A Revision of CROSS Security: Proofs and Attacks for Multi-Round Fiat-Shamir Signatures

Michele Battagliola[1], Riccardo Longo[2], Federico Pintore[3], Edoardo Signorini[4], and Giovanni Tognolini[3]

[1] Università Politecnica delle Marche, Ancona, Italy
[2] Fondazione Bruno Kessler, Center for Cybersecurity, Trento, Italy
[3] Università di Trento, Trento, Italy
[4] Telsy, Turin, Italy

Abstract. Signature schemes from multi-round interactive proofs are becoming increasingly relevant in post-quantum cryptography. A prominent example is CROSS, recently admitted to the second round of the NIST on-ramp standardisation process for post-quantum digital signatures. While the security of these constructions relies on the Fiat-Shamir transform, in the case of CROSS the use of the fixed-weight parallel-repetition optimisation makes the security analysis fuzzier than usual. A recent work has shown that the fixed-weight parallel repetition of a multi-round interactive proof is still knowledge sound, but no matching result appears to be known for the non-interactive version.
In this paper we provide two main results. First, we explicitly prove the EUF-CMA security of CROSS, filling a gap in the literature. We do this by showing that, in general, the Fiat-Shamir transform of an HVZK and knowledge-sound multi-round interactive proof is EUF-CMA secure. Second, we present a novel forgery attack on signatures obtained from fixed-weight repetitions of 5-round interactive proofs, substantially improving upon a previous attack on parallel repetitions due to Kales and Zaverucha. Our new attack has particular relevance for CROSS, as it shows that several parameter sets achieve a significantly lower security level than claimed, with reductions up to 24% in the worst case.

1 Introduction

Post-Quantum Digital Signatures. The need to identify quantum-resistant alternatives to existing public-key schemes has triggered the design of many new cryptosystems [32], including digital signatures. While the first NIST standards covering key encapsulation and signatures have already been made public, the situation with the latter is not considered fully satisfactory, so NIST has launched an "on-ramp" process to standardise new signature schemes [30], with the second round candidates announced in October 2024 [31]. Out of the fourteen selected signatures, more than half of them are based on the Fiat-Shamir heuristic [1, 3, 4, 5, 6, 10, 11, 19, 23]. Among them, CROSS stands out as a promising code-based alternative to the standardised SPHINCS$^+$ [26]. In fact, CROSS enjoys noticeably smaller signature sizes, small public keys, and appears to be easily tunable for different security levels and efficiency targets.

Fiat-Shamir Transform. Introduced by Fiat and Shamir in [24], the Fiat-Shamir transform allows to turn any public-coin[5] interactive proof into a digital signature. Informally, the Fiat-Shamir transform replaces random challenges sent by the verifier with outputs of some hash functions, thus removing the need for interaction. A Fiat-Shamir digital signature *inherits* the main security properties of the starting interactive proof, which almost always enjoys special soundness. In particular, the security of the resulting signature relates to the knowledge error determined by the special soundness, albeit with a notoriously loose reduction [33].

Many interactive proofs, with notable exceptions like the one at the base of SQISign [22], only have very small (often binary) challenge spaces, which results in a big knowledge error. A common way to reduce it is by performing t parallel repetitions of the base interactive proof.

The Fixed-Weight Optimisation. Parallel repetitions have, however, a significant impact on signature size. To mitigate this, a widespread optimisation consists in using *fixed-weight challenge vectors*: when different challenges have significantly different response sizes, the idea is to "maximise" the number of challenges having a short response, i.e. choosing a special challenge $\tilde{c}$ which appears a specific number of times, leading to a response of fixed and "minimal" size. In order to preserve security, when using this technique it is necessary to increase the number of repetitions compared to plain parallel repetition, but the tradeoff with signature size is typically very favourable. This optimisation has been successfully used in many signature schemes, such as [10, 12, 17, 18, 21, 25, 34].

The security of this solution is well understood in the case of a 3-round, public-coin, 2-special-sound interactive proof, and only recently it was proven that the fixed-weight optimisation preserves knowledge soundness in the general case of multi-round interactive proofs [13]. This is exactly the case for CROSS, which is derived from a fixed-weight repetition of 5-round interactive proof.

For the non-interactive case, the Fiat-Shamir transform applied to a parallel repetition of a $(k_1, \ldots, k_\mu)$-special-sound interactive proof was recently analysed in [9]. However, for fixed-weight parallel repetitions, the picture is much fuzzier and less explored, to the extent that the choice of the original parameters of CROSS were based on an attack adapted from a result by Kales and Zaverucha [28] for plain parallel repetitions.

In light of this, the following questions naturally arise:

> *Is CROSS EUF-CMA secure? Is there a way to improve the attack from Kales and Zaverucha by exploiting the extra structure of fixed-weight parallel repetitions?*

Our Contribution. We positively answer the two questions above by explicitly proving that CROSS is EUF-CMA secure and by presenting a novel forgery attack on it. For the former, we prove a more general result, i.e. that the Fiat-Shamir transform of any interactive proof having negligible knowledge error yields an EUF-CMA signature, with a security loss of at most $\binom{Q}{\mu}$, where

[5] All verifier's random choices are made public.

$2\mu + 1$ is the number of rounds and Q is the number of signature queries the adversary is allowed to do. Thanks to the results in [13], this has as a direct consequence the fact that signatures obtained from fixed-weight parallel repetitions of $(k_1, \ldots, k_\mu)$-special-sound interactive proofs are EUF-CMA secure, thus formally proving the security of CROSS.

For the novel forgery attack against CROSS that we present, we substantially improve the forgery technique by Kales and Zaverucha [28] for signatures based on $q2$-identification schemes, i.e. $(2, 2)$-special-sound 5-pass interactive proofs having the first challenge space of cardinality q and the second one of cardinality 2. The original attack exploits the fact that, in this context, the second challenge can be repeatedly guessed without modifying the commitment and the first-round challenge. In CROSS specifications [10] this strategy is adapted to exploit the fixed-weight distribution of the second challenge. However, we show that, when the distribution of the second challenge is highly unbalanced, sticking to the fixed weight when guessing the occurrences of the special challenge $\tilde{c}$ is not optimal. In fact, a better result can be achieved by using a slightly higher weight when guessing the challenge. This might be counterintuitive, but actually aligns with the optimal cheating probability derived for the interactive protocol in [13].

Applying the newly proposed attack to the CROSS scheme and its parameter sets submitted to the first round of the NIST "on-ramp" process, we find a significant reduction in the security of two of the three versions provided for each security level, with a security loss of up to 24% compared to what is claimed. Specifically, both the "balanced" and "small" variants employ highly unbalanced distributions in their second challenge, making them particularly susceptible to our attack. The "fast" variant, however, maintains its original security target. Fixed-weight parameters should therefore be re-chosen to reach the required security level. This should mitigate the security impact of our result, as the underlying hard problems have not been affected in any way by our work. In Table 1 it is reported the computational complexity of our attack for the affected parameter sets, demonstrating practical implications for the security of CROSS.

Organisation. In Section 2 we provide some preliminaries and definitions on interactive proofs, digital signatures and the Fiat-Shamir transform. Next, in Section 3 we prove our main result, showing the EUF-CMA security of any signature obtained by applying the Fiat-Shamir transform to an interactive-proof having negligible knowledge error. Then, in Section 4 we describe our forgery attack, that improves the one by Kales and Zaverucha [28], showing that CROSS's security is lower than claimed. Finally, in Section 5 we draw some conclusions and propose some additional research directions.

2 Preliminaries

Notation. We denote by $\mathbb{N}^*$ the set of non-zero natural numbers. For a finite set X, we write $|X|$ for the cardinality of X. We denote by $\{0,1\}^*$ the set of binary strings of arbitrary length.

Table 1. An overview of the cost of our forgery attack compared to the attack considered in [10] for choosing the first-round parameters for the "balanced" and "small" parameter sets of CROSS. Complexities are given as $\log_2$ of the estimated gate count.

Parameter Set		Known Forgery	Our Forgery	Loss
CROSS-R-SDP 1	balanced	128.01	**120.46**	**6%**
	small	128.00	**97.48**	**24%**
CROSS-R-SDP 3	balanced	192.07	**179.67**	**6%**
	small	192.02	**156.37**	**19%**
CROSS-R-SDP 5	balanced	256.01	**240.82**	**6%**
	small	255.22	**217.15**	**15%**
CROSS-R-SDP (G) 1	balanced	128.13	**122.72**	**4%**
	small	128.01	**108.22**	**15%**
CROSS-R-SDP (G) 3	balanced	192.03	189.83	1%
	small	192.03	**167.56**	**13%**
CROSS-R-SDP (G) 5	balanced	256.08	252.70	1%
	small	256.03	**228.58**	**11%**

Where not otherwise specified, each algorithm is probabilistic polynomial-time (PPT). For a deterministic algorithm A, we write $y \leftarrow \mathsf{A}(x)$ to denote the assignment to y of the output of A on input x. If A is probabilistic, we write $y \leftarrow\!\$\ \mathsf{A}(x)$. To make the algorithm's use of random coins r explicit, we write $\mathsf{A}(x; r)$. In a pseudocode, each variable assignment is done by either deterministic assignment ($\leftarrow$) or probabilistic assignment ($\leftarrow\!\$$), while the symbol = is reserved for equality testing. Furthermore, we use the symbol $\perp$ to denote a failure, e.g. $\perp \leftarrow \mathsf{A}(x)$.

For a set S, we write $s \leftarrow\!\$\ S$ to denote sampling from the uniform distribution over S.

For an adversary $\mathcal{A}$ and an arbitrary function F, we write $\mathcal{A}^{\mathsf{F}}$ (resp., $\mathcal{A}^{\mathsf{OF}}$) to denote the execution of $\mathcal{A}$ with access (resp., with oracle access) to F.

2.1 Multi-Round Interactive Proofs

Definition 1 (Binary relation). *A* binary relation *is a finite set $R \subseteq X \times Y$, where $X, Y \subseteq \{0,1\}^*$. Given $(x, y) \in R$, we say that y is a* witness *for the* statement *x. The set $L_R = \{x \in X \mid \exists y \in Y \text{ s.t. } (x, y) \in R\}$ is called the set of* true statements *for R, or its* language.

For a binary relation R we can (informally) define Experiment 1 against an adversary $\mathcal{A}$.[6]

[6] To be more precise, the game should depend on the parameter λ, and we should define R as a family of relations.

Experiment 1: $\mathsf{Exp}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda)$
1: $(x, y) \leftarrow\!\$\ R$
2: $y' \leftarrow\!\$\ \mathcal{A}(x)$
3: **return** $(x, y') \in R$

Definition 2 (Hard Binary Relation). *Let $\mathcal{A}$ be an adversary playing the* hard-relation *experiment $\mathsf{Exp}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda)$ (Experiment 1) against a binary relation R. We define the advantage of $\mathcal{A}$ in the experiment as $\mathbf{Adv}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda) = \Pr\left[\mathsf{Exp}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda) = 1\right]$. We say that R is* hard *if and only if $\mathbf{Adv}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda)$ is negligible in λ for every probabilistic polynomial-time adversary $\mathcal{A}$.*

Definition 3 (Interactive Proof). *An* interactive proof *$(\mathcal{P}, \mathcal{V})$ for a binary relation $R \subseteq X \times Y$ is an interactive protocol between two probabilistic polynomial-time machines $\mathcal{P}$ and $\mathcal{V}$. The* prover *$\mathcal{P}$ takes as input a pair $(x, y) \in R$ while the* verifier *$\mathcal{V}$ takes as input x. At the end of the protocol, $\mathcal{V}$ either accepts (outputs 1) or rejects (outputs 0). We denote the output of the protocol with $(\mathcal{P}(y), \mathcal{V})(x)$. Furthermore, we say that a* transcript*, i.e. the set of all messages exchanged in a protocol execution, is* accepting *(*rejecting*) if $\mathcal{V}$ accepts (rejects, respectively).*

Definition 4 (Public-Coin). *An interactive proof $(\mathcal{P}, \mathcal{V})$ is* public-coin *if all $\mathcal{V}$'s random choices are made public.*

If an interactive proof is public-coin, the verifier needs to send to the prover only their random choices. For this reason, we call *challenges* the messages ch sent by the verifier and *challenge set* the set Ch from which the verifier's messages are sampled. In the case of a $(2\mu + 1)$-round interactive proof, we define the challenge set Ch of the protocol as the Cartesian product of μ *round challenge sets* $\mathsf{Ch}^{[i]}$, with $i \in \{1, \ldots, \mu\}$, meaning that the challenge for the i-th round is sampled from $\mathsf{Ch}^{[i]}$. When $\mu = 1$, and thus the rounds are only 3, we use the name *Sigma protocol*.

