# Provable Security of SHA-256-Based LMS Revisited in the Ideal-Cipher Model

Chun Guo[123(✉)], Hao Cheng[123(✉)], Bart Preneel[4(✉)], and Meiqin Wang[1235(✉)]

[1] School of Cyber Science and Technology, Shandong University, Qingdao, China
[2] State Key Laboratory of Cryptography and Digital Economy Security, Shandong University, Qingdao, 266237, China
[3] Key Laboratory of Cryptologic Technology and Information Security of Ministry of Education, Shandong University, Qingdao, Shandong, 266237, China
[4] imec-COSIC, KU Leuven, Leuven, Belgium
[5] Quan Cheng Shandong Laboratory, Jinan, China
chun.guo.sc@gmail.com,{mqwang}@sdu.edu.cn

**Abstract.** The hash-based Leighton-Micali Signature (LMS) has been standardized by IETF, NIST and ISO. State-of-the-art non-quantum security proofs for LMS have been given by Katz (SSR 2016) and Fluhrer (Eprint 2017/553). However, the actual structure of SHA-256(/192) is the (chopped) Merkle-Damgård construction using a blockcipher-based Davies-Meyer function, and precise security bounds of SHA-256-based LMS_SHA256_M32 and SHA-256/192-based LMS_SHA256_M24 should be derived in the *ideal-cipher model (ICM)*. This was *not* considered in Katz and Fluhrer's proofs. In fact, by leveraging structural properties of (chopped) Merkle-Damgård, we exhibit forgery attacks against LMS_SHA256_M32 and LMS_SHA256_M24 that expose the inapplicability of Katz's and Fluhrer's random-oracle bounds to the concrete SHA-256 constructions. The precise security guarantees of LMS_SHA256_M32 and LMS_SHA256_M24 thus remain unknown (and are also missing in the standards).

To bridge the structural gap, we provide ICM-based security proofs for the additive-Davies-Meyer abstractions of LMS_SHA256_M32 and LMS_SHA256_M24. The security level of LMS_SHA256_M32 is roughly $256 - \log_2 \widehat{B}$ bits, where $\widehat{B}$ is the number of blocks in SHA-256 calls internally made by LMS_SHA256_M32 and $\widehat{B} \ll 2^{128}$. The security level of LMS_SHA256_M24 is roughly $\min\{192 - \log_2 \widehat{\Gamma}, 256 - \log_2 \widehat{B}\}$ bits, where $\widehat{\Gamma}$ is the number of SHA-256/192 calls internally made by LMS_SHA256_M24 and $\widehat{B}$ is the number of blocks in them, and $\widehat{B} \ll 2^{128}$ and $\widehat{B}\widehat{\Gamma} \ll 2^{192}$. In conclusion, LMS_SHA256_M32 and LMS_SHA256_M24 ensure a moderate (yet not "full") level of security.

## 1 Introduction

**Hash-based Signature LMS.** Hash-based signatures constitute an important category of post-quantum cryptography. A number of hash-based signatures have been standardized in the past decade. Among the stateful

schemes, the Leighton-Micali Signature (LMS) [39] is present in many standards and is widely deployed. LMS and its hierarchical variant HSS were specified in RFC 8554 [41], with additional parameter sets subsequently defined in RFC 9858 [21]. LMS/HSS has further been incorporated into several IETF Standards Track specifications [31,30,24]. LMS/HSS is also standardized as NIST SP 800-208 [11] and ISO/IEC 14888-4:2024 [32]. Its engineering adoption is also substantial: LMS/HSS support has been incorporated into widely used cryptographic software, including OpenSSL, OpenJDK, wolfCrypt, and Caliptra [50,46,52,9]. Moreover, LMS-based secure-boot or firmware-authentication chains have appeared in commercial product lines [45,10,7].

LMS is constructed by combining a variant of Winternitz one-time signature [37,44,42] and several Merkle-trees [44,42]. Both components are built from a hash function $\mathsf{h}$. Instantiating $\mathsf{h}$ with the 32-byte output hash function standard SHA-256, RFC 8554 specifies a group of five instances LMS_SHA256_M32_H$h$, $h \in \{5, 10, 15, 20, 25\}$. The security-critical values, including the Merkle-tree node and the underlying Winternitz-chain state, are also 32 bytes, while $h$ specifies the Merkle-tree height. NIST SP 800-208 and RFC 9858 add three additional groups of instances, including:

- Group LMS_SHA256_M24_H$h$, $h \in \{5, 10, 15, 20, 25\}$, obtained by instantiating $\mathsf{h}$ with the 24-byte output hash standard SHA-256/192;
- Group LMS_SHAKE_M32_H$h$, $h \in \{5, 10, 15, 20, 25\}$, obtained by instantiating $\mathsf{h}$ with the 32-byte output hash standard SHAKE256/256;
- The group LMS_SHAKE_M24_H$h$, $h \in \{5, 10, 15, 20, 25\}$, is obtained by instantiating $\mathsf{h}$ with the 24-byte output hash standard SHAKE256/192.

For simplicity, we will use LMS_SHA256 to refer to LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$ indifferently.

**Provable Security of LMS, and Gaps.** State-of-the-art (non-quantum) security proofs for LMS have been given by Katz [33] and Fluhrer [20].[6]

*Katz's bound in VIL RO Model.* Modeling the hash function as a monolithic variable-input-length (VIL) random oracle (RO) $\mathbf{RO} : \{0,1\}^* \to \{0,1\}^n$, Katz [33, Theorem 2] proved that any adversary making $T$ classical queries to the VIL RO has success probability at most $O(T/2^n)$ to find a forgery for LMS in the multi-user setting, provided that the users' LMS *identifiers are distinct*. However, as emphasized in many papers [14,18,49], the structural properties of practical hash functions, including SHA-256 and SHAKE256/256, may affect such VIL-RO-based results and must be revisited separately.

Fortunately, SHAKE256/256 and SHAKE256/192 adopt the sponge construction [4], which is *indifferentiable* from a VIL RO in the random permutation model [5,38]. Gathering Katz's LMS bound [33, Theorem 2] and Lefevre's

[6] In the quantum regime, Eaton [19] modeled the hash as a monolithic variable-input-length random oracle and proved quantum unforgeability for LMS.

sponge indifferentiability bound [38] (and using composition [40,49]) yields optimal security: if the adversary, resp. the LMS, makes $T$, resp. $\widehat{B}$ random permutation queries during the experiment, then the forgery probability is at most $O(\frac{T}{2^{256}} + \frac{(T+\widehat{B})^2}{2^{512}})$, resp. $O(\frac{T}{2^{192}} + \frac{(T+\widehat{B})^2}{2^{512}})$, in the multi-user setting.[7]

The landscape becomes significantly different for SHA-256 and SHA-256/192. In detail, SHA-256 follows the *strengthened Merkle-Damgård with an additive Davies-Meyer construction* $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ [43,15]: the complete input $U$ is first padded by appending a block containing the length of $U$ and then digested by the cascade construction using an underlying fixed-input-length (FIL) compression function $\mathsf{DM} : \{0,1\}^r \times \{0,1\}^s \mapsto \{0,1\}^s$. The compression function $\mathsf{DM}$ is constructed from the blockcipher SHACAL-2 : $\{0,1\}^{512} \times \{0,1\}^{256} \mapsto \{0,1\}^{256}$ [29] by defining $\mathsf{DM}(W, S_{i-1}) :=$ SHACAL-2$(W, S_{i-1}) \boxplus S_{i-1}$, where $\boxplus$ denotes component-wise addition modulo $2^{32}$ [48, Sect. 6.2.2]. The entire construction is illustrated in Fig. 1 (top). Unfortunately, due to the famous length-extension attack, the construction $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ is *not indifferentiable* [14] from a VIL RO (also mentioned in [21, Sect. 8]). By this, Katz's VIL RO-based provable result [33, Theorem 2] does *not* imply security of LMS_SHA256_M32_H$h$.

On the other hand, SHA-256/192 follows the *chopped Merkle-Damgård with Davies-Meyer construction* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U)$ [14], which computes $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(U)$ but outputs only its leftmost $n = 192 < s$ bits. By [14,27], $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$ is indifferentiable from a VIL RO. With SHA-256/192 parameters, however, the indifferentiability bound is $O(T/2^{64} + T^2/2^{256})$ [27, Theorem 2]. With [33, Theorem 2], this only ensures 64-bit unforgeability security of LMS_SHA256_M24_H$h$, which is insufficient even for NIST's security level 1 (128-bit).

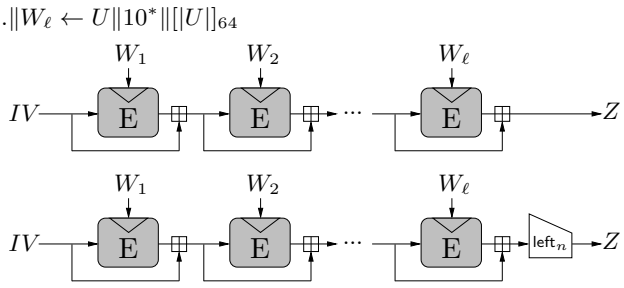



**Fig. 1.** (Top) The structure of SHA-256: strengthened Merkle-Damgård with the additive Davies-Meyer construction $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$; (Bottom) the structure of SHA-256/192: chopped Merkle-Damgård with the additive Davies-Meyer construction $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$. $\boxplus$ denote component-wise addition modulo $2^{32}$ [48, Sect. 6.2.2].

[7] Katz's LMS$^{\mathbf{RO}}$ bound [33] only covers the *distinct user identifier* setting. In Appendix E, we adapt [33] to the *random user identifier* setting and show how it implies 256, resp. 192, bit security for LMS_SHAKE_M32_H$h$, resp. LMS_SHAKE_M24_H$h$.

*Fluhrer.* Fluhrer [20] has already observed the structural gap between "real" LMS and Katz's model. To bridge the gap, Fluhrer employed the *injectively-padded Merkle-Damgård construction* $\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}$ (as shown in Fig. 2) as a refined model for SHA-256. The construction $\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}$ can be derived from the aforementioned Merkle-Damgård construction by: (i) replacing the Davies-Meyer function $\mathsf{DM}$ with a compression function with no accessible internal structure, $\mathbf{F} : \{0,1\}^r \times \{0,1\}^s \to \{0,1\}^s$, and (ii) replacing the strengthened padding with a simpler injective padding $\mathsf{ipad}(U) := U \| sf(U)$. Assuming $\mathbf{F}$ to be a *public fixed-input-length (FIL) RO*, Fluhrer proved security for LMS_SHA256_M32_H$h$ via a dedicated proof [20, Corollary 1].

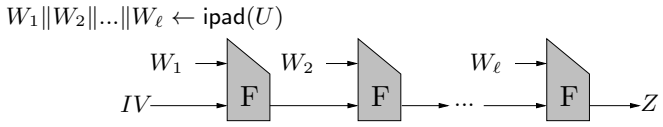



**Fig. 2.** Injectively-padded Merkle-Damgård using a compression function $\mathbf{F}$ with no accessible internal structure, which is employed by Fluhrer [20] as a model of SHA-256.

However, there remains a remarkable gap between the Davies-Meyer compression function $\mathsf{DM}$ in SHA-256 and the FIL RO (with no accessible internal structure) used in [20]. In fact, the structural properties of $\mathsf{DM}^{\text{SHACAL-2}}$ enable non-trivial second preimage attacks in the multi-target setting [16,23,22], which, as will be seen, can be extended to forgery attacks against LMS_SHA256_M32_H$h$ with success probability much higher than Fluhrer's security conclusions [20, Corollary 1]. By this, Fluhrer's results do not apply to LMS_SHA256_M32_H$h$.[8]

*Public indifferentiability.* In many signature constructions, hash functions only process *public inputs* (in particular, the signed message), and the construction $\mathsf{MD}^{\mathsf{DM}^{\mathbf{E}},\mathsf{lpad}}$ is indifferentiable from such a public-use random oracle [18] and this enables deriving security proofs for SHA-256-based signature constructions. However, in LMS (and the underlying Winternitz OTS scheme), the signature secret key is the input of some hash calls, meaning that [18] is *inapplicable*.

*Conclusion, and our question.* In conclusion, provable security of structurally faithful additive-Davies-Meyer models (i.e., $\text{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\text{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$) for LMS_SHA256 remains an open problem. Since LMS_SHA256 has been included in standards [31,30,24,11,32] and has been widely adopted in [50,46,52,9,45,10,7], it is urgent to bridge this gap. We remark that this is challenging: it is fairly easy

[8] As shown recently in [28], Fluhrer's results were *flawed* even if we view $\mathbf{F}$ as a FIL RO, because they neglected the influence of the second-preimage challenge-message length. Fluhrer's conclusions were also contradicted by concrete attacks [28].

to prove an $O(T^2/2^s)$ bound (see page 23), but by using the various sophisticated hash-input separations, LMS is clearly pursuing a much better bound, and we are not aware of a proof of such a bound.

### 1.1 Our Contributions

By exploiting the structural properties of the (chopped) Merkle-Damgård with $\mathsf{DM}^{\text{SHACAL-2}}$ construction, we first exhibit two forgery attacks against LMS_SHA256_M32_H$h$ to contradict Fluhrer's provable results [20]. We also exhibit attacks against LMS_SHA256_M24_H$h$. The attacks further demonstrate the necessity of end-to-end security proofs for the fine-grained models $\text{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\text{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$, in the ideal-cipher model (ICM); our work fills this gap. Our provable bounds are tight for "reasonable" parameters. Our provable security results for $\text{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\text{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ are easily lifted to the (more realistic) query preprocessing model, as well as HSS.

**Forgery Attacks Leveraging Merkle-Damgård Structures.** Our attacks employ two different ideas.

(i) **Fixed-point-based attack:** Gauravaram and Knudsen [23] proposed an attack against enhanced target collision security of the $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ construction, which chooses multiple challenge messages of the same length and uses inverse blockcipher queries $\mathbf{E}_W^{-1}(0^s)$ to create random fixed-points of $\mathsf{DM}$ to "hit" $s$-bit final states of challenge hash-calls. We note that this idea can be applied to find second preimages for $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ and $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$ of certain forms in the LMS setting, and this further yields a forgery for LMS. We slightly extend their idea by removing the restriction of using identical-length challenge messages, achieving applicability to a *known-message* setting.
When the selected signed-message targets yield $\Theta(\Gamma)$ distinct $s$-bit final states, the attack achieves success probability

$$\Omega\left(\min\left\{1, \frac{T\Gamma}{2^{256}}\right\}\right),$$

Fluhrer attempted to prove multi-user security for LMS_SHA256_M32_H$h$ and concluded that its forgery probability is at most $2^{-129}$ even if $\widehat{B} + T = 2^{120}$ [20, Corollary 1]. Consider collecting $\Gamma = 2^{55}$ targets from $2^{30}$ height-25 LMS users, such that every signed-message is short. Using only $T = 2^{115}$ inverse-blockcipher queries, the fixed-point attack already has success-probability $\Gamma T/2^{256} = 2^{-86}$. Even for the largest standardized M32 Winternitz parameter set, the complete key-generation and signing work for these users is below roughly $2^{70}$ blockcipher queries; hence $\widehat{B} + T < 2^{116} \ll 2^{120}$. This illustrates why inverse blockcipher queries and the structural properties of $\mathsf{DM}^{\mathbf{E}}$ must be represented explicitly in the security model.
This also shows that RFC 8554 [41, Sect. 9], which cites Fluhrer's conclusion, *should be revised*. NIST SP 800-208 [11, Appendix C.3] also cites [20] without mentioning Fluhrer's conclusion, and this *may need revision* as well.

(ii) **Diamond-based attack:** we combine the use of diamond structure in [2,47] (diamond was originally proposed in [34,36]), the expandable-message technique of Kelsey-Schneier [35] and the multi-target internal-state technique of Andreeva et al. [1] to obtain forgery attacks against LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$ that outperform our fixed-point-based attack.

For LMS_SHA256_M32_H$h$, when the selected signed-message paths yield $\Theta(B)$ distinct $s$-bit internal states and $T = \Omega\big(2^{128}\sqrt{I}\big)$, the attack achieves success probability

$$\Omega\left(\min\left\{1, \frac{TB}{2^{256}}\right\}\right).$$

For LMS_SHA256_M24_H$h$, we further give an explicit LMS-specific terminal-padding realization that directly targets the chopped outputs, which (for $\Theta(B)$ distinct states and $T = \Omega\big(2^{128}\sqrt{I}\big)$) achieves success probability

$$\Omega\left(\min\left\{1, \max\left\{\frac{TI}{2^{192}}, \frac{TB}{2^{256}}\right\}\right\}\right).$$

While the basic ideas are not new, the applicability to LMS has not been stated explicitly to our knowledge. The diamond-based attack has better success probability in general, but it usually yields not-too-short forgery (see Sect. 3.2 for details). The fixed-point-based attack is useful when both signed and forged messages are restricted to be very short, as may occur in constrained applications.

**Security proofs for LMS$^{\mathsf{sMD}^{\mathbf{DM^E}}}$ and LMS$^{\mathsf{ChopMD}_n^{\mathbf{DM^E}}}$.** We follow the same multi-user unforgeability definition as [33,20]: the adversary succeeds by producing a forgery valid under at least one of the involved user keys. Assume that (i) there are $u$ users with uniformly and independently sampled identifiers; (ii) the adversary makes at most $T$ ideal-cipher queries, and (iii) during the experiment, LMS_SHA256_M32_H$h$, respectively LMS_SHA256_M24_H$h$, internally makes at most $\widehat{I}$ hash-calls and at most $\widehat{B}$ Davies-Meyer evaluations.[9] Under these conditions we prove:

(i) When $T + \widehat{B} \leq 2^{255}$, the forgery probability for LMS_SHA256_M32_H$h$ is at most

$$\frac{26\widehat{B}T + 12\widehat{B}^2}{2^{256}}. \tag{1}$$

[9] In symmetric cryptography, $T$ and $\widehat{B}$ are usually interpreted as time and data complexity.

(ii) When $T + \widehat{B} \leq 2^{191}$, the forgery probability for LMS_SHA256_M24_H$h$ is at most

$$\frac{u^{\mu+1}}{(\mu+1)!\,2^{128\mu}} + \frac{8\mu T + 2\mu\widehat{B} + 4\widehat{B}\widehat{\Gamma}}{2^{192}} + \frac{589824T + 22\widehat{B}T + 8\widehat{B}^2}{2^{256}}$$
$$+ \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^{192}}, \frac{2\mu(T+\widehat{B})^3}{2^{448}}\right\} + \min\left\{\frac{2\widehat{\Gamma}T}{2^{192}}, \frac{2\mu T(T+\widehat{B})^2}{2^{448}}\right\}$$
$$+ \min\left\{\frac{2\widehat{\Gamma}T}{2^{192}}, \frac{1}{2^{192}} + \frac{18560\widehat{\Gamma}T(T+\widehat{B})}{2^{448}}\right\}$$
$$+ \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^{192}}, \frac{1}{2^{192}} + \frac{12\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2^{704}} + \frac{772\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})}{2^{448}}\right\}, \quad (2)$$

where $\mu$ can be any threshold such that $1 \leq \mu \leq \widehat{\Gamma}$;

In Eq. (1), when $B = \Theta(\widehat{B})$, the term $O(\widehat{B}T/2^{256})$ matches the success probability of the aforementioned diamond attack. The term $O(\widehat{B}^2/2^{256})$ appears to be proof overhead, but this may not be the bottleneck since $\widehat{B}$ is not under the adversary's full control and may not be too large in reality. For example, when $u = 2^{40}$ and signed messages have at most $2^{35}$ blocks on average, it can be calculated that $\widehat{B} \leqslant 2^{100}$, so that $\widehat{B}^2/2^{256} \leqslant 2^{-56}$ remains very small. In conclusion, following [3, Sect. 5.6.1.1] and when $\widehat{B} \ll 2^{128}$, the security of LMS_SHA256_M32_H$h$ is of roughly $256 - \log_2 \widehat{B}$ bits (ignoring constant factors), in the sense that constant advantage becomes possible at $T \approx 2^{256}/\widehat{B}$. Importantly, this goes *beyond the 128-bit birthday security* when $\widehat{B}$ is small.

In Eq. (2), assume $u \leq 2^{64}$, setting $\mu = 2$ then makes $\frac{u^{\mu+1}}{(\mu+1)!2^{128\mu}} \leq 2^{-64}$; but the other terms remain substantially more complicated. To illustrate more clearly, we set $\ell_{\max} = 2^{55} + 1$ (as in SHA-256/192) and numerically solve the $T$ and $\widehat{B}$ values satisfying Eq. (2)$= 1/2$ for four choices of average hash-input block length $\widehat{\ell} = \widehat{B}/\widehat{\Gamma} \in \{5, 2^{15}, 2^{30}, 2^{40}\}$, and draw the values as four (red solid) curves in Fig. 3. To ease comparison, we also calculate $T$ and $\widehat{B}$ that make Eq. (1)$= 1/2$ and draw similar (blue dashed) curves in Fig. 3. With the pragmatic approximation $B \approx \widehat{B}$ and $\Gamma \approx \widehat{\Gamma}$ and for each $B$, we also calculate $T$ that makes the better of our two attacks achieve success probability $1/2$, and draw similar (black) curves in Fig. 3.

An important feature of Eq. (2) is that *when $\widehat{\Gamma}$ is small*, the (huge) latter terms in the four minimums *do not dominate*. Consequently,

(i) When $\widehat{\Gamma}$ is sufficiently small such that every minimum yields the former term $O(\widehat{\Gamma}(T+\widehat{B})/2^{192})$, the bound becomes

$$\frac{16T + 4\widehat{B} + 8\widehat{B}\widehat{\Gamma} + 8\widehat{\Gamma}T}{2^{192}} + \frac{589824T + 22\widehat{B}T + 8\widehat{B}^2}{2^{256}}.$$

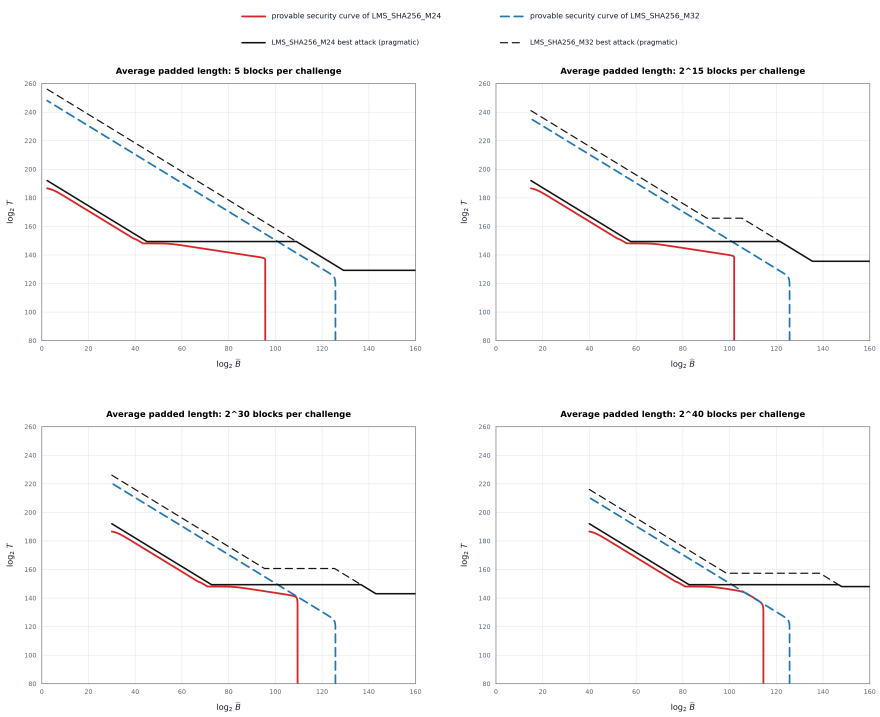



**Fig. 3.** Complexity curves for LMS_SHA256_M24_H$h$ provable security (red solid) and best attack (black solid) and LMS_SHA256_M32_H$h$ provable security (blue dashed) and best attack (black dashed), targeting 1/2-success probability. The four panels use average padded lengths $\widehat{\ell} = \widehat{B}/\widehat{\Gamma} \in \{5, 2^{15}, 2^{30}, 2^{40}\}$ blocks per challenge. For this illustrative comparison only, we use the pragmatic approximation $B \approx \widehat{B}$ and $\Gamma \approx \widehat{\Gamma}$. Curves for $\widehat{B} < \widehat{\ell}$ are omitted because they would give fewer than one challenge, which is meaningless. For LMS_SHA256_M24_H$h$, parameters below the red solid line are secure, while parameters beyond the black solid line are insecure. For LMS_SHA256_M32_H$h$, parameters below the blue dashed line are secure, while parameters beyond the black dashed line are insecure.

Hence, when $\widehat{B} \ll 2^{128}$, ignoring constant factors, LMS_SHA256_M24_H$h$ security is roughly $\min\{192 - \log_2 \widehat{\Gamma}, 256 - \log_2 \widehat{B}\}$ bits. This corresponds to the left halves of the red solid curves in Fig. 3.

