simulator Sim runs in time $t_{\mathsf{Sim}} = 3t_{\mathsf{exp}}$, and the extractor $\mathsf{WitnessExt}$ performs only arithmetic operations in the ring $\mathbb{Z}_p$, and hence for our purposes its running time is dominated by that of the other algorithms under consideration.

Let $\mathcal{D} = \{\mathcal{D}_\lambda\}_{\lambda\in\mathbb{N}}$ be the distribution which outputs pairs of the form $((\mathbb{G},p,g,h),w)$ where $(\mathbb{G},p,g) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $w \leftarrow \mathbb{Z}_p$ and $h = g^w$. It is well-known that the hardness of the relation

$\mathcal{R}_{\mathsf{2DLog}}$ with respect to Gen is tightly implied by the hardness of the discrete logarithm relation with respect to $\mathcal{D}$. That is, for any algorithm A there exists an algorithm B such that $\mathsf{T}_{A,\mathsf{Gen}}$ and $\mathsf{T}_{B,\mathcal{D}}$ are identically distributed[7] and

$$\Pr\left[g^w = h \;\middle|\; \begin{array}{c}(\mathbb{G},p,g) \leftarrow \mathsf{GroupGen}(1^\lambda)\\ h \leftarrow \mathbb{G}\\ w \leftarrow A(\mathbb{G},p,g,h)\end{array}\right] = \Pr\left[g^{w_1}\cdot g_2^{w_2} = h \;\middle|\; \begin{array}{c}(\mathbb{G},p,g) \leftarrow \mathsf{GroupGen}(1^\lambda)\\ g_2,h \leftarrow \mathbb{G}\\ (w_1,w_2) \leftarrow B(\mathbb{G},p,g,g_2,h)\end{array}\right].$$

It immediately follows that if the discrete logarithm relation is 2-moment hard, then the $\mathcal{R}_{\mathsf{2DLog}}$ relation is 2-moment $(\Delta = 1, \omega = 1/2)$-hard, where the parameter $\omega = 1/2$ comes from the fact that the witness space $\mathcal{W}_\lambda$ of $\mathcal{R}_{\mathsf{2DLog}}$ is of size p^2 where p is the order of the group. Hence, the following theorem which establishes concrete security bounds for the Okamoto identification scheme follows immediately from Corollary 4.2.

Theorem 6.5. *Let $t_{\bar{\mathsf{P}}} = t_{\bar{\mathsf{P}}}(\lambda)$ and $q_{\bar{\mathsf{P}}} = q_{\bar{\mathsf{P}}}(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. If the discrete logarithm problem is 2-moment hard with respect to Gen, then for any malicious prover $\bar{\mathsf{P}}$ that runs in time $t_{\bar{\mathsf{P}}}$ and issues $q_{\bar{\mathsf{P}}}$ transcript-generation queries it holds that*

$$\mathbf{Adv}^{\mathsf{PA\text{-}IMP}}_{\mathcal{ID}_{\mathsf{Okamoto}},\bar{\mathsf{P}}}(\lambda) \leq \left(\frac{(16(t_{\bar{\mathsf{P}}} + 3(q_{\bar{\mathsf{P}}}+1)\cdot t_{\exp})^2}{2^\lambda}\right)^{\frac{2}{3}} + \frac{2}{2^\lambda},$$

for all sufficiently large $\lambda \in \mathbb{N}$.

The Okamoto signature scheme is obtained from $\mathcal{ID}_{\mathsf{Okamoto}}$ via the Fiat-Shamir transform relative to hash function H, as described in Section 5. Therefore, the following theorem which establishes concrete security bounds for the Okamoto signature scheme, is an immediate corollary of Theorem 5.1.