Throughout this work we assume that, within an execution of an interactive proof $(\mathcal{P}, \mathcal{V})$, the prover $\mathcal{P}$ always sends the first and the last message. We also consider $\mu + 1$ PPT algorithms associated with the prover, namely $\mathcal{P}_0, \ldots, \mathcal{P}_\mu$, which are assumed to share states. In particular, the prover first sends an initial *commitment* $\mathsf{com} \leftarrow \mathcal{P}_0(y)$ from a suitable space Com, and subsequently undertakes μ mutual exchanges with the verifier. In the i-th exchange, with $i \in \{1, \ldots, \mu\}$, the verifier sends a random *challenge* $\mathsf{ch}^{[i]}$ from a challenge set $\mathsf{Ch}^{[i]}$ to which the prover replies with a *response* $\mathsf{rsp}^{[i]}$ from a response set $\mathsf{Rsp}^{[i]}$. Each response $\mathsf{rsp}^{[i]}$ is obtained as the output of an algorithm $\mathcal{P}_i$ that takes as input the witness y, the commitment com and the previous challenge-response pairs $\{(\mathsf{ch}^{[j]}, \mathsf{rsp}^{[j]})\}_{j=1}^{i-1}$. Note that the number of communication rounds is odd, i.e. of the form $2\mu + 1$ with $\mu \in \mathbb{N}^*$. We refer to an interactive proof having $2\mu + 1$ communication rounds with the name $(2\mu + 1)$*-round interactive proof*. We depict such a protocol in Figure 1.

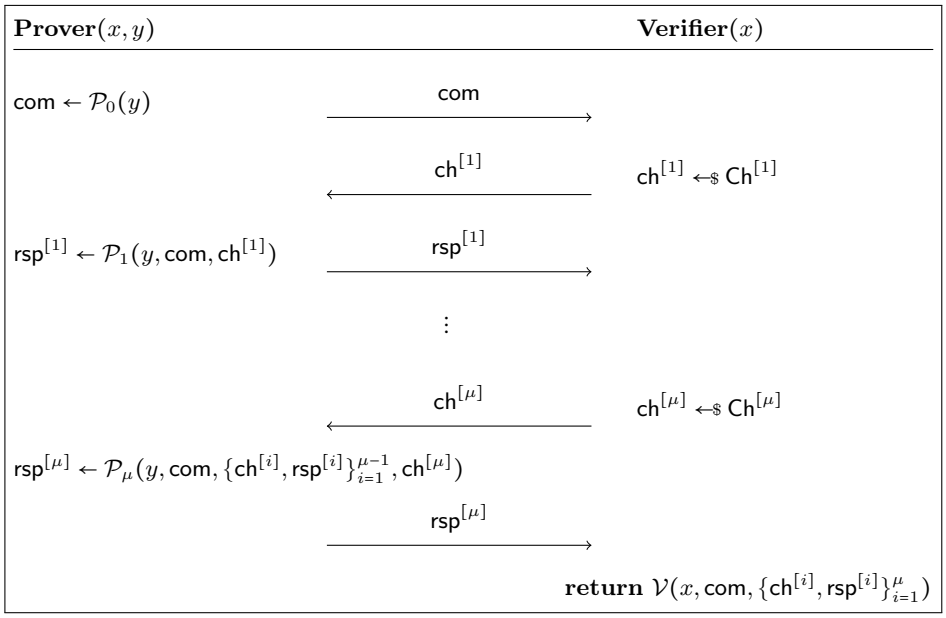

Fig. 1. Public-Coin $(2\mu + 1)$-Round Interactive Proof

Commonly, an interactive proof is required to satisfy completeness and soundness, as per definitions below.

Definition 5 (Completeness). *An interactive proof* $(\mathcal{P}, \mathcal{V})$ *for a binary relation* $R \subseteq X \times Y$ *is* complete *if, for every* $(x, y) \in R$*, we have:*

$$\Pr[(\mathcal{P}(y), \mathcal{V})(x) = 0] \leq \rho(x),$$

where the value $\rho(x)$ *— called* completeness error *— is negligible (in* $|x|$*). If* $\rho(x) = 0$ *for all* $x \in L_R$*, the protocol is said to be* perfectly complete.

Definition 6 (Soundness). *An interactive proof* $(\mathcal{P}, \mathcal{V})$ *for a binary relation* $R \subseteq X \times Y$ *is* sound *if, for every* $x \notin L_R$ *and a (potentially-dishonest) prover* $\mathcal{P}^*$*, we have:*

$$\Pr[(\mathcal{P}^*, \mathcal{V})(x) = 1] \leq \sigma(x),$$

where the value $\sigma(x)$ *— called* soundness error *— is negligible (in* $|x|$*).*

We note that an interactive proof which satisfies both the previous properties allows a prover $\mathcal{P}$ to convince the verifier $\mathcal{V}$ that a statement x is true. To prove $\mathcal{P}$'s knowledge of a witness y such that $(x, y) \in R$, the following stronger feature is required.

Definition 7 (Knowledge Soundness). *An interactive proof* $(\mathcal{P}, \mathcal{V})$ *for a binary relation* $R \subseteq X \times Y$ *is* knowledge sound*, with* knowledge error κ*, if there*

exists an algorithm $\mathcal{E}$ that, given as input any $x \in X$ and rewindable oracle access to a (potentially-dishonest) prover $\mathcal{P}^*$, runs in an expected polynomial time (in $|x|$) and outputs a witness $y \in Y$ for x with probability:

$$\Pr\Big[(x, \mathcal{E}^{\mathcal{P}^*}(x)) \in R\Big] \geq \frac{\varepsilon(x, \mathcal{P}^*) - \kappa(x)}{\mathsf{poly}(|x|)},$$

where $\varepsilon(x, \mathcal{P}^*) = \Pr\big[(\mathcal{P}^*, \mathcal{V})(x) = 1\big]$. The algorithm $\mathcal{E}$ is called knowledge extractor*.*

Definition 8 (Proof of Knowledge). *An interactive proof $(\mathcal{P}, \mathcal{V})$ for a binary relation $R \subseteq X \times Y$ which satisfies both completeness with completeness error ρ and knowledge soundness with knowledge error κ is a* proof of knowledge *if there exists a positive-definite polynomial p over the integers such that $1 - \rho(x) \geq \kappa(x) + \frac{1}{p(|x|)}$ for all $x \in X$.*

A common strategy to prove the knowledge soundness of a public-coin interactive proof is showing that it enjoys special soundness. Informally, this means showing that there is an extracting algorithm which can compute a witness given enough accepting transcripts relative to a true statement x. In the following, we recall the simple concept of k-out-of-N special soundness for Sigma protocols. For the general notion of $(k_1, \ldots, k_\mu)$-out-of-$(N_1, \ldots, N_\mu)$ special soundness for $(2\mu + 1)$-rounds, we refer to the more extensive presentation of [13].

Definition 9 (k-out-of-N Special Soundness). *Let $k, N \in \mathbb{N}^*$. A Sigma protocol $(\mathcal{P}, \mathcal{V})$ for a relation R, with challenge set of cardinality $N \geq k$, is k-out-of-N* special sound *if there exists a polynomial time algorithm that, on input a statement x and k accepting transcripts $(\mathsf{com}, \mathsf{ch}_1, \mathsf{rsp}_1), \ldots, (\mathsf{com}, \mathsf{ch}_k, \mathsf{rsp}_k)$ with common first message com and pairwise distinct challenges $\mathsf{ch}_1, \ldots, \mathsf{ch}_k$, outputs a witness y for x. When N is clear from the context, we also say $(\mathcal{P}, \mathcal{V})$ is k-special sound.*

Definition 10 (Honest-Verifier Zero-Knowledge (HVZK)). *An interactive proof $(\mathcal{P}, \mathcal{V})$ for a binary relation R is* computationally (resp., statistically/perfectly) honest-verifier zero-knowledge (HVZK) *if there exists a PPT algorithm S (called* simulator*), such that for any $x \in L_R$, S produces a transcript which is computationally (resp., statistically/perfectly) indistinguishable from the distribution of the transcripts obtained through the interaction of $\mathcal{P}$ and $\mathcal{V}$.*

Definition 11 (Min-Entropy of Messages). *Let $(\mathcal{P}, \mathcal{V})$ be a proof system for a hard relation $R \subseteq X \times Y$, $(x, y) \in R$ and λ be the security parameter. Let us consider the set of all possible messages associated to y that the prover sends:*

$$M(y) = \left\{\mathcal{P}(x, y; r) \mid r \leftarrow\$ \{0,1\}^\lambda\right\}.$$

We define the maximum probability of a message appearing in any round of $M(y)$ as:

$$\alpha(y) = \max_{a \in M(y), 1 \leq i \leq \mu} \left\{\Pr\big[\mathcal{P}(x, y; r)[i] = a[i] \,\big|\, r \leftarrow\$ \{0,1\}^\lambda\big]\right\},$$

Experiment 2: $\mathsf{Exp}^{\text{EUF-CMA}}_{\mathsf{DS},\mathcal{F}}$

$(\mathsf{pk},\mathsf{sk}) \leftarrow\!\$\ \mathsf{KGen}(\lambda)$
$\mathsf{ST} \leftarrow \varnothing$
$(\mathsf{msg},\sigma) \leftarrow\!\$\ \mathcal{F}^{\mathsf{O},\mathsf{OSign}}(\mathsf{pk})$
if $\mathsf{msg} \in \mathsf{ST}$ **then**
  return 0
return $\mathsf{Vrfy}(\mathsf{pk},\mathsf{msg},\sigma)$

$\mathsf{OSign}(\mathsf{msg})$:
$\sigma \leftarrow\!\$\ \mathsf{Sign}(\mathsf{sk},\mathsf{msg})$
$\mathsf{ST} \leftarrow \mathsf{ST} \cup \{\mathsf{msg}\}$
return σ

where $a[i]$ denotes the message relative to the i-th round of $a \in M(y)$.

Then the min-entropy *function associated to $(\mathcal{P},\mathcal{V})$ is defined as follows:*

$$\beta(\lambda) = \min_{(x,y)\in R} \left\{ \log_2 \frac{1}{\alpha(y)} \right\}.$$

Remark 1. If β is not super-logarithmic in λ, we can consider the generalised randomised version of the Fiat-Shamir transform, introduced in [2, Construction 3.1]: during every round, the prover picks a random s_i of appropriate length such that the min entropy of $a[i] \parallel s_i$ is super-logarithmic and sets $a[i] \leftarrow a[i] \parallel s_i$. From now on, for the sake of readability, we limit our analysis to the case of β being super-logarithmic.

2.2 Digital Signatures

Digital Signature Schemes are cryptosystems used to provide *integrity*, *authenticity* and *non-repudiation* to digital data.

Definition 12 (Digital Signature Scheme). *A* digital signature scheme DS *is defined by a tuple of polynomial-time algorithms* $\mathsf{DS} = (\mathsf{KGen},\mathsf{Sign},\mathsf{Vrfy})$*, where the first two are probabilistic and the third is deterministic. In particular, we have:*

- $(\mathsf{pk},\mathsf{sk}) \leftarrow\!\$\ \mathsf{KGen}(\lambda)$*: on input a security parameter λ, it outputs a public key* pk *and the corresponding secret key* sk*.*
- $\sigma \leftarrow\!\$\ \mathsf{Sign}(\mathsf{sk},\mathsf{msg})$*: on input a private key* sk *and a message* msg*, the algorithm outputs a signature σ.*
- $1/0 \leftarrow \mathsf{Vrfy}(\mathsf{pk},\mathsf{msg},\sigma)$*: it takes as input a public key* pk*, a message* msg *and a signature σ, and outputs 1 (accept) or 0 (reject).*

We ask that an honestly generated signature is always verified, i.e. for every security parameter λ and message msg*,*

$$\Pr\left[\mathsf{Vrfy}(\mathsf{pk},\mathsf{msg},\sigma) = 1 \;\middle|\; \begin{array}{c}(\mathsf{pk},\mathsf{sk}) \leftarrow\!\$\ \mathsf{KGen}(\lambda)\\ \sigma \leftarrow\!\$\ \mathsf{Sign}(\mathsf{sk},\mathsf{msg})\end{array}\right] = 1.$$

The most standard security property for a digital signature scheme is *unforgeability under chosen-message attack*, where an adversary has as many couples message-signature as it wishes and is asked to produce a forgery (i.e. a valid signature without having direct access to the private key).