(ii) On the other hand, when $\widehat{\Gamma}$ is sufficiently large such that every minimum yields the latter term, the bound becomes

$$\frac{16T + 4\widehat{B} + 4\widehat{B}\widehat{\Gamma} + 2}{2^{192}} + \frac{589824T + 22\widehat{B}T + 8\widehat{B}^2}{2^{256}} + \frac{12\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2^{704}} + \frac{4(T+\widehat{B})^2(2T+\widehat{B}) + (18560 + 772\ell_{\max})\widehat{\Gamma}T(T+\widehat{B})}{2^{448}}.$$

Provided that $\widehat{B} \ll 2^{128}$ and $\widehat{B}\widehat{\Gamma} \ll 2^{192}$, so that the terms independent of $T$ remain negligible, the security of LMS_SHA256_M24_H$h$ in this regime is, ignoring constant factors, roughly $\min\left\{149,\ 256 - \log_2 \widehat{B},\ 162 - \frac{1}{4}\log_2 \widehat{\Gamma}\right\}$ bits.

We stress that the above simplifications are (unavoidably) informal due to the complicated appearance of Eq. (2).

*Discussion.* By the attacks, LMS_SHA256_M32_H$h$, resp. LMS_SHA256_M24_H$h$, does not achieve "full" 256, resp. 192, bit security, which seems to be the goal of the complicated prefix system and can be proven for LMS_SHAKE_M32_H$h$, resp. LMS_SHAKE_M24_H$h$. Thus, SHAKE-based LMS instances are preferable. In applications that desire "full" security but cannot afford the 1600-bit state of SHAKE, the simplest solution is to use SHA-512/256, an instance of $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$ with $s = 512$ and $n = 256$. By [14,27], the indifferentiability bound of SHA-512/256 is $O((T + \widehat{B})/2^{256} + (T + \widehat{B})^2/2^{512})$ [27, Theorem 2]; therefore, with [33, Theorem 2] or Appendix E, $\text{LMS}^{\text{SHA-512/256}}$ ensures 256-bit security.

*Remark on ICM.* We remind the reader that our bounds are proved in the strong ideal-cipher model. Their application to the standardized LMS_SHA256 is therefore heuristic rather than a standard-model guarantee. The bounds nevertheless distinguish adversarial ideal-cipher queries from the cipher calls internally made by LMS, and thus expose the different effects of these two resources.

**Query-preprocessing security, and HSS.** The multi-user unforgeability definition [33,20] used in our main proofs gives the adversary access to the ideal primitive only after the LMS key generation has been invoked for the $u$ users. In practice, however, SHA-256 and SHAKE have been fixed and standardized for decades, and an adversary has evaluated them long before generating a particular user's key. It is thus natural to study *preprocessing security* of LMS.

To this end, note that by [25, Lemma 1], our results imply security of the same independent-secret LMS model in the *query-preprocessing* model of Ghoshal and Tessaro [25, Lemma 1]. Concretely, if the adversary makes at most $T_{\text{pre}}$ primitive queries before the $u$ key generation invocations and at most $T_{\text{post}}$ primitive queries afterwards, then setting $T := T_{\text{pre}} + T_{\text{post}}$ in Eqs. (1) and (2) yields the corresponding preprocessing bounds. Namely, with preprocessing, the total query complexity for attacking this model is not decreased.[10]

We finally follow Fluhrer [20, Sect. 3] and show that our provable bounds on LMS_SHA256 can be lifted to imply the same level of security for HSS, provided that all component LMS keys of all HSS users are generated "once-for-all".

[10] A more popular model [51,17,13,12] limits the length of preprocessing results rather than preprocessing queries, in which LMS security remains an interesting open question.

**Open questions.** We leave further tightening our bounds and extending our treatments to the quantum setting as an important open problem. Regarding the latter, a big obstacle is the security loss incurred by the current compressed permutation/blockcipher oracle techniques [8].

### 1.2 Organization

Sect. 2 gives the notation and models for LMS; Sect. 3 gives the technical overview. In the appendix, Sect. A presents our attacks; Sect. B presents a general framework for security proofs of LMS, and then Sect. C and D prove security for models of LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$. Sect. E addresses random oracle-based LMS. Finally, Sect. G and F discuss query-preprocessing and HSS respectively.

## 2 Preliminaries

Let $\{0,1\}^*$ be the set of all finite bit strings, including the empty string. For $X \in \{0,1\}^*$, let $|X|$ denote its bit length. For two bit strings $X$ and $Y$, $X\|Y$ denotes their concatenation. Let $0^i$ be the string of $i$ zero bits. We write $\mathsf{left}_i(X)$ (resp. $\mathsf{right}_i(X)$) to denote the $i$ leftmost (resp. rightmost) bits of $X$. Given a bit string $X \in \{0,1\}^{\ell r}$, write its decomposition into $r$-bit blocks as $X = X_1\| \cdots \|X_\ell$, $X_i \in \{0,1\}^r$ for $i \in \{1,\ldots,\ell\}$. When $X = \mathsf{lpad}(U)$ is a padded hash input, we normally denote its blocks by $W_1,\ldots,W_\ell$, so that $\mathsf{lpad}(U) = W_1\| \cdots \|W_\ell$. For non-negative integers $i, m$ with $i < 2^m$, we let $[i]_m$ denote the $m$-bit representation of $i$. If $X$ is uniformly distributed over a set $\mathcal{S}$, we write $X \xleftarrow{\$} \mathcal{S}$. Throughout this paper, 8080, 8181, 8282 and 8383 are hexadecimal words.

**Ideal-cipher.** We call a blockcipher with $r$-bit keys and $s$-bit inputs/outputs a $(r,s)$-blockcipher, and denote by $\mathcal{E}_{r,s}$ the set of all $(r,s)$-blockciphers. A blockcipher $\mathbf{E} \xleftarrow{\$} \mathcal{E}_{r,s}$ sampled from $\mathcal{E}_{r,s}$ is an ideal $(r,s)$-blockcipher. Our security proof will be in the ideal-cipher model, in which an ideal $(r,s)$-blockcipher is sampled and made public to all entities, including the adversaries.

### 2.1 Structures of SHA-256 and SHA-256/192 and Models

SHA-256 and SHA-256/192 use a blockcipher SHACAL-2 : $\{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{256}$ [29] for hashing. For generality, we consider a blockcipher $\mathbf{E} : \{0,1\}^r \times \{0,1\}^s \to \{0,1\}^s$ defined on two parameters $r$ and $s$, and generalize components and structures of SHA-256 and SHA-256/192 using $\mathbf{E}$.

**Length-padding.** SHA-256 and SHA-256/192 employ a special type of suffix-free padding function [48, Sect. 5.1.1], which we call *length padding*. Generalizing to using $r$ as block length, on input $U \in \{0,1\}^{<2^{64}}$, $\mathsf{lpad} : \{0,1\}^{<2^{64}} \to (\{0,1\}^r)^+$ first appends the bit 1 to the end of $U$, followed by $x$ zero bits, where $x$ is the

smallest non-negative solution to $|U| + 1 + x \equiv (r - 64) \pmod{r}$, and then appends the 64-bit encoding of $|U|$. We denote its unpadding function by $\mathsf{lpad}^{-1}$, where $\mathsf{lpad}^{-1}(\overline{U})$ returns $\perp$ whenever $\overline{U}$ is not a valid padded string. As a crucial property, writing the padded input as $\mathsf{lpad}(U) = W_1 \| \cdots \| W_\ell$, the 64-bit encoding of $|U|$ always lies in the final padding block $W_\ell$. Setting $r := 512$ yields the concrete padding function used in SHA-256 and SHA-256/192.

Since $|U|$ must be encoded in 64 bits, $\mathsf{lpad}$ limits $|U| < 2^{64}$, which is also required by FIPS PUB 180-4 [48, Sects. 5.1.1 and 6.2]. For SHA-256, this means a padded input $\mathsf{lpad}(U)$ has at most $\frac{|\mathsf{lpad}(U)|}{512} = \left\lceil \frac{|U|+65}{512} \right\rceil \leq 2^{55} + 1$ 512-bit blocks. When defined upon $r$, $\ell_{\max}$ is instead kept symbolic: over its full domain $|U| < 2^{64}$, its largest possible value is $\lceil (2^{64} + 64)/r \rceil$.

**Strengthened/Chopped Merkle-Damgård with Davies-Meyer construction.** SHA-256 and SHA-256/192 are defined over 256-bit state. For generality, we describe their structures using a general parameter $s$ to denote the state size. To this end, identify an $s$-bit chaining state with an element of

$$\mathbb{G}_s := (\mathbb{Z}/2^{32}\mathbb{Z})^{s/32}.$$

For $X, Y \in \mathbb{G}_s$, write $X \boxplus Y$ and $X \boxminus Y$ for componentwise addition and subtraction modulo $2^{32}$. For every positive multiple $n$ of 32 with $n \leq s$, put $\mathbb{G}_n := (\mathbb{Z}/2^{32}\mathbb{Z})^{n/32}$ and use the same symbols for its componentwise operations. The projection $\mathsf{left}_n$ retains complete 32-bit words and is compatible with both operations; in particular,

$$\mathsf{left}_n(X \boxplus Y) = \mathsf{left}_n(X) \boxplus \mathsf{left}_n(Y), \qquad \mathsf{left}_n(X \boxminus Y) = \mathsf{left}_n(X) \boxminus \mathsf{left}_n(Y).$$

For every fixed operand, addition and subtraction are permutations of the state space. Moreover, for fixed $X \in \mathbb{G}_s$ and every $z \in \mathbb{G}_n$, exactly $2^{s-n}$ values $Y$ satisfy $\mathsf{left}_n(X \boxplus Y) = z$; symmetrically, for fixed $Y$, exactly the same number of values $X$ satisfy this equality.

With the above notation, SHA-256 follows the *strengthened Merkle-Damgård with Davies-Meyer construction* $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ [43,15] built upon the blockcipher $\mathbf{E} : \{0,1\}^r \times \{0,1\}^s \to \{0,1\}^s$, the message-length padding function $\mathsf{lpad}$ and a fixed public initialization vector $IV \in \{0,1\}^s$. It consists of three steps:

1. Pad the complete input $U$ with $\mathsf{lpad}$ to obtain the padded input $\overline{U} \leftarrow \mathsf{lpad}(U) \in (\{0,1\}^r)^*$;
2. Write $\overline{U}$ as $\overline{U} = W_1 \| \cdots \| W_\ell$, where every $W_i$ is a $r$-bit block;
3. Set $S_0 := IV$ and, for $i = 1, \ldots, \ell$, compute the Merkle-Damgård chaining values
$$S_i := \mathsf{DM}^{\mathbf{E}}(W_i, S_{i-1}) = \mathbf{E}(W_i, S_{i-1}) \boxplus S_{i-1}.$$
The full hash output is $Z := S_\ell$.

The function SHA-256/192 follows the *chopped Merkle-Damgård with Davies-Meyer construction* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U)$ [14]. The construction $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U)$ outputs a

digest of $n < s$ bits, and is defined upon $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ by

$$\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U) := \mathsf{left}_n\big(\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(U)\big) = \mathsf{left}_n(S_\ell).$$

The constructions $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ and $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$ are illustrated in Fig. 1 (top) and (bottom) respectively.

### 2.2 LMS Signature

LMS is a two level signature scheme, where a one time signature (LM-OTS) is used to sign the message, while a Merkle tree signs the LM-OTS public key. In RFC 8554, the LM-OTS and LMS algorithms and the LMS public key include "typecodes" to determine the parameters $(m, w)$ and $(m, h)$ in use. For clarity, our abstract description makes the parameters $(m, w)$ and $(m, h)$ explicit in the algorithm inputs and public keys.

**General Parameterized Models.** While the concrete instances motivating this paper use the parameters of RFC 8554 and SHA-256, we introduce general models $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ based on $\mathbf{E} : \{0,1\}^r \times \{0,1\}^s \to \{0,1\}^s$ for generality. The two models are families indexed by the parameter tuple

$$(\iota, n, s, r).$$

Here $\iota$ is the bit length of an LMS user identifier, $n$ is the bit length of every LMS hash value, LM-OTS private element, and signing randomizer, and $s$ and $r$ are respectively the chaining-state length and the block length of the iterated hash. The encoded fields $[q]_{32}$, $[d]_{16}$, $[v]_8$, and the 16-bit domain-separation constants retain their RFC widths; the chosen values of $(h, w)$ must therefore keep the corresponding indices within those fields.

The model $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ using $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ has $n = s$, while $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ has $0 < n < s$. To be compatible with the modular addition feed-forward in SHA, $n$ and $s$ are positive multiples of 32, while $\iota$ and $r$ are byte-aligned.

For subsequent elaborations, let $\mathsf{h} : \{0,1\}^* \to \{0,1\}^{8m}$ be a hash function and put $n := 8m$. We first describe $\text{LMS-OTS}^{\mathsf{h}}$, the LMS-OTS scheme based on $\mathsf{h}$, and then use it to describe $\mathrm{LMS}^{\mathsf{h}}$, the LMS scheme based on $\mathsf{h}$.

**Description of $\text{LMS-OTS}^{\mathsf{h}}$.** For $w \in \{1, 2, 4, 8\}$, put $e \leftarrow 2^w - 1$. The $n$-bit output of $\mathsf{h}$ can be viewed as a sequence of $\frac{n}{w}$ integers, each $w$ bits long. Set $\xi \leftarrow \left\lceil \left( \lfloor \log_2 \left( (2^w - 1) \cdot \frac{n}{w} \right) \rfloor + 1 \right) / w \right\rceil$ and $p \leftarrow \frac{n}{w} + \xi$. Define a function $\mathsf{checksum} : (\{0,1\}^w)^{n/w} \to \{0,1\}^{w\xi}$ as follows:

$$\mathsf{checksum}(D[0], \ldots, D[n/w - 1]) := \sum_{i=0}^{\frac{n}{w}-1} (2^w - 1 - D[i]) \ ,$$

where each $D[i] \in \{0,1\}^w$ is viewed as an integer in $\{0, \ldots, 2^w - 1\}$ and the result is expressed as an integer using exactly $w\xi$ bits. Define the LM-OTS scheme as follows.

*Key-generation algorithm* LMSOTSGen. LMSOTSGen takes as input $I, (m, w), q$, where $m = n/8$ is the hash-output length in bytes, $I \in \{0,1\}^{\iota}$ is an identifier, $w \in \{1, 2, 4, 8\}$ controls the space/time trade-off, and $q$ is a 32-bit diversification factor. Here $\iota$ is independent of both $s$ and $n$; the standardized LMS parameter sets use $\iota = 128$. We require the resulting Winternitz chain count to satisfy $p \leq 2^{15}$, so every chain index is separated from the reserved 16-bit domain-separation values. Set $n \leftarrow 8m$, and determine $\xi$ and $p$ from $(m, w)$ as above. The algorithm proceeds as follows:

(1) Choose $p$ uniform values $X_0^0, \ldots, X_0^{p-1} \in \{0,1\}^n$.
(2) For $d = 0$ to $p-1$ and $i = 0$ to $e-1$, compute

$$X_{i+1}^d \leftarrow \mathsf{h}\big(I \parallel [q]_{32} \parallel [d]_{16} \parallel [i]_8 \parallel X_i^d\big).$$

(3) Compute $pk \leftarrow \mathsf{h}(I \parallel [q]_{32} \parallel [8080]_{16} \parallel X_e^0 \parallel \ \ldots \ \parallel X_e^{p-1})$.

The public key is $pk$, and the private key is $sk = \big(X_0^0, \ldots, X_0^{p-1}\big)$.

Sampling $X_0^0, \ldots, X_0^{p-1}$ from $\{0,1\}^n$ uniformly and independently in Step (1) follows RFC 8554 [41]. NIST SP 800-208 [11] profiles LMS using the pseudorandom key-generation method of RFC 8554, Appendix A. Consequently, when applying our provable bounds to NIST SP 800-208, one should add the pseudorandom security bounds of the method of RFC 8554 to obtain the final bounds. The formal discussion is out of the scope of this paper.

*Signing algorithm* LMSOTSSign. It inputs a private key $sk = \big(X_0^0, \ldots, X_0^{p-1}\big)$, a message $M \in \{0,1\}^*$ as usual, as well as $I, (m, w), q$ as above. It uses the following steps:

(1) Choose $R$ uniformly at random from $\{0,1\}^n$.
(2) Compute $D \leftarrow \mathsf{h}(I \parallel [q]_{32} \parallel [8181]_{16} \parallel R \parallel M)$ and $c \leftarrow \mathsf{checksum}(D)$. Set $V \leftarrow D \| c$, and parse $V$ as a sequence of $w$-bit integers $V_0, \ldots, V_{p-1}$
(3) For $d = 0$ to $p-1$, do the following. For $i = 0$ to $V_d - 1$, compute

$$X_{i+1}^d \leftarrow \mathsf{h}\big(I \parallel [q]_{32} \parallel [d]_{16} \parallel [i]_8 \parallel X_i^d\big).$$

After the inner loop, set $\sigma_d \leftarrow X_{V_d}^d$.
(4) Return the LM-OTS signature $\sigma = (R, \sigma_0, \ldots, \sigma_{p-1})$.

*Verification algorithm* LMSOTSVrfy. Verification takes as input a message $M \in \{0,1\}^*$, an LM-OTS signature $\sigma = (R, \sigma_0, \ldots, \sigma_{p-1})$, and $I, (m, w), q$ as above. It involves the following steps:

(1) Compute $D \leftarrow \mathsf{h}(I \parallel [q]_{32} \parallel [8181]_{16} \parallel R \parallel M)$ and $c \leftarrow \mathsf{checksum}(D)$. Set $V \leftarrow D \| c$, and parse $V$ as a sequence of $w$-bit integers $V_0, \ldots, V_{p-1}$
(2) For $d = 0$ to $p-1$, set $X_{V_d}^d \leftarrow \sigma_d$, and then, for $i = V_d$ to $e-1$, compute

$$X_{i+1}^d \leftarrow \mathsf{h}\big(I \| [q]_{32} \| [d]_{16} \| [i]_8 \| X_i^d\big).$$

(3) Output $\mathsf{h}(I \parallel [q]_{32} \parallel [8080]_{16} \parallel X_e^0 \parallel \ \ldots \ \parallel X_e^{p-1})$.

We note that, in contrast to the usual convention, LMSOTSVrfy returns a string rather than a bit and does not take a public key as input. A signature $\sigma$ on some message $M$ is valid relative to some fixed public key $pk$ if the output of LMSOTSVrfy is equal to $pk$.

One can verify that correctness holds in the following sense: for any $I, q$, and $(sk, pk)$ output by $\mathsf{LMSOTSGen}(I, (m, w), q)$, and any message $M$, we have:

$$\mathsf{LMSOTSVrfy}\big(M, \mathsf{LMSOTSSign}(sk, M, I, (m, w), q), I, (m, w), q\big) = pk\,.$$

**Description of LMS$^{\mathsf{h}}$.** An instance of the LMS$^{\mathsf{h}}$ scheme is defined by computing a Merkle tree of height $h$ using $2^h$ LM-OTS public keys at the leaves. We give a formal definition now.

*Key-generation algorithm* LMSGen. LMSGen takes as input the hash length $m = n/8$, the parameter pair $(h, w)$, and a value $I \in \{0,1\}^{\iota}$. It proceeds as follows:

(1) For $q = 0, \ldots, 2^h - 1$, compute $(pk_q, sk_q) \leftarrow \mathsf{LMSOTSGen}(I, (m, w), q)$.
(2) For $\rho = 2^h, \ldots, 2^{h+1} - 1$, set $T[\rho] \leftarrow \mathsf{h}(I \parallel [\rho]_{32} \parallel [8282]_{16} \parallel pk_{\rho - 2^h})$.
(3) For $\rho = 2^h - 1, \ldots, 1$, set $T[\rho] \leftarrow \mathsf{h}(I \parallel [\rho]_{32} \parallel [8383]_{16} \parallel T[2\rho] \parallel T[2\rho + 1])$.

The public key is $pk = (m, h, w, I, T[1])$, and the private key is

$$sk = \big(sk_0, \ldots, sk_{2^h - 1}, T[1..2^{h+1} - 1]\big).$$

*Signing algorithm* LMSSign takes a private key $\big(sk_0, \ldots, sk_{2^h-1}, T[1..2^{h+1}-1]\big)$, an integer $0 \le q < 2^h$, a message $M \in \{0,1\}^*$, an identifier $I$ and a pair of parameters $(m, w)$ as above as inputs. The algorithm proceeds as follows:

(1) Compute $\sigma \leftarrow \mathsf{LMSOTSSign}(sk_q, M, I, (m, w), q)$.
(2) Set $p_0, \ldots, p_{h-1}$ as $p_i \leftarrow T[\lfloor (q + 2^h)/2^i \rfloor \oplus 1]$.
(3) Return the LMS signature $\Sigma = (q, \sigma, p_0, \ldots, p_{h-1})$.

*Verification algorithm* LMSVrfy. LMSVrfy inputs a public key $(m, h, w, I, T)$, a message $M \in \{0,1\}^*$, and an LMS signature $\Sigma = (q, \sigma, p_0, \ldots, p_{h-1})$, where $\sigma = (R, \sigma_0, \ldots, \sigma_{p-1})$. The algorithm proceeds as follows:

(1) Compute $pk \leftarrow \mathsf{LMSOTSVrfy}(M, \sigma, I, (m, w), q)$.
(2) Set $\rho \leftarrow q + 2^h$, and compute $T[\rho] = \mathsf{h}\,(I \| [\rho]_{32} \| [8282]_{16} \| pk)$.
(3) For $i = 1..h$, set $\rho \leftarrow \lfloor (q + 2^h)/2^i \rfloor$, and compute $T[\rho]$ as follows:
   - $T[\rho] \leftarrow \mathsf{h}(I \parallel [\rho]_{32} \parallel [8383]_{16} \parallel T[2\rho] \parallel p_{i-1})$ if $\lfloor q/2^{i-1} \rfloor$ is even
   - $T[\rho] \leftarrow \mathsf{h}(I \parallel [\rho]_{32} \parallel [8383]_{16} \parallel p_{i-1} \parallel T[2\rho + 1])$ if $\lfloor q/2^{i-1} \rfloor$ is odd
4. Return 1 if and only if $T[1] = T$.

This verification procedure works because, if the signature is valid, all $T$ elements computed during the verification procedure match the corresponding $T$ elements of the private key.

**The** m-unforge **experiment** $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathbf{Sign}^{\mathbf{hP}},\mathscr{A},m,u}$:[11]

1 **Initialization:**
  (1) Set $n \leftarrow 8m$, sample an ideal oracle $\mathbf{p} \xleftarrow{\$} \mathcal{PS}$;
  (2) For $\mathsf{i} = 1, \ldots, u$, select the public parameters $(h^{\mathsf{i}}, w^{\mathsf{i}})$ independently of the ideal oracle, the user identifiers, and all key-generation randomness. Sample $I^{\mathsf{i}} \xleftarrow{\$} \{0,1\}^{\iota}$ and invoke $(pk_{\mathsf{i}}, sk_{\mathsf{i}}) \leftarrow \text{SIGN}^{\mathbf{P}}.\mathsf{Gen}(m, (h^{\mathsf{i}}, w^{\mathsf{i}}), I^{\mathsf{i}})$.

2 **Interaction:** $(\mathsf{i}^*, M^*, \Sigma^*) \leftarrow \mathscr{A}^{\mathsf{muSign}^{\mathbf{P}},\mathbf{P}}((I^1, pk_1), ..., (I^u, pk_u))$, where the oracle $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}, q, M^{\mathsf{i},q})$ returns $\text{SIGN}^{\mathbf{P}}.\mathsf{Sign}(sk_{\mathsf{i}}, q, M^{\mathsf{i},q}, I^{\mathsf{i}})$ when $1 \le \mathsf{i} \le u$, $0 \le q < 2^{h^{\mathsf{i}}}$ and $M^{\mathsf{i},q}$ lies in the message domain of SIGN, and $\perp$ otherwise.
  – **Adversarial restriction**: for each valid pair $(\mathsf{i}, q)$, $\mathscr{A}$ makes at most one query of the form $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}, q, \star)$.

3 **Finalization:** Parse $\Sigma^* = (q^*, \sigma^*, p^*_0, \ldots, p^*_{h^{\mathsf{i}^*}-1})$, output 1 if and only if: (i) $1 \le \mathsf{i}^* \le u$, (ii) $0 \le q^* < 2^{h^{\mathsf{i}^*}}$, (iii) $M^*$ lies in the message domain of SIGN, (iv) no signing query $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}^*, q^*, M^*)$ was made before, and (v) $\text{SIGN}^{\mathbf{P}}.\mathsf{Vrfy}(pk_{\mathsf{i}^*}, M^*, \Sigma^*) = 1$.

**Fig. 4:** Experiment for multi-user unforgeability of LMS-like signatures.

**Implicit restrictions on $(\iota, n, s, r)$.** Standard LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$ have $(\iota, n, s, r) = (128, 256, 256, 512)$ and $(128, 192, 256, 512)$ respectively. The choices not only affect concrete security, but also fix the "block layout" of involved hash-calls. For example, in both LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$, every call in $\mathsf{LMSOTSGen}$ Step (2) makes exactly 1 call to SHACAL-2, while every call in $\mathsf{LMSGen}$ Step (3) makes exactly 2 calls to SHACAL-2. For simplicity, our security proofs follow these particular "block layouts". Our proofs thus require additional restrictions on $(\iota, n, s, r)$. However, to simplify presentations, we *implicitly* assume that the parameter tuple $(\iota, n, s, r)$ in use *results in* the same "block layouts" as LMS_SHA256_M32_H$h$ and LMS_SHA256_M24_H$h$.