Theorem 6.6. *Let $t_F = t_F(\lambda)$, $q_{\mathsf{H}} = q_{\mathsf{H}}(\lambda)$ and $q_{\mathsf{Sign}} = q_{\mathsf{Sign}}(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. If the discrete logarithm problem is 2-moment hard with respect to Gen, and the hash function H is modeled as a random oracle, then for every t_F-time algorithm F that issues q_{H} oracle queries and q_{Sign} signing queries it holds that*

$$\mathbf{Adv}^{\mathsf{Forge}}_{\mathcal{SIG}_{\mathcal{ID}_{\mathsf{Okamoto}},\mathsf{H},F}}(\lambda) \leq \left(\frac{q_{\mathsf{H}}\cdot(16(t_F + 3(q_{\mathsf{Sign}}+1)\cdot t_{\mathsf{exp}}))^2}{2^\lambda}\right)^{\frac{2}{3}} + 2\cdot\left(\frac{(q_{\mathsf{Sign}}+1)\cdot q_{\mathsf{H}}^2 + 1}{2^\lambda}\right)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

References

[AAB+02] M. Abdalla, J. H. An, M. Bellare, and C. Namprempre. From identification to signatures via the Fiat-Shamir transform: Minimizing assumptions for security and forward-security. In *Advances in Cryptology – EUROCRYPT '02*, pages 418–433, 2002.

[BCC+16] J. Bootle, A. Cerulli, P. Chaidos, J. Groth, and C. Petit. Efficient zero-knowledge arguments for arithmetic circuits in the discrete log setting. In *Advances in Cryptology – EUROCRYPT '16*, pages 327–357, 2016.

[7] To be precise, the running time $\mathsf{T}_{B,\mathcal{D}}$ of B is distributed as $\mathsf{T}_{A,\mathsf{Gen}} + 2t_{\mathsf{exp}}$, since B performs two exponentiations and invokes A once. For simplicity of presentation, we assume that the term $2t_{\mathsf{exp}}$ is subsumed by $\mathsf{T}_{A,\mathsf{Gen}}$.

[BD20] M. Bellare and W. Dai. The multi-base discrete logarithm problem: Tight reductions and non-rewinding proofs for Schnorr identification and signatures. In *Progress in Cryptology – INDOCRYPT '20*, pages 529–552, 2020.

[BN06] M. Bellare and G. Neven. Multi-signatures in the plain public-key model and a general forking lemma. In *Proceedings of the ACM Conference on Computer and Communications Security*, pages 390–399, 2006.

[FJS14] N. Fleischhacker, T. Jager, and D. Schröder. On tight security proofs for Schnorr signatures. In *Advances in Cryptology – ASIACRYPT '14*, pages 512–531, 2014.

[FKL18] G. Fuchsbauer, E. Kiltz, and J. Loss. The algebraic group model and its applications. In *Advances in Cryptology – CRYPTO '18*, pages 33–62, 2018.

[FPS20] G. Fuchsbauer, A. Plouviez, and Y. Seurin. Blind Schnorr signatures and signed ElGamal encryption in the algebraic group model. In *Advances in Cryptology – EUROCRYPT '20*, pages 63–95, 2020.

[FS86] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *Advances in Cryptology – CRYPTO '86*, pages 186–194, 1986.

[GBL08] S. Garg, R. Bhaskar, and S. V. Lokam. Improved bounds on security reductions for discrete log based signatures. In *Advances in Cryptology – CRYPTO '08*, pages 93–107, 2008.

[Gol04] O. Goldreich. Foundations of Cryptography – Volume 2: Basic Applications. Cambridge University Press, 2004.

[JT20] J. Jaeger and S. Tessaro. Expected-time cryptography: Generic techniques and applications to concrete soundness. In *Proceedings of the 18th Theory of Cryptography Conference*, pages 414–443, 2020.

[KMP16] E. Kiltz, D. Masny, and J. Pan. Optimal security proofs for signatures from identification schemes. In *Advances in Cryptology – CRYPTO '16*, pages 33–61, 2016.

[Oka92] T. Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In *Advances in Cryptology – CRYPTO '92*, pages 31–53, 1992.

[PS00] D. Pointcheval and J. Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13:361–396, 2000.