Definition 13 (Existential unforgeability under chosen-message attack). *Let* $\mathsf{DS} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vrfy})$ *be a digital signature scheme, let* O *be a random oracle and let* $\mathcal{F}$ *be a forger. We define the advantage of* $\mathcal{F}$ *playing the* EUF-CMA *experiment* $\mathsf{Exp}^{\text{EUF-CMA}}_{\mathsf{DS},\mathcal{F}}$ *(Experiment 2) against* DS *in the random oracle model as:*

$$\mathbf{Adv}^{\text{EUF-CMA}}_{\mathsf{DS},\mathcal{F}}(\lambda) = \Pr\left[\mathsf{Exp}^{\text{EUF-CMA}}_{\mathsf{DS},\mathcal{F}}(\lambda) = 1\right].$$

We say that DS *is* existential unforgeable against chosen-message *attacks if* $\mathbf{Adv}^{\text{EUF-CMA}}_{\mathsf{DS},\mathcal{F}}(\lambda)$ *is negligible in* λ *for every probabilistic polynomial-time forger* $\mathcal{F}$*.*

Remark 2. The forger's access to the random oracle O is not strictly required, and its relevance depends on whether the security of the digital signature is given in the Random Oracle Model (ROM) or in the standard model. Since we are interested in digital signatures obtained by applying the Fiat-Shamir transform (see Section 2.3) we included it in the definition.

2.3 The Fiat-Shamir Transform

Firstly introduced in [24], the Fiat-Shamir transform is a widespread heuristic, used to design digital signature schemes starting from public-coin interactive proofs. Intuitively, the idea is to replace the challenge-communication steps with a hash function, evaluated on a suitable input. Formally, we have the following definition:

Definition 14 (Fiat-Shamir Signature). *Let* λ *be a security parameter and let* $(\mathcal{P}, \mathcal{V})$ *be a knowledge sound* $(2\mu+1)$*-round interactive proof system for a hard relation* $R \subseteq X \times Y$ *with knowledge error* $\kappa = \mathsf{negl}(\lambda)$*. Let* $\mathsf{H}_1, \ldots, \mathsf{H}_\mu$ *be cryptographic hash functions with* $\mathsf{H}_i\colon \{0,1\}^* \to \mathsf{Ch}^{[i]}$*. The signature scheme obtained from* $(\mathcal{P}, \mathcal{V})$ *by applying the Fiat-Shamir transform is a triple of algorithms* $\mathsf{FS}[(\mathcal{P}, \mathcal{V})] = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vrfy})$ *as detailed in Algorithm 1.*

It is well-known that for Sigma protocols with high min-entropy, a digital signature obtained with the Fiat-Shamir transform is EUF-CMA secure in the ROM [2, Theorem 3.3].

Remark 3. When proving the security of a signature based on a multi-round interactive proof, each H_i is considered a different Random Oracle. For this reason, in the real implementation, different hash functions should be used. Real implementations, however, sometimes use a single hash function H for all the Random Oracle and One and "separate" the domain, for example adding i_ℓ as a prefix, where i_ℓ denotes the representation of integer i as a bit-string of fixed length ℓ [15]:

$$\mathsf{H}_i(\mathsf{msg}) = \mathsf{H}(i_\ell \parallel \mathsf{msg}).$$

In the following we implicitly assume that all the random oracles queries are done by using domain separation and we limit ourselves to simply denotes the message msg, since the exact H_i can be deduced by the number of couples $(\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]})$ in msg.

Algorithm 1 Fiat-Shamir Transformation of a $(2\mu+1)$-Interactive Proof

$\mathsf{KGen}(\lambda)$:
1: $(x, y) \leftarrow\$ R$
2: $\mathsf{sk} \leftarrow y$
3: $\mathsf{pk} \leftarrow x$
4: **return** $(\mathsf{pk}, \mathsf{sk})$

$\mathsf{Sign}(\mathsf{sk}, \mathsf{msg})$:
1: $\mathsf{com} \leftarrow \mathcal{P}_0(\mathsf{sk})$
2: **for** $i \in \{1, \dots, \mu\}$ **do**
3: $\quad \mathsf{ch}^{[i]} \leftarrow \mathsf{H}_i(\mathsf{com}, \{\mathsf{ch}^{[j]}, \mathsf{rsp}^{[j]}\}_{j=1}^{i-1}, \mathsf{msg})$
4: $\quad \mathsf{rsp}^{[i]} \leftarrow \mathcal{P}_i(\mathsf{sk}, \mathsf{com}, \{\mathsf{ch}^{[j]}, \mathsf{rsp}^{[j]}\}_{j=1}^{i-1}, \mathsf{ch}^{[i]})$
5: **return** $\sigma \leftarrow (\mathsf{com}, \mathsf{rsp}^{[1]}, \dots, \mathsf{rsp}^{[\mu]})$

$\mathsf{Vrfy}(\mathsf{pk}, \mathsf{msg}, \sigma)$:
1: $(\mathsf{com}, \mathsf{rsp}^{[1]}, \dots, \mathsf{rsp}^{[\mu]}) \leftarrow \sigma$
2: **for** $i \in \{1, \dots, \mu\}$ **do**
3: $\quad \mathsf{ch}^{[i]} \leftarrow \mathsf{H}_i(\mathsf{com}, \{\mathsf{ch}^{[j]}, \mathsf{rsp}^{[j]}\}_{j=1}^{i-1}, \mathsf{msg})$
4: **return** $\mathcal{V}(x, \mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{\mu})$

2.4 Parallel Repetition and Fixed-Weight Optimisation

When the knowledge error of a $(2\mu+1)$-round knowledge sound interactive proof $(\mathcal{P}, \mathcal{V})$ is not negligible, a common way to decrease it is to repeat the protocol in parallel multiple times, i.e. the prover and the verifier run t parallel executions of the protocol and the verifier accepts if the resulting t transcripts are all accepting. We denote by $(\mathcal{P}^t, \mathcal{V}^t)$ the t-fold parallel repetition of $(\mathcal{P}, \mathcal{V})$. While this technique has been analysed since the 1990s [14], it was only in 2022 that Attema and Fehr [7] proved that the t-fold parallel repetition of any $(k_1, \dots, k_\mu)$-special-sound $(2\mu+1)$-round public-coin interactive proof optimally reduces the knowledge error from κ down to κ^t.

Protocols obtained by using parallel repetitions usually have the drawback of having big transcripts. When responses to different challenges have very unbalanced sizes and compactness is a bigger concern than computational efficiency, it can be beneficial to use *fixed-weight challenges*, that also may come with the additional feature of making the transcript size constant.

Definition 15 (Weight). *Let* Ch *be a finite set,* $t \in \mathbb{N}^*$ *and* $\tilde{c} \in \mathsf{Ch}$*. For an element* $c = (c_1, \dots, c_t) \in \mathsf{Ch}^t$*, we define the* weight *of* c *with respect to* $\tilde{c}$ *as:*

$$\mathsf{wt}_{\tilde{c}}(c) := |\{j \in \{1, \dots, t\} : c_j = \tilde{c}\}|.$$

Definition 16. *Let* $t, w, \mu \in \mathbb{N}^*$ *such that* $t \geq w$*, let* $\mathsf{Ch}^{[1]}, \dots, \mathsf{Ch}^{[\mu]}$ *be finite sets and let* $\tilde{c} \in \mathsf{Ch}^{[\mu]}$*. Given* $\mathsf{Ch} = \mathsf{Ch}^{[1]} \times \dots \times \mathsf{Ch}^{[\mu]}$*, we denote by* $\mathsf{Ch}_{\tilde{c}}^{t,w}$ *the set of elements* $\mathbf{c} \in \mathsf{Ch}^t$ *for which* $\mathsf{wt}_{\tilde{c}}\left(\mathbf{c}_1^{[\mu]}, \dots, \mathbf{c}_t^{[\mu]}\right) = w$*, i.e.*

$$\mathsf{Ch}_{\tilde{c}}^{t,w} := \left\{\mathbf{c} \in \mathsf{Ch}^t : \mathsf{wt}_{\tilde{c}}\left(\left(\mathbf{c}_1^{[\mu]}, \dots, \mathbf{c}_t^{[\mu]}\right)\right) = w\right\}.$$

When $\tilde{c}$ *is clear from the context, we will simplify the notation and write* $\mathsf{Ch}^{t,w}$ *instead of* $\mathsf{Ch}_{\tilde{c}}^{t,w}$*. Furthermore, when* Ch *is not a Cartesian product but a simple*

set (i.e. $\mu = 1$ and so $\mathsf{Ch} = \mathsf{Ch}^{[\mu]}$), we will simply denote by $\mathsf{Ch}_{\tilde{c}}^{t,w}$ the set:

$$\left\{c = (c_1, \ldots, c_t) \in \mathsf{Ch}^t : \mathsf{wt}_{\tilde{c}}(c) = w\right\}.$$

Definition 17 (Fixed-weight Repetition). *Let $k_1, \ldots, k_\mu$, $N_1, \ldots, N_\mu \in \mathbb{N}^*$, $R \subseteq X \times Y$ be a binary relation and $(\mathcal{P}, \mathcal{V})$ be a $(2\mu+1)$-round public-coin interactive proof for R, where $\mathcal{V}$ samples i-th challenges ($i \in \{1, \ldots, \mu\}$) from a set $\mathsf{Ch}^{[i]}$ of cardinality $N_i \geq k_i$. Therefore, the challenge set of $(\mathcal{P}, \mathcal{V})$ is $\mathsf{Ch} = \prod_{i=1}^{\mu} \mathsf{Ch}^{[i]}$. Let $\tilde{c}$ be a given element of $\mathsf{Ch}^{[\mu]}$. A (t, w)-*fixed-weight parallel repetition *of $(\mathcal{P}, \mathcal{V})$ with respect to $\tilde{c}$, which we denote by $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$, is a t-fold parallel repetition of $(\mathcal{P}, \mathcal{V})$ whose challenge set is $\mathsf{Ch}_{\tilde{c}}^{t,w}$.*

Throughout this work, we will consider fixed-weight repetitions only for $(2\mu+1)$-round public-coin interactive proofs for which there exists a unique element $\tilde{c} \in \mathsf{Ch}^{[\mu]}$ such that, for every possible $\mathbf{c} = (c^{[1]}, \ldots, c^{[\mu]}) \in \mathsf{Ch}^{[1]} \times \cdots \times \mathsf{Ch}^{[\mu]}$, the response size when $c^{[\mu]} = \tilde{c}$ is significantly higher than when $c^{[\mu]} \neq \tilde{c}$. Under this assumption, a fixed-weight repetition can lead to a more compact protocol compared to a plain parallel repetition, as it was shown in [10, 12, 21, 25, 34].

Remark 4. In Definition 17, we consider the fixed element $\tilde{c}$ as an element of $\mathsf{Ch}^{[\mu]}$ rather than of the challenge set of previous rounds or a Cartesian product of (a subset of) them. This is consistent with the concrete instances of the fixed-weight technique that have appeared so far (see the list above).