### 2.3 Multi-user Unforgeability Definition for LMS-like Schemes

In the main body of this paper, we follow the multi-user unforgeability definition used by Katz [33] and Fluhrer [20]. We use the experiment in Fig. 4, which unifies both the ROM and ICM instances. It is parameterized by an ideal-oracle $\mathbf{p}$ sampled from a corresponding space $\mathcal{PS}$ and by an LMS-like signature scheme SIGN that uses $\mathbf{p}$. Thus, in the ICM used in Sect. C and D, $\mathbf{p} = \mathbf{E}$ comprises both forward and inverse access to $(\mathbf{E}, \mathbf{E}^{-1})$, whereas in the ROM used in Appendix E, $\mathbf{p} = \mathbf{RO}$ comprises the single random-oracle interface.

[11] The parameter $m \in \{32, 24\}$ is an input to the LMS key-generation algorithm [41] and determines the output length and concrete security level. We fix $m$ in each experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{SIGN}^{\mathbf{hP}},\mathscr{A},m,u}$ and regard it as a parameter of the experiment.

## 3 Technical Overview

### 3.1 Graph Representation of Ideal-Cipher Query Transcript

This section focuses on $\mathrm{LMS}^{\mathsf{h}^{\mathbf{E}}}$, i.e., LMS model using ideal-cipher-based hash functions (except for the proof Framework in Sect. 3.4). For convenience, we first introduce a graph representation of an ideal-cipher query transcript. Concretely, during the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\textsc{Sign}^{\mathsf{h}^{\mathbf{E}}},\mathscr{A},m,u}$, every ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ or $X \leftarrow \mathbf{E}_W^{-1}(Y)$ accesses a unique triple $(W; X, Y)$, called an *ideal-cipher record*, or simply a *record*, and it then holds $X \boxplus Y = \mathsf{DM}^{\mathbf{E}}(W, X)$. We view every such Davies-Meyer evaluation as a labeled directed edge $\big(X \xrightarrow{W} X \boxplus Y\big)$ and $\{0,1\}^s$ as the vertex set; together, these constitute the *transcript graph* $\mathcal{G}$. Adjacent edges then form labeled *paths* in the transcript graph, and every evaluated hash input $Z \leftarrow \mathsf{h}^{\mathbf{E}}(U)$ yields a path $\big(IV \xrightarrow{\mathsf{lpad}(U)} Z \| R\big)$ for some $R \in \{0,1\}^{s-n}$. A path $\big(S_1 \xrightarrow{MM} S_2\big)$ is of *length-d*, if $|MM| = d \times r$, i.e., $MM$ consists of $d$ $r$-bit blocks.

### 3.2 Our Attack Approach

This section considers $\mathsf{h}^{\mathbf{E}} \in \{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}, \mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}\}$. Following the notation in Sect. 2.3, select $\Gamma$ queries $\Sigma^j \leftarrow \mathsf{muSign}^{\mathbf{E}}(\mathsf{i}_j, q^j, M^j)$, $j \in \{1, \ldots, \Gamma\}$. Put

$$\mathsf{prefix}_j := I^{\mathsf{i}_j} \| [q^j]_{32} \| [8181]_{16} \,.$$

During the signing query, $\text{LMS-OTS}^{\mathsf{h}^{\mathbf{E}}}$ samples $R^j \xleftarrow{\$} \{0,1\}^n$, computes $D^j \leftarrow \mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_j \| R^j \| M^j)$, and signs this digest. Consequently, a fixed-prefix second preimage consists of an index $j \in \{1, \ldots, \Gamma\}$ and strings $R', M'$ satisfying

$$M' \neq M^j, \qquad \mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_j \| R' \| M') = \mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_j \| R^j \| M^j) \,.$$

Outputting $M'$ and replacing the randomizer $R^j$ in $\Sigma^j$ by $R'$ then yields a forgery for user $\mathsf{i}_j$. The attacker may choose among the $\Gamma$ targets, but its second preimage must retain the prefix of the target it ultimately breaks; hence this is a *fixed-prefix, multi-target second-preimage problem*.

Without the fixed-prefix requirement, $\Gamma$ distinct target digests give the usual multi-target scale $\Omega(\min\{1, \Gamma T/2^n\})$. LMS's domain-separated prefixes are intended to prevent a candidate for one target from being reused against all the others. If $\mathsf{h}$ is a monolithic RO, a fixed-prefix search making $T$ RO queries has probability $O(T/2^n)$, as long as prefix repetitions are rare. If $\mathsf{h}^{\mathbf{E}} \in \{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}, \mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}\}$, however, generic second-preimage techniques can be adapted and combined with LMS formatting to bypass this separation.

For the subsequent elaborations, assume that the $\Gamma$ selected signed-message hashing calls have $W_1^{(j)} \| \cdots \| W_{\ell_j}^{(j)} = \mathsf{lpad}(\mathsf{prefix}_j \| R^j \| M^j)$, and let

$$S_0^{(j)} = IV, \quad S_i^{(j)} = \mathsf{DM}^{\mathbf{E}}(W_i^{(j)}, S_{i-1}^{(j)}) \,.$$

**Finding FPSpr using Random Fixed-Points.** A crucial property of the additive Davies-Meyer function $\mathsf{DM}^{\mathbf{E}}(W,S) := \mathbf{E}_W(S) \boxplus S$ is that an inverse blockcipher query $S_W^{\mathsf{fp}} \leftarrow \mathbf{E}_W^{-1}(0^s)$ yields a random fixed point of $\mathsf{DM}^{\mathbf{E}}$. Once $S_W^{\mathsf{fp}} = S_{\ell_{j^*}}^{(j^*)}$ for some $j^*$, "hashing" the padded input $W_1^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)} \| W$ also yields $S_{\ell_{j^*}}^{(j^*)}$ as the final state, and thus produces the same hash output as $\mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_{j^*} \| R^{j^*} \| M^{j^*})$.

However, to have $\mathsf{lpad}^{-1}(W_1^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)} \| W) \neq \perp$, $W$ must be *padding-compatible*, i.e., it must properly encode the bit length of the extended unpadded input. Following the definition of length-padding in Sect. 2.1, we use a byte-aligned terminal profile that reserves $1 + 7 + 64 = 72$ bits of the final block for the delimiter, seven zero bits, and the 64-bit length field. It therefore leaves $r-72$ message bits in that block (for LMS_SHA256, we have $r = 512$ and $r-72 = 440$).

With this in mind, in another closely related setting (enhanced target collision resistance), Gauravaram and Knudsen [23] use equal-length challenge messages and padding-compatible fixed-point blocks. Their ideas can be applied to $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$. Concretely, for $D \in \{0,1\}^{r-72}$ define

$$W_D := D \| 1 \| 0^7 \| [\ell_1 r + (r-72)]_{64}.$$

Thus $W_D$ has exactly $r$ bits and carries the correct length field, provided that $\ell_1 r + (r-72) < 2^{64}$. Query $S_D^{\mathsf{fp}} \leftarrow \mathbf{E}_{W_D}^{-1}(0^s)$, and choose the $\Gamma$ signed messages so that their padded inputs have one common length $\ell_1 = \cdots = \ell_\Gamma$. Then $S_D^{\mathsf{fp}} = S_{\ell_j}^{(j)}$ gives a valid extension for $\mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_j \| R^j \| M^j)$. If a constant fraction of these targets have distinct final chaining states, then $T$ distinct fresh inverse ideal-cipher queries achieve a success probability of roughly $\Omega(\min\{1, \Gamma T/2^s\})$.

We extend this idea to avoid the common-length restriction and fit into short-message restrictions. Concretely, assume that the application (e.g., an IoT protocol) sets $\ell_{\mathrm{app}}$, $2 \leq \ell_{\mathrm{app}} \leq \ell_{\max}$, as an upper bound on the padded length of both signed and forged inputs. Then, choose $L$ as the largest positive integer satisfying $L \leq \ell_{\mathrm{app}} - 1$ and $Lr + (r-72) < 2^{64}$, and repeatedly pick distinct $D \in \{0,1\}^{r-72}$ and set $W_D := D \| 1 \| 0^7 \| [Lr + r - 72]_{64}$ to query $X_D \leftarrow \mathbf{E}_{W_D}^{-1}(0^s)$. Once $X_D = S_{\ell_{j^*}}^{(j^*)}$ for any $j^*$ with padded length $\ell_{j^*} \leq L$, it means "hashing" the padded input

$$W_1^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)} \| \underbrace{W_D \| \cdots \| W_D}_{L - \ell_{j^*} + 1 \text{ copies}}$$

also yields $S_{\ell_{j^*}}^{(j^*)}$ as the final state. This leverages the property that a fixed-point block can be arbitrarily repeated in such a chain without affecting its final state. Moreover, (i) The old padding block $W_{\ell_{j^*}}^{(j^*)}$ and every copy of $W_D$ except the last one are treated simply as message data; only the last copy of $W_D$ is interpreted as the terminal padding block. This means the length information is properly encoded. (ii) $W_1^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}$ naturally contains *the right prefix value.*

Consequently,

$$\mathsf{prefix}_{j^*}\|R^{j^*}\|M' = \mathsf{lpad}^{-1}(W_1^{(j^*)}\|\cdots\|W_{\ell_{j^*}}^{(j^*)}\|\underbrace{W_D\|\cdots\|W_D}_{L-\ell_{j^*}+1 \text{ copies}})$$

provides an FPSpr for $\mathsf{prefix}_{j^*}\|R^{j^*}\|M^{j^*}$.

Let $\gamma$ be the number of distinct final chaining states among the qualifying targets with $\ell_j \le L$. Thus, $T$ fresh inverse trials succeed with probability roughly $\Omega(\min\{1, \gamma T/2^s\})$. When a constant fraction of the selected targets qualify and have distinct final states, so that $\gamma = \Theta(\Gamma)$, this yields $\Omega(\min\{1, \Gamma T/2^s\})$.

**Finding FPSpr using Diamonds and Expandable-Message.** For simplicity, assume that $\Gamma = 2^d$ for some integer $d$. Put $B := \sum_{j=1}^{\Gamma} \ell_j$ for the number of padded blocks in these selected signed-message inputs. The preceding fixed-point attack can use at most $\Gamma$ targets, whereas the following attacks can exploit (most of) the $B$ internal states.

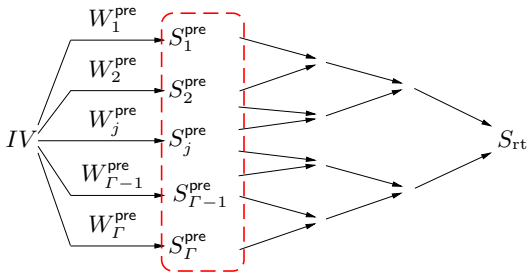



Fig. 5: Two steps in the **Constructing a diamond** phase of the diamond-based attack. Every rightward arrow represents a Davies-Meyer evaluation. The red dashed rectangle groups the states $S_1^{\mathsf{pre}}, \ldots, S_\Gamma^{\mathsf{pre}}$ produced in Step 1. These states form the leaf set of the binary merge tree in Step 2, although their placement in the tree need not preserve the index order $1, \ldots, \Gamma$. Every vertex with indegree 2 consists of a Davies-Meyer collision and consumes about $2^{s/2}$ queries.

Concretely, for $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$ without chopping, the attack consists of three phases.

(i) **Constructing a diamond.** For every target index $j$, choose an $r$-bit first block $W_j^{\mathsf{pre}} = \mathsf{prefix}_j\|\star$ and compute $S_j^{\mathsf{pre}} \leftarrow \mathsf{DM}^{\mathbf{E}}(W_j^{\mathsf{pre}}, IV)$. Construct a balanced chosen-start diamond [34,36,2,47] that "merges" these states into a single $s$-bit state $S_{\mathrm{rt}}$. Concretely, such a diamond is a balanced binary tree taking $S_1^{\mathsf{pre}}, ..., S_\Gamma^{\mathsf{pre}}$ as leaves and $S_{\mathrm{rt}}$ as root, as depicted in Fig. 5.
This binary tree has $2^d - 1 = \Gamma - 1$ vertexes with indegree 2, and each of them corresponds to a $s$-bit Davies-Meyer collision. However, by a dedicated

approach, the binary tree or diamond can be constructed using $\widetilde{O}(2^{s/2}\sqrt{\Gamma})$ ideal-cipher queries [36, Sect. 2].
Once such a structure is constructed, it provides a length-$(d+1)$ path

$$\left(IV \xrightarrow{W_j^{\mathsf{pre}} \| msg_j} S_{\mathrm{rt}}\right)$$

for every $j$, where $W_j^{\mathsf{pre}} \| msg_j$ preserves the LMS prefix required for the $j$ th target. By this, the $\Gamma$ prefixes are "linked" to the same $s$-bit state $S_{\mathrm{rt}}$, and subsequently constructed edges $\left(S_{\mathrm{rt}} \xrightarrow{\star} \star\right)$ would simultaneously address all the prefixes.

(ii) **Constructing an expandable-message.** Put $k = \lfloor \log_2 \ell_{\min} \rfloor$, where $\ell_{\min} = \min_j \ell_j$. Starting from $S_0^{\mathsf{exp}} := S_{\mathrm{rt}}$, construct the Kelsey-Schneier expandable message [35], as shown in Fig. 6 (Top). Concretely, at level $t \in \{1, \ldots, k\}$, use a birthday search to construct a length-1 path and a length-$(2^{k-t}+1)$ path from $S_{t-1}^{\mathsf{exp}}$ to the same state $S_t^{\mathsf{exp}}$.
Once completed, for every

$$b \in \{k, k+1, \ldots, k+2^k-1\},$$

choosing the length-1 or longer path independently at the $k$ levels yields a length-$b$ path $\left(S_{\mathrm{rt}} \xrightarrow{E^{(b)}} S_k^{\mathsf{exp}}\right)$, where $|E^{(b)}| = br$. It costs $O(k2^{s/2} + 2^k)$ ideal-cipher queries to construct this expandable message [35, Sect. 3.2].

(iii) **Seeking a full-state bridge.** Try fresh "bridge blocks" $W_r^{\mathsf{bridge}}$ for $r = 1, 2, \ldots$, until

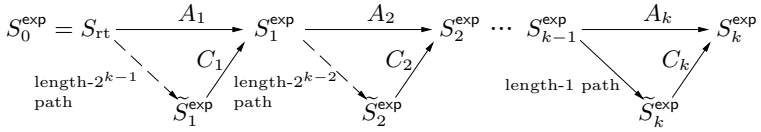
$$\mathsf{DM}^{\mathbf{E}}(W_{r^*}^{\mathsf{bridge}}, S_k^{\mathsf{exp}}) = S_{i^*}^{(j^*)}$$

for some challenge index $j^* \in \{1, ..., \Gamma\}$ and block index $i^*$ satisfying $d+k+2 \le i^* \le \min\{d+k+2^k+1, \ell_{j^*}-2\}$, as shown in Fig. 6 (Bottom). Namely, the target-prefix path $\left(IV \xrightarrow{W_1^{(j^*)} \| \cdots \| W_{i^*}^{(j^*)}} S_{i^*}^{(j^*)}\right)$ is of length-$i^*$, while the remaining suffix path $\left(S_{i^*}^{(j^*)} \xrightarrow{W_{i^*+1}^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}} S_{\ell_{j^*}}^{(j^*)}\right)$ is of length-$(\ell_{j^*} - i^*)$. By restricting $i^* \le \ell_{j^*} - 2$, we ensure that $\left(S_{i^*}^{(j^*)} \xrightarrow{W_{i^*+1}^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}} S_{\ell_{j^*}}^{(j^*)}\right)$ contains the final two blocks $W_{\ell_{j^*}-1}^{(j^*)}$ and $W_{\ell_{j^*}}^{(j^*)}$, which may carry SHA padding suffix of the $j^*$ th challenge, unchanged.
Once such a "hit" is found, it then chooses the corresponding diamond branch and expandable-message length and forms

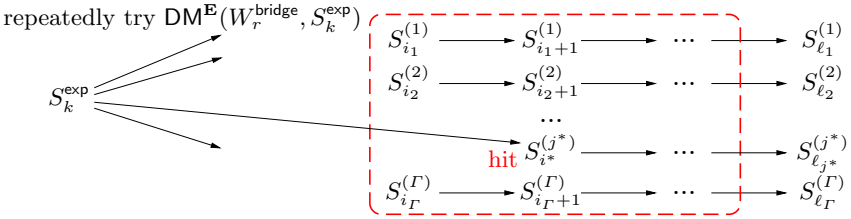
$$\overline{U}' = \underbrace{W_{j^*}^{\mathsf{pre}}}_{1 \text{ block}} \| \underbrace{msg_{j^*}}_{d \text{ blocks}} \| \underbrace{E^{(i^*-d-2)}}_{\text{choose to be } i^*-d-2 \text{ blocks}} \| \underbrace{W_{r^*}^{\mathsf{bridge}}}_{1 \text{ block}} \| \underbrace{W_{i^*+1}^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}}_{\ell_{j^*}-i^* \text{ blocks}}.$$

This string begins with $\mathsf{prefix}_{j^*}$ and reaches the target state $S_{i^*}^{(j^*)}$ before "copying" the challenge suffix $\left(S_{i^*}^{(j^*)} \xrightarrow{W_{i^*+1}^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}} S_{\ell_{j^*}}^{(j^*)}\right)$. Moreover, its

block count is $1 + d + (i^* - d - 2) + 1 + (\ell_{j^*} - i^*) = \ell_{j^*}$. Hence the length information encoded in $W^{(j^*)}_{\ell_{j^*}-1} \| W^{(j^*)}_{\ell_{j^*}}$ remains correct. Consequently, unpadding $\overline{U}'$ yields a valid fixed-prefix second preimage and an LMS forgery.




phase 2: Constructing an expandable-message




phase 3: Seeking a full-state bridge

Fig. 6: The **Constructing an expandable-message** and **Seeking a full-state bridge** phases of the diamond-based attack. Every dash lines represent a sequence of Davies-Meyer evaluations. Every vertex with indegree 2 consists of a Davies-Meyer collision and consumes about $2^{s/2}$ queries. The red dashed rectangle denotes the set of valid "hitting targets" in **Seeking a full-state bridge** phase.

It can be seen that

$$N_{\mathrm{hit}} := \left| \left\{ S_i^{(j)} : 1 \le j \le \Gamma,\ d + k + 2 \le i \le \min\{d + k + 2^k + 1, \ell_j - 2\} \right\} \right|$$

counts the number of distinct internal states that can function as valid "hitting targets" in **Seeking a full-state bridge** phase. Conditional on successful diamond and expandable-message setup and on that $W_r^{\mathsf{bridge}}$ is "fresh", each "bridge trial" $\mathsf{DM}^{\mathbf{E}}(W_r^{\mathsf{bridge}}, S_k^{\mathsf{exp}})$ produces a $s$-bit random output. Consequently, with $T$ "bridge trials", the probability to have a "hit" is roughly $\Omega(\min\{1, TN_{\mathrm{hit}}/2^s\})$.

Not every internal state on every challenge path can serve as a valid "hitting target". Nevertheless, when $\ell_1, ..., \ell_\Gamma$ are comparable and sufficiently large and (informally) state collisions are rare, we can roughly have $N_{\mathrm{hit}} = \Theta(B)$. Therefore, as long as the adversarial query complexity $T$ is sufficient to cover **Constructing a diamond** and **Constructing an expandable-message** (which roughly requires $T = \widetilde{\Omega}(2^{s/2}\sqrt{\Gamma} + k2^{s/2} + 2^k)$), the final success probability is roughly $\Omega(\min\{1, TN_{\mathrm{hit}}/2^s\})$.

For the chopped construction, the preceding full-state bridge attack remains applicable. A probably more effective attack instead directly targets the $\Gamma$ chopped message digests. Concretely, the attack follows the above **Constructing a diamond** phase to construct the prefix-merging diamond. Then, in the **Seeking terminal-hit phase**, it samples $T$ values $D \in \{0,1\}^{r-72}$ uniformly without replacement, defines the terminal padded block

$$W_D^{\mathrm{term}} := D \| 1 \| 0^7 \| [(1+d)r + (r-72)]_{64}$$

for each of them, and computes $S_D^{\mathrm{term}} \leftarrow \mathsf{DM}^{\mathbf{E}}(W_D^{\mathrm{term}}, S_{\mathrm{rt}})$ to seek $\mathsf{left}_n\big(S_D^{\mathrm{term}}\big) = \mathsf{left}_n\big(S_{\ell_{j^*}}^{(j^*)}\big)$ for some challenge index $j^*$. Once such a "hit" is found, the string $\overline{U}'' = W_{j^*}^{\mathsf{pre}} \| msg_{j^*} \| W_D^{\mathrm{term}}$ is a valid padded hash input, since $W_D^{\mathrm{term}}$ encodes the correct SHA padding suffix. Moreover, $\overline{U}''$ begins with $\mathsf{prefix}_{j^*}$. Therefore, unpadding $\overline{U}''$ yields a fixed-prefix second preimage for the $j^*$ th challenge.

If the chosen targets contain $\Theta(\Gamma)$ distinct digests, then $T$ terminal trials roughly gives "hitting" probability $\Omega(\min\{1, \Gamma T/2^n\})$.

As long as the adversarial query complexity $T$ is sufficient to cover **Constructing a diamond** and **Constructing an expandable-message** (which roughly requires $T = \widetilde{\Omega}(2^{s/2}\sqrt{\Gamma} + k2^{s/2} + 2^k)$), the attacker can choose the better attack approach among seeking a full-state bridge and seeking terminal-hit phase, and the better of them achieves success probability

$$\Omega\left(\min\left\{1,\ \max\left\{\frac{BT}{2^s}, \frac{\Gamma T}{2^n}\right\}\right\}\right).$$

### 3.3 Fluhrer's Proof Approach and Gaps

As mentioned in the introduction, Fluhrer [20] attempted to prove m-unforge security for $\mathrm{LMS}^{\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}}$ using a random function-based Merkle-Damgård construction $\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}$. To this end, Fluhrer [20] introduced a two-step proof approach: first, proving a *multi-target fixed-prefix (second-)preimage (FPMSpr)* security statement for $\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}$ [20, Theorem 1, Corollary 1]; second, deriving generic security for $\mathrm{LMS}^{\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}}$ from the FPMSpr statement for $\mathsf{MD}^{\mathbf{F},\mathsf{ipad}}$.

Despite the structural gap between $\mathbf{F}$ and $\mathsf{DM}^{\mathbf{E}}$, it is still natural to ask if Fluhrer's two-step approach remains applicable to $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$. However, this is hindered by an additional conceptual gap. In Fluhrer's FPMSpr experiment [20, Sect. 2.1], the $i$ th so-called Group-1 challenge hash-input is generated by *sampling* both its secret input $K_i$ and its *target* $Z_i$ independently and uniformly, and then *programming the random compression function* $\mathbf{F}$ so that

$$\mathbf{F}\big(IV, \mathsf{ipad}(\mathsf{prefix}_{1,i} \| K_i)\big) = Z_i\,.$$

One of the adversarial winning conditions is to either recover $K_i$ or find a second preimage for $K_i$ under the prefix $\mathsf{prefix}_{1,i}$. It turns out that Group-1 challenge inputs are intended to capture the secrecy of Winternitz chain states. However,

this is not how Winternitz chain states are generated in LM-OTS. Starting from a sampled secret $X_{d,0} = x_d$, every Winternitz-chain value

$$X_{d,v+1} = \mathsf{MD}^{\mathbf{F},\mathsf{ipad}}\big(I\|[q]_{32}\|[d]_{16}\|[v]_8\|X_{d,v}\big) = \mathbf{F}\big(IV, \mathsf{ipad}(I\|[q]_{32}\|[d]_{16}\|[v]_8\|X_{d,v})\big)$$

is computed by honestly calling $\mathbf{F}$. While replacing both $X_{d,v}$ and $X_{d,v+1}$ by uniformly random strings may yield indistinguishable distributions, the formal argument is somewhat intricate and is not included in [20].

### 3.4 Our Proof Approach

Motivated by the above discussion, we directly analyze the hash-calls that are actually made during LMS key generation, signing, and verification in the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\text{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$, which avoids the gap in [20]. Our proof also consists of two steps. The first step is general for any instantiation of $\mathsf{h}^{\mathbf{P}}$: it reduces successful forgery to two events $\mathsf{BadGuess}_{\mathsf{h}^{\mathbf{P}}}$ and $\mathsf{BadFPMSpr}_{\mathsf{h}^{\mathbf{P}}}$ on the hash-calls in $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$. The second step is to bound the probabilities of the events $\mathsf{BadGuess}_{\mathsf{h}^{\mathbf{P}}}$ and $\mathsf{BadFPMSpr}_{\mathsf{h}^{\mathbf{P}}}$ for $\mathsf{h}^{\mathbf{P}} \in \{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}, \mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}\}$. These are elaborated in the following three subsections in turn.

**A Proof Framework for LMS-like Signatures.** Following Fluhrer [20], we normalize $\mathscr{A}$ as follows. After $\mathscr{A}$ outputs forgery $\big(\mathsf{i}^*, M^*, (q^*, \sigma^*, p_0^*, \ldots, p_{h-1}^*)\big)$, we require $\mathscr{A}$ to: (i) run $\mathsf{LMSVrfy}(pk_{\mathsf{i}^*}, M^*, \Sigma^*)$, to "materialize" all primitive queries needed to verify its eventual forgery; (ii) query $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}^*, q^*, M^{\mathsf{i}^*,q^*})$ for arbitrary $M^{\mathsf{i}^*,q^*} \neq M^*$, if no query of the form $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}^*, q^*, \star)$ has been made. The ideal-cipher queries needed for these auxiliary calls are included in its query complexity $T$ (thus, this proof-level convention for $T$ is slightly more inclusive than the informal convention used in Sect. 1.1). This normalization simplifies the subsequent proof presentation.