[PV05] P. Paillier and D. Vergnaud. Discrete-log-based signatures may not be equivalent to discrete log. In *Advances in Cryptology – ASIACRYPT '05*, pages 1–20, 2005.

[Sch89] C. Schnorr. Efficient identification and signatures for smart cards. In *Advances in Cryptology – CRYPTO '89*, pages 239–252, 1989.

[Sch91] C. Schnorr. Efficient signature generation by smart cards. *Journal of Cryptology*, 4(3):161–174, 1991.

[Seu12] Y. Seurin. On the exact security of Schnorr-type signatures in the random oracle model. In *Advances in Cryptology – EUROCRYPT '12*, pages 554–571, 2012.

[Sho97] V. Shoup. Lower bounds for discrete logarithms and related problems. In *Advances in Cryptology – EUROCRYPT '97*, pages 256–266, 1997.

A Additional Proofs

A.1 Proof of Claim 4.4

Consider the functions $f, g : [0,1] \to \mathbb{R}$ defined as follows:

$$f(z) = z \cdot (1 - (1-z)^B)$$
$$g(z) = \begin{cases} \frac{1}{2} \cdot B \cdot z^2 & 0 \le z < \frac{1}{B} \\ z - \frac{1}{2 \cdot B} & \text{otherwise} \end{cases}$$

Note that our goal is to lower bound $\mathbb{E}_{\mathsf{st}}\left[f(\widetilde{\epsilon}(\mathsf{st}))\right]$. We will do so in two steps: First we will show that it is sufficient to lower bound $\mathbb{E}_{\mathsf{st}}\left[g(\widetilde{\epsilon}(\mathsf{st}))\right]$, and then we will prove such a bound.

Claim A.1. *$g(z) \le f(z)$ for all $z \in [0,1]$.*

Proof. Using Taylor approximation we obtain

$$(1-z)^B \le 1 - B \cdot z + \frac{1}{2} \cdot B^2 \cdot z^2.$$

Therefore,

$$\begin{aligned} f(z) &\ge z \cdot \left(B \cdot z - \frac{1}{2} \cdot B^2 \cdot z^2 \right) \\ &= B \cdot z^2 \left(1 - \frac{1}{2} \cdot B \cdot z \right). \end{aligned}$$

Hence, for all $z < 1/B$ it holds that

$$\begin{aligned} f(z) &\ge B \cdot z^2 \left(1 - \frac{1}{2} \cdot B \cdot \frac{1}{B} \right) \\ &= \frac{1}{2} \cdot B \cdot z^2 \\ &= g(z). \end{aligned}$$

We differentiate f and observe that for all $1 \ge z \ge 1/B$ it holds that

$$\begin{aligned} f'(z) &= 1 - (1-z)^B + B \cdot z \cdot (1-z)^{B-1} \\ &= 1 + (1-z)^{B-1} \cdot ((B+1) \cdot z - 1) \\ &\ge 1 + (1-z)^{B-1} \cdot \left(\frac{B+1}{B} - 1 \right) \\ &\ge 1 \\ &= g'(z). \end{aligned}$$

Since $f(1/B) \ge g(1/B)$, this implies that $g(z) \le f(z)$ for all $z \in [1/B, 1]$ and the claim follows. ■

Claim A.1 implies that

$$\mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st}) \cdot \left(1 - (1 - \widetilde{\epsilon}(\mathsf{st}))^B \right) \right] = \mathbb{E}_{\mathsf{st}}\left[f(\widetilde{\epsilon}(\mathsf{st})) \right] \ge \mathbb{E}_{\mathsf{st}}\left[g(\widetilde{\epsilon}(\mathsf{st})) \right].$$

The derivative of g is

$$g'(z) = \begin{cases} B \cdot z & 0 \le z < \frac{1}{B} \\ 1 & \text{otherwise} \end{cases}$$

which is a non-decreasing function on $[0,1]$ and hence g is convex in this interval. Therefore, Jensen's inequality implies that

$$\mathbb{E}_{\mathsf{st}}\left[g(\widetilde{\epsilon}(\mathsf{st}))\right] \geq g\left(\mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st})\right]\right) \geq g\left(\max\left\{0, \epsilon - \frac{1}{|\mathcal{C}_\lambda|}\right\}\right).$$