In [13], it was proven that the fixed-weight optimisation produces a knowledge sound interactive proof for any μ. In particular:

Theorem 1 ([13] Fixed-Weight Repetition of a $(k_1, \ldots, k_\mu)$-Special-Sound Multi-Round Interactive Proof). *Let $(\mathcal{P}, \mathcal{V})$ be a $(k_1, \ldots, k_\mu)$-special-sound $(2\mu+1)$-round interactive proof having challenge sets $\mathsf{Ch}^{[1]}, \ldots, \mathsf{Ch}^{[\mu]}$, where $\mathsf{Ch}^{[i]}$ has cardinality $N_i \geq k_i$. Let $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$ be the (t, w)-fixed-weight repetition of $(\mathcal{P}, \mathcal{V})$, where $w, t \in \mathbb{N}^*$ and $1 \leq w \leq t$. Then $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$ is knowledge sound with knowledge error $\kappa_{t,w}$, where $\kappa_{t,w}$ is the maximum, taken over $\alpha \in \{0, \ldots, t\}$, of the expression:*

$$\frac{\sum_{\ell=\max(0,w-t+\alpha)}^{\min(w,\alpha)} \binom{\alpha}{\ell}\binom{t-\alpha}{w-\ell} Z_0^{\ell} (Z_1 - Z_0)^{\alpha-\ell} (Z_2)^{w-\ell} (Z_1 - Z_2)^{t-\alpha-w+\ell}}{\binom{t}{w}(N_\mu - 1)^{t-w}(\prod_{i=1}^{\mu-1} N_i)^t}, \tag{1}$$

where Z_0, Z_1, Z_2 are defined as follows:

$$\begin{aligned}
Z_0 &:= \prod_{\ell=1}^{\mu-1} N_\ell, \\
Z_1 &:= \sum_{\ell=1}^{\mu} \left(\prod_{j=\ell+1}^{\mu} N_j \right) (k_\ell - 1) \left(\prod_{j=1}^{\ell-1} (N_j - k_j + 1) \right), \\
Z_2 &:= \sum_{\ell=1}^{\mu-1} \left(\prod_{j=\ell+1}^{\mu-1} N_j \right) (k_\ell - 1) \left(\prod_{j=1}^{\ell-1} (N_j - k_j + 1) \right).
\end{aligned}$$

Experiment 3: $\mathsf{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$

$(x, y) \leftarrow\!\$\ R$
$\mathsf{com} \leftarrow\!\$\ \mathcal{I}^{\mathsf{OTrGen}}(x)$
for $i \leftarrow 1, \ldots, \mu$ **do**
  $\mathsf{ch}^{[i]} \leftarrow\!\$\ \mathsf{Ch}^{[i]}$
  $\mathsf{rsp}^{[i]} \leftarrow\!\$\ \mathcal{I}(\mathsf{com}, \{\mathsf{ch}^{[j]}\}_{j\leq i}, \{\mathsf{rsp}^{[j]}\}_{j<i})$
return $\mathcal{V}(x, \mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{\mu})$

$\mathsf{OTrGen}(x)$:
  return $(\mathcal{P}(y), \mathcal{V})(x)$

3 Unforgeability of Fiat-Shamir Signatures from Fixed-Weight Repetitions of Multi-Round Interactive Proofs

In this section, we prove that a digital signature scheme obtained by applying the Fiat-Shamir transform to an interactive proof having negligible knowledge error is EUF-CMA. As a direct consequence, following from the results in [13], we also obtain that the Fiat-Shamir transform of a (t, w)-fixed-weight repetition of a $(k_1, \ldots, k_\mu)$-special-sound protocol is EUF-CMA as long as t, w are chosen such that the knowledge error is negligible.

3.1 Security Proof

We will use the notion of security against impersonation under passive attack, which we recall below.

Definition 18 (Security against impersonation under passive attack). *Let $R \subseteq X \times Y$ be a binary relation, $(\mathcal{P}, \mathcal{V})$ a $(2\mu+1)$-round interactive proof for R and $\mathcal{I}$ an impersonator. We define the advantage of $\mathcal{I}$ playing the experiment $\mathsf{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$ (Experiment 3) against $(\mathcal{P}, \mathcal{V})$ as:*

$$\mathbf{Adv}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda) = \Pr\left[\mathsf{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda) = 1\right].$$

We say that $(\mathcal{P}, \mathcal{V})$ is polynomially-secure against impersonation under passive attack if $\mathbf{Adv}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$ is negligible in λ for every probabilistic polynomial-time impersonator $\mathcal{I}$.

In the following, we prove that any public-coin interactive proof which is knowledge sound (Definition 7) and HVZK (Definition 10) is also polynomially-secure against impersonation under passive attack.

Theorem 2. *Let $(\mathcal{P}, \mathcal{V})$ be a $(2\mu+1)$-round interactive proof for a hard binary relation $R \subseteq X \times Y$ which is HVZK and knowledge sound with negligible knowledge error κ. Let $\mathcal{I}$ be an impersonator against $(\mathcal{P}, \mathcal{V})$ (Definition 18). Then there exists an adversary $\mathcal{A}$ against the hard binary relation R such that:*

$$\mathbf{Adv}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda) \leq \mathsf{poly}(|x|) \cdot \mathbf{Adv}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda) + \kappa,$$

and the expected running time of $\mathcal{A}$ is approximately that of $\mathcal{I}$.

Proof. Consider the impersonation experiment $\mathsf{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$ of Definition 18, and an impersonator $\mathcal{I}$. We now show a hard-relation adversary $\mathcal{A}$ who exploits $\mathcal{I}$ as a subroutine — perfectly simulating $\mathsf{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$ — in order to build a dishonest prover $\mathcal{P}^*$ against $(\mathcal{P},\mathcal{V})$.

At the start of the hard-relation experiment (Experiment 1), $\mathcal{A}$ receives a statement $x \in X$ and forwards it to $\mathcal{I}$. The transcript oracle OTrGen is simulated by running the HVZK simulator S of $(\mathcal{P},\mathcal{V})$. Transcripts produced in this way and those produced by OTrGen are identically distributed by definition. Notice that the success probability $\varepsilon(x,\mathcal{P}^*)$ of $\mathcal{P}^*$ is equal to $\mathbf{Adv}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$. Since $(\mathcal{P},\mathcal{V})$ is knowledge sound with knowledge error κ, there exists a knowledge extractor $\mathcal{E}$ that, on input $x \in X$ and rewindable oracle access to $\mathcal{P}^*$, outputs a witness $y \in Y$ for x with probability at least $\frac{\varepsilon(x,\mathcal{P}^*)-\kappa}{\mathsf{poly}(|x|)}$. Thus, we have:

$$\mathbf{Adv}^{\mathsf{H-REL}}_{R,\mathcal{A}}(\lambda) \geq \frac{\mathbf{Adv}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda) - \kappa}{\mathsf{poly}(|x|)}. \qquad \square$$

Corollary 1. *Let $(\mathcal{P},\mathcal{V})$ be a HVZK $(k_1,\ldots,k_\mu)$-special-sound $(2\mu+1)$-round interactive proof and $(\mathcal{P}^{t,w},\mathcal{V}^{t,w})$ be the (t,w)-fixed-weight repetition of $(\mathcal{P},\mathcal{V})$, where $w,t \in \mathbb{N}^*, 1 \leq w \leq t$, are chosen such that the knowledge error κ (Equation (1)) is negligible in λ. Then $(\mathcal{P}^{t,w},\mathcal{V}^{t,w})$ is secure against impersonation under passive attack.*

Proof. As shown in [13] $(\mathcal{P}^{t,w},\mathcal{V}^{t,w})$ is knowledge sound with knowledge error κ as in Equation (1). Since $\kappa = \mathsf{negl}(\lambda)$ by hypothesis, the result follows from the previous theorem. $\square$

Our goal is to show that the application of the Fiat-Shamir transform to a multi-round interactive proof, which is secure as per Definition 18, produces an EUF-CMA signature scheme, with a security loss proportional to $\binom{Q}{\mu}$, where Q is the number of hash queries the adversary is allowed to do.

Theorem 3. *Let $(\mathcal{P},\mathcal{V})$ be a $(2\mu+1)$-round interactive proof having super-polynomially large challenge sets $\mathsf{Ch}^{[1]},\ldots,\mathsf{Ch}^{[\mu]}$. If $(\mathcal{P},\mathcal{V})$ is secure against impersonation under passive attack, then the signature scheme $\mathsf{FS}[(\mathcal{P},\mathcal{V})]$, obtained by applying the Fiat-Shamir transform, is EUF-CMA.*

Proof. The proof is very similar to the one in [2]. We first provide a general idea of the proof, then we detail it.

Overview. Let $\mathcal{F}$ be a forger for the signature scheme $\mathsf{FS}[(\mathcal{P},\mathcal{V})]$, and let $Q(\lambda)$ and $Q_s(\lambda)$ be the number of hash and sign queries, respectively, that $\mathcal{F}$ is allowed to do (for the sake of readability, in the following we omit the dependency on λ). Our goal is to build an impersonator $\mathcal{I}$ for the interactive proof system $(\mathcal{P},\mathcal{V})$. The impersonator $\mathcal{I}$ interacts with the challenger of the experiment $\mathrm{Exp}^{\mathsf{IMP}}_{(\mathcal{P},\mathcal{V}),\mathcal{I}}(\lambda)$ and has access to transcripts obtained from the oracle OTrGen. In order to exploit $\mathcal{F}$ as a subroutine, $\mathcal{I}$ simulates the challenger of the experiment

$\mathrm{Exp}^{\text{EUF-CMA}}_{\mathsf{FS}[(\mathcal{P},\mathcal{V})],\mathcal{F}}(\lambda)$ with which $\mathcal{F}$ interacts. To do so, $\mathcal{I}$ needs to answer both *sign* and *random-oracle* queries.
We make some assumptions on $\mathcal{F}$:

- All the random-oracle queries are well-formed, i.e. they are of the form:

$$(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{k}, \mathsf{msg}),$$

  for some message msg, $\mathsf{com} \in \mathsf{Com}$, $\mathsf{ch}^{[i]} \in \mathsf{Ch}^{[i]}$, $\mathsf{rsp}^{[i]} \in \mathsf{Rsp}^{[i]}$ and $k < \mu$. [7] For the sake of readability, when the exact content of the random-oracle query is not relevant we simply write $x^{[k]}$ in place of it. Notice that the dependence from k is necessary to properly identify the correct output space.
- Before outputting a forgery $\sigma = (\mathsf{com}, \mathsf{rsp}^{[1]}, \ldots, \mathsf{rsp}^{[\mu]})$ on a message msg, the forger has made all the random-oracle queries $(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{k}, \mathsf{msg})$ for $k = 0, \ldots, \mu - 1$.
- The forger $\mathcal{F}$ has made the random-oracle queries of the previous point sequentially, and following the conventional order.

These hypotheses are the same as in [2], except for the last one about the sequentiality of the queries, that instead is typical of the multi-round setting. However, since σ is a forgery, we have that all the intermediate challenges are evaluations of a hash function. By contradiction, let us suppose that the forger does not perform the queries in order and let j be such that the query for $(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{j-1}, \mathsf{msg})$ is after the one for $(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{j}, \mathsf{msg})$. By construction, $\mathsf{ch}^{[j]} = \mathsf{H}_j(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{j-1}, \mathsf{msg})$ and, since the challenge space is super-polynomially large, $\mathcal{F}$ has negligible probability of guessing it.

Initialisation. $\mathcal{I}$ initialises the random-oracle-query counter $\mathsf{hc} = 0$ and the sign-query counter $\mathsf{sc} = 0$. $\mathcal{I}$ also initialises the hash table $\mathsf{HT} = \varnothing$, and the sign table $\mathsf{ST} = \varnothing$, then randomly generates an increasing ordered list of *forge pointers* $\mathsf{FP} \subset \{1, \ldots, Q\}$ with cardinality μ, and a forge pointer counter $\mathsf{fpc} = 1$. These forge pointers will be used by $\mathcal{I}$ during the simulation to try to guess when $\mathcal{F}$ is doing a query related to the forgery.

$\mathcal{I}$ receives the public statement x in the impersonation game and sets it as the public key for the digital signature $\mathsf{FS}[(\mathcal{P}, \mathcal{V})]$, i.e. $\mathsf{pk} = x$. $\mathcal{I}$ forwards this information to $\mathcal{F}$.

Training phase. Now $\mathcal{F}$ can perform Q random-oracle queries and Q_s sign queries to $\mathcal{I}$. In the first case $\mathcal{I}$ uses the hash table HT to answer, while in the second case $\mathcal{I}$ uses one of the transcripts obtained from the oracle OTrGen. Specifically, the simulation works as follows (if at any times $\mathsf{sc} > Q_s$ or $\mathsf{hc} > Q$ then the simulation aborts):

- *$\mathcal{F}$ performs a hash query with input $x^{[k]} \in \{0,1\}^*$*: we have two cases

[7] See Remark 3 for a discussion about domain separation and how to identify a random oracle correctly.