Then, every hash-call made by $\text{LMS}^{\mathsf{h}^{\mathbf{P}}}$ during $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$ is viewed as a "challenge" to $\mathscr{A}$. By LMS construction, every such call is of the form $\mathsf{h}^{\mathbf{P}}(\mathsf{prefix}\|M)$ for some specific choice of $\mathsf{prefix}$. The aforementioned two events are defined on these hash-calls.

Event $\mathsf{BadGuess}_{\mathsf{h}^{\mathbf{P}}}$ occurs when the *adversarially visible* $\mathbf{p}$ queries and answers allow the evaluation of some challenge hash-call in Winternitz chains that are still supposed to be hidden to $\mathscr{A}$. Note that all calls in Winternitz chains are originally hidden to $\mathscr{A}$, but some of them are subsequently given to $\mathscr{A}$ during signing. For this, we dynamically maintain a set $\mathcal{HS}$ for these hidden calls.

Event $\mathsf{BadFPMSpr}_{\mathsf{h}^{\mathbf{P}}}$ occurs when the $\mathbf{p}$ queries and answers appeared during $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$ allow the evaluation of some hash-call $Z = \mathsf{h}^{\mathbf{P}}(\mathsf{prefix}\|M')$, such that there exists a challenge hash-call $Z = \mathsf{h}^{\mathbf{P}}(\mathsf{prefix}\|M)$ with the same prefix and output and $M' \neq M$.

Following a valid forgery backwards through the Merkle authentication path and then through the Winternitz chains, we can prove that $\mathscr{A}$ succeeds in forging only if either $\mathsf{BadGuess}_{\mathsf{h}^{\mathbf{P}}}$ or $\mathsf{BadFPMSpr}_{\mathsf{h}^{\mathbf{P}}}$ occurs. Briefly, at the first "position"

where the forged computation differs from its honest reference, the forgery either supplies an undisclosed Winternitz predecessor (indicating $\mathsf{BadGuess}_{\mathsf{hP}}$) or supplies a fixed-prefix second preimage for some challenge hash-call (indicating $\mathsf{BadFPMSpr}_{\mathsf{hP}}$). This is Lemma 1 in Sect. B.

**The Simpler Case: $\mathbf{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$.** In $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, the hash output length equals the chaining-state length, $n = s$. We follow the graph representation in Sect. 3.1.

We identify several simpler bad events in $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$ to help bound the probabilities of $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, particularly the latter. A naïve approach is to define an event BadCol that occurs as long as two distinct records $(W; X, Y)$ and $(W'; X', Y')$ have $X \boxplus Y = X' \boxplus Y'$, and it is fairly easy to prove $\neg\mathsf{BadCol} \Rightarrow \neg\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$. However, this results in an inferior bound $O\big((T + \widehat{B})^2/2^s\big)$, and more fine-grained bad events are needed.

To this end, we distinguish *honest ideal-cipher queries* from *adversarial ideal-cipher queries*: an honest ideal-cipher query is made internally by $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, whereas an adversarial ideal-cipher query is made directly by the (normalized) adversary $\mathscr{A}$.

An ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$, resp. $X \leftarrow \mathbf{E}_W^{-1}(Y)$, is *fresh*, if there is no record of the form $(W; X, \star)$, resp. $(W; \star, Y)$, right before this query is made. Such a fresh ideal-cipher query *materializes* a record $(W; X, Y)$ and fixes the record's *first-materialization time*. Otherwise, the ideal-cipher query does not materialize new record. Consequently, a record may be accessed by several honest or adversarial ideal-cipher queries.

With the above terminology, we identify three events. BadHonest concerns honest ideal-cipher queries: it occurs if during the first evaluation of a collected "challenge" hash-call,[12] an honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ finds its corresponding record $(W; X, Y)$ already present, has $X \boxplus Y \in \{IV, X\}$ or has $X \boxplus Y \in \{X', X' \boxplus Y'\}$ for a previously materialized record $(W'; X', Y')$. BadFwd and BadBwd concern adversarial ideal-cipher queries: BadFwd occurs if a fresh adversarial ideal-cipher query in the forward direction $Y \leftarrow \mathbf{E}_W(X)$ finds $X \boxplus Y = IV$ or $X \boxplus Y \in \{X', X' \boxplus Y'\}$ for a previously materialized record $(W'; X', Y')$ accessed by an honest ideal-cipher query for a challenge, while BadBwd occurs if a fresh adversarial ideal-cipher query in the backward direction $X \leftarrow \mathbf{E}_W^{-1}(Y)$ finds $X = IV$ or $X \boxplus Y = IV$ or $X \in \{X', X' \boxplus Y'\}$ or $X \boxplus Y \in \{X', X' \boxplus Y'\}$ for a previously materialized record $(W'; X', Y')$ accessed by an honest ideal-cipher query for a challenge.

The intuitions behind BadHonest are as follows. By LMS construction, in every challenge hash-call, the key $W_1$ of the first underlying ideal-cipher query $\mathbf{E}_{W_1}(IV)$ contains an associated $s$-bit (pseudo)random "anchor" value: in the

[12] We emphasize *first evaluation* because by LMS construction, many challenge calls are to be evaluated during both key generation and signing, and the latter evaluations necessarily access records already materialized during key generation.

first hash-call of a Winternitz chain, this "anchor" is an initial chain value sampled in Step (1) of $\mathsf{LMSOTSGen}$; in a message-hashing call $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}\|R\|M)$, this "anchor" is the $s$-bit randomizer $R$; in every other hash-call, this "anchor" is the $s$-bit output of a previous hash-call, which is pseudorandom as long as $\mathsf{BadHonest}$ never occurred w.r.t. previous hash-calls. Therefore, the first query $\mathbf{E}_{W_1}(IV)$ collides with existing record with probability $O\big((T+\widehat{B})/2^s\big)$.

If no such record exists, the query $Y_1 \leftarrow \mathbf{E}_{W_1}(IV)$ is fresh and yields an $s$-bit random output, and the probability to observe the other events on $IV \boxplus Y_1$ is $O\big((T+\widehat{B})/2^s\big)$. Moreover, if this hash-call internally makes more than 1 ideal-cipher queries, then the next query $\mathbf{E}_{W_2}(IV \boxplus Y_1)$ uses a pseudorandom $s$-bit "plaintext" input, and the same bound $O\big((T+\widehat{B})/2^s\big)$ can be established in a similar vein to the above. By iterating this argument, the bound $O\big((T+\widehat{B})/2^s\big)$ can be proven for all honest ideal-cipher queries underlying one challenge hash-call. A union bound over all $\widehat{B}$ honest queries yields the bound $O\big((T+\widehat{B})\widehat{B}/2^s\big)$ on the probability of $\mathsf{BadHonest}$.

We stress that the above argument is merely an informal overview. Our formal analysis of $\mathsf{BadHonest}$ in Appendix C.3 follows a much more complicated approach insisted by GPT 5.6 Sol Ultra, which first proves a general lemma (Lemma 2 (i)) stating that the "anchor" is uniform in at least $2^s - T - \widehat{B}$ possibilities conditioned on the previous query transcript, and then applies this lemma to analyze $\mathsf{BadHonest}$.

On the other hand, $\mathsf{BadFwd}$ and $\mathsf{BadBwd}$ (only) concern collisions among an honest ideal-cipher query and an adversarial query, and the probability is clearly $O\big(T\widehat{B}/2^s\big)$. Due to the diamond attack (see Sect. 3.2), $O\big(T\widehat{B}/2^s\big)$ has been the best possible result. Since none of the events concern collision among adversarial queries, we avoid $O(T^2/2^s)$ compared to the aforementioned naïve approach.

If $\mathsf{BadHonest}$ never occurred, the only possibility to have $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ is that an adversarial query $\mathbf{E}_W(IV)$ or $\mathbf{E}_W^{-1}(Y)$ has $W$ "hit" a hidden hash-call in Winternitz chains. By construction, every hash-call in Winternitz chains consists of a single honest ideal-cipher query. By the aforementioned intuitions on $\mathsf{BadHonest}$, every such ideal-cipher query has an $s$-bit (pseudo)random "anchor" in its key. Recall that hidden hash-calls have not been disclosed to $\mathscr{A}$ during signing queries. This means conditioned on the *adversarial-known* ideal-cipher query transcript, the "anchor" in every hidden hash-call remains uniform in $\approx 2^s$ possibilities, and thus the bound $O\big(T\widehat{B}/2^s\big)$ after taking union over $T$ adversarial queries. Again, our formal analysis in Appendix C.4 follows a complicated approach insisted by GPT 5.6 Sol Ultra that first proves the conditional uniformity of the "anchor" (Lemma 2 (ii)) and then applies this lemma to analyze $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$.

It remains to prove $\neg(\mathsf{BadHonest} \vee \mathsf{BadFwd} \vee \mathsf{BadBwd}) \Rightarrow \neg\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$. With the graph representation, the occurrence of $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ implies the

existence of a challenge hash-call path $\big(IV \xrightarrow{\mathsf{prefix}\|M} S\big)$ and an alternative path $\big(IV \xrightarrow{\mathsf{prefix}\|M'} S\big)$, $M' \neq M$. Assume that the states in $\big(IV \xrightarrow{\mathsf{prefix}\|M} S\big)$ are

$$S_0 = IV, S_1, ..., S_{\ell-1}, S_\ell.$$

Delete the maximal common initial sequence of the two paths, and follow the alternative path until its first re-entry into the honest state set $\{S_0, ..., S_\ell\}$: assume that this re-entry enters the state $S_j$. The alternative path thus contains a record $(W'; X', Y')$ with $X' \boxplus Y' = S_j$, which means $S_j \neq IV$, otherwise one of the bad events occurs. This further means that the challenge path also contains a record $(W_j; S_{j-1}, Y_j)$ with $S_{j-1} \boxplus Y_j = S_j$.

Now, if $(W'; X', Y')$ is materialized before $(W_j; S_{j-1}, Y_j)$, then the query $\mathbf{E}_{W_j}(S_{j-1})$ in the challenge hash-call causes $\mathsf{BadHonest}$ (either $\mathbf{E}_{W_j}(S_{j-1})$ is non-fresh, or $\mathbf{E}_{W_j}(S_{j-1})$ materializing $(W_j; S_{j-1}, Y_j)$ with $S_{j-1} \boxplus Y_j = X' \boxplus Y'$). If $(W'; X', Y')$ is materialized later, then $S_{j-1} \boxplus Y_j = X' \boxplus Y'$ means it causes $\mathsf{BadHonest}$, $\mathsf{BadFwd}$ or $\mathsf{BadBwd}$, depending on whether $(W'; X', Y')$ is materialized by an honest query or an adversarial query. This proves $\neg(\mathsf{BadHonest} \vee \mathsf{BadFwd} \vee \mathsf{BadBwd}) \Rightarrow \neg\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$. Injecting the above terms into the proceeding proof framework yields the final bound. Formally,

**Theorem 1 (Multi-user unforgeability of $\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$).** *During the experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$*, assume that: (i) the adversary* $\mathscr{A}$ *makes at most* $T$ *adversarial ideal-cipher queries to* $\mathbf{E}$ *(including the auxiliary queries for our proof); (ii)* $\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$ *internally makes at most* $\widehat{\Gamma}$ *calls to* $\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}$*, which results in at most* $\widehat{B}$ *Davies-Meyer calls (it thus holds* $1 \leq \widehat{\Gamma} \leq \widehat{B}$*). Further assume* $T + \widehat{B} \leq 2^s/2$*. Then, the construction* $\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$ *has*

$$\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\big] \leq \frac{26\widehat{B}T + 12\widehat{B}^2}{2^s}.$$

**What Changes for $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}$.** For the case of $n < s$, a straightforward idea is to redefine the events $\mathsf{BadHonest}$, $\mathsf{BadFwd}$ and $\mathsf{BadBwd}$ used for $\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$ to address leftmost $n$-bit collisions instead of $s$-bit collisions. However, this would give rise to an inferior bound $O\big((T + \widehat{B})\widehat{B}/2^n\big)$. The central refinement is therefore to "charge" an $n$-bit collision probability only if genuinely needed.

To elaborate, we use the following idea: we follow the aforementioned overview for $\mathrm{LMS}^{\mathsf{sMD}^{\mathrm{DM}^{\mathbf{E}}}}$ to see where it fails for $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}$ and how to remedy.

First, the events $\mathsf{BadFwd}$, $\mathsf{BadBwd}$ and $\mathsf{BadHonest}$ are retained. $\mathsf{BadFwd}$ and $\mathsf{BadBwd}$ contribute the term $O\big(T\widehat{B}/2^s\big)$ to the bound. For $\mathsf{BadHonest}$, the argument for the non-first underlying ideal-cipher queries is also non-changed and contributes $O\big((T + \widehat{B})\widehat{B}/2^s\big)$. However, for the first ideal-cipher query $\mathbf{E}_{W_1}(IV)$

in every challenge hash-call, the (pseudo)random "anchor" value in the key $W_1$ is of only $n$ bits, and the same argument as before yields the inferior term $O\big((T+\widehat{B})\widehat{B}/2^n\big)$.

Fortunately, $W_1$ also contains the prefix value of this challenge hash-call. The prefix of LMS is designed to be: (i) distinct hash-calls made for the same user are using distinct prefix values; (ii) prefixes from distinct users contain distinct identifiers that are uniform and independent. By this, we introduce another event $\mathsf{BadID}_\mu$ for that some prefix value appears in $\mu+1$ or more users, where $\mu$ functions as a threshold. The probability of $\mathsf{BadID}_\mu$ is bounded to $\frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}}$.

Conditioned on $\neg\mathsf{BadID}_\mu$, for every first query $\mathbf{E}_{W_1}(IV)$ we only need to consider the probability that $W_1$ collides with the subset of records $(W;X,Y)$ with $W$ sharing the same prefix with $W_1$. Eventually, the probability is reduced to $O\big(\mu(T+\widehat{B})/2^n\big)$, which is perfect since $\mu$ can be very small.

Similarly, conditioned on $\neg\mathsf{BadID}_\mu$, the probability of $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}$ is bounded to $O\big(\mu(T+\widehat{B})/2^n+(T+\widehat{B})\widehat{B}/2^s\big)$ (Sect. D.7), which is better than the term $O\big((T+\widehat{B})\widehat{B}/2^n\big)$ derived without $\neg\mathsf{BadID}_\mu$.

Next, we split the event $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}$ into sub-events: $\mathsf{BadHonSpr}_n$ covers an alternative (second preimage) path first completed by a later challenge hash-call; $\mathsf{PreChSpr}$ covers an alternative path already present when the honest challenge terminal query record is materialized (and its leftmost $n$ bits collide with the alternative path); and $\mathsf{PostChSpr}$ covers an alternative path completed afterwards by an adversarial ideal-cipher query. It can be proven that

$$\neg(\mathsf{BadHonest}\vee\mathsf{BadFwd}\vee\mathsf{BadBwd}\vee\mathsf{BadHonSpr}_n\vee\mathsf{PreChSpr}\vee\mathsf{PostChSpr})\\ \Longrightarrow\neg\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}.$$

For $\mathsf{BadHonSpr}_n$, conditioned on $\neg\mathsf{BadHonest}$, it can be proved that the newly materialized record $(W;X,Y)$ occurs exactly once in the alternative path and must be its terminal record. Thus, $\mathsf{BadHonSpr}_n$ occurs only if $\mathsf{left}_n(X\boxplus Y)=Z$ for some challenge path $\big(IV\xrightarrow{\mathsf{prefix}\|M}Z\|R\big)$, with probability $O(\widehat{B}\widehat{\Gamma}/2^n)$ (page 66), which is better than the straightforward bound $O(\widehat{B}^2/2^n)$.

For the remaining, we gain motivations from [28, Appendix F.2]: they considered LMS using a permutation-based hash, restrict the number of collisions $\big((W;X,Y),(W';X',Y')\big)$, $X\boxplus Y=X'\boxplus Y'$, to a small integer threshold $C$, and then prove $C$-dependent combinatorial properties for the transcript graph, which may be helpful in our case. But this approach only yields an essentially-birthday term $O\big(\big((T+\widehat{B})^2/2^s\big)^C\big)$. To remedy, we replace $C$ with the quantity *excess indegree* $\kappa(\mathcal{G})$ of the transcript graph $\mathcal{G}$. Concretely, let $d^-(S)$ be the indegree of the vertex $S$ in $\mathcal{G}$, then

$$\kappa(\mathcal{G}):=\sum_{S\in\mathcal{G}}\max\{d^-(S)-1,0\}.$$

Using exponential-moment, it can be proven

$$\mathbb{E}[\kappa(\mathcal{G})] \le \frac{(T+\widehat{B})(T+\widehat{B}-1)}{2(2^s-T-\widehat{B})}, \quad \Pr\left[\kappa(\mathcal{G}) > \frac{\mathrm{e}(T+\widehat{B})(T+\widehat{B}-1)}{2(2^s-T-\widehat{B})}+n.\right] \le \frac{1}{2^n}.$$

Then, adapting [28, Lemma 6] yields: (i) the paths from $IV$ to a particular vertex begin with at most $\kappa(\mathcal{G})+1$ distinct prefix values, (ii) the total number of pairs $(P,S)$, where $P$ is a collected prefix value and $S$ is a vertex reachable from $IV$ by a path whose label begins with $P$, is at most $(\kappa(\mathcal{G})+1)(T+\widehat{B})$, and (iii) at most $(\kappa(\mathcal{G})+1)(\ell_{\max}-1)$ starting states distinct from a prescribed terminal state can reach it through a sufficiently short old tail. If the terminal state itself is additionally admitted through a non-empty cycle, this last bound increases by at most one. These facts are stated in Lemma 6 in Sect. D.4.

By this, we have two approaches to bound $\mathsf{PreChSpr}$. First, directly counting target challenges yields an $O(\widehat{\Gamma}(T+\widehat{B})/2^n)$ bound. Second, using prefix multiplicity $\mu$ and excess indegree yields the term $O(\mu(T+\widehat{B})/2^n+\mu(T+\widehat{B})^3/2^{n+s})$. Taking the minimal yields (formal analysis is in Sect. D.8)

$$O\bigg(\min\Big\{\frac{\widehat{\Gamma}(T+\widehat{B})}{2^n}, \frac{\mu(T+\widehat{B})}{2^n}+\frac{\mu(T+\widehat{B})^3}{2^{n+s}}\Big\}\bigg).$$

Subsequently, during organizing the final bound, the latter term $O(\mu(T+\widehat{B})/2^n)$ is extracted from the minimum, and the remaining constitutes the first minimum in Eq. (2) (and in Theorem 2 below).

For $\mathsf{PostChSpr}$, the straightforward approach again yields $O(\widehat{\Gamma}(T+\widehat{B})/2^n)$. To have a fine-grained bound, we distinguish three cases: (i) an adversarial forward query $Y \leftarrow \mathbf{E}_W(X)$ first causes $\mathsf{PostChSpr}$, and the resulting record $(W;X,Y)$ is the terminal record in the alternative path; (ii) an adversarial backward query $X \leftarrow \mathbf{E}_W^{-1}(Y)$ first causes $\mathsf{PostChSpr}$, and $(W;X,Y)$ is the terminal record in the alternative path; (iii) the adversarial record $(W;X,Y)$ is non-terminal in the alternative path. This time, the analysis produces a very complicated term, and we could only refer to Lemma 12 in Sect. D.9 for details. Gathering all the above terms eventually yields:

**Theorem 2 (Multi-user unforgeability of $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$).** *During the experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$*, assume that: (i) the adversary* $\mathscr{A}$ *makes at most* $T$ *adversarial ideal-cipher queries to the ideal-cipher* $\mathbf{E}$ *(including the auxiliary queries for our proof); (ii)* $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ *internally makes at most* $\widehat{\Gamma}$ *calls to* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$*, which results in at most* $\widehat{B}$ *Davies-Meyer calls; (iii) for every hash-call* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U)$*, the padded input* $\mathsf{lpad}(U)$ *has at most* $\ell_{\max}$ $r$*-bit blocks. Put* $Q := T+\widehat{B}$*, and further assume* $Q \le 2^{n-1}$*. Then, for every integer*

*threshold* $\mu \in \{1, ..., \widehat{\Gamma}\}$*, the construction* $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ *has*

$$\begin{aligned}
&\Pr\left[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\right] \\
&\leq \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{8\mu T + 2\mu\widehat{B} + 4\widehat{B}\widehat{\Gamma}}{2^n} + \frac{16n^2T + 22\widehat{B}T + 8\widehat{B}^2}{2^s} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n}, \frac{2\mu(T+\widehat{B})^3}{2^{n+s}}\right\} + \min\left\{\frac{2\widehat{\Gamma}T}{2^n}, \frac{2\mu T(T+\widehat{B})^2}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}T}{2^n}, \frac{1}{2^n} + \frac{(128+96n)\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n}, \frac{1}{2^n} + \frac{12\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2^{2s+n}} + \frac{4(n+1)\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\}.
\end{aligned}$$

## References


1. Andreeva, E., Bouillaguet, C., Dunkelman, O., Fouque, P.A., Hoch, J.J., Kelsey, J., Shamir, A., Zimmer, S.: New Second-Preimage Attacks on Hash Functions. Journal of Cryptology **29**(4), 657–696 (Oct 2016). https://doi.org/10.1007/s00145-015-9206-4
2. Antonov, S.: Round 3 Official Comment: SPHINCS+. Post to the NIST PQC Forum (Apr 2022), 20 April 2022, https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/FVItvyRea28/m/mGaRi5iZBwAJ
3. Barker, E.: Recommendation for Key Management: Part 1 – General. NIST Special Publication 800-57 Part 1 Rev. 5, National Institute of Standards and Technology (May 2020). https://doi.org/10.6028/NIST.SP.800-57pt1r5, https://doi.org/10.6028/NIST.SP.800-57pt1r5
4. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Sponge Functions. In: ECRYPT hash workshop. vol. 2007 (2007)
5. Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the Indifferentiability of the Sponge Construction. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 181–197 (Apr 2008). https://doi.org/10.1007/978-3-540-78967-3_11
6. Blackburn, S.R., Stinson, D.R., Upadhyay, J.: On the complexity of the herding attack and some related attacks on hash functions **64**(1-2), 171–193 (2012). https://doi.org/10.1007/s10623-010-9481-x
7. Broadcom Inc.: Emulex SecureHBA Delivers the Industry's First Autonomous In-Flight Encryption with Post-Quantum Cryptography. Solution Brief (2025), https://docs.broadcom.com/doc/securehba-encryption-sb
8. Carolan, J.: Compressed Permutation Oracles. Cryptology ePrint Archive, Paper 2025/1734 (2025), https://eprint.iacr.org/2025/1734, Appear at ACM STOC 2026.
9. CHIPS Alliance: Caliptra: Firmware Image Format and Verification. Caliptra Specification and Open-Source Root-of-Trust Implementation (2026), https://github.com/chipsalliance/Caliptra/blob/main/doc/Caliptra.md
10. Cisco Systems: BIOS Protection: LMS-Signed Bootloader. Trustworthy Systems Guide, Cisco IOS XE (2026), https://www.cisco.com/c/en/us/td/docs/switches/lan/c9000/sec-crypto/tws/trustworthy-systems-guide/bios-protection.html

11. Cooper, D.A., Apon, D.C., Dang, Q.H., Davidson, M.S., Dworkin, M.J., Miller, C.A., et al.: Recommendation for Stateful Hash-Based Signature Schemes. NIST Special Publication **800**(208), 800–208 (2020)
12. Coretti, S., Dodis, Y., Guo, S.: Non-Uniform Bounds in the Random-Permutation, Ideal-Cipher, and Generic-Group Models. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 693–721 (Aug 2018). https://doi.org/10.1007/978-3-319-96884-1_23
13. Coretti, S., Dodis, Y., Guo, S., Steinberger, J.P.: Random Oracles and Non-uniformity. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 227–258 (Apr / May 2018). https://doi.org/10.1007/978-3-319-78381-9_9
14. Coron, J.S., Dodis, Y., Malinaud, C., Puniya, P.: Merkle-Damgård Revisited: How to Construct a Hash Function. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 430–448 (Aug 2005). https://doi.org/10.1007/11535218_26
15. Damgård, I.: A Design Principle for Hash Functions. In: Brassard, G. (ed.) CRYPTO'89. LNCS, vol. 435, pp. 416–427 (Aug 1990). https://doi.org/10.1007/0-387-34805-0_39
16. Dean, R.D.: Formal Aspects of Mobile Code Security. Princeton University (1999)
17. Dodis, Y., Guo, S., Katz, J.: Fixing Cracks in the Concrete: Random Oracles with Auxiliary Input, Revisited. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part II. LNCS, vol. 10211, pp. 473–495 (Apr / May 2017). https://doi.org/10.1007/978-3-319-56614-6_16
18. Dodis, Y., Ristenpart, T., Shrimpton, T.: Salvaging Merkle-Damgård for Practical Applications. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 371–388 (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_22
19. Eaton, E.: Leighton-Micali Hash-Based Signatures in the Quantum Random-Oracle Model. Cryptology ePrint Archive, Report 2017/607 (2017), https://eprint.iacr.org/2017/607
20. Fluhrer, S.: Further Analysis of a Proposed Hash-Based Signature Standard. Cryptology ePrint Archive, Report 2017/553 (2017), https://eprint.iacr.org/2017/553
21. Fluhrer, S., Dang, Q.: RFC 9858: Additional Parameter Sets for HSS LMS Hash-Based Signatures (2025)
22. Gauravaram, P., Hirose, S., Stebila, D.: Security Analysis of a Design Variant of Randomized Hashing. In: Batten, L., Kim, D.S., Zhang, X., Li, G. (eds.) Applications and Techniques in Information Security - 8th International Conference, ATIS 2017, Auckland, New Zealand, July 6-7, 2017, Proceedings. Communications in Computer and Information Science, vol. 719, pp. 14–22. Springer (2017). https://doi.org/10.1007/978-981-10-5421-1_2, https://doi.org/10.1007/978-981-10-5421-1_2
23. Gauravaram, P., Knudsen, L.R.: On Randomizing Hash Functions to Strengthen the Security of Digital Signatures. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 88–105 (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_5
24. Geest, D.V., Bashiri, K., Fluhrer, S., Gazdag, S., Kousidis, S.: RFC 9802: Use of the HSS and XMSS Hash-Based Signature Algorithms in Internet X.509 Public Key Infrastructure (2025)
25. Ghoshal, A., Tessaro, S.: The Query-Complexity of Preprocessing Attacks. pp. 482–513. LNCS (2023). https://doi.org/10.1007/978-3-031-38545-2_16
26. Guido, B., Joan, D., Michaël, P., Gilles, V.: Cryptographic Sponge Functions. 2011-STMicroelectronics NXP Semiconductors, Version 0.1 January 14 (2011)