Now, since $\epsilon - 1/|\mathcal{C}_\lambda| < \epsilon \leq 1/B$, we obtain

$$\mathbb{E}_{\mathsf{st}}\left[\widetilde{\epsilon}(\mathsf{st}) \cdot \left(1 - (1 - \widetilde{\epsilon}(\mathsf{st}))^B\right)\right] \geq \frac{1}{2} \cdot B \cdot \left(\epsilon - \frac{1}{|\mathcal{C}_\lambda|}\right)^2.$$

■

A.2 Proof of Claim 5.3

Consider the functions $f, g : [0,1] \to \mathbb{R}$ defined as follows:

$$f(z) = z \cdot (1 - (1-z)^B)$$
$$g(z) = \begin{cases} \frac{1}{2} \cdot B \cdot z^2 & 0 \leq z < \frac{1}{B} \\ z - \frac{1}{2 \cdot B} & otherwise \end{cases}$$

As in the proof of Claim 4.4, it holds that

$$\begin{aligned} \mathbb{E}\left[\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1]) \cdot \left(1 - (1 - \widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1]))^B\right)\right] &= \mathbb{E}\left[f\left(\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1])\right)\right] \\ &\geq \mathbb{E}\left[g(\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1]))\right] \\ &\geq g\left(\mathbb{E}\left[\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1])\right]\right). \end{aligned}$$

Recall that for each $i \in [q_{\mathsf{H}}]$, we use the notation $\epsilon_i = \Pr\left[\mathsf{V}(x, m_0, \alpha_0, \beta_0, \gamma_0) = 1 \ \wedge \ I_0 = i\right]$ and $\widetilde{\epsilon}_i = \mathbb{E}\left[\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1])\right]$. By definition of $\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1])$, it holds that

$$\begin{aligned} \widetilde{\epsilon}_i &\geq \max\left\{0, \Pr\left[\mathsf{V}(\mathsf{vk}, m_0, \alpha_0, \beta_0, \gamma_0) = 1 \ \wedge \ I_0 = i\right] - \frac{q_{\mathsf{H}}}{|\mathcal{C}_\lambda|} - q_{\mathsf{Sign}} \cdot q_{\mathsf{H}} \cdot \delta\right\} \\ &= \max\left\{0, \epsilon_i - \frac{q_{\mathsf{H}}}{|\mathcal{C}_\lambda|} - q_{\mathsf{Sign}} \cdot q_{\mathsf{H}} \cdot \delta\right\}. \end{aligned}$$

Since for every $i \in [q_{\mathsf{H}}]$ it holds that $\epsilon_i \leq \epsilon \leq 1/B$ we obtain $\widetilde{\epsilon}_i \leq \epsilon_i \leq 1/B$. Hence, by the definition of g we obtain

$$\begin{aligned} \mathbb{E}\left[\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1]) \cdot \left(1 - (1 - \widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1]))^B\right)\right] &\geq g\left(\mathbb{E}\left[\widetilde{\epsilon}_i(x, r, \vec{\tau}, \vec{y}[i-1])\right]\right) \\ &= \frac{1}{2} \cdot B \cdot \widetilde{\epsilon}_i^2. \end{aligned}$$

■

A.3 Proof of Claim 6.1

We start by proving that $\mathcal{ID}_{\mathsf{Schnorr}}$ is simulatable. Consider the algorithm Sim which takes as input a pair $(1^\lambda, x)$ and is defined as follows:

1. Parse x as $((\mathbb{G}, g, p), h)$.
2. Sample $\beta, \gamma \leftarrow \mathbb{Z}_p$.

3. Compute $\alpha = g^{\gamma} \cdot h^{-\beta}$.
4. Output (α, β, γ).

Observe that distribution $\{(x, \mathsf{Sim}(1^\lambda, x))\}_{\lambda \in \mathbb{N}}$ is identical to the distribution $\{(x, (\alpha', \beta', \gamma'))\}_{\lambda \in \mathbb{N}}$ where $(x = ((\mathbb{G}, g, p), h), w) \leftarrow \mathsf{Gen}(1^\lambda)$, $(\alpha', \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$, $\beta' \leftarrow \mathbb{Z}_p$ and $\gamma' \leftarrow \mathsf{P}_2(\mathsf{st}, \beta)$.