1. If $\mathsf{HT}[x^{[k]}]$ is already defined, $\mathcal{I}$ returns it.
2. Otherwise, $\mathcal{I}$ increases the counter hc by 1, then,
   - if $\mathsf{hc} \notin \mathsf{FP}$, $\mathcal{I}$ picks uniformly at random $d \in \mathsf{Ch}^{[k+1]}$, sends it to $\mathcal{F}$ and sets $\mathsf{HT}[x^{[k]}] \leftarrow d$.
   - If $\mathsf{hc} \in \mathsf{FP}$ then $\mathcal{I}$ checks whether $k = \mathsf{FP}[\mathsf{fpc}]$. If so, it parses $x^{[k]}$ as $(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{k}, \mathsf{msg})$, sends $\mathsf{rsp}^{[k]}$ to the verifier in the impersonation game as the $(2k+1)$-th move of the impersonation attempt and receives back from the verifier a challenge $\mathsf{ch}^{[k+1]}$. Otherwise, $\mathcal{I}$ pads $x^{[k]}$ so that it has the form $(\mathsf{com}, \{\mathsf{ch}^{[i]}, \mathsf{rsp}^{[i]}\}_{i=1}^{\mathsf{FP}[\mathsf{fpc}]}, \mathsf{msg})$ and then proceeds in the same way. Then, $\mathcal{I}$ sets $\mathsf{HT}[x^{[k]}] \leftarrow \mathsf{ch}^{[k+1]}$, sends it to $\mathcal{F}$ and increase fpc by one.

   This procedure allows $\mathcal{I}$ to perfectly simulate the random oracle.

– *$\mathcal{F}$ performs a sign query for message* msg: $\mathcal{I}$ increases the sign-query counter sc by one, picks an unused transcript and parses it as the signature of msg, defining all the hash values according to it, and storing them in HT. Notice that in this step $\mathcal{I}$ may need to overwrite an entry in HT, we will show later that this happens with negligible probability.

In order to state that $\mathcal{I}$ correctly simulates the experiment, it remains to show that the simulation fails only with negligible probability and to bound the success probability of $\mathcal{I}$.

Simulation Failure. We now focus on the cases in which the simulation may fail, and we find an upper bound on the probability that such failure happens. We have shown that the simulation of $\mathrm{Exp}^{\text{EUF-CMA}}_{\mathsf{FS}[(\mathcal{P},\mathcal{V})],\mathcal{F}}(\lambda)$ fails only if $\mathcal{I}$ is forced to overwrite the hash table HT during a sign query performed by $\mathcal{F}$. The overwriting of HT during a sign query might refer to a previous hash query or to a previous sign query. In particular, during the i-th sign query we have already set at most $Q + \mu(i-1)$ entries of HT, where the term $\mu(i-1)$ is due to the fact that in each sign query we modify μ entries in the hash table. Thus, the failure probability is bounded by:

$$\mu \frac{\mu + Q + \mu(i-1)}{2^{\beta(\lambda)}},$$

where $\beta(\lambda)$ is the min-entropy (Definition 11) associated with $(\mathcal{P}, \mathcal{V})$.

Then, the overall failure probability is, at most:

$$\mu \sum_{i=1}^{Q_s} \frac{\mu + Q + \mu(i-1)}{2^{\beta(\lambda)}} = \mu \frac{\mu Q_s + Q_s Q}{2^{\beta(\lambda)}} + \frac{\mu^2 Q_s (Q_s - 1)}{2 \cdot 2^{\beta(\lambda)}} \leq \mu^2 Q_s \frac{Q_s + Q}{2^{\beta(\lambda)}}. \quad (2)$$

Exploit of $\mathcal{F}$'s Forgery. Once $\mathcal{F}$ has concluded the training phase, $\mathcal{F}$ outputs a forgery $\sigma = (\mathsf{com}, \mathsf{rsp}^{[1]}, \ldots, \mathsf{rsp}^{[\mu]})$ for a message $\widehat{\mathsf{msg}}$ not previously queried. Then $\mathcal{I}$ concludes its impersonation attempt by sending the message $\mathsf{rsp}^{[\mu]}$ as response to the last challenge $\mathsf{ch}^{[\mu]}$ received.

Note that if $\mathcal{I}$ guesses all the indexes in FP correctly, then the impersonator succeeds if and only if the forgery verifies.

Evaluation of $\mathcal{I}$'s Advantage. From our assumptions, we have that the forger $\mathcal{F}$ must perform all hash queries involved in the forgery among the Q hash queries it is allowed to perform during the training phase, and that it needs to perform them in order. Therefore, with probability:

$$\Pr[\mathcal{I} \text{ guesses } \mathsf{FP} \mid \mathcal{I} \text{ simulates}] = \binom{Q}{\mu}^{-1},$$

the impersonator guesses the right set FP. If $\mathcal{I}$ correctly simulates the experiment $\mathrm{Exp}^{\text{EUF-CMA}}_{\mathsf{FS}[(\mathcal{P},\mathcal{V})],\mathcal{F}}(\lambda)$, $\mathcal{F}$ wins the simulated experiment, while interacting with $\mathcal{I}$, with the same non-negligible probability:

$$\Pr\left[\mathcal{F}^{\mathcal{I}} \text{ wins } \middle| \mathcal{I} \text{ simulates}\right] = \Pr\left[\mathrm{Exp}^{\text{EUF-CMA}}_{\mathsf{FS}[(\mathcal{P},\mathcal{V})],\mathcal{F}}(\lambda) = 1\right] = \epsilon(\lambda).$$

Finally, we can find a lower bound to the probability of success of the impersonator $\mathcal{I}$ in playing the experiment:

$$\binom{Q}{\mu}^{-1}\left(\epsilon(\lambda) - \mu^2 Q_s \frac{Q_s + Q}{2^{\beta(\lambda)}}\right),$$

which is non-negligible in the security parameter λ. [8] □

Corollary 2. *Let $(\mathcal{P},\mathcal{V})$ be a HVZK $(k_1,\ldots,k_\mu)$-special-sound $(2\mu+1)$-round interactive proof and $(\mathcal{P}^{t,w},\mathcal{V}^{t,w})$ be the (t,w)-fixed-weight repetition of $(\mathcal{P},\mathcal{V})$, where $w,t \in \mathbb{N}^*, 1 \le w \le t$, are chosen such that the knowledge error κ (Equation (1)) is negligible in λ. Then the digital signature $\mathsf{FS}[(\mathcal{P}^{t,w},\mathcal{V}^{t,w})]$ obtained by the application of the Fiat-Shamir transform is EUF-CMA secure.*

Proof. It is a straightforward application of Corollary 1 and Theorem 3. In fact, by Corollary 1 we have that $(\mathcal{P}^{t,w},\mathcal{V}^{t,w})$ is secure against impersonation under passive attack. Thus, by Theorem 3 we have that the obtained signature is EUF-CMA secure. □

Remark 5. At the basis of the proof of Theorem 2 there is the extractor defined in [13, Lemma 3], which works in expected polynomial time. Even if expected polynomial time is acceptable in many contexts [27, 29], it is often preferable to work in *strict* polynomial time, since most of the hard problems are stated with respect of polynomial-time adversaries.

In order to make the problem more concrete, we briefly summarise how the extractor works in the context of 3-round interactive proof[9], as per [7]. Let $(\mathcal{P},\mathcal{V})$ be a k-special-sound interactive proof with challenge space Ch. Given an adversary $\mathcal{A}$ for it, the considered extractor — denoted by $\mathcal{E}_k(\mathsf{Ch})^{\mathcal{A}}$ — has the

[8] The above equation is a consequence of [16, Lemma 2]. Indeed, the only difference between the real execution of $\mathsf{Exp}^{\text{EUF-CMA}}_{\mathsf{FS}[(\mathcal{P},\mathcal{V})],\mathcal{F}}$ and the simulation is the failure probability.

[9] The extractor for the general case of $(k_1,\ldots,k_\mu)$-special-sound interactive proofs is a recursive application of this simpler extractor.

goal of finding k couples of accepting transcripts $\{\mathsf{ch}_i, \mathsf{rsp}_i\}_{i=1}^{k}$. The idea is to define a base extractor that output a single accepting transcript $(\mathsf{ch}_1, \mathsf{rsp}_1)$, then the whole extractor $\mathcal{E}_k$ is defined recursively: first it runs the base extractor, then removes the extracted challenge from the challenge space and runs $\mathcal{E}_{k-1}$ on the new challenge space. In details:

- $\mathcal{E}_1(\mathsf{Ch})^{\mathcal{A}}$ samples uniformly at random a challenge $\mathsf{ch}_1 \in \mathsf{Ch}$ and gets from the adversary $\mathsf{rsp}_1 \leftarrow \mathcal{A}(\mathsf{ch}_1)$. If $\mathsf{Vrfy}(\mathsf{ch}_1, \mathsf{rsp}_1) = 1$, then it returns $(\mathsf{ch}_1, \mathsf{rsp}_1)$, otherwise it returns $\perp$.
- $\mathcal{E}_k(\mathsf{Ch})^{\mathcal{A}}$, with $k > 1$. First, it runs the base extractor $\mathcal{E}_1(\mathsf{Ch})^{\mathcal{A}}$. If $\mathcal{E}_1(\mathsf{Ch})^{\mathcal{A}}$ fails and returns $\perp$, then $\mathcal{E}_k(\mathsf{Ch})^{\mathcal{A}}$ fails as well and returns $\perp$. Otherwise, if $\mathcal{E}_1(\mathsf{Ch})^{\mathcal{A}}$ successfully returns a couple $(\mathsf{ch}_1, \mathsf{rsp}_1)$, it removes ch_1 from the challenge space, defining $\mathsf{Ch}' = \mathsf{Ch} \setminus \{\mathsf{ch}_1\}$ and runs $\mathcal{E}_{k-1}^{\mathcal{A}}(\mathsf{Ch}')$, with the goal of obtaining the remaining $k-1$ transcripts. If $\mathcal{E}_{k-1}^{\mathcal{A}}(\mathsf{Ch}')$ successfully returns $k-1$ couples $(\mathsf{ch}_2, \mathsf{rsp}_2), \ldots, (\mathsf{ch}_k, \mathsf{rsp}_k)$, then $\mathcal{E}_k(\mathsf{Ch})^{\mathcal{A}}$ returns $(\mathsf{ch}_1, \mathsf{rsp}_1), \ldots, (\mathsf{ch}_k, \mathsf{rsp}_k)$, otherwise it flips a coin: if the coin returns heads, then $\mathcal{E}_k^{\mathcal{A}}(\mathsf{Ch})$ returns $\perp$, otherwise it runs $\mathcal{E}_{k-1}^{\mathcal{A}}(\mathsf{Ch}')$ once more, repeating the process above.

It is clear that the only cases where the above extractor runs in super-polynomial time are when $\mathcal{E}_{k-1}^{\mathcal{A}}(\mathsf{Ch}')$ returns $\perp$ and the coin lands on tails for a super-polynomial number z of consecutive instances. The probability of this happening is upper-bounded by the probability of the coin landing on tails for that number of instances. Thus, defined p as the (non-negligible) probability that the coin lands on tail, we have that this happens with probability p^z, which is clearly negligible.

Now, let us consider the following modification of the extractor above: after a polynomial number of tries, the extractor $\mathcal{E}_k^{\mathcal{A}}(\mathsf{Ch})$ halts and returns $\perp$. Clearly, in this way the extractor runs in (strict) polynomial time, however we need to adjust the success probability.

In particular, the new success probability is:

$$\Pr\big[\perp \neq \mathcal{E}_k^{\mathcal{A}}(\mathsf{Ch})\big](1 - \Pr\big[\mathcal{E}_k^{\mathcal{A}}(\mathsf{Ch}) \textit{ runs in super-polynomial time}\big])$$

As proved above, the probability that $\mathcal{E}_k^{\mathcal{A}}(\mathsf{Ch})$ is allowed to run for more than a polynomial number of iterations is, at most, p^z, thus the loss in success probability is negligible.