27. Gunsing, A.: Block-Cipher-Based Tree Hashing. pp. 205–233. LNCS (2022). https://doi.org/10.1007/978-3-031-15985-5_8
28. Guo, C., Hu, K., Jiang, S., Fan, Y., Fu, Y., Preneel, B., Wang, M.: Permutation-Based Hash from Non-Idealized Assumptions: Adding Feed-Forward to Sponge. Cryptology ePrint Archive, Paper 2025/1006 (2025), https://eprint.iacr.org/2025/1006, appeared at CRYPTO 2026.
29. Handschuh, H., Naccache, D.: SHACAL (-submission to NESSIE-). In: Proceedings of First Open NESSIE Workshop (2000), http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.3.4066&rep=rep1&type=pdf
30. Housley, R.: RFC 9708: Use of the HSS/LMS Hash-Based Signature Algorithm in the Cryptographic Message Syntax (CMS) (2025)
31. Housley, R., Security, V.: RFC 8778: Use of the HSS/LMS Hash-Based Signature Algorithm with CBOR Object Signing and Encryption (COSE) (2020)
32. Information technology – Security techniques – Digital signatures with appendix – Part 4: Stateful hash-based mechanisms. International Standard ISO/IEC 14888-4:2024, International Organization for Standardization, Geneva, Switzerland (2024)
33. Katz, J.: Analysis of a Proposed Hash-Based Signature Standard. In: International conference on research in security standardisation. pp. 261–273. Springer (2016)
34. Kelsey, J., Kohno, T.: Herding Hash Functions and the Nostradamus Attack. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 183–200 (May / Jun 2006). https://doi.org/10.1007/11761679_12
35. Kelsey, J., Schneier, B.: Second Preimages on n-Bit Hash Functions for Much Less than 2n Work. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 474–490 (May 2005). https://doi.org/10.1007/11426639_28
36. Kortelainen, T., Kortelainen, J.: On Diamond Structures and Trojan Message Attacks. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part II. LNCS, vol. 8270, pp. 524–539 (Dec 2013). https://doi.org/10.1007/978-3-642-42045-0_27
37. Lamport, L.: Constructing digital signatures from a one way function (1979)
38. Lefevre, C.: Indifferentiability of the Sponge Construction with a Restricted Number of Message Blocks. IACR Trans. Symm. Cryptol. **2023**(1), 224–243 (2023). https://doi.org/10.46586/tosc.v2023.i1.224-243
39. Leighton, F.T., Micali, S.: Large Provably Fast and Secure Digital Signature Schemes Based on Secure Hash Functions. U.S. Patent 5,432,852 (Jul 11 1995)
40. Maurer, U.M., Renner, R., Holenstein, C.: Indifferentiability, Impossibility Results on Reductions, and Applications to the Random Oracle Methodology. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 21–39 (Feb 2004). https://doi.org/10.1007/978-3-540-24638-1_2
41. McGrew, D., Curcio, M., Fluhrer, S.: RFC 8554: Leighton-Micali Hash-Based Signatures (2019)
42. Merkle, R.C.: A Certified Digital Signature. In: Brassard, G. (ed.) CRYPTO'89. LNCS, vol. 435, pp. 218–238 (Aug 1990). https://doi.org/10.1007/0-387-34805-0_21
43. Merkle, R.C.: One Way Hash Functions and DES. In: Brassard, G. (ed.) CRYPTO'89. LNCS, vol. 435, pp. 428–446 (Aug 1990). https://doi.org/10.1007/0-387-34805-0_40
44. Merkle, R.C.: Secrecy, authentication, and public key systems. Stanford university (1979)
45. Moor Insights & Strategy: Securing Data in the Agentic AI Era. Tech. rep., Moor Insights & Strategy (Jun 2025), https://www.hpe.com/psnow/downloadDoc/

A%20Research%20Brief%20by%20Moor%20Insights%20-%20Securing%20data%20in%20the%20agentic%20AI%20era-a00150066enw.pdf, research brief hosted by Hewlett Packard Enterprise
46. Oracle: JDK Providers Documentation: The SUN Provider. Java Platform, Standard Edition 25 Documentation (2025), https://docs.oracle.com/en/java/javase/25/security/oracle-providers.html
47. Perlner, R.A., Kelsey, J., Cooper, D.A.: Breaking Category Five SPHINCS$^+$ with SHA-256. pp. 501–522 (2022). https://doi.org/10.1007/978-3-031-17234-2_23
48. Pub, F.: Secure Hash Standard (SHS). Fips pub **180**(4), 2012 (2012)
49. Ristenpart, T., Shacham, H., Shrimpton, T.: Careful with Composition: Limitations of the Indifferentiability Framework. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 487–506 (May 2011). https://doi.org/10.1007/978-3-642-20465-4_27
50. The OpenSSL Project: EVP_PKEY-LMS: Leighton–Micali Signature (LMS) Key Type and Algorithm Support. OpenSSL 3.6 Documentation (2025), https://docs.openssl.org/3.6/man7/EVP_PKEY-LMS/
51. Unruh, D.: Random Oracles and Auxiliary Input. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 205–223 (Aug 2007). https://doi.org/10.1007/978-3-540-74143-5_12
52. wolfSSL Inc.: wolfCrypt Implementations of LMS/HSS and XMSS/XMSSMT Signatures: Build Options and Benchmarks (Jun 2024), https://www.wolfssl.com/

# A Attack Against $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$- and $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$-based LMS

First, in Sect. A.1 we show that finding a fixed-prefix second preimage of a particular form for $\mathsf{h}^{\mathbf{E}}$ yields a forgery for $\mathrm{LMS}^{\mathsf{h}^{\mathbf{E}}}$. We then give two complementary ways to construct the required fixed-prefix second preimage: Sect. A.2 adapts a fixed-point-based attack of Gauravaram et al. [23,22]; Sect. A.3 adapts the diamond-based attack of [2,47] to the model $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and gives a direct-terminal variant for the chopped-output model $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$. Finally, Table 1 summarizes their fixed-resource success bounds.

Throughout this section, $\Gamma$ denotes the number of selected signed-message challenge calls and $B$ denotes the total number of padded blocks in those selected inputs. By contrast, $\widehat{\Gamma}$ and $\widehat{B}$ retain their meanings in the security theorems: they upper-bound all challenge calls of the five types and all honest Davies-Meyer calls, respectively. Thus $\Gamma \le \widehat{\Gamma}$ and $B \le \widehat{B}$. Moreover, we fix a positive integer $T$ and consider attacks in which each phase makes at most $T$ ideal-cipher queries, so that we can derive lower bounds on success probabilities for attacks with $\Theta(T)$ queries.

## A.1 A Fixed-Prefix Second Preimage Yields an LMS Forgery

Consider the multi-user unforgeability notion defined in Sect. 2.3. Suppose that, during the **Interaction** phase, $\mathscr{A}^{\mathsf{muSign}^{\mathbf{E}},\mathbf{E},\mathbf{E}^{-1}}$ makes $\Gamma$ signing queries $\Sigma^j \leftarrow \mathsf{muSign}^{\mathbf{E}}(\mathsf{i}_j, q^j, M^j)$, $j = 1, \ldots, \Gamma$, and receives

$$\Sigma^j = \left(q^j, (R^j, \sigma_0^{(j)}, \ldots, \sigma_{p-1}^{(j)}), p_0^{(j)}, \ldots, p_{h-1}^{(j)}\right).$$

Let $I^{\mathsf{i}_j}$ be the identifier of user $\mathsf{i}_j$ and define $\mathsf{prefix}_j := I^{\mathsf{i}_j} \| [q^j]_{32} \| [8181]_{16}$: this means $|\mathsf{prefix}_j| = \iota + 32 + 16 = \iota + 48$. Constructing an LMS forgery reduces to solving a fixed-prefix multi-target second-preimage problem for the $\Gamma$ challenge calls

$$D^j \leftarrow \mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_j \| R^j \| M^j), \qquad j = 1, \ldots, \Gamma.$$

Concretely, suppose that the adversary finds an index $j^* \in \{1, \ldots, \Gamma\}$ and values $R', M'$ satisfying $M' \neq M^{j^*}$, $R' \| M' \neq R^{j^*} \| M^{j^*}$, and

$$\mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_{j^*} \| R' \| M') = \mathsf{h}^{\mathbf{E}}(\mathsf{prefix}_{j^*} \| R^{j^*} \| M^{j^*}) \,. \tag{3}$$

It can then output

$$\left(\mathsf{i}_{j^*}, M', \Sigma' = \left(q^{j^*}, (R', \sigma_0^{(j^*)}, \ldots, \sigma_{p-1}^{(j^*)}), p_0^{(j^*)}, \ldots, p_{h-1}^{(j^*)}\right)\right).$$

This is a valid forgery for user $\mathsf{i}_{j^*}$. Indeed, Eq. (3) implies that the verifier obtains the same value $D^{j^*}$ in Step 1 of $\mathsf{LMSOTSVrfy}$. Hence the checksum, all Winternitz digits $V_0, \ldots, V_{p-1}$, and the reconstructed LM-OTS public key remain unchanged. The resulting LMS leaf is therefore unchanged as well. Reusing $\Sigma^{j^*}$'s authentication path consequently reconstructs the same Merkle root, so $\mathsf{LMSVrfy}(pk_{\mathsf{i}_{j^*}}, M', \Sigma') = 1$. The condition $M' \neq M^{j^*}$ is required because changing only the LM-OTS randomizer $R$ would not satisfy the winning condition in Fig. 4.

### A.2 Attack By Appending Fixed-Point Blocks

**Targets and Parameters.** Consider an IoT application whose message domain is defined solely by the following length restriction, enforced by both signing and verification: every allowed $M$ satisfies

$$\left|\mathsf{lpad}(I \| [q]_{32} \| [8181]_{16} \| R \| M)\right| \leq \ell_{\mathrm{app}} r$$

for some very small $\ell_{\mathrm{app}} \in \{2, ..., \ell_{\max}\}$ (e.g., $\ell_{\mathrm{app}} = 5$). In this regime, by appending fixed-point blocks to one of the challenge inputs, our attack below produces a forged padded input having at most $\ell_{\mathrm{app}}$ blocks. Choose $L$ as the largest positive integer satisfying

$$L \leq \ell_{\mathrm{app}} - 1 \qquad \text{and} \qquad Lr + (r - 72) < 2^{64}.$$

Thus the forged padded input constructed below has $L + 1 \leq \ell_{\mathrm{app}}$ blocks, and its unpadded length is representable by the 64-bit length field.

Let the attacker have signatures corresponding to the $\Gamma$ selected signed-message challenge inputs

$$\mathsf{lpad}(\mathsf{prefix}_j \| R^j \| M^j) = W_1^{(j)} \| \cdots \| W_{\ell_j}^{(j)}, \qquad 1 \leq j \leq \Gamma,$$

and let

$$S_0^{(j)} = IV, \quad S_i^{(j)} = \mathsf{DM}^{\mathbf{E}}(W_i^{(j)}, S_{i-1}^{(j)}).$$

The prefixes $\mathsf{prefix}_j$ and the padded lengths $\ell_j$ may all be different. Let

$$B := \sum_{j=1}^{\Gamma} \ell_j$$

be the total number of blocks in the padded challenge inputs used by this attack. Recomputing all their final chaining states costs at most $B$ adversarial ideal-cipher queries. The analysis below uses a selected subcollection satisfying $B \le T$; if the available signed inputs contain more blocks, the attacker first chooses such a subcollection, and all subsequent occurrences of $\Gamma$, $B$, and $\gamma$ refer to that subcollection.

For every $D \in \{0,1\}^{r-72}$, define the $r$-bit block

$$W_D := D \| 1 \| 0^7 \| [Lr + (r-72)]_{64}. \tag{4}$$

The block is exactly the suffix padding appended to an unpadded string of length $Lr + (r-72)$ bits. Moreover, there are $2^{r-72}$ choices for $D$, which is $2^{512-72} = 2^{440}$ for the concrete SHA-256 parameter $r = 512$.

**Attack Process.** The attack proceeds as follows.

1. Sample uniformly and without replacement $T$ values $D \in \{0,1\}^{r-72}$, where $T < 2^{r-72}$, and make inverse (adversarial) ideal-cipher queries $X_D \leftarrow \mathbf{E}^{-1}_{W_D}(0^s)$.
   Let $\mathsf{BadKey}$ be the event one of the aforementioned adversarial ideal-cipher queries $\mathbf{E}^{-1}_{W_D}(0^s)$ is not *key-fresh*, i.e., before that inverse query is made, there has been an ideal-cipher query under the key $W_D$. Before this search the attacker makes no adversarial ideal-cipher queries except target recomputation; by this and by the sampling-without-replacement strategy, those queries could only replay keys among the at most $\widehat{B}$ keys in LMS internal ideal-cipher queries. Condition on the pre-search transcript $\mathsf{Tr}_{\mathrm{pre}}$, and let $h_{\mathrm{pre}} \le \widehat{B}$ be the number of its distinct keys belonging to the structured $2^{r-72}$-element family in Eq. (4), it then holds

$$\Pr\left[\mathsf{BadKey} \mid \mathsf{Tr}_{\mathrm{pre}}\right] = 1 - \frac{\binom{2^{r-72}-h_{\mathrm{pre}}}{T}}{\binom{2^{r-72}}{T}} \le \frac{T h_{\mathrm{pre}}}{2^{r-72}-T+1} \le \frac{T\widehat{B}}{2^{r-72}-T+1}\,. \tag{5}$$

   The equality is for every fixed pre-search transcript. Since the last quantity is deterministic, averaging gives the same upper bound on $\Pr[\mathsf{BadKey}]$.
2. If $X_D = S^{(j^*)}_{\ell_{j^*}}$ for some $j^*$ with $\ell_{j^*} \le L$, then $\mathbf{E}_{W_D}(S^{(j^*)}_{\ell_{j^*}}) = 0^s$, so the corresponding Davies-Meyer call is a self-loop, i.e., $\mathsf{DM}^{\mathbf{E}}(W_D, S^{(j^*)}_{\ell_{j^*}}) = S^{(j^*)}_{\ell_{j^*}}$. Abort if no such pair $(D, j^*)$ exists.
3. Insert $L - \ell_{j^*}$ (which can be zero) copies of the fixed-point block between the old padded string and $D$, and regard the resulting string as a new unpadded input:

$$U' := W_1^{(j^*)} \| \cdots \| W^{(j^*)}_{\ell_{j^*}} \| \underbrace{W_D \| \cdots \| W_D}_{L-\ell_{j^*} \text{ copies}} \| D\,.$$

   Because the old padded string begins with $\mathsf{prefix}_{j^*} \| R^{j^*} \| M^{j^*}$, $U'$ has a unique parsing $U' = \mathsf{prefix}_{j^*} \| R^{j^*} \| M'$ with $M' \neq M^{j^*}$. Output $M'$ together with the previously obtained LMS signature.

   By Eq. (4), the new padded encoding is

$$\mathsf{lpad}(U') = W_1^{(j^*)} \| \cdots \| W^{(j^*)}_{\ell_{j^*}} \| \underbrace{W_D \| \cdots \| W_D}_{L-\ell_{j^*}+1 \text{ copies}}\,.$$

Indeed, the length of $U'$ is

$$\ell_{j^*} r + (L - \ell_{j^*}) r + (r-72) = Lr + (r-72)\,,$$

hence its final padding block is exactly $W_D$. The resulting padded input has $L+1 \le \ell_{\mathrm{app}}$ blocks and is thus valid. The first $\ell_{j^*}$ calls reach $S^{(j^*)}_{\ell_{j^*}}$, and every one of the remaining $L+1-\ell_{j^*}$ calls leaves that state unchanged. Hence

$$\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*} \| R^{j^*} \| M') = \mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*} \| R^{j^*} \| M^{j^*}).$$

The original signature consequently verifies for the different message $M'$. Equality holds for the complete chaining state, so the same attack also applies to $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$.

**Resources and Success Probability.** No additional ideal-cipher query is needed for the repeated copies of $W_D$. Hence target recomputation uses at most $B$ queries and the fixed-point search uses at most $T$ queries. Under $B \le T$, the entire attack makes at most

$$B + T \le 2T = O(T)$$

adversarial ideal-cipher queries.

Conditioned on $\neg\mathsf{BadKey}$, the blocks $W_D$ are distinct fresh ideal-cipher keys, and the resulting values $X_D$ are therefore independent and uniform in $\{0,1\}^s$. Put

$$\gamma := \big|\{S^{(j)}_{\ell_j} : 1 \le j \le \Gamma,\ \ell_j \le L\}\big|.$$

Namely, $\gamma$ is the number of *distinct* terminal states of challenge calls whose padded inputs are short enough (to admit the extension). The attack then satisfies

$$\begin{aligned}
\Pr[\mathsf{Forge}_{\mathrm{ext}}] &\ge 1 - \Big(1 - \frac{\gamma}{2^s}\Big)^T - \Pr\big[\mathsf{BadKey}\big] \\
&\ge 1 - \exp\left\{-\frac{T\gamma}{2^s}\right\} - \Pr\big[\mathsf{BadKey}\big] \\
&\ge \frac{1}{2}\min\left\{1, \frac{T\gamma}{2^s}\right\} - \frac{T\widehat{B}}{2^{r-72} - T + 1}\,. \qquad (6)
\end{aligned}$$

As long as $\widehat{B} = o\big(\gamma 2^{r-72-s}\big)$ and $T\gamma \le \widehat{B}T < 2^s$, the success probability is $\Omega(T\gamma/2^s)$.

When every selected challenge input has a constant number of padded blocks and $\gamma = \Theta(\Gamma)$, as expected for random known-message samples whose lengths satisfy the stated application profile and for suitable chosen-message samples, we have $B = \Theta(\Gamma)$. Provided $B \le T$ and $\Pr\big[\mathsf{BadKey}\big]$ in Eq. (6) is negligible, the lower bound is then $\Omega(TB/2^s)$.

### A.3 Attacks Using the Diamond Structure

**Targets, Parameters, and Query Budgets.** Following Sect. A.2, select $\Gamma$ signed-message challenge inputs

$$\mathsf{lpad}(\mathsf{prefix}_j \| R^j \| M^j) = W_1^{(j)} \| \cdots \| W_{\ell_j}^{(j)}, \qquad 1 \le j \le \Gamma,$$

and write their chaining values as

$$S_0^{(j)} = IV, \qquad S_i^{(j)} = \mathsf{DM}^{\mathbf{E}}(W_i^{(j)}, S_{i-1}^{(j)}).$$

The prefixes $\mathsf{prefix}_j$ and the padded lengths $\ell_j$ may all be different. Collisions among $\mathsf{prefix}_1, \ldots, \mathsf{prefix}_\Gamma$ would increase attack efficiency. However, for simplicity, this section does not credit such collisions and gives a worst-case setup analysis. Also let

$$B := \sum_{j=1}^{\Gamma} \ell_j, \qquad \ell_{\min} := \min_j \ell_j.$$

Choose the target set so that $B \le T$. If more targets are available, the attacker selects a subcollection satisfying this condition; all quantities below refer to that subcollection. For simplicity, take $\Gamma = 2^d$ and put $k := \lfloor \log_2 \ell_{\min} \rfloor$. We assume that the lengths differ by at most a fixed constant factor and that

$$\ell_{\min} \ge 2(d + k + 3).$$

These length assumptions are retained for both variants analyzed here.

*Candidate blocks and sampling.* For every selected target $j$, recall from Sect. A.1 that $|\mathsf{prefix}_j| = \iota + 48$, and write

$$W_1^{(j)} = \mathsf{prefix}_j \| R^j \| \eta_j, \qquad \eta_j \in \{0,1\}^{r-\iota-48-n}.$$

The candidate set for the attack's first block on branch $j$ is

$$\mathcal{W}_j^{\mathsf{pre}} := \big\{ \mathsf{prefix}_j \| R' \| \eta' : R' \in \{0,1\}^n,\ \eta' \in \{0,1\}^{r-\iota-48-n} \setminus \{\eta_j\} \big\}. \tag{7}$$

Its size, the same for every $j$, is

$$F_{\mathrm{pre}} := |\mathcal{W}_j^{\mathsf{pre}}| = 2^n \big(2^{r-\iota-48-n} - 1\big) = 2^{r-\iota-48} - 2^n.$$

By the block-layout assumption in Sect. 2.2, each hash call in $\mathsf{LMSOTSGen}$ Step (2) uses one call to $\mathbf{E}$. Thus $\iota + n + 121 \le r$, so $r - \iota - 48 - n \ge 73$ and $F_{\mathrm{pre}} > 0$.

Each new block is sampled uniformly from the candidate set specified for its phase, excluding all blocks previously sampled by the attack. Rejection sampling implements this exclusion. If no candidate remains, the attack returns failure. The common budget and sampling conditions are

$$B \le T, \qquad \Gamma \le T, \qquad \Gamma \le F_{\mathrm{pre}}, \qquad 4T < 2^r. \tag{8}$$

They prevent exhaustion of the prefix candidate sets and of the full $r$-bit candidate set during the bounded full-state attack. The direct-terminal variant has an additional candidate-set condition, stated with that variant below.

A newly sampled block $W$ may still equal a key used in a prior honest or adversarial ideal-cipher query. Since $W$ is the key in $\mathbf{E}(W, S)$, this is key reuse even if the state input $S$ differs. The attack does not filter candidates using the experiment's query history. In the analysis, such reuse is charged to $\mathsf{BadKey}_{\mathrm{dia}}$ or $\mathsf{BadKey}_{\mathrm{chop}}$. These events concern a sampled block's first use in the construction; replaying a selected block when evaluating the final message is not counted.

**Full-State Bridge Attack with Fixed Budgets.** The attacker first recomputes the selected target paths and stores their chaining values, using at most $B$ ideal-cipher queries. It then runs the following three phases (as overviewed in Sect. 3.2), each with its own cap of $T$ queries. A phase stops as soon as its required output is obtained. If another query would exceed that phase's cap before completion, the entire attack returns failure.

*Phase 1: Constructing a diamond.* For $j = 1, \ldots, \Gamma$, sample $W_j^{\mathsf{pre}} \in \mathcal{W}_j^{\mathsf{pre}}$ and compute $S_j^{\mathsf{pre}} = \mathsf{DM}^{\mathbf{E}}(W_j^{\mathsf{pre}}, IV)$. We next use the remaining $T-\Gamma$ queries to construct a balanced chosen-start diamond [34,36] in which every $S_j^{\mathsf{pre}}$ has a $d$-block path $\big(S_j^{\mathsf{pre}} \xrightarrow{msg_j} S_{\mathrm{rt}}\big)$ to one common state $S_{\mathrm{rt}}$.

For completeness, we provide a detailed description of the process. The construction proceeds level by level. At level $t \in \{0, \ldots, d-1\}$, write the $q$ live nodes as

$$(\mathcal{L}_1, S_1^{\mathsf{dia}}), \ldots, (\mathcal{L}_q, S_q^{\mathsf{dia}}),$$

where $\mathcal{L}_i$ is the set of original target indices whose paths have been merged into the $i$-th live node, with current state $S_i^{\mathsf{dia}}$; and every $S_j^{\mathsf{pre}}$ with $j \in \mathcal{L}_i$ has a path of the same length to $S_i^{\mathsf{dia}}$. Initially $q = \Gamma$, $\mathcal{L}_i = \{i\}$, and $S_i^{\mathsf{dia}} = S_i^{\mathsf{pre}}$.

(i) While the total number of ideal-cipher queries made in Step 1 remains below $T$, select the live labels in balanced order. For label $i$ and its local trial number $a$, sample $W_{t,i,a}^{\mathsf{dia}} \in \{0,1\}^r$ under the sampling rule above, make one ideal-cipher query, and store

$$(S^{\mathsf{cand}}, t, i, a, W_{t,i,a}^{\mathsf{dia}}), \qquad S^{\mathsf{cand}} := \mathsf{DM}^{\mathbf{E}}(W_{t,i,a}^{\mathsf{dia}}, S_i^{\mathsf{dia}}).$$

(ii) Whenever two stored records with different labels $i, j$ have the same endpoint, insert an edge $\{i, j\}$ in the collision graph, labelled with the two sampled blocks and their common endpoint $S_{i,j}^{\mathsf{dia},+}$. Thus

$$\mathsf{DM}^{\mathbf{E}}(W_{t,i,a_i}^{\mathsf{dia}}, S_i^{\mathsf{dia}}) = \mathsf{DM}^{\mathbf{E}}(W_{t,j,a_j}^{\mathsf{dia}}, S_j^{\mathsf{dia}}) = S_{i,j}^{\mathsf{dia},+}.$$

(iii) Once the graph contains a matching covering all live labels, replace each matched pair by $(\mathcal{L}_i \cup \mathcal{L}_j, S_{i,j}^{\mathsf{dia},+})$. Append the corresponding edge block to each path below each child. Every path grows by one block and the number of live nodes becomes $q/2$. If the cap is reached without such a matching, return failure.