Per the special soundness property, consider the algorithm $\mathsf{WitnessExt}$ which takes as input an instance $x = ((\mathbb{G}, g, p), h)$ and a pair (α, β, γ) and $(\alpha, \beta', \gamma')$ of accepting transcripts such that $\beta' \neq \beta$ and is defined as follows:

1. Parse x as $((\mathbb{G}, g, p), h)$.
2. Output $w^* = (\gamma - \gamma')/(\beta - \beta')$.

Since the two transcripts are accepting, we have $g^{\gamma} \cdot h^{-\beta} = \alpha = g^{\gamma'} \cdot h^{-\beta'}$. This implies that $h = g^{(\gamma - \gamma')/(\beta - \beta')} = g^{w^*}$, implying that w^* is the discrete logarithm of h with respect to g. ■

A.4 Proof of Claim 6.4

We start by proving that $\mathcal{ID}_{\mathsf{Okamoto}}$ is simulatable. Consider the algorithm Sim which takes as input a pair $(1^\lambda, x)$ and is defined as follows:

1. Parse x as $((\mathbb{G}, g, p), g_2, h)$.
2. Sample $\beta, \gamma_1, \gamma_2 \leftarrow \mathbb{Z}_p$.
3. Compute $\alpha = g^{\gamma_1} \cdot g_2^{\gamma_2} \cdot h^{-\beta}$.
4. Output (α, β, γ), where $\gamma = (\gamma_1, \gamma_2)$.

Observe that distribution $\{(x, \mathsf{Sim}(1^\lambda, x))\}_{\lambda \in \mathbb{N}}$ is identical to the distribution $\{(x, (\alpha', \beta', \gamma'))\}_{\lambda \in \mathbb{N}}$ where $(x = ((\mathbb{G}, g, p), g_2, h), w) \leftarrow \mathsf{Gen}(1^\lambda)$, $(\alpha', \mathsf{st}) \leftarrow \mathsf{P}_1(x, w)$, $\beta' \leftarrow \mathbb{Z}_p$ and $\gamma' \leftarrow \mathsf{P}_2(\mathsf{st}, \beta)$.

Per the special soundness property, consider the algorithm $\mathsf{WitnessExt}$ which takes as input an instance $x = ((\mathbb{G}, g, p), g_2 h)$ and a pair (α, β, γ) and $(\alpha, \beta', \gamma')$ of accepting transcripts such that $\beta' \neq \beta$ and is defined as follows:

1. Parse x as $((\mathbb{G}, g, p), g_2, h)$, γ as (γ_1, γ_2) and γ' as (γ_1', γ_2').
2. Compute $w_i^* = (\gamma_i - \gamma_i')/(\beta - \beta')$ for $i \in \{1, 2\}$.
3. Output $w^* = (w_1^*, w_2^*)$.

Since the two transcripts are accepting, we have $g^{\gamma_1} \cdot g_2^{\gamma_2} \cdot h^{-\beta} = \alpha = g^{\gamma_1'} \cdot g_2^{\gamma_2'} \cdot h^{-\beta'}$. This implies that $h = g^{(\gamma_1 - \gamma_1')/(\beta - \beta')} \cdot g_2^{(\gamma_2 - \gamma_2')/(\beta - \beta')} = g^{w_1^*} \cdot g_2^{w_2^*}$, implying that indeed $(x, w^*) \in \mathcal{R}_{\mathsf{2DLOG}}$. ■