4 A Novel Forgery for q2-Identification Schemes

In this section, we describe a novel forgery attack for the signature schemes obtained from a fixed-weight repetition of a q2-identification scheme [20], i.e. a $(2,2)$-out-of-$(q,2)$-special-sound interactive proof[10]. The forgery is a generalisation of the attack described for CROSS [10], which in turn is obtained from the attack in [28] for the plain parallel repetition of q2 schemes.

[10] In the following we assume that $\mathsf{Ch}^{[2]} = \{0,1\}$ and that, in the fixed-weight repetition, the value 1 appears exactly w times in the second challenge.

In Section 4.3 we also show how this attack impacts first round parameters of CROSS [10].

4.1 Forgery Attack on the Interactive Protocol

Let $(\mathcal{P}, \mathcal{V})$ be a $q2$-identification scheme, and consider its (t, w)-fixed-weight repetition $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$. Before presenting the forgery attack for the digital signature obtained by applying Fiat-Shamir on $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$, we briefly comment on the forgery attack for the underlying interactive proof described in [10, Proposition 17]. There, the authors describe a forgery running in average time $\mathcal{O}\left(\frac{1}{P(t,w,q)}\right)$, where:

$$P(t, w, q) = \sum_{\ell=0}^{\min\{w, t-w\}} \frac{\binom{w}{\ell}\binom{t-w}{\ell}}{\binom{t}{w}} q^{-2l}. \tag{3}$$

Notice that $P(t, w, q)$ describes the cheating probability of a dishonest prover attacking the interactive proof $(\mathcal{P}^{t,w}, \mathcal{V}^{t,w})$. As discussed in Section 2.4, in [13] it is proved that such an interactive proof is knowledge sound, and an explicit expression for the optimal cheating probability of a dishonest prover is given in Theorem 1. By applying this result to $q2$-identification schemes, where $\mu = 2, N_1 = q, N_2 = 2$, and $k_1 = k_2 = 2$, we obtain that the optimal cheating probability is given by:

$$\max_{\alpha \in \{0,\dots,t\}} \sum_{\ell=\max\{0, w-t+\alpha\}}^{\min\{w,\alpha\}} \frac{\binom{\alpha}{\ell}\binom{t-\alpha}{w-\ell}}{\binom{t}{w}} q^{-(\alpha-\ell)-(w-\ell)}. \tag{4}$$

It is easy to show that the expression in Equation (3) coincides with that in Equation (4) when $\alpha = w$. Unfortunately, this value for α is not optimal when w is larger than $t/2$, thus the adversary's cheating probability is underestimated.

On a high-level, the original strategy of [10, Proposition 17] is as follows. The adversary wins if in each parallel execution it is able to guess one of the two individual challenges. This is a consequence of the existence of an efficient strategy for the adversary that has some peculiarities. Namely, in each round the adversary can prepare a response for which, if the next challenge is correctly guessed, it can complete the protocol and have the verifier to accept, no matter what challenges are provided in the remaining rounds. The existence of such a strategy is a property enjoyed by CROSS and most $(k_1, \dots, k_\mu)$-special-sound protocols. This property can be formalised in general with the notion of *special-unsoundness* [8], but in the scope of our attack it will be expressed in more details in the next section, with the notion of *piecewise simulatability*.

Since we are considering t parallel repetitions of $(\mathcal{P}, \mathcal{V})$ where the second challenge is a binary string of length t and weight w, the adversary can try to guess w executions in which they expect the second individual challenge to be 1. If for one execution the choice is wrong, the adversary can still win by guessing the first individual challenge, which happens with probability q^{-1}. Assuming that the adversary's choice was wrong for ℓ of the selected w executions, it will have

to guess the first individual challenges for 2ℓ executions[11]. Since there are $\binom{w}{\ell}$ different ways of choosing ℓ wrong positions among the selected w, and $\binom{t-w}{\ell}$ ways of finding these positions among the remaining $t-w$, the obtained cheating probability is that of Equation (3).

This strategy appears to be optimal when $w \approx t/2$, where the maximum for Equation (4) is reached for $\alpha = w$. However, a more general approach can be obtained by allowing the adversary to select $\alpha \in \{0, \ldots, t\}$ executions in which it guesses a value 1 for the second individual challenge. In particular, when $w \approx t$, choosing $\alpha > w$ results in a better strategy. In fact, making mistakes in a few positions is more efficient than guessing all the positions for the 1-entries in the second challenge.

Proposition 1. *Consider the (t, w)-fixed-weight repetition of a $(2, 2)$-out-of-$(q, 2)$-special-sound interactive proof. A dishonest prover can convince a verifier if, for all parallel executions, they either guess the first or the second individual challenge (or both) correctly. If the prover selects $\alpha \geq w$ executions for the fixed-weight element in the second challenge, the attack runs in average time $\mathcal{O}\left(\frac{1}{P_\alpha(t,w,q)}\right)$, where*

$$P_\alpha(t, w, q) = \sum_{\ell=\max\{0, w-t+\alpha\}}^{\min\{w,\alpha\}} \frac{\binom{\alpha}{\ell}\binom{t-\alpha}{w-\ell}}{\binom{t}{w}} q^{-(\alpha-\ell)-(w-\ell)}.$$

The overall cost of the attack is estimated by optimizing over $\alpha \in \{0, \ldots, t\}$.

Proof. As discussed above, the cheating strategy associated with Theorem 1 for $(k_1, \ldots, k_\mu)$-special-sound interactive proofs is optimal. By substituting the parameters of $q2$-identification schemes, namely $\mu = 2, N_1 = q, N_2 = 2$, and $k_1 = k_2 = 2$, in the expression of Equation (1), we immediately obtain $P_\alpha(t, w, q)$. □

4.2 Forgery Attack on the Signature Scheme

We now describe a forgery attack for a signature scheme obtained by applying the Fiat-Shamir transform on the (t, w)-fixed-weight repetition of a $q2$-identification scheme. The forgery is based on the attack described in [10, Proposition 18], which exploits the fact that the second round challenge can be repeatedly guessed without modifying the initial commitment and the first round challenge. In this way, the attack can be split into two phases. In the first phase, the adversary tries to guess the value of the first challenge for at least t^* parallel executions of the protocol. Then, in the second phase, the adversary tries to guess the second challenge for the remaining incorrect executions. As in the interactive case, this strategy can be improved for fixed-weight challenges by selecting $\alpha \geq w$ parallel

[11] For each of the ℓ wrong positions for the choice of 1s, there is a corresponding wrong choice among the 0s.

executions where the fixed-weight elements appears inside the second challenge, i.e. guessing $\mathsf{ch}^{[2]}$ from $(\mathsf{Ch}^{[2]})^{t,\alpha}$ instead of $(\mathsf{Ch}^{[2]})^{t,w}$.

To carry out the attack, the adversary must be able to generate valid transcripts after correctly guessing either the first or the second challenge. This capability is slightly stronger than that granted by the HVZK simulator S of Definition 10, where S takes as input only a public key. This property can be obtained by requiring the base interactive proof to be *piecewise simulatable* [28]. Informally, this means that the HVZK simulator can be split into two algorithms (in two different ways). By doing so, it is possible to first produce a partial transcript from the first algorithm, then complete it by giving one of the challenges as input to the second algorithm.

Definition 19 (Piecewise Simulatability). *A HVZK 5-round interactive proof is* piecewise simulatable *if there exist probabilistic-polynomial time algorithms $A = (A_1, A_2)$ and $B = (B_1, B_2)$, defined as follows:*

Simulator A:
1: $T_1 = (\mathsf{com}, \mathsf{ch}^{[1]}, \mathsf{rsp}^{[1]}) \leftarrow\$ A_1(\mathsf{pk})$
2: $\mathsf{rsp}^{[2]} \leftarrow\$ A_2(\mathsf{pk}, T_1, \widetilde{\mathsf{ch}}^{[2]})$
3: $T \leftarrow (\mathsf{com}, \mathsf{ch}^{[1]}, \mathsf{rsp}^{[1]}, \widetilde{\mathsf{ch}}^{[2]}, \mathsf{rsp}^{[2]})$

Simulator B:
1: $T_1 = \mathsf{com} \leftarrow\$ B_1(\mathsf{pk})$
2: $(\mathsf{rsp}^{[1]}, \mathsf{rsp}^{[2]}) \leftarrow\$ B_2(\mathsf{pk}, T_1, \widetilde{\mathsf{ch}}^{[1]}, \mathsf{ch}^{[2]})$
3: $T' \leftarrow (\mathsf{com}, \widetilde{\mathsf{ch}}^{[1]}, \mathsf{rsp}^{[1]}, \mathsf{ch}^{[2]}, \mathsf{rsp}^{[2]})$

where T and T' are distributed as the output of the HVZK simulator $S(\mathsf{pk})$, for all $ch^{[2]} \in \mathsf{Ch}^{[2]}$ and when $\widetilde{\mathsf{ch}}^{[1]}$ (resp., $\widetilde{\mathsf{ch}}^{[2]}$) is chosen uniformly at random from $\mathsf{Ch}^{[1]}$ (resp., $\mathsf{Ch}^{[2]}$).

If the interactive proof satisfies Definition 19, Simulator A can be employed to produce a valid transcript on a random second challenge $\widetilde{\mathsf{ch}}^{[2]}$ for a given prefix $(\mathsf{com}, \mathsf{ch}^{[1]}, \mathsf{rsp}^{[1]})$. On the other hand, Simulator B can be employed to produce a valid transcript on a random first challenge $\widetilde{\mathsf{ch}}^{[1]}$. Compared with the original definition of [28], algorithm B_2 has been modified so that it can also accept as input the second challenge $\mathsf{ch}^{[2]}$. By making no assumption on the distribution of $\mathsf{ch}^{[2]}$, we capture the ability of the adversary to attempt to produce a valid transcript without changing the commitment and the first challenge. Although this is a stronger notion, it is fulfilled by CROSS[12] and the $q2$-identification schemes investigated in [28].

We are now ready to describe the anticipated forgery attack, which is detailed in Algorithm 2. There, the adversary has access to the user's public key pk and uses Algorithms $(A_1, A_2), (B_1, B_2)$ to produce a forgery for an arbitrary message msg.

Proposition 2. *By optimizing over the choice of t^* and α, the forgery of Algorithm 2 runs on average time*

$$\mathcal{O}\left(\min_{t^* \in \{0,\dots,t\}} \left\{ \frac{1}{P_1(t, t^*, q)} + \frac{1}{P_2(t, t^*, w, q)} \right\}\right),$$

[12] In the forgery procedure of [10, Proposition 18], a detailed description of the required algorithms is provided.

Algorithm 2 Forgery attack

Let $t^* \in \{0,\dots,t\}$ be the number of executions where the first challenged is guessed, and $\alpha \geq w$ be the number of executions where the fixed-weight element is chosen in the second challenge. Let $\mathsf{H}_1, \mathsf{H}_2$ be cryptographic hash functions from $\{0,1\}^*$ to $(\mathsf{Ch}^{[1]})^t, (\mathsf{Ch}^{[2]})^{t,w}$, respectively.