If all $d$ levels are completed, their last node defines $S_{\mathrm{rt}}$, and the stored edges give a $d$-block string $msg_j$ for every target:

$$\big(S_j^{\mathsf{pre}} \xrightarrow{msg_j} S_{\mathrm{rt}}\big), \qquad j = 1, \ldots, \Gamma.$$

For $\Gamma = 1$, set $S_{\mathrm{rt}} := S_1^{\mathsf{pre}}$ and take $msg_1$ to be empty. Phase 1 then uses exactly one query, for the initial prefix block, and no merging queries.

*Phase 2: Constructing an expandable-message.* Starting from $S_0^{\mathsf{exp}} := S_{\mathrm{rt}}$, construct $k$ "collision gadgets": each of which consists of a short and a long message branch that lead from the same starting state to the same ending state.

For $t = 1, \ldots, k$, put $d_t := 2^{k-t}$. If the total number of ideal-cipher queries made in this phase has exceeded $T - d_t$, return failure. Otherwise, sample $d_t$ blocks from $\{0,1\}^r$ to form a $d_t r$-bit message-fragment $D_t$. Starting from $S_{t-1}^{\mathsf{exp}}$, apply $\mathsf{DM}^{\mathbf{E}}$ iteratively to these blocks, using one ideal-cipher query per block, to obtain the path $\big(S_{t-1}^{\mathsf{exp}} \xrightarrow{D_t} \widetilde{S}_t^{\mathsf{exp}}\big)$.

Next, seek a collision $\mathsf{DM}^{\mathbf{E}}(A_t, S_{t-1}^{\mathsf{exp}}) = \mathsf{DM}^{\mathbf{E}}(C_t, \widetilde{S}_t^{\mathsf{exp}})$. For this, initialize two empty lists and perform the following search:

(i) Sample a new candidate block $A$ according to the sampling rule above, compute $Z_A \leftarrow \mathsf{DM}^{\mathbf{E}}(A, S_{t-1}^{\mathsf{exp}})$, and store $(A, Z_A)$ in the first list.

(ii) Sample a new candidate block $C$ according to the same sampling rule, compute $Z_C \leftarrow \mathsf{DM}^{\mathbf{E}}(C, \widetilde{S}_t^{\mathsf{exp}})$, and store $(C, Z_C)$ in the second list.

(iii) Repeat the two types of sampling and querying actions alternately. After each query, check whether its endpoint already appears in the other list. As soon as pairs $(A, Z_A)$ and $(C, Z_C)$ with $Z_A = Z_C$ are found, set $A_t := A$ and $C_t := C$ and stop the search. If this phase's total query budget $T$ is exhausted without such a match, return failure.

The one-block branch $A_t$ and the $(d_t+1)$-block branch $D_t\|C_t$ then reach the same state. If all $k$ pairs are completed, choosing one branch from each pair gives, for every $b \in \{k, \ldots, k+2^k-1\}$, a $b$-block string $E^{(b)}$ satisfying

$$\big(S_{\mathrm{rt}} \xrightarrow{E^{(b)}} S_k^{\mathsf{exp}}\big).$$

*Phase 3: Seeking a full-state bridge.* For every target $j$, define

$$\mathcal{I}_j := \big\{i : d+k+2 \le i \le \min\{d+k+2^k+1, \ell_j - 2\}\big\}.$$

These are the positions on target path $j$ at which the bridge can be attached. For $i \in \mathcal{I}_j$, the required expandable-message length $i-d-2$ lies in $\{k, \ldots, k+2^k-1\}$. The upper bound $i \le \ell_j - 2$ leaves the final two target blocks, which may contain the SHA length encoding, untouched. The length assumptions give

$$\ell_{\min}/2 < 2^k \le \ell_{\min}, \qquad \ell_j - d - k - 3 \ge \ell_{\min}/2,$$

and hence, with constants uniform over the selected targets,

$$|\mathcal{I}_j| = \min\{2^k, \ell_j - d - k - 3\} = \Theta(\ell_{\min}) = \Theta(\ell_j).$$

For $r = 1, \ldots, T$, sample a bridge block $W_r^{\mathsf{bridge}} \in \{0,1\}^r$ under the common sampling rule and make one query from $S_k^{\mathsf{exp}}$. Stop at the first trial $r^*$ for which some target $j^*$ and $i^* \in \mathcal{I}_{j^*}$ satisfy

$$\mathsf{DM}^{\mathbf{E}}(W_{r^*}^{\mathsf{bridge}}, S_k^{\mathsf{exp}}) = S_{i^*}^{(j^*)}.$$

If no such hit occurs in $T$ trials, return failure. Otherwise, select the stored prefix branch and the expandable-message length and output

$$\overline{U}' := W_{j^*}^{\mathsf{pre}} \| msg_{j^*} \| E^{(i^*-d-2)} \| W_{r^*}^{\mathsf{bridge}} \| W_{i^*+1}^{(j^*)} \| \cdots \| W_{\ell_{j^*}}^{(j^*)}.$$

Its block count is

$$1 + d + (i^* - d - 2) + 1 + (\ell_{j^*} - i^*) = \ell_{j^*}.$$

Thus the copied length encoding remains correct. The first block begins with $\mathsf{prefix}_{j^*}$, the diamond and expandable message reach $S_k^{\mathsf{exp}}$, the bridge reaches $S_{i^*}^{(j^*)}$, and the copied suffix reaches the original final state. Parsing $\mathsf{prefix}_{j^*}\|R'\|M' \leftarrow \mathsf{lpad}^{-1}(\overline{U}')$ gives

$$\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*}\|R'\|M') = \mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*}\|R^{j^*}\|M^{j^*}).$$

Since $W_{j^*}^{\mathsf{pre}}$ is sampled from $\mathcal{W}_{j^*}^{\mathsf{pre}}$ defined in Eq. (7), its message portion $\eta'$ differs from $\eta_{j^*}$. Hence $M' \neq M^{j^*}$ deterministically, and this yields an LMS forgery by

Sect. A.1. It can be seen assembling the forgery requires no additional ideal-cipher query. Every execution, successful or not, uses at most $B + 3T \le 4T$ queries.

For $n < s$, it is easy to see

$$\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*} \| R' \| M') = \mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_{j^*} \| R^{j^*} \| M^{j^*})$$

holds as well, so this "full-state bridge" is one of two available diamond attacks in the chopped-output case.

**Success Probability of the Budgeted Full-State Attack.** Fix the selected challenge inputs and the challenge paths obtained by recomputing them. We analyze the success probability of the subsequent attack for these fixed challenges. Let

$$N_{\mathrm{hit}} := \left| \left\{ S_i^{(j)} : 1 \le j \le \Gamma,\ i \in \mathcal{I}_j \right\} \right|$$

be the number of distinct target states *at* the possible bridge positions. The total number of these positions, before identifying equal states, is

$$\widehat{B}_{\mathrm{el}} := \sum_{j=1}^{\Gamma} \left| \mathcal{I}_j \right| = \sum_{j=1}^{\Gamma} \max\left\{0, \min\{2^k, \ell_j - d - k - 3\}\right\}.$$

Thus $N_{\mathrm{hit}} \le \widehat{B}_{\mathrm{el}} \le B \le \widehat{B}$. Since $|\mathcal{I}_j| = \Theta(\ell_j)$ for every selected target, summing over $j$ gives $\widehat{B}_{\mathrm{el}} = \Theta(B)$.

*Probability of sampling non-fresh keys.* Define $\mathsf{BadKey}_{\mathrm{dia}}$ to occur if a block sampled in Phases 1–3, on its first use in these phases, is a key under which an honest or adversarial ideal-cipher query has already been made. The attack's own sampled blocks are pairwise distinct, so this event records a collision with the pre-existing query history. The attack makes no other ideal-cipher queries before these phases except to recompute the selected honest target paths. Hence that history uses at most $\widehat{B}$ different keys. Replays of selected blocks in the final message are not included in the event.

Phase 1 samples $\Gamma$ prefix blocks from sets of size $F_{\mathrm{pre}}$. Besides, the three phases together sample at most $3T$ other blocks from the full space $\{0,1\}^r$. Sequential uniform sampling and a union bound give

$$\Pr\left[\mathsf{BadKey}_{\mathrm{dia}}\right] \le \min\left\{1, \frac{\Gamma \widehat{B}}{F_{\mathrm{pre}} - \Gamma + 1} + \frac{3T\widehat{B}}{2^r - 4T + 1}\right\}. \tag{9}$$

The second denominator also allows for the at most $\Gamma \le T$ prefix blocks. The conditions in Eq. (8) make both denominators positive. For SHA-256 parameters, this bound is

$$(1 + o(1))\left(\frac{\Gamma \widehat{B}}{2^{336}} + \frac{3T\widehat{B}}{2^{512}}\right).$$

*Success probabilities of Phases 1 and 2.* To lower-bound the probability that Phases 1 and 2 each finish within $T$ ideal-cipher queries, we introduce an auxiliary experiment. This experiment follows the same order of evaluations as Phases 1 and 2, without imposing the query limits. Whenever a new Davies-Meyer evaluation would be performed, the auxiliary experiment generates an independent uniform value in $\{0,1\}^s$ as its endpoint and counts this as one query, without sampling a message block. It keeps the same collision tests and conditions for completing each phase.

Let $Q_{\mathrm{dia}}$ and $Q_{\mathrm{exp}}$ be the random query counts until phase 1 and 2 succeed in this auxiliary experiment. At a diamond level with $q$ live labels, the batched birthday-and-matching analysis [6, Sects. 3–4] gives an expected query count of $\widetilde{O}(q+2^{s/2}\sqrt{q})$, where $\widetilde{O}$ hides logarithmic factors. Including the initial prefix queries and summing over the levels gives

$$\mathbb{E}\big[Q_{\mathrm{dia}}\big] \le \Gamma + \sum_{t=0}^{d-1} \widetilde{O}\left(\frac{\Gamma}{2^t} + 2^{s/2}\sqrt{\frac{\Gamma}{2^t}}\right) = \widetilde{O}\Big(\Gamma + 2^{s/2}\sqrt{\Gamma}\Big).$$

For $\Gamma = 1$, $Q_{\mathrm{dia}} = 1$ exactly. In the usual range $\Gamma \le 2^s$, the displayed upper bound simplifies to $\widetilde{O}(2^{s/2}\sqrt{\Gamma})$.

For the expandable message, processing $D_1, \ldots, D_k$ uses $\sum_{t=1}^{k} d_t = 2^k - 1$ queries. The $k$ two-list collision searches have total expected cost $\Theta(k2^{s/2})$, following the expandable-message construction of Kelsey and Schneier [35, Sect. 3.2]. Thus

$$\mathbb{E}\big[Q_{\mathrm{exp}}\big] = \Theta(k2^{s/2} + 2^k).$$

Markov's inequality and a union bound yield

$$\Pr[Q_{\mathrm{dia}} > T \text{ or } Q_{\mathrm{exp}} > T] \le \frac{\mathbb{E}\big[Q_{\mathrm{dia}}\big] + \mathbb{E}\big[Q_{\mathrm{exp}}\big]}{T}.$$

Consequently, both "$T$-capped" attack phases succeed in the auxiliary experiment with probability at least

$$p_{\mathrm{set}}(T) := \max\left\{0, 1 - \frac{\mathbb{E}\big[Q_{\mathrm{dia}}\big] + \mathbb{E}\big[Q_{\mathrm{exp}}\big]}{T}\right\}. \tag{10}$$

*Bridge trials and the success bound.* In the auxiliary experiment, conditional on any successful construction transcript, the $T$ potential bridge trials have independent uniform endpoints. Stopping at the first hit gives the same success event as examining all $T$ trials. Its probability is exactly

$$1 - \left(1 - \frac{N_{\mathrm{hit}}}{2^s}\right)^T.$$

Multiplying the phase 1 and 2 success lower bound $p_{\mathrm{set}}(T)$ yields the total success lower bound.

To transfer the result to the real "$T$-capped" attack experiment, we couple the real and auxiliary executions by using the same endpoint whenever the real construction first queries a previously unqueried key. This endpoint is uniform because the corresponding ideal-cipher permutation has not yet been queried, and because the relevant candidate sets cannot be exhausted within the $T$ queries (only $\Gamma$ prefix blocks

and fewer than $4T$ blocks in total are requested). The two executions then agree until $\mathsf{BadKey}_{\mathrm{dia}}$ occurs. Therefore, in the real "$T$-capped" attack experiment, the attack success probability is at least

$$\begin{aligned}\Pr[\mathsf{Forge}_{\mathrm{dia}}] &\geq p_{\mathrm{set}}(T)\left[1-\left(1-\frac{N_{\mathrm{hit}}}{2^s}\right)^T\right]-\Pr[\mathsf{BadKey}_{\mathrm{dia}}]\\ &\geq p_{\mathrm{set}}(T)\left(1-\exp\left\{-\frac{TN_{\mathrm{hit}}}{2^s}\right\}\right)-\Pr[\mathsf{BadKey}_{\mathrm{dia}}]. \qquad (11)\end{aligned}$$

If a constant fraction of the states at the possible bridge positions have distinct values, then there is an absolute constant $c_{\mathrm{el}}>0$ such that $N_{\mathrm{hit}}\geq c_{\mathrm{el}}B$, which means

$$\Pr[\mathsf{Forge}_{\mathrm{dia}}] \geq p_{\mathrm{set}}(T)\left(1-\exp\left\{-\frac{c_{\mathrm{el}}TB}{2^s}\right\}\right)-\Pr[\mathsf{BadKey}_{\mathrm{dia}}]. \qquad (12)$$

For $c_{\mathrm{el}}TB\leq 2^s$, the conditional success probability of Phase 3 in the auxiliary experiment, given successful completion of Phases 1 and 2, is $\Omega(TB/2^s)$. Subject to the sampling conditions above, a positive constant success lower bound follows when

$$T\geq 2\big(\mathbb{E}\big[Q_{\mathrm{dia}}\big]+\mathbb{E}\big[Q_{\mathrm{exp}}\big]\big),\qquad TB=\Omega(2^s),\qquad B\leq T,$$

and $\Pr[\mathsf{BadKey}_{\mathrm{dia}}]$ is at most half the positive term in Eq. (12).

The bound has a simpler form. Concretely, let $\overline{Q}_{\mathrm{set}} := \mathbb{E}\big[Q_{\mathrm{dia}}\big]+\mathbb{E}\big[Q_{\mathrm{exp}}\big] = \widetilde{O}\Big(\Gamma+2^{s/2}\sqrt{\Gamma}+(\log_2\ell_{\min})2^{s/2}+\ell_{\min}\Big)$. Since we assumed $N_{\mathrm{hit}}\geq c_{\mathrm{el}}B$ and since $B=\Theta(\Gamma\ell_{\min})$, the bound can be written as

$$\Pr[\mathsf{Forge}_{\mathrm{dia}}]\geq \max\left\{0,1-\frac{\overline{Q}_{\mathrm{set}}}{T}\right\}\left(1-\exp\left\{-\Omega\left(\frac{T\Gamma\ell_{\min}}{2^s}\right)\right\}\right)-\Pr[\mathsf{BadKey}_{\mathrm{dia}}].$$

In particular, if $B\leq T$, $T\geq 2(\mathbb{E}\big[Q_{\mathrm{dia}}\big]+\mathbb{E}\big[Q_{\mathrm{exp}}\big])$, and $N_{\mathrm{hit}}\geq c_{\mathrm{el}}B$, then the same attack, with its deterministic cap of $4T$, satisfies for an absolute constant $c_{\mathrm{dia}}>0$,

$$\Pr[\mathsf{Forge}_{\mathrm{dia}}]\geq c_{\mathrm{dia}}\min\left\{1,\frac{TB}{2^s}\right\}-\Pr[\mathsf{BadKey}_{\mathrm{dia}}].$$

When $B=\Theta(\widehat{B})$, this matches the corresponding theorem term in regimes where the key-reuse correction is sufficiently small compared with the positive term. Otherwise, when $B\ll\widehat{B}$, there would be a gap since the attack does not "fully" utilize all the $\widehat{B}$ blocks in the challenge hash-calls (we refer to Sect. B for elaborations on these calls).

**Chopped Outputs with Fixed Budgets.** For $n<s$, the full-state bridge attack already described is valid with the same query bound of $4T$, since equality of the full final states implies equality of their leftmost $n$ bits. Alternatively, we can also use the following direct-terminal attack, as overviewed in Sect. 3.2.

Concretely, after recomputing the challenge calls, run only the aforementioned phase 1 *Constructing a diamond* with at most $T$ queries, then run a new phase 2 *Seeking terminal-hit phase* that consists of at most $T$ terminal trials. There is no expandable-message or internal-bridge phase.

*Phase 1: Constructing a diamond.* This phase is the same as phase 1 of the full-state case. If Phase 1 fails to be completed within $T$ queries, return failure immediately. Otherwise, once completed, it supplies branches $W_j^{\mathsf{pre}} \| msg_j$ that all reach $S_{\mathrm{rt}}$ after $1+d$ blocks. In addition to Eq. (8), require

$$d + 2 \le \ell_{\max}, \qquad (1+d)r + (r-72) < 2^{64}, \qquad 2T < 2^{r-72} + 1. \tag{13}$$

The first two conditions ensure a valid input length and a representable 64-bit length field. The last ensures enough terminal candidates even after excluding blocks sampled during Phase 1. For $D \in \{0,1\}^{r-72}$, put

$$W_D^{\mathrm{term}} := D \| 1 \| 0^7 \| \big[(1+d)r + (r-72)\big]_{64}.$$

Thus $W_j^{\mathsf{pre}} \| msg_j \| W_D^{\mathrm{term}}$ is a valid padded input beginning with $\mathsf{prefix}_j$, whose unpadded bit length is $(1+d)r + (r-72)$.

*Phase 2: Seeking terminal-hit phase.* Sample $T$ block $D$ uniformly from the remaining candidates whose blocks $W_D^{\mathrm{term}}$ have not been sampled by the attack before, and make the single query

$$\mathbf{E}_{W_D^{\mathrm{term}}}(S_{\mathrm{rt}}), \qquad S_D^{\mathrm{term}} := \mathsf{DM}^{\mathbf{E}}(W_D^{\mathrm{term}}, S_{\mathrm{rt}}).$$

No candidate is filtered using the experimental query history. Stop at the first trial for which some target $j^*$ satisfies

$$\mathsf{left}_n(S_D^{\mathrm{term}}) = \mathsf{left}_n(S_{\ell_{j^*}}^{(j^*)}) = D^{j^*}.$$

If all $T$ trials miss, return failure. On a hit, output

$$\overline{U}'' := W_{j^*}^{\mathsf{pre}} \| msg_{j^*} \| W_D^{\mathrm{term}}, \qquad \mathsf{prefix}_{j^*} \| R' \| M' := \mathsf{Ipad}^{-1}(\overline{U}'').$$

The digest matches the selected target, while the changed message-bit window in $W_{j^*}^{\mathsf{pre}}$ ensures $M' \neq M^{j^*}$. This is a fixed-prefix second preimage and hence yields a forgery. Each terminal trial costs one query regardless of the target lengths. The attack therefore has the deterministic query bound $B + 2T \le 3T$.

*Success probability.* Let

$$N_{\mathrm{out}} := \Big|\Big\{\mathsf{left}_n(S_{\ell_j}^{(j)}) : 1 \le j \le \Gamma\Big\}\Big| = \Big|\Big\{D^j : 1 \le j \le \Gamma\Big\}\Big|$$

be the number of distinct target digests. Define $\mathsf{BadKey}_{\mathrm{chop}}$ in the same way as $\mathsf{BadKey}_{\mathrm{dia}}$, now for the prefix, diamond, and terminal queries of this variant. There are $\Gamma$ prefix blocks, at most $T$ other diamond blocks, and at most $T$ terminal blocks from a set of size $2^{r-72}$. Sequential sampling gives

$$\Pr\big[\mathsf{BadKey}_{\mathrm{chop}}\big] \le \min\left\{1, \frac{\Gamma\widehat{B}}{F_{\mathrm{pre}} - \Gamma + 1} + \frac{T\widehat{B}}{2^r - 2T + 1} + \frac{T\widehat{B}}{2^{r-72} - 2T + 1}\right\}, \tag{14}$$

All denominators are positive under the stated conditions. In particular, after at most $T$ Phase 1 blocks have been excluded, the last condition in Eq. (13) leaves enough candidates for all $T$ terminal trials.

Let $Q_{\mathrm{dia}}$ be the number of ideal-cipher queries made in phase 1. Following the above analysis of $p_{\mathrm{set}}(T)$ in Eq. (10), phase 1 succeeds with probability at least

$$p_{\mathrm{dia}}(T) := \max\left\{0, 1 - \frac{\mathbb{E}\big[Q_{\mathrm{dia}}\big]}{T}\right\}.$$

We also consider an auxiliary experiment in which the $T$ potential terminal outputs are independent uniform $n$-bit values. Each output is shared by all prefix branches, so a trial succeeds if it hits any of the $N_{\mathrm{out}}$ distinct target digests. Coupling to the real bounded execution and subtracting the reuse correction gives

$$\Pr\big[\mathsf{Forge}_{\mathrm{chop}}\big] \geq p_{\mathrm{dia}}(T)\left[1 - \left(1 - \frac{N_{\mathrm{out}}}{2^n}\right)^T\right] - \Pr\big[\mathsf{BadKey}_{\mathrm{chop}}\big]. \tag{15}$$

The executions agree until $\mathsf{BadKey}_{\mathrm{chop}}$ first occurs. If the selected digests are distinct, $N_{\mathrm{out}} = \Gamma$ and the terminal-search factor is at least

$$1 - \exp\left\{-\frac{\Gamma T}{2^n}\right\}.$$

Subject to the sampling and input-domain conditions above, constant success follows when

$$T = \widetilde{O}(\Gamma + 2^{s/2}\sqrt{\Gamma}), \qquad \Gamma T = \Omega(2^n), \qquad B \leq T,$$

and $\Pr\big[\mathsf{BadKey}_{\mathrm{chop}}\big]$ is at most half the positive constant furnished by the two factors in Eq. (15).

*Choosing an attack strategy.* When $n < s$, the attacker may choose the strategy with a higher successes probability, given its resources. Only the chosen variant is run; either choice fits within the common total cap of $4T$. Hence, when both variants' conditions hold,

$$\begin{aligned}\Pr[\mathsf{Forge}_{\mathrm{chop,dia}}] \geq \max\Bigg\{ & p_{\mathrm{set}}(T)\left[1 - \left(1 - \frac{N_{\mathrm{hit}}}{2^s}\right)^T\right] - \Pr[\mathsf{BadKey}_{\mathrm{dia}}], \\ & p_{\mathrm{dia}}(T)\left[1 - \left(1 - \frac{N_{\mathrm{out}}}{2^n}\right)^T\right] - \Pr\big[\mathsf{BadKey}_{\mathrm{chop}}\big]\Bigg\}. \end{aligned} \tag{16}$$

Suppose $B \leq T$, $T \geq 2(\mathbb{E}\big[Q_{\mathrm{dia}}\big] + \mathbb{E}\big[Q_{\mathrm{exp}}\big])$, $N_{\mathrm{hit}} = \Theta(B)$, and $N_{\mathrm{out}} = \Theta(\Gamma)$, with each key-reuse correction smaller than a fixed fraction of its corresponding positive term. Then Eq. (16) gives, up to constant and logarithmic factors,

$$\Pr[\mathsf{Forge}_{\mathrm{chop,dia}}] = \Omega\left(\min\left\{1, \max\left\{\frac{TB}{2^s}, \frac{T\Gamma}{2^n}\right\}\right\}\right). \tag{17}$$

The terminal contribution matches a theorem term in the total challenge count $\widehat{\Gamma}$ only if $\Gamma = \Theta(\widehat{\Gamma})$.

**Table 1.** Fixed-resource estimates of the forgery-success lower bounds for an $O(T)$ ideal-cipher-query attack budget, $\Gamma$ selected signed-message challenge calls, and $B$ padded blocks in their inputs. All hidden-key-reuse corrections, absolute constants, and rounding effects are omitted. A displayed bound is subject to the setup and target-length and distinctness conditions stated below the table. When $n = s$, the direct-terminal attack also applies, and this gives the full-output terminal entry.

| Attack | Full output: $n = s$ | Chopped output: $n < s$ |
|---|---|---|
| Terminal-padding fixed-point extension | $\Omega\Big(\min\Big\{1, \frac{T\Gamma}{2^s}\Big\}\Big)$ | $\Omega\Big(\min\Big\{1, \frac{T\Gamma}{2^s}\Big\}\Big)$ |
| Prefix diamond with an expandable message and a full-state bridge | $\Omega\Big(p_{\mathrm{set}}(T)\min\Big\{1, \frac{TB}{2^s}\Big\}\Big)$ | $\Omega\Big(p_{\mathrm{set}}(T)\min\Big\{1, \frac{TB}{2^s}\Big\}\Big)$ |
| Prefix diamond with a direct terminal trial | $\Omega\Big(p_{\mathrm{dia}}(T)\min\Big\{1, \frac{T\Gamma}{2^s}\Big\}\Big)$ | $\Omega\Big(p_{\mathrm{dia}}(T)\min\Big\{1, \frac{T\Gamma}{2^n}\Big\}\Big)$ |

The terminal-padding row assumes $\gamma = \Theta(\Gamma)$. The full-state bridge row assumes that the number of distinct internal states at positions $i \in \mathcal{I}_j$ is $\Theta(B)$, while the two terminal-contact rows assume $\Theta(\Gamma)$ distinct target outputs. The factors $p_{\mathrm{dia}}(T)$ and $p_{\mathrm{set}}(T)$ account for the capped prefix-diamond and expandable-message setup phases. For the setup estimate used here, one diamond leaf is used for each selected target; possible savings from coincident prefixes are not credited. All rows that recompute the selected challenge paths require $B = O(T)$. These rows match bounds written with the theorem-wide quantities $\widehat{B}$ and $\widehat{\Gamma}$ only when, as applicable, $B = \Theta(\widehat{B})$ and $\Gamma = \Theta(\widehat{\Gamma})$.