Forge($\mathsf{pk}, \mathsf{msg}$):

1: **repeat**
2: **for** $i \in \{1,\dots,t\}$ **do**
3: $T_i^{[1]} \leftarrow (\mathsf{com}_i, \widetilde{\mathsf{ch}}_i^{[1]}, \mathsf{rsp}_i^{[1]}) \leftarrow\$ A_1(\mathsf{pk})$ ▷ Guess values for the first challenge
4: $(\mathsf{ch}_1^{[1]},\dots,\mathsf{ch}_t^{[1]}) \leftarrow \mathsf{H}_1(\mathsf{pk}, \{\mathsf{com}_i\}_{i=1}^t, \mathsf{msg})$
5: $S \leftarrow \{i \in \{1,\dots,t\} \mid \mathsf{ch}_i^{[1]} = \widetilde{\mathsf{ch}}_i^{[1]}\}$
6: **until** $|S| \geq t^*$
7: $(\widetilde{\mathsf{ch}}_1^{[2]},\dots,\widetilde{\mathsf{ch}}_t^{[2]}) \leftarrow\$ (\mathsf{Ch}^{[2]})^{t,\alpha}$ ▷ Guess values for the second challenge
8: **for** $i \in S$ **do**
9: $\widetilde{\mathsf{rsp}}_i^{[1]} \leftarrow \mathsf{rsp}_i^{[1]}$
10: **repeat**
11: **for** $i \notin S$ **do**
12: $(\widetilde{\mathsf{rsp}}_i^{[1]}, \widetilde{\mathsf{rsp}}_i^{[2]}) \leftarrow\$ B_2(\mathsf{pk}, \mathsf{com}_i, \mathsf{ch}_i^{[1]}, \widetilde{\mathsf{ch}}_i^{[2]})$
13: $(\mathsf{ch}_1^{[2]},\dots,\mathsf{ch}_t^{[2]}) \leftarrow \mathsf{H}_2(\mathsf{pk}, \{\mathsf{com}_i, \mathsf{ch}_i^{[1]}, \widetilde{\mathsf{rsp}}_i^{[1]}\}_{i=1}^t, \mathsf{msg})$
14: **until** $\mathsf{ch}_i^{[2]} = \widetilde{\mathsf{ch}}_i^{[2]}$ for all $i \notin S$
15: **for** $i \in S$ **do**
16: $\widetilde{\mathsf{rsp}}_i^{[2]} \leftarrow\$ A_2(\mathsf{pk}, T_i^{[1]}, \mathsf{ch}_i^{[2]})$
17: **return** $\sigma \leftarrow (\{\mathsf{com}_i, \widetilde{\mathsf{rsp}}_i^{[1]}, \widetilde{\mathsf{rsp}}_i^{[2]}\}_{i=1}^t)$

where

$$P_1(t,t^*,q) = \sum_{j=t^*}^{t} \binom{t}{j}\left(\frac{1}{q}\right)^j\left(1-\frac{1}{q}\right)^{t-j},$$

$$P_2(t,t^*,w,q) = \max_{\alpha\in\{w,\dots,t\}} \sum_{j=t^*}^{t} \frac{\binom{t}{j}\left(\frac{1}{q}\right)^j\left(1-\frac{1}{q}\right)^{t-j}}{P_1(t,t^*,q)} \sum_{w^*=\max\{0,\alpha-j\}}^{\min\{t-j,\alpha\}} \frac{\binom{t-j}{w^*}\binom{j}{\alpha-w^*}}{\binom{t}{\alpha}} \frac{\binom{j}{w-w^*}}{\binom{t}{w}}.$$

Proof. Complexity estimation for the forgery attack is essentially the same as that described in [10, Proposition 18], except that in guessing the values of the second challenge, the adversary can now choose a number $\alpha \geq w$ of parallel executions for the fixed-weight elements (Line 7).

The algorithm iterates over the first loop (Lines 1 to 6) until the choices on the first challenge are valid for at least t^* parallel executions. These prefixes of the transcript are obtained by repeatedly executing the simulator A_1. Once this is obtained, the algorithm freezes the individual commitments and the first challenge. Then, it starts making attempts for the second challenge, and it only stops when the latter is correctly generated for the remaining $t - t^*$ executions (Lines 10 to 14). For each attempt, the algorithm executes the probabilistic algorithm B_2, obtaining fresh values for the first response and, consequently, ensuring new values for the second challenges $(\mathsf{ch}_1^{[2]},\dots,\mathsf{ch}_t^{[2]})$ on Line 13. By

doing this, the commitments prepared in the initial loop remain unchanged. This procedure gets repeated until the second challenge is suitably chosen. Namely, in every execution where the attacker did not guess the correct value for the first challenge, the value for the second challenge must be correctly guessed (Line 14).

The total cost of the attack is the sum of the costs for the two phases. The probability that the initial guess $(\widetilde{\mathsf{ch}}_1^{[1]}, \ldots, \widetilde{\mathsf{ch}}_t^{[1]})$ is valid, i.e. that it matches in at least t^* positions with $(\mathsf{ch}_1^{[1]}, \ldots, \mathsf{ch}_t^{[1]})$ generated on Line 4, is

$$P_1(t, t^*, q) = \sum_{j=t^*}^{t} \binom{t}{j} \left(\frac{1}{q}\right)^j \left(1 - \frac{1}{q}\right)^{t-j}.$$

Consequently, the average cost for the first loop is $O\left(\frac{1}{P_1(t,t^*,q)}\right)$.

We now consider the second loop. Let S denote the set of indices i for which $\mathsf{ch}_i^{[1]} = \widetilde{\mathsf{ch}}_i^{[1]}$ and its complement by $\bar{S}$. We define $j = |S|$ and we notice that

$$\Pr[\,|S| = j \mid |S| \geq t^*\,] = \frac{\binom{t}{j}\left(\frac{1}{q}\right)^j \left(1 - \frac{1}{q}\right)^{t-j}}{P_1(t, t^*, q)}.$$

Denote by $\mathsf{ch}_S^{[2]}$ (resp., $\widetilde{\mathsf{ch}}_S^{[2]}$) the vector formed by the coordinates of $\mathsf{ch}^{[2]}$ (resp., $\widetilde{\mathsf{ch}}^{[2]}$) which are indexed by S. Analogously, we denote by $\mathsf{ch}_{\bar{S}}^{[2]}$ (resp., $\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}$) the vector formed by the coordinates of $\mathsf{ch}^{[2]}$ (resp., $\widetilde{\mathsf{ch}}^{[2]}$) which are not indexed by S. For the second loop to halt, $\mathsf{ch}^{[2]}$ must be such that $\mathsf{ch}_{\bar{S}}^{[2]} = \widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}$. Let w^* denote the number of 1-guesses for the individual executions indexed by $\bar{S}$; that is, w^* is the Hamming weight of $\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}$. Recall that, in guessing $\widetilde{\mathsf{ch}}^{[2]}$ the adversary chooses $\alpha \geq w$ position for the 1-entries (Line 7). It follows that

$$\Pr\left[\mathsf{wt}(\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}) = w^*\right] = \frac{\binom{t-j}{w^*}\binom{j}{\alpha - w^*}}{\binom{t}{\alpha}}.$$

The probability that a generated $\mathsf{ch}^{[2]}$ is valid, i.e. $\mathsf{ch}_{\bar{S}}^{[2]} = \widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}$, is

$$\begin{aligned}
\Pr\left[\mathsf{ch}^{[2]} \text{ is valid} \,\middle|\, \mathsf{wt}(\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}) = w^*\right] &= \frac{\left|\left\{\, \mathsf{ch}^{[2]} \in \{0,1\}^t \mid \mathsf{wt}(\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}) = w^*, \mathsf{ch}_{\bar{S}}^{[2]} = \widetilde{\mathsf{ch}}_{\bar{S}}^{[2]} \,\right\}\right|}{\binom{t}{w}} \\
&= \frac{\left|\left\{\, \mathsf{ch}^{[2]} \in \{0,1\}^t \mid \mathsf{wt}(\mathsf{ch}_S^{[2]}) = w - w^*, \mathsf{ch}_{\bar{S}}^{[2]} = \widetilde{\mathsf{ch}}_{\bar{S}}^{[2]} \,\right\}\right|}{\binom{t}{w}} \\
&= \frac{\binom{j}{w - w^*}}{\binom{t}{w}}.
\end{aligned}$$

Putting everything together, we have that in each execution of the second loop, $\mathsf{ch}^{[2]}$ is correctly guessed with an average probability of

$$\begin{aligned} P_2(t,t^*,w,q) &= \max_{\alpha\in\{w,\dots,t\}} \sum_{j=t^*}^{t} \Pr[|S|=j \mid |S|\geq t^*] \\ &\quad \cdot \sum_{w^*=\max\{0,\alpha-j\}}^{\min\{t-j,\alpha\}} \Pr\left[\mathsf{wt}(\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]})=w^*\right] \cdot \Pr\left[\mathsf{ch}^{[2]} \text{ is valid} \,\middle|\, \mathsf{wt}(\widetilde{\mathsf{ch}}_{\bar{S}}^{[2]}=w^*)\right] \\ &= \max_{\alpha\in\{w,\dots,t\}} \sum_{j=t^*}^{t} \frac{\binom{t}{j}\left(\frac{1}{q}\right)^j\left(1-\frac{1}{q}\right)^{t-j}}{P_1(t,t^*,q)} \sum_{w^*=\max\{0,\alpha-j\}}^{\min\{t-j,w\}} \frac{\binom{t-j}{w^*}\binom{j}{\alpha-w^*}}{\binom{t}{\alpha}} \frac{\binom{j}{w-w^*}}{\binom{t}{w}}. \end{aligned}$$

The overall cost of the attack is estimated by summing the costs for both phases and optimizing over t^*, that is

$$\min_{t^*\in\{0,\dots,t\}} \left\{ \frac{1}{P_1(t,t^*,q)} + \frac{1}{P_2(t,t^*,w,q)} \right\}. \qquad \square$$

4.3 Application to CROSS

CROSS [10] is a code-based signature scheme submitted to the "on-ramp" NIST competition, which was recently admitted to the second round. The signature scheme is obtained by applying the Fiat-Shamir transform to a parallel repetition of a 5-round interactive proof. This interactive proof — which we will denote by Π_{CR} — is a q2-identification scheme.

CROSS specifications provide parameters for two variants: R-SDP and R-SDP (G). In particular, for each variant, different parameter sets are provided for NIST security categories 1, 3, and 5. In turn, for each security category, three parameter sets are proposed, aiming at three distinct optimisation corners: computational speed in signature and verification ("fast" parameters), signature size ("small" parameters), and a balanced version which aims for a balance between the previous two ("balanced" parameters).

Every parameter set employs the fixed-weight optimisation on the second challenge of Π_{CR}^t. In particular, the second challenge from $\mathsf{Ch}^{[2]}$ has always a given weight w. The modification of Π_{CR}^t obtained via this optimisation is denoted by $\Pi_{\mathsf{CR}}^{t,w}$. Within CROSS specifications, the fixed-weight element is 1. The three different parameter sets for each variant of the protocol and each NIST security category are chosen accordingly to the ratio between w and t. For the "fast" variant, we have $w \approx t/2$ and t slightly larger than the security parameter λ. In the "balanced" and "small" versions, we instead have that w is close to t and $t \gg \lambda$. The choice of fixed-weight parameters (t,w) in [10] is done such that the complexity of the best forgery attack against CROSS exceeds 2^λ, and the value of t is the minimum possible.

In the parameter sets of CROSS, submitted to the first round of the NIST "on-ramp" process, the choice of the parameters t,w is made by evaluating the complexity of a forgery that does not exploit the fixed-weight of the second

Table 2. Cost of our forgery attack (Proposition 2) compared to the attack considered in [10] for choosing the parameters of CROSS as submitted to the first round of the NIST "on-ramp" process [30]. Complexities are given as $\log_2$ of the estimated gate count. t^* and α show the optimal choices for the attack parameters.

Parameters					CROSS Forgery		Our Forgery		
Set	**Optim.**	p	t	w	**Compl.**	t^*	**Compl.**	t^*	α
CROSS-R-SDP 1	fast	127	163	85	128.06	35	128.05	35	86
	balanced	127	252	212	128.01	40	**120.46**	38	227
	small	127	960	938	128.00	65	**97.48**	55	960
CROSS-R-SDP 3	fast	127	245	127	192.08	52	192.05	52	128
	balanced	127	398	340	192.07	61	**179.67**	59	365
	small	127	945	907	192.02	83	**156.37**	73	944
CROSS-R-SDP 5	fast	127	327	169	256.06	70	256.03	70	171
	balanced	127	327	169	256.01	81	**240.82**	78	459
	small	127	968	912	255.22	101	**217.15**	91	957
CROSS-R-SDP (G) 1	fast	509	153	79	128.06	24	128.06	24	79
	balanced	509	243	206	128.13	27	**122.72**	26	216
	small	509	871	850	128.01	38	**108.22**	34	867
CROSS-R-SDP (G) 3	fast	509	230	123	192.03	37	191.98	37	125
	balanced	509	255	176	192.03	37	<u>189.83</u>	37	184
	small	509	949	914	192.03	53	**167.56**	48	937
CROSS-R-SDP (G) 5	fast	509	306	157	256.01	49	256.00	49	158
	balanced	509	356	257	256.08	51	<u>252.70</u>	50	270
	small	509	996	945	256.03	66	**228.58**	61	974

challenge. Applying the forgery described in the previous section and evaluating its complexity with the expression of Proposition 2, we can observe that the previous strategy is not optimal, and when $w \approx t$ (e.g., for "balanced" and "small" parameter sets) the CROSS parameters do not achieve the expected security level (Table 2). In particular, Proposition 2 leads to a more efficient adversary for any choice of parameters t and w. For the R-SDP variant of CROSS, the "balanced" parameterisation loses 6% of security margin on average across all NIST security categories, while the "small" parameterisation loses 19% on average. The largest security loss is incurred by the "small" parameterisation for NIST security category 1, with a loss of 30 bits compared to the security target of 128 bits. For the R-SDP (G) variant, the incurred security loss is lower due to an increased value for q, with an average reduction of 2% for the "balanced" parametrisation and of 13% for the "small" parametrisation. Again, the largest security loss is incurred by the "small" parameterisation for NIST security category 1, with a loss of 20 bits compared to the security target of 128 bits.

Artifacts. Scripts for reproducing the attack costs, including all presented tables, are available at `https://github.com/edoars/revise-cross-parameters`.

5 Conclusions

In this work we provided an explicit proof of the EUF-CMA security of CROSS, a signature scheme currently in the second round of the NIST "on-ramp" standardisation process for post-quantum signatures. We did that by proving that the Fiat-Shamir transform of any interactive proof with negligible knowledge error yields an EUF-CMA secure signature scheme, with a security loss of at most $\binom{Q}{\mu}$, where Q is the number of signature queries and $2\mu + 1$ is the number of rounds.

As a second contribution, we presented a novel forgery attack against signatures based on $q2$-identification schemes, significantly improving upon previous results. When applied to CROSS, our attack demonstrates that some parameter sets achieve lower security levels than originally claimed, with reductions of up to 24% in the worst case. This has practical implications for CROSS's parameter selection.

Several interesting directions remain open for future research. First, while we proved an upper bound for the security loss of the Fiat-Shamir transform, we do not have a matching lower bound that proves the optimality of our attack. Finding such a bound would provide a complete picture of the exact security guarantees provided by Fiat-Shamir signatures obtained from fixed-weight repetitions of multi-round interactive proofs.

A second direction concerns the requirements for our forgery attack. Currently, the attack requires the underlying interactive proof to satisfy piecewise simulatability, a stronger property than typically needed for Fiat-Shamir signatures. Although this property is satisfied by CROSS and most 5-round interactive proofs in the literature, it would be valuable to understand how the attack could be adapted to protocols with different security properties. Of particular interest would be investigating the impact of *early abort capabilities* [28] in intermediate rounds, as this could potentially prove certain protocols more resistant to our forgery technique.

Acknowledgements

This work has been partially supported by project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union — NextGenerationEU.

The first author is supported by the Italian Ministry of University's PRIN 2022 program under the "Mathematical Primitives for Post Quantum Digital Signatures" (P2022J4HRR) and "POst quantum Identification and eNcryption primiTives: dEsign and Realization (POINTER)" (2022M2JLF2) projects funded by the European Union — Next Generation EU.

References

[1] N. Aaraj et al. *PERK*. Tech. rep. available at `https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures`. National Institute of Standards and Technology, 2023.

[2] 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: *EUROCRYPT 2002*. Ed. by L. R. Knudsen. Vol. 2332. LNCS. Springer, Berlin, Heidelberg, 2002, pp. 418–433. DOI: 10.1007/3-540-46035-7_28.

[3] G. Adj, L. Rivera-Zamarripa, J. Verbel, E. Bellini, S. Barbero, A. Esser, C. Sanna, and F. Zweydinger. *MiRitH — MinRank in the Head*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[4] C. Aguilar-Melchor et al. *SDitH — Syndrome Decoding in the Head*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[5] N. Aragon, M. Bardet, L. Bidoux, J. Chi-Domínguez, V. Dyseryn, T. Feneuil, P. Gaborit, R. Neveu, M. Rivain, and J. Tillich. *MIRA*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[6] N. Aragon et al. *RYDE*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[7] T. Attema and S. Fehr. "Parallel Repetition of $(k_1, \ldots, k_\mu)$-Special-Sound Multi-round Interactive Proofs". In: *CRYPTO 2022, Part I*. Ed. by Y. Dodis and T. Shrimpton. Vol. 13507. LNCS. Springer, Cham, Aug. 2022, pp. 415–443. DOI: 10.1007/978-3-031-15802-5_15.

[8] T. Attema, S. Fehr, and M. Klooß. "Fiat-Shamir Transformation of Multi-round Interactive Proofs". In: *TCC 2022, Part I*. Ed. by E. Kiltz and V. Vaikuntanathan. Vol. 13747. LNCS. Springer, Cham, Nov. 2022, pp. 113–142. DOI: 10.1007/978-3-031-22318-1_5.

[9] T. Attema, S. Fehr, and M. Klooß. "Fiat-Shamir Transformation of Multi-Round Interactive Proofs (Extended Version)". In: *Journal of Cryptology* 36.4 (Oct. 2023), p. 36. DOI: 10.1007/s00145-023-09478-y.

[10] M. Baldi et al. *CROSS — Codes and Restricted Objects Signature Scheme*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[11] M. Baldi et al. *LESS — Linear Equivalence Signature Scheme*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[12] A. Barenghi, J.-F. Biasse, E. Persichetti, and P. Santini. "LESS-FM: Fine-Tuning Signatures from the Code Equivalence Problem". In: *Post-Quantum Cryptography - 12th International Workshop, PQCrypto 2021*. Ed. by J. H.

Cheon and J.-P. Tillich. Springer, Cham, 2021, pp. 23–43. DOI: 10.1007/978-3-030-81293-5_2.

[13] M. Battagliola, R. Longo, F. Pintore, E. Signorini, and G. Tognolini. *Security of Fixed-Weight Repetitions of Special-Sound Multi-Round Proofs*. Cryptology ePrint Archive, Report 2024/884. 2024. URL: https://eprint.iacr.org/2024/884.

[14] M. Bellare, R. Impagliazzo, and M. Naor. "Does Parallel Repetition Lower the Error in Computationally Sound Protocols?" In: *38th FOCS*. IEEE Computer Society Press, Oct. 1997, pp. 374–383. DOI: 10.1109/SFCS.1997.646126.

[15] M. Bellare and P. Rogaway. "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols". In: *ACM CCS 93*. Ed. by D. E. Denning, R. Pyle, R. Ganesan, R. S. Sandhu, and V. Ashby. ACM Press, Nov. 1993, pp. 62–73. DOI: 10.1145/168588.168596.

[16] M. Bellare and P. Rogaway. "The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs". In: *EUROCRYPT 2006*. Ed. by S. Vaudenay. Vol. 4004. LNCS. Springer, Berlin, Heidelberg, 2006, pp. 409–426. DOI: 10.1007/11761679_25.

[17] W. Beullens, S. Dobson, S. Katsumata, Y.-F. Lai, and F. Pintore. "Group signatures and more from isogenies and lattices: generic, simple, and efficient". In: *DCC* 91.6 (2023), pp. 2141–2200. DOI: 10.1007/s10623-023-01192-x.

[18] W. Beullens, S. Katsumata, and F. Pintore. "Calamari and Falafl: Logarithmic (Linkable) Ring Signatures from Isogenies and Lattices". In: *ASIACRYPT 2020, Part II*. Ed. by S. Moriai and H. Wang. Vol. 12492. LNCS. Springer, Cham, Dec. 2020, pp. 464–492. DOI: 10.1007/978-3-030-64834-3_16.

[19] J. Chavez-Saab et al. *SQIsign*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[20] M.-S. Chen, A. Hülsing, J. Rijneveld, S. Samardjiska, and P. Schwabe. "From 5-Pass MQ-Based Identification to MQ-Based Signatures". In: *ASIACRYPT 2016, Part II*. Ed. by J. H. Cheon and T. Takagi. Vol. 10032. LNCS. Springer, Berlin, Heidelberg, Dec. 2016, pp. 135–165. DOI: 10.1007/978-3-662-53890-6_5.

[21] T. Chou, R. Niederhagen, E. Persichetti, T. H. Randrianarisoa, K. Reijnders, S. Samardjiska, and M. Trimoska. "Take Your MEDS: Digital Signatures from Matrix Code Equivalence". In: *AFRICACRYPT 23*. Ed. by N. El Mrabet, L. De Feo, and S. Duquesne. Vol. 14064. LNCS. Springer, Cham, July 2023, pp. 28–52. DOI: 10.1007/978-3-031-37679-5_2.

[22] L. De Feo, D. Kohel, A. Leroux, C. Petit, and B. Wesolowski. "SQISign: Compact Post-quantum Signatures from Quaternions and Isogenies". In: *ASIACRYPT 2020, Part I*. Ed. by S. Moriai and H. Wang. Vol. 12491. LNCS. Springer, Cham, Dec. 2020, pp. 64–93. DOI: 10.1007/978-3-030-64837-4_3.

[23] T. Feneuil and M. Rivain. *MQOM — MQ on my Mind*. Tech. rep. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures. National Institute of Standards and Technology, 2023.

[24] A. Fiat and A. Shamir. "How to Prove Yourself: Practical Solutions to Identification and Signature Problems". In: *CRYPTO'86*. Ed. by A. M. Odlyzko. Vol. 263. LNCS. Springer, Berlin, Heidelberg, Aug. 1987, pp. 186–194. DOI: 10.1007/3-540-47721-7_12.

[25] W. Ghantous, F. Pintore, and M. Veroni. "Efficiency of SIDH-based signatures (yes, SIDH)". In: *J. Math. Cryptol.* 18.1 (2024). DOI: 10.1515/JMC-2023-0023. URL: https://doi.org/10.1515/jmc-2023-0023.

[26] A. Hülsing et al. *SPHINCS$^+$*. Tech. rep. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022. National Institute of Standards and Technology, 2022.

[27] J. Jaeger and S. Tessaro. "Expected-Time Cryptography: Generic Techniques and Applications to Concrete Soundness". In: *TCC 2020, Part III*. Ed. by R. Pass and K. Pietrzak. Vol. 12552. LNCS. Springer, Cham, Nov. 2020, pp. 414–443. DOI: 10.1007/978-3-030-64381-2_15.

[28] D. Kales and G. Zaverucha. "An Attack on Some Signature Schemes Constructed from Five-Pass Identification Schemes". In: *CANS 20*. Ed. by S. Krenn, H. Shulman, and S. Vaudenay. Vol. 12579. LNCS. Springer, Cham, Dec. 2020, pp. 3–22. DOI: 10.1007/978-3-030-65411-5_1.

[29] J. Katz and Y. Lindell. "Handling Expected Polynomial-Time Strategies in Simulation-Based Security Proofs". In: *Journal of Cryptology* 21.3 (July 2008), pp. 303–349. DOI: 10.1007/s00145-007-9004-8.

[30] NIST. *Call for Additional Digital Signature Schemes for the Post-Quantum Cryptography Standardization Process*. URL: https://csrc.nist.gov/projects/pqc-dig-sig/standardization/call-for-proposals. 2023.

[31] NIST. *Post-Quantum Cryptography: Additional Digital Signature Schemes. Round 2 Additional Signatures*. URL: https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures. 2024.

[32] NIST. *Post-Quantum Cryptography Standardization*. URL: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography. 2017.

[33] D. Pointcheval and J. Stern. "Security Arguments for Digital Signatures and Blind Signatures". In: *Journal of Cryptology* 13.3 (June 2000), pp. 361–396. DOI: 10.1007/s001450010003.

[34] L. Ran, S. Samardjiska, and M. Trimoska. "Algebraic Algorithm for the Alternating Trilinear Form Equivalence Problem". In: *Code-Based Cryptography - 11th International Workshop, CBCrypto 2023, Lyon, France, April 22-23, 2023, Revised Selected Papers*. Ed. by A. Esser and P. Santini. Vol. 14311. Lecture Notes in Computer Science. Springer, 2023, pp. 84–103. DOI: 10.1007/978-3-031-46495-9_5.