# B Security Proof Framework for General LMS$^{\mathsf{h}}$

In the following subsections, we first set up the query histories and normalize the adversary. We then trace the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$, collect the relevant hash calls, and mark the calls hidden from the adversary. Finally, we introduce $\mathsf{BadGuess}_{\mathsf{hP}}$ and $\mathsf{BadFPMSpr}_{\mathsf{hP}}$ and reduce multi-user unforgeability of LMS$^{\mathsf{hP}}$ to their non-occurrence.

**Query History and Normalizing Adversary.** At any time in the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$, let the *experimental* $\mathbf{p}$*-query history* contain all $\mathbf{p}$-queries that occur during $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$ and their responses. Note that this includes both adversarial queries (made by $\mathscr{A}$) and honest queries (made by LMS$^{\mathsf{hP}}$). Let the *adversarial* $\mathbf{p}$*-query history* contain all $\mathbf{p}$-queries *known to the adversary* $\mathscr{A}$ and their responses. Note that this includes both the queries directly made by $\mathscr{A}$ and those made by LMS$^{\mathsf{hP}}$ and then disclosed to $\mathscr{A}$ during processing signing queries.

In all of our proofs, we "normalize" the adversary as follows. First, before outputting its forgery, we ask $\mathscr{A}$ to make all $\mathbf{p}$-queries required to run $\mathsf{LMSVrfy}$ on that forgery. Second, for the forgery $(\mathsf{i}^*, M^*, \Sigma^*)$, $\Sigma^* = (q^*, \sigma^*, p^*_0, \ldots, p^*_{h^{\mathsf{i}^*}-1})$, if $\mathscr{A}$ never made signing queries of the form $\mathsf{muSign}(\mathsf{i}^*, q^*, \star)$, then we ask $\mathscr{A}$ to make an additional query to $\mathsf{muSign}(\mathsf{i}^*, q^*, M^{\mathsf{i}^*,q^*})$ for an admissible message $M^{\mathsf{i}^*,q^*} \neq M^*$, as a "proof reference". The $\mathbf{p}$-queries induced by these additional actions are included in the afore-

mentioned experimental and adversarial **p**-query history as well, and all induced costs are charged to the corresponding resource budgets.

**Collecting Involved Hash-Calls.** We trace the *honest hash-calls* made by $\mathrm{LMS}^{\mathsf{h}^{\mathbf{P}}}$ during the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$, set notational conventions and mark Winternitz-chain values that have not been disclosed to the adversary as *hidden*. For the latter, we maintain a set $\mathcal{HS}$, initialize it to empty, and add records to and remove records from it as the experiment proceeds ($\mathcal{HS}$ is thus time-dependent).

First, as described in Fig. 4, during the **Initialization** phase of $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$, $\mathsf{Gen}(m,(h^{\mathsf{i}},w^{\mathsf{i}}),I^{\mathsf{i}})$ is invoked for all $\mathsf{i}\in\{1,...,u\}$, and it internally makes a number of calls to GenOTS and $\mathsf{h}^{\mathbf{P}}$ for constructing Merkle trees. For every user $\mathsf{i}$, let $p^{\mathsf{i}}$ be the LM-OTS parameter determined by $(m,w^{\mathsf{i}})$ as in Sect. 2.2. Concretely,

- For every user index $\mathsf{i}$, LM-OTS leaf index $q$, chain index $d\in\{0,\ldots,p^{\mathsf{i}}-1\}$, and chain step $v\in\{0,\ldots,e^{\mathsf{i}}-1\}$, let $X_0^{\mathsf{i},q,d}\leftarrow x_d$, and let
$$X_{v+1}^{\mathsf{i},q,d}\leftarrow \mathsf{h}^{\mathbf{P}}\big(I^{\mathsf{i}}\|[q]_{32}\|[d]_{16}\|[v]_8\parallel X_v^{\mathsf{i},q,d}\big) \tag{18}$$
be the honest hash-calls occurring in the Winternitz chain in Step (2). We add each coordinate 4-tuple $(\mathsf{i},q,d,v)$ to $\mathcal{HS}$. Note that, as described below, some of them will be removed from $\mathcal{HS}$ during subsequent signing queries. Moreover, each such call is a new *Winternitz-chain challenge* hash-call with prefix $I^{\mathsf{i}}\|[q]_{32}\|[d]_{16}\|[v]_8$.
- For every user index $\mathsf{i}$ and LM-OTS leaf index $q$, after the final Winternitz-chain values $X_{e^{\mathsf{i}}}^{\mathsf{i},q,0},\ldots,X_{e^{\mathsf{i}}}^{\mathsf{i},q,p^{\mathsf{i}}-1}$ have been computed, the call
$$pk_q^{\mathsf{i}}\leftarrow \mathsf{h}^{\mathbf{P}}\big(I^{\mathsf{i}}\|[q]_{32}\|[8080]_{16}\parallel X_{e^{\mathsf{i}}}^{\mathsf{i},q,0}\|\cdots\|X_{e^{\mathsf{i}}}^{\mathsf{i},q,p^{\mathsf{i}}-1}\big) \tag{19}$$
becomes a new *8080 challenge* with prefix $I^{\mathsf{i}}\|[q]_{32}\|[8080]_{16}$.
The Winternitz-chain terminals $X_{e^{\mathsf{i}}}^{\mathsf{i},q,0},\cdots,X_{e^{\mathsf{i}}}^{\mathsf{i},q,p^{\mathsf{i}}-1}$ may be obtained by $\mathscr{A}$ after $\mathscr{A}$ queries $\mathsf{muSign}(\mathsf{i},q,\star)$ for all $q$. We thus directly add the **p**-queries and responses underlying the call in Eq. (19) to the adversarial query history.
- For every user index $\mathsf{i}$ and every leaf index $\rho\in\{2^{h^{\mathsf{i}}},\ldots,2^{h^{\mathsf{i}}+1}-1\}$ in the $\mathsf{i}$-th user's Merkle tree, the call
$$T[\rho]^{\mathsf{i}}\leftarrow \mathsf{h}^{\mathbf{P}}\big(I^{\mathsf{i}}\|[\rho]_{32}\|[8282]_{16}\parallel pk_{\rho-2^{h^{\mathsf{i}}}}^{\mathsf{i}}\big) \tag{20}$$
becomes a new *8282 challenge* with prefix $I^{\mathsf{i}}\|[\rho]_{32}\|[8282]_{16}$. We also add the **p**-queries and responses underlying this call to the adversarial query history.
- For every user index $\mathsf{i}$ and every internal node index $\rho\in\{1,\ldots,2^{h^{\mathsf{i}}}-1\}$ in the $\mathsf{i}$-th user's Merkle tree, the call
$$T[\rho]^{\mathsf{i}}\leftarrow \mathsf{h}^{\mathbf{P}}\big(I^{\mathsf{i}}\|[\rho]_{32}\|[8383]_{16}\|T[2\rho]^{\mathsf{i}}\|T[2\rho+1]^{\mathsf{i}}\big) \tag{21}$$
becomes a *8383 challenge* with prefix $I^{\mathsf{i}}\|[\rho]_{32}\|[8383]_{16}$. We also add the **p**-queries and responses underlying this call to the adversarial query history.

Then, during the **Interaction** phase of $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{h}^{\mathbf{P}}},\mathscr{A},m,u}$, each adversarial signing query creates a new challenge call and *removes some coordinate 4-tuples from* $\mathcal{HS}$. Concretely, consider a user index $\mathsf{i}$ and a LM-OTS leaf index $q$. Since initialized, all

the coordinate 4-tuples of the form $(\mathsf{i}, q, \star, \star)$ have been in $\mathcal{HS}$. By the restrictions in Fig. 4, $\mathscr{A}$ never queried $\mathsf{muSign}(\mathsf{i}, q, \star)$ before, which means all the 4-tuples $(\mathsf{i}, q, \star, \star)$ are still in $\mathcal{HS}$, till $\mathscr{A}$ queries $\mathsf{muSign}(\mathsf{i}, q, M^{\mathsf{i},q})$. Then:

- After sampling $R \xleftarrow{\$} \{0,1\}^n$, the signer evaluates

$$D^{\mathsf{i},q} \leftarrow \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \parallel R \| M^{\mathsf{i},q}\big), \tag{22}$$

  and this becomes a new *8181 challenge* with prefix $I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16}$, randomizer $R$, and adversarially chosen signed message $M^{\mathsf{i},q}$.
- Subsequently, for the $d$-th Winternitz chain, the signature reveals $\sigma^{\mathsf{i},q,d} = X_{V_d}^{\mathsf{i},q,d}$ to $\mathscr{A}$. We now distinguish two cases:
  - Case 1: $V_d > 0$. Then, starting from $X_{V_d}^{\mathsf{i},q,d}$, every $X_v^{\mathsf{i},q,d}$ with $v \geq V_d$ can be computed by $\mathscr{A}$ and are thus "not hidden" any more. Therefore, the corresponding 4-tuples $(\mathsf{i}, q, d, v)$, $v \geq V_d$, are removed from $\mathcal{HS}$. On the other hand, we add the $\mathbf{p}$-queries and responses underlying the $e^{\mathsf{i}} - V_d$ corresponding hash-calls to the adversarial query history.
  - Case 2: $V_d = 0$. In this case, the $d$-th chain becomes completely computable by $\mathscr{A}$, and: (i) all the 4-tuples $(\mathsf{i}, q, d, v)$, $v \in \{0, ..., e^{\mathsf{i}}\}$, are removed from $\mathcal{HS}$; (ii) the $\mathbf{p}$-queries and responses underlying all the $e^{\mathsf{i}}$ hash-calls are added to the adversarial query history.

**Event $\mathsf{BadGuess}_{\mathsf{hP}}$** concerns the aforementioned "unrecoverability" of the still-hidden Winternitz-chain values marked by the coordinates in $\mathcal{HS}$. Concretely, $\mathsf{BadGuess}_{\mathsf{hP}}$ occurs, if at any time during $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathsf{hP},\mathscr{A},m,u}$ there exists a coordinate 4-tuple $(\mathsf{i}, q, d, v)$ that is still in $\mathcal{HS}$, but the corresponding hash-call $\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X_v^{\mathsf{i},q,d}\big)$ can be evaluated by using the $\mathbf{p}$-queries and responses in the adversarial $\mathbf{p}$-query history. This notion is a bit informal, and its concrete meanings will be clarified in the corresponding sections.

**Event $\mathsf{BadFPMSpr}_{\mathsf{hP}}$** occurs if $\mathscr{A}$ finds a fixed-prefix second preimage of a required form for one of the aforementioned challenge hash-calls, and the fixed-prefix second preimage can be evaluated by using the $\mathbf{p}$-queries and responses in the experimental query history. Concretely, $\mathsf{BadFPMSpr}_{\mathsf{hP}}$ occurs if any of the following conditions holds:

(i) There exists a collected Winternitz-chain challenge at coordinate $(\mathsf{i}, q, d, v)$, and $\mathscr{A}$ finds $M^\star$ with $|M^\star| = n$, $M^\star \neq X_v^{\mathsf{i},q,d}$, and

$$\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| M^\star\big) = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X_v^{\mathsf{i},q,d}\big).$$

(ii) There exists a collected 8080 challenge at coordinate $(\mathsf{i}, q)$, and $\mathscr{A}$ finds $M^\star$ with $|M^\star| = p^{\mathsf{i}} n$, $M^\star \neq X_{e^{\mathsf{i}}}^{\mathsf{i},q,0} \| \cdots \| X_{e^{\mathsf{i}}}^{\mathsf{i},q,p^{\mathsf{i}}-1}$ and

$$\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| M^\star\big) = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| X_{e^{\mathsf{i}}}^{\mathsf{i},q,0} \| \cdots \| X_{e^{\mathsf{i}}}^{\mathsf{i},q,p^{\mathsf{i}}-1}\big).$$

(iii) There exists a collected 8282 challenge at coordinate $(\mathsf{i}, \rho)$, and $\mathscr{A}$ finds $M^\star$ with $|M^\star| = n$, $M^\star \neq pk_{\rho-2^{h^{\mathsf{i}}}}^{\mathsf{i}}$ and

$$\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8282]_{16} \| M^\star\big) = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8282]_{16} \| pk_{\rho-2^{h^{\mathsf{i}}}}^{\mathsf{i}}\big).$$

(iv) There exists a collected 8383 challenge at coordinate $(\mathsf{i}, \rho)$, and $\mathscr{A}$ finds $M^\star$ with $|M^\star| = 2n$, $M^\star \neq T[2\rho]^{\mathsf{i}} \| T[2\rho+1]^{\mathsf{i}}$ and

$$\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8383]_{16} \| M^\star\big) = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8383]_{16} \| T[2\rho]^{\mathsf{i}} \| T[2\rho+1]^{\mathsf{i}}\big).$$

(v) There exists a collected 8181 challenge at coordinate $(\mathsf{i}, q)$, and $\mathscr{A}$ finds $M^\star \neq R^{\mathsf{i},q} \| M^{\mathsf{i},q}$ such that $M^\star = R' \| M'$ with $|R'| = n$, $M'$ in the message domain of SIGN, and

$$\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \| M^\star\big) = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \| R^{\mathsf{i},q} \| M^{\mathsf{i},q}\big).$$

**From Forgery to Second Preimage.** Formally, we prove the following lemma.

**Lemma 1.** *The experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$ *outputting 1 indicates that either* BadGuess *or* BadFPMSpr *occurred during* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$. *Consequently,*

$$\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{LMS}^{\mathsf{hP}},\mathscr{A},m,u} \Rightarrow 1\big] \leq \Pr\big[\mathsf{BadGuess} \vee \mathsf{BadFPMSpr}\big]. \tag{23}$$

*Proof.* Suppose that $\mathscr{A}$ outputs a valid forgery $(\mathsf{i}^\star, M^\star, \Sigma^\star)$, and let $e^{\mathsf{i}^\star} := 2^{w^{\mathsf{i}^\star}} - 1$ be the parameter of the i-th user. Write

$$pk_{\mathsf{i}^\star} = \big(m, h^{\mathsf{i}^\star}, w^{\mathsf{i}^\star}, I^{\mathsf{i}^\star}, T[1]^{\mathsf{i}^\star}\big), \qquad \Sigma^\star = \big(q^\star, \sigma^\star, p^\star_0, \ldots, p^\star_{h^{\mathsf{i}^\star}-1}\big),$$

where $\sigma^\star = \big(R^\star, \sigma^\star_0, \ldots, \sigma^\star_{p^{\mathsf{i}^\star}-1}\big)$. Assume that $\Sigma \leftarrow \mathsf{muSign}^{\mathsf{P}}(\mathsf{i}^\star, q^\star, M)$ is the "reference" signing query for the pair $(\mathsf{i}^\star, q^\star)$—it necessarily exists, since we focus on normalized adversaries. Write

$$\Sigma = \big(q^\star, \sigma, p_0, \ldots, p_{h^{\mathsf{i}^\star}-1}\big), \qquad \sigma = \big(R, \sigma_0, \ldots, \sigma_{p^{\mathsf{i}^\star}-1}\big).$$

Run $\mathsf{LMSVrfy}$ on $(pk_{\mathsf{i}^\star}, M^\star, \Sigma^\star)$ and $(pk_{\mathsf{i}^\star}, M, \Sigma)$ in parallel: both computations end at the public Merkle-tree root. We compare the two computations backwards from that root.

As notations, put

$$\rho_0 := q^\star + 2^{h^{\mathsf{i}^\star}}, \qquad \rho_{i+1} := \lfloor \rho_i / 2 \rfloor \quad (0 \leq i < h^{\mathsf{i}^\star}),$$

and let $Z^\star_i$ and $Z_i$ be, respectively, the forged and reference leaf values at coordinate $(\mathsf{i}^\star, \rho_i)$, $0 \leq i \leq h^{\mathsf{i}^\star}$. With the above notations, it can be seen that, for every $i \in \{0, \ldots, h^{\mathsf{i}^\star} - 1\}$, the hash-calls in the Merkle trees of the two verification processes are

$$\begin{aligned} Z^\star_{i+1} &:= \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_{i+1}]_{32} \| [8383]_{16} \| \mathsf{Pair}^\star_i\big), \\ Z_{i+1} &:= \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_{i+1}]_{32} \| [8383]_{16} \| \mathsf{Pair}_i\big), \end{aligned}$$

where

$$\mathsf{Pair}^\star_i := \begin{cases} Z^\star_i \| p^\star_i, & \text{if } \rho_i \text{ is even,} \\ p^\star_i \| Z^\star_i, & \text{if } \rho_i \text{ is odd,} \end{cases} \qquad \mathsf{Pair}_i := \begin{cases} Z_i \| p_i, & \text{if } \rho_i \text{ is even,} \\ p_i \| Z_i, & \text{if } \rho_i \text{ is odd.} \end{cases}$$

Since both verification processes return 1, it holds $Z^\star_{h^{\mathsf{i}^\star}} = Z_{h^{\mathsf{i}^\star}}$. Depending on where the values differ, we distinguish three cases as follows.

*Case 1:* $\mathsf{Pair}_i^\star \neq \mathsf{Pair}_i$ *for some* $i \in \{0, \ldots, h^{\mathsf{i}^\star} - 1\}$. Consider the largest $i$ with $\mathsf{Pair}_i^\star \neq \mathsf{Pair}_i$. It then holds

$$Z_{i+1}^\star = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_{i+1}]_{32} \| [8383]_{16} \| \mathsf{Pair}_i^\star\big) = Z_{i+1} = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_{i+1}]_{32} \| [8383]_{16} \| \mathsf{Pair}_i\big),$$

which indicates the occurrence of $\mathsf{BadFPMSpr}_{\mathsf{hP}}$.

*Case 2:* $\mathsf{Pair}_i^\star = \mathsf{Pair}_i$ *for all* $i \in \{0, \ldots, h^{\mathsf{i}^\star} - 1\}$, but the LM-OTS public keys or terminal Winternitz values differ. Then it holds $Z_0^\star = Z_0$. In the verification processes, the two values are derived via

$$Z_0^\star = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_0]_{32} \| [8282]_{16} \| pk^\star\big) = Z_0 = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [\rho_0]_{32} \| [8282]_{16} \| pk\big),$$

where $pk^\star$ and $pk$ are the two reconstructed LM-OTS public keys. If $pk^\star \neq pk$, then $Z_0^\star = Z_0$ indicates $\mathsf{BadFPMSpr}_{\mathsf{hP}}$. Otherwise, let $X_{e^{\mathsf{i}^\star}}^{\star,0}, \ldots, X_{e^{\mathsf{i}^\star}}^{\star,p^{\mathsf{i}^\star}-1}$ and $X_{e^{\mathsf{i}^\star}}^{\mathsf{i}^\star,q^\star,0}, \ldots, X_{e^{\mathsf{i}^\star}}^{\mathsf{i}^\star,q^\star,p^{\mathsf{i}^\star}-1}$ be, respectively, the forged and honest terminal Winternitz values in the two verification processes. If the two terminal-value vectors differ, then $pk^\star = pk$ means

$$\begin{aligned}&\mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [8080]_{16} \| X_{e^{\mathsf{i}^\star}}^{\star,0} \| \cdots \| X_{e^{\mathsf{i}^\star}}^{\star,p^{\mathsf{i}^\star}-1}\big)\\ &\quad = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [8080]_{16} \| X_{e^{\mathsf{i}^\star}}^{\mathsf{i}^\star,q^\star,0} \| \cdots \| X_{e^{\mathsf{i}^\star}}^{\mathsf{i}^\star,q^\star,p^{\mathsf{i}^\star}-1}\big),\end{aligned}$$

and thus $\mathsf{BadFPMSpr}_{\mathsf{hP}}$ occurs.

*Case 3: all terminal Winternitz values agree.* In this case, let

$$\begin{aligned}D^\star &:= \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [8181]_{16} \| R^\star \| M^\star\big),\\ D &:= \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [8181]_{16} \| R \| M\big)\end{aligned}$$

be the forged and reference digests, and let $(V_0, \ldots, V_{p^{\mathsf{i}^\star}-1})$ and $(V_0^\star, \ldots, V_{p^{\mathsf{i}^\star}-1}^\star)$ be their digit vectors after appending the Winternitz checksum. If $D^\star = D$, then the two 8181 inputs are distinct because $M^\star \neq M$, and $\mathsf{BadFPMSpr}_{\mathsf{hP}}$ occurs.

Suppose therefore that $D^\star \neq D$. The checksum construction guarantees an index $d$ such that $V_d^\star < V_d$. Indeed, either a forged message digit is smaller, or all forged message digits are at least their reference counterparts with one strict inequality; in the latter case, the forged checksum is smaller and one of its digits is smaller. In particular, $V_d \geq 1$.

For this chain, write the honest states as $X_i := X_i^{\mathsf{i}^\star,q^\star,d}$, and define the forged verification states by $X_{V_d^\star}^\star := \sigma_d^\star$,

$$X_{i+1}^\star := \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [d]_{16} \| [i]_8 \| X_i^\star\big), \qquad V_d^\star \leq i < e^{\mathsf{i}^\star}.$$

The terminal values agree, $X_{e^{\mathsf{i}^\star}} = X_{e^{\mathsf{i}^\star}}^\star$, by the preceding comparison of the LM-OTS public-key inputs. We further distinguish two subcases.

- Subcase 3.1: there exists $i \in \{V_d - 1, \ldots, e^{\mathsf{i}^\star} - 1\}$ such that $X_i \neq X_i^\star$. Let $i_0$ be the largest such index. Then $X_{i_0} \neq X_{i_0}^\star$, whereas $X_{i_0+1} = X_{i_0+1}^\star$. Therefore,

  $$X_{i_0+1} = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [d]_{16} \| [i_0]_8 \| X_{i_0}\big) = X_{i_0+1}^\star = \mathsf{h}^{\mathsf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [d]_{16} \| [i_0]_8 \| X_{i_0}^\star\big).$$

  This again indicates the occurrence of $\mathsf{BadFPMSpr}_{\mathsf{hP}}$.

– Subcase 3.2: $X_i = X_i^\star$ for all $i \in \{V_d-1, \ldots, e^{\mathsf{i}^\star}-1\}$. In particular, $X^\star_{V_d-1} = X_{V_d-1}$. Because $V_d \geq 1$, the coordinate $(\mathsf{i}^\star, q^\star, d, V_d - 1)$ was inserted into $\mathcal{HS}$ during initialization and is not removed by the reference signing update, which removes only the coordinates with $v \geq V_d$. It therefore remains in $\mathcal{HS}$ after $\mathsf{muSign}^{\mathbf{P}}(\mathsf{i}^\star, q^\star, M)$. Meanwhile, since $V_d^\star < V_d$ and since we have normalized the adversary, the $\mathbf{p}$-queries needed to evaluate $\mathsf{h}^{\mathbf{P}}\big(I^{\mathsf{i}^\star} \| [q^\star]_{32} \| [d]_{16} \| [V_d - 1]_8 \| X^\star_{V_d-1}\big)$ have been in the adversarial query history. Hence $X^\star_{V_d-1} = X_{V_d-1}$ indicates the occurrence of $\mathsf{BadGuess}_{\mathsf{hP}}$.

This completes the proof. □

# C Detailed Analysis for $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$

This section applies the framework in Sect. B to analyze $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$. For convenience, we recall Theorem 1 below.

**Theorem 3 (Multi-user unforgeability of $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$).** *During the experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$*, assume that: (i) the adversary* $\mathscr{A}$ *makes at most* $T$ *adversarial ideal-cipher queries to* $\mathbf{E}$ *(including the auxiliary queries for our proof); (ii)* $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ *internally makes at most* $\widehat{\Gamma}$ *calls to* $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}$*, which results in at most* $\widehat{B}$ *Davies-Meyer calls (it thus holds* $1 \leq \widehat{\Gamma} \leq \widehat{B}$*). Further assume* $T + \widehat{B} \leq 2^s/2$*. Then, the construction* $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ *has*

$$\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\big] \leq \frac{26\widehat{B}T + 12\widehat{B}^2}{2^s}, \tag{24}$$

To prove Theorem 3, we first introduce useful tools on conditional expectations in Sect. C.1. We then focus on $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$: first elaborate on useful properties of the challenge hash-calls in Sect. C.2, then introduce the event $\mathsf{BadHonest}$ in Sect. C.3, then bound the probabilities of $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ and $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ in Sect. C.4 and C.5 respectively. We finally apply Lemma 1 in Sect. C.6 to complete the proof.

## C.1 Law of Iterated Expectations

Let $(\Omega, \mathcal{F}, \Pr)$ be a discrete probability space, and let $\mathcal{G} \subseteq \mathcal{H} \subseteq \mathcal{F}$. The *law of iterated expectations* states that the expectation $\mathbb{E}[X \mid \mathcal{G}]$ of a random variable $X$ conditioned on $\mathcal{G}$ equals the expectation of another conditional expectation $\mathbb{E}[X \mid \mathcal{H}]$ conditioned on $\mathcal{G}$:

$$\mathbb{E}\big[\mathbb{E}[X \mid \mathcal{H}] \,|\, \mathcal{G}\big] = \mathbb{E}\big[X \mid \mathcal{G}\big]. \tag{25}$$

It has an unconditional corollary $\mathbb{E}\big[\mathbb{E}[X \mid \mathcal{G}]\big] = \mathbb{E}[X]$. Both identities can be proven by the law of total probability.

Moreover, if $Y$ is $\mathcal{G}$-measurable and the relevant expectations are finite, then

$$\mathbb{E}\big[XY \mid \mathcal{G}\big] = Y\,\mathbb{E}\big[X \mid \mathcal{G}\big], \tag{26}$$

$$\mathbb{E}[XY] = \mathbb{E}\big[Y\,\mathbb{E}[X \mid \mathcal{G}]\big]. \tag{27}$$

For an event $E$, we use the notation

$$\Pr\big[E \mid \mathcal{G}\big] := \mathbb{E}\big[\mathbf{1}_E \mid \mathcal{G}\big].$$

Consequently,

$$\Pr\big[E\big] = \mathbb{E}\big[\mathbf{1}_E\big] = \mathbb{E}\big[\Pr[E \mid \mathcal{G}]\big]. \tag{28}$$

**Ideal-Cipher Query Records and Transcripts.** We maintain a *transcript*, in the form of a list $\mathcal{Q}$, for the ideal-cipher queries and responses appeared during an execution of $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\text{LMS}^{\text{sMD}^{\text{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$. An ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ or $X \leftarrow \mathbf{E}_W^{-1}(Y)$ is called *fresh*, if neither $Y \leftarrow \mathbf{E}_W(X)$ nor $X \leftarrow \mathbf{E}_W^{-1}(Y)$ occurred before. As the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\text{LMS}^{\text{sMD}^{\text{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$ proceeds, every fresh query $Y \leftarrow \mathbf{E}_W(X)$ or $X \leftarrow \mathbf{E}_W^{-1}(Y)$ "materializes" an *ideal-cipher query record* (or simply a *record*) $(W; X, Y)$ in $\mathcal{Q}$. On the other hand, every non-fresh query may be viewed as a mere "lookup" of a corresponding record in $\mathcal{Q}$. Records in $\mathcal{Q}$ are maintained in the order of their appearance. Hence, the $k$ th record $(W_k; X_k, Y_k)$ in $\mathcal{Q}$ indicates that the $k$ th *fresh* ideal-cipher query in $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\text{LMS}^{\text{sMD}^{\text{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$ is either $Y_k \leftarrow \mathbf{E}_{W_k}(X_k)$ or $X_k \leftarrow \mathbf{E}_{W_k}^{-1}(Y_k)$.

## C.2 Properties of the Challenge Calls

We use the five challenge types and the time-dependent set $\mathcal{HS}$ exactly as defined in Sect. B.

For the parameterized full-state model $n = s$, the challenge hash-calls collected in Sect. B have the following properties.

First, each collected challenge call is associated with a unique *coordinate tuple*. Furthermore, for every user index $\mathfrak{i} \in \{1, ..., u\}$, each collected challenge call is associated with a unique prefix value:

(i) Each Winternitz-chain challenge is associated with a unique coordinate triple $(q, d, v)$, and the corresponding prefix value is $I^{\mathfrak{i}} \| [q]_{32} \| [d]_{16} \| [v]_8$. Moreover, we can use the coordinate 4-tuple $(\mathfrak{i}, q, d, v)$ to refer to this call;
(ii) Each 8080 challenge is associated with a unique coordinate $q$, and the corresponding prefix is $I^{\mathfrak{i}} \| [q]_{32} \| [8080]_{16}$. Moreover, we can use the coordinate triple $(\mathfrak{i}, q, 8080)$ to refer to this call;
(iii) Each 8282 challenge is associated with a unique coordinate $\rho$, and the corresponding prefix is $I^{\mathfrak{i}} \| [\rho]_{32} \| [8282]_{16}$. Moreover, we can use the coordinate triple $(\mathfrak{i}, \rho, 8282)$ to refer to this call;
(iv) Each 8383 challenge is associated with a unique coordinate $\rho$, and the corresponding prefix is $I^{\mathfrak{i}} \| [\rho]_{32} \| [8383]_{16}$. Moreover, we can use the coordinate triple $(\mathfrak{i}, \rho, 8383)$ to refer to this prefix;
(v) Each 8181 challenge is associated with a unique coordinate $q$, and the corresponding prefix is $I^{\mathfrak{i}} \| [q]_{32} \| [8181]_{16}$. Moreover, we can use the coordinate triple $(\mathfrak{i}, q, 8181)$ to refer to this call.

Since $0 \leq d < p^{\mathfrak{i}} \leq 2^{15}$, the 16-bit chain-index field is disjoint from the reserved values $[8080]_{16}, [8181]_{16}, [8282]_{16}, [8383]_{16}$. It follows that the prefix values collected for the same user index are both *distinct* and *prefix-free*. This prefix-freeness means that

one cannot misinterpret a hash evaluation with a longer prefix as an evaluation with another shorter prefix.

By the above, we denote by $\mathsf{prefix}_{\mathbf{ct}}$ the prefix value at the *coordinate tuple* $\mathbf{ct}$. This means:

$$\mathsf{prefix}_{\mathbf{ct}} := \begin{cases} I^{\mathrm{i}}\|[q]_{32}\|[d]_{16}\|[v]_8, & \text{if } \mathbf{ct} = (\mathrm{i}, q, d, v), \\ I^{\mathrm{i}}\|[q]_{32}\|[8080]_{16}, & \text{if } \mathbf{ct} = (\mathrm{i}, q, 8080), \\ I^{\mathrm{i}}\|[q]_{32}\|[8181]_{16}, & \text{if } \mathbf{ct} = (\mathrm{i}, q, 8181), \\ I^{\mathrm{i}}\|[\rho]_{32}\|[8282]_{16}, & \text{if } \mathbf{ct} = (\mathrm{i}, \rho, 8282), \\ I^{\mathrm{i}}\|[\rho]_{32}\|[8383]_{16}, & \text{if } \mathbf{ct} = (\mathrm{i}, \rho, 8383). \end{cases} \tag{29}$$

**"Block layout" of challenge calls.** As mentioned in Sect. 2.2, while we are using a general parameter tuple $(\iota, n, s, r)$, for the "block layout" of challenge hash-calls we simply follow the LMS_SHA256_M32_H$h$. For this, for LMS_SHA256_M32_H$h$, where $n = s = 256$, the "block layout" is as follows.

- For each coordinate 4-tuple $(\mathrm{i}, q, d, v)$, define
$$W_v^{\mathrm{i},q,d} := \mathsf{lpad}\big(I^{\mathrm{i}}\|[q]_{32}\|[d]_{16}\|[v]_8\|X_v^{\mathrm{i},q,d}\big).$$
The unpadded input has length $128+32+16+8+256 = 440$ bits. Since $440+65 = 505 \le 512$, it follows that $|W_v^{\mathrm{i},q,d}| = r = 512$, and the challenge call is evaluated via a **single** ideal-cipher call $X_{v+1}^{\mathrm{i},q,d} \leftarrow \mathbf{E}_{W_v^{\mathrm{i},q,d}}(IV) \boxplus IV$.
- For each coordinate triple $(\mathrm{i}, \rho, 8282)$, define
$$W_{8282}^{\mathrm{i},\rho} := \mathsf{lpad}\big(I^{\mathrm{i}}\|[\rho]_{32}\|[8282]_{16}\|pk^{\mathrm{i}}_{\rho-2^{h^{\mathrm{i}}}}\big).$$
The unpadded input has length $128 + 32 + 16 + 256 = 432$ bits. Since $432 + 65 = 497 \le 512$, the 8282 challenge is evaluated via a **single** ideal-cipher-call $T[\rho]^{\mathrm{i}} \leftarrow \mathbf{E}_{W_{8282}^{\mathrm{i},\rho}}(IV) \boxplus IV$.
- For each coordinate triple $(\mathrm{i}, \rho, 8383)$, the unpadded input has length $176+2\cdot256 = 688$ bits. Its padded block length is $\lceil (688+65)/512 \rceil = 2$. The first padded block is
$$\overline{U}_{8383}^{\mathrm{i},\rho}[1] := I^{\mathrm{i}}\|[\rho]_{32}\|[8383]_{16}\|T[2\rho]^{\mathrm{i}}\|\mathsf{left}_{80}\big(T[2\rho+1]^{\mathrm{i}}\big),$$
and the first ideal-cipher call is $\mathbf{E}_{\overline{U}_{8383}^{\mathrm{i},\rho}[1]}(IV)$.
- For each coordinate triple $(\mathrm{i}, q, 8080)$, the unpadded input has length $176 + 256p^{\mathrm{i}}$ bits. This is the longest key-generation hash evaluation. Its first padded block is
$$\overline{U}_{8080}^{\mathrm{i},q}[1] := I^{\mathrm{i}}\|[q]_{32}\|[8080]_{16}\|X_{e^{\mathrm{i}}}^{\mathrm{i},q,0}\|\mathsf{left}_{80}\big(X_{e^{\mathrm{i}}}^{\mathrm{i},q,1}\big),$$
and the first ideal-cipher call is $\mathbf{E}_{\overline{U}_{8080}^{\mathrm{i},q}[1]}(IV)$.

### C.3 Event BadHonest and Its Probability

The event $\mathsf{BadHonest}$ occurs if, when $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ internally evaluates a challenge hash-call for the first time,[13] $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ makes an honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ for which at least one of the following holds:

[13] Note that by LMS design, many challenge-calls are intended to be evaluated by $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ during both key generation and signing.

(i) before this honest ideal-cipher query, the record $(W; X, Y)$ has already been present in $\mathcal{Q}$;
(ii) its Davies-Meyer endpoint satisfies $X \boxplus Y \in \{IV, X\}$;
(iii) there is another, distinct record $(W'; X', Y')$ already present in $\mathcal{Q}$ such that $X \boxplus Y \in \{X', X' \boxplus Y'\}$.

Clause (iii) includes records "materialized" by earlier adversarial and honest ideal-cipher queries.

The definition emphasizes "evaluates for the first time" because a type of non-fresh honest query is *designed* to occur, which must be excluded from Clause (i). Concretely, the signing algorithm will revisit records "materialized" during key generation when it recomputes a Winternitz-chain prefix (even if the corresponding coordinates are in $\mathcal{HS}$). The involved honest queries are certainly non-fresh. Since it is not part of the first evaluation of that challenge hash-call, it does not cause BadHonest Clause (i). In contrast, if an honest ideal-cipher query made during the first evaluation of another collected challenge hash-call accesses an already materialized record, clause (i) of BadHonest occurs.

Below we first formalize a lemma regarding the conditional distribution of the "anchor" values in challenge-calls. We then use this lemma to bound $\Pr[\mathsf{BadHonest}]$.

**"Anchor" values entering first challenge blocks.** Denote by $\gamma_{\mathrm{ch}}$ the actual number of challenge calls occurring in the execution. Let $\mathcal{T}_{\mathrm{ch}}$ be the set of their coordinate tuples, listed in order of appearance as

$$\mathbf{ct}_1, \ldots, \mathbf{ct}_{\gamma_{\mathrm{ch}}}.$$

As mentioned in Sect. 3.3, in every challenge hash-call, the key $W_1$ of the first underlying ideal-cipher query $\mathbf{E}_{W_1}(IV)$ contains an associated $s$-bit (pseudo)random "anchor" value. For each $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, denote this value by $\mathsf{anch}_{\mathbf{ct}}$. Concretely,

$$\mathsf{anch}_{\mathbf{ct}} := \begin{cases} X_v^{\mathrm{i},q,d}, & \text{if } \mathbf{ct} = (\mathrm{i}, q, d, v), \\ X_{e^{\mathrm{i}}}^{\mathrm{i},q,0}, & \text{if } \mathbf{ct} = (\mathrm{i}, q, 8080), \\ R^{\mathrm{i},q}, & \text{if } \mathbf{ct} = (\mathrm{i}, q, 8181), \\ pk^{\mathrm{i}}_{\rho-2^{h^{\mathrm{i}}}}, & \text{if } \mathbf{ct} = (\mathrm{i}, \rho, 8282), \\ T[2\rho]^{\mathrm{i}}, & \text{if } \mathbf{ct} = (\mathrm{i}, \rho, 8383). \end{cases} \tag{30}$$

For every $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, fix all "non-anchor" bits of its first block, and let $K_{\mathbf{ct}}(a)$ denote the resulting first-block key when the "anchor" is set to $a \in \{0,1\}^s$. Since the complete "anchor" occupies a fixed bit position within the first block, the map $a \mapsto K_{\mathbf{ct}}(a)$ is injective.

We equip each execution with a *global clock* that assigns distinct time indices to successive actions (including ideal-cipher queries, random sampling, and updates to auxiliary variables or query histories) and allows us to refer unambiguously to the execution history immediately before or after any action. Let $\tau_{\mathrm{HB}}$ and $\tau_{\mathrm{G}}$ be the first occurrence times of BadHonest and $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ on this clock, with value $\infty$ if the corresponding event never occurs.

For each $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, let $\theta_{\mathbf{ct}}$ be the time immediately before the first honest ideal-cipher query of the challenge call at coordinate $\mathbf{ct}$. Let $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ denote the execution history at time $\theta_{\mathbf{ct}}$, with the anchor value $\mathsf{anch}_{\mathbf{ct}}$, the query response producing it (if

any), and all copies or metadata that would reveal these values removed. Concretely, if $\mathsf{anch}_{\mathbf{ct}}$ is a direct secret or randomizer, $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ omits its value; if $\mathsf{anch}_{\mathbf{ct}}$ is derived from another "producer ideal-cipher query", $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ omits the "producer query's" response whose Davies-Meyer endpoint is $\mathsf{anch}_{\mathbf{ct}}$. Subject to these omissions, $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ retains the schedule, identifiers, and all other query keys, inputs, public data, and assigned table entries available no later than $\theta_{\mathbf{ct}}$. In the derived-anchor case, it also retains the (already fixed) "producer query's" key and input.

For the $t$ th adversarial ideal-cipher query, let $\vartheta_t$ be the time after its direction, key, and input or output have been fixed but before the query is processed. Let $\mathcal{V}^{+}_{t}$ be the complete adversarial view at this time: it includes the fixed query parameters, all preceding public keys, signing queries and responses, and the adversarial ideal-cipher history, but excludes records that have neither been accessed by an adversarial ideal-cipher query nor been transferred (due to signing) to the adversarial ideal-cipher history.

**Lemma 2 (Conditional bounds for full-state anchor values).** *Assume $T + \widehat{B} < 2^s$. Fix an arbitrary identifier vector $(I^1, ..., I^u)$.*

*(i) For every $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, condition on an arbitrary fixed value of the redacted history $\mathcal{F}^{\circ}_{\mathbf{ct}-}$. Then, for every $\mathcal{F}^{\circ}_{\mathbf{ct}-}$-measurable "target value" $\widehat{a} \in \{0,1\}^s$,*

$$\Pr\left[\tau_{\mathrm{HB}} \geq \theta_{\mathbf{ct}} \ \wedge \ \mathsf{anch}_{\mathbf{ct}} = \widehat{a} \ \middle|\ \mathcal{F}^{\circ}_{\mathbf{ct}-}\right] \leq \frac{1}{2^s - T - \widehat{B}}. \tag{31}$$

*Informally, at time $\theta_{\mathbf{ct}}$ (when the first ideal-cipher query of the challenge-call at $\mathbf{ct}$ occurs), consider the whole execution history: removing the "anchor" value $\mathsf{anch}_{\mathbf{ct}}$ and the ideal-cipher query response that generates $\mathsf{anch}_{\mathbf{ct}}$ from this execution history yields the "redacted" execution history $\mathcal{F}^{\circ}_{\mathbf{ct}-}$. Then, given this "redacted" history $\mathcal{F}^{\circ}_{\mathbf{ct}-}$, the value $\mathsf{anch}_{\mathbf{ct}}$ remains "unpredictable". In particular, the probability that: (i) $\mathsf{BadHonest}$ has not yet occurred before the first ideal-cipher query of this challenge call is executed, and (ii) the hidden "anchor" value of this challenge call equals $\widehat{a}$, is $O(1/2^s)$.*

*(ii) Consider an adversarial ideal-cipher query selected adaptively from the information available immediately before it is processed. Let $t$ be its possibly random index among the adversarial ideal-cipher queries, and let $\vartheta_t$ denote the time immediately before it is processed. Here, $\mathcal{V}^{+}_{t}$ denotes the complete view available to the adversary at time $\vartheta_t$. Further, let the Winternitz coordinate $(\mathrm{i}, q, d, v)$ and $\widehat{x} \in \{0,1\}^s$ be $\mathcal{V}^{+}_{t}$-measurable.*[14] *Put*

$$\mathsf{G}_t := \{\tau_{\mathrm{HB}} \geq \vartheta_t\} \cap \{\tau_{\mathrm{G}} \geq \vartheta_t\}.$$

*Then*

$$\Pr\left[\mathsf{G}_t \ \wedge \ \left\{(\mathrm{i}, q, d, v) \in \mathcal{HS} \text{ at } \vartheta_t\right\} \ \wedge \ \left\{X_v^{\mathrm{i},q,d} = \widehat{x}\right\} \ \middle|\ \mathcal{V}^{+}_{t}\right] \leq \frac{1}{2^s - T - \widehat{B}}. \tag{32}$$

*Namely, the probability that: (i) neither $\mathsf{BadHonest}$ nor $\mathsf{BadGuess}$ has occurred before the $t$-th adversarial ideal-cipher query is processed (the meaning of $\mathsf{G}_t$), and (ii) at time $\vartheta_t$, $(\mathrm{i}, q, d, v)$ remains "hidden" to $\mathscr{A}$, and (iii) $X_v^{\mathrm{i},q,d}$, the "anchor" value of $(\mathrm{i}, q, d, v)$, equals a value $\widehat{x}$ that can be computed by $\mathscr{A}$ from its view.*

*Proof.* We prove the two clauses in turn.

[14] A random variable is *$\mathcal{F}$-measurable* if its value is completely determined by the information in $\mathcal{F}$.

*Proof of Clause (i).* First consider Eq. (31). If $\mathsf{anch}_{\mathbf{ct}}$ is $X_0^{\mathrm{i},q,d}$ ($\mathbf{ct} = (\mathrm{i}, q, d, 0)$) or $R^{\mathrm{i},q}$ ($\mathbf{ct} = (\mathrm{i}, q, 8181)$), then it is directly sampled, and is uniform in $\{0,1\}^s$ and independent of the "redacted" history $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ (since $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ does not contain $\mathsf{anch}_{\mathbf{ct}}$). Consequently, for every fixed $\widehat{a} \in \{0,1\}^s$,

$$\Pr[\mathsf{anch}_{\mathbf{ct}} = \widehat{a} \mid \mathcal{F}^{\circ}_{\mathbf{ct}-}] = \frac{1}{2^s} \leq \frac{1}{2^s - T - \widehat{B}}.$$

Adding the condition $\tau_{\mathrm{HB}} \geq \theta_{\mathbf{ct}}$ can only decrease the probability; hence the claimed bound follows.

For the remaining, suppose that $\mathsf{anch}_{\mathbf{ct}}$ is derived. On each fixed value of $\mathcal{F}^{\circ}_{\mathbf{ct}-}$, let $(W^-; X^-, Y^-)$ be ideal-cipher query record with $X^- \boxplus Y^- = \mathsf{anch}_{\mathbf{ct}}$, i.e., the record that "produces" $\mathsf{anch}_{\mathbf{ct}}$. By construction, this means $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ internally makes a query $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$.

If $(W^-; X^-, Y^-)$ has been "materialized" before $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ querying $\mathbf{E}_{W^-}(X^-)$, then clause (i) of $\mathsf{BadHonest}$ has already occurred before $\theta_{\mathbf{ct}}$, and the left-hand side of Eq. (31) is zero.

Otherwise, the "producer query" $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$ is fresh. At the time of sampling, $Y^-$ is uniform among the outputs not yet assigned under $W^-$.

Let $r_-$ be the number of other outputs under $W^-$ fixed by $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ at time $\theta_{\mathbf{ct}}$. Then $r_- \leq T + \widehat{B}$. Note that these may include records "materialized" after the "producer query" $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$. We next show that, conditional on $\mathcal{F}^{\circ}_{\mathbf{ct}-}$, $Y^-$ is uniform among the remaining $2^s - r_-$ possible values. This essentially requires to prove that the information, particularly the new ideal-cipher query records, added to $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ between processing $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$ and time $\theta_{\mathbf{ct}}$ does not significantly affect the uniformity of $Y^-$.

By construction of LMS, no adversarial ideal-cipher query can occur during this interval, since key generation and each signing invocation execute without interleaved adversarial queries. Regarding honest ideal-cipher queries, we distinguish four cases:

(i) For a Winternitz challenge with $v > 0$, the anchor is produced by the chain step labeled $[v-1]_8$ and is next used by the step labeled $[v]_8$; these distinct labels also separate the producer and consumer keys.
(ii) For an 8080 challenge, the anchor is the terminal value of chain $d = 0$, and the remaining Winternitz chains are computed without using this value.
(iii) For an 8282 challenge, the anchor is the output of the corresponding 8080 call and is first used as an input by the corresponding leaf-hashing call.
(iv) For an 8383 challenge, the anchor is the completed left-child value; any intervening computations before the parent hash call are performed without using this value.

Therefore, during the interval after $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$ is processed but before time $\theta_{\mathbf{ct}}$, the keys and inputs of the intervening honest ideal-cipher queries are computed from the fixed public parameters and the other chain or tree values retained in $\mathcal{F}^{\circ}_{\mathbf{ct}-}$. Replacing $\mathsf{anch}_{\mathbf{ct}}$ by another candidate compatible with the retained table entries leaves these query parameters and their execution order unchanged. Moreover, $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ removes $Y^-$, its endpoint $\mathsf{anch}_{\mathbf{ct}}$, and all copies or metadata that would reveal them.

If the first evaluation of another collected challenge accesses the record $(W^-; X^-, Y^-)$ after $Y^- \leftarrow \mathbf{E}_{W^-}(X^-)$ is processed but before time $\theta_{\mathbf{ct}}$, then $\mathsf{BadHonest}$ has already occurred, and the probability being bounded is zero. Otherwise, fix the information

retained in $\mathcal{F}^{\circ}_{\mathbf{ct}-}$. The other table entries under $W^-$ exclude their $r_-$ assigned outputs as possible values of $Y^-$. Conditional on the retained table entries, every remaining candidate value of $Y^-$ allows the same number of permutation completions. The intervening computations remain valid and equally likely for every such choice. Hence the remaining candidate values are equally likely given $\mathcal{F}^{\circ}_{\mathbf{ct}-}$.

Consequently, conditional on $\mathcal{F}^{\circ}_{\mathbf{ct}-}$, $Y^-$ is uniform among the $2^s - r_-$ compatible outputs. Since $Y \mapsto X^- \boxplus Y$ is a bijection, the anchor $\mathsf{anch}_{\mathbf{ct}} = X^- \boxplus Y^-$ is likewise uniform over $2^s - r_-$ possible values. By these,

$$\Pr\big[\tau_{\mathrm{HB}} \geq \theta_{\mathbf{ct}} \ \wedge \ \mathsf{anch}_{\mathbf{ct}} = \widehat{a} \ \big|\ \mathcal{F}^{\circ}_{\mathbf{ct}-}\big] \leq \frac{1}{2^s - r_-} \leq \frac{1}{2^s - T - \widehat{B}}.$$

The conditioning above is valid for all five coordinate types. This proves Clause (i).

*Proof of Clause (ii).* We prove the bound for the joint event in Eq. (32).

*Step 1: Fix the adversarial view and identify the hidden assignment.* Fix a value of $\mathcal{V}^+_t$. This fixes the selected coordinate $(\mathsf{i}, q, d, v)$ and the target value $\widehat{x}$. Let

$$\mathsf{E} := \mathsf{G}_t \cap \big\{(\mathsf{i}, q, d, v) \in \mathcal{HS} \text{ at } \vartheta_t\big\} \cap \big\{X_v^{\mathsf{i},q,d} = \widehat{x}\big\}$$

denote the joint event in Eq. (32). For each possible value $x \in \{0,1\}^s$ of the hidden Winternitz value, define its "consumer key" by

$$W_{(\mathsf{i},q,d,v)}(x) := \mathsf{lpad}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| x\big).$$

Different values of $x$ give different "consumer keys". Initialization has already evaluated this Winternitz step. The "consumer record" at coordinate $(\mathsf{i}, q, d, v)$ has input $IV$ and stored response

$$Y^+ = \mathbf{E}_{W_{(\mathsf{i},q,d,v)}(X_v^{\mathsf{i},q,d})}(IV).$$

When $v > 0$, the preceding Winternitz step is the "producer": its record $(W^-; IV, Y^-)$ satisfies $X_v^{\mathsf{i},q,d} = IV \boxplus Y^-$. Its key $W^-$ is fixed by $X_{v-1}^{\mathsf{i},q,d}$ and the label $[v-1]_8$. In contrast, every candidate "consumer key" has the label $[v]_8$, so the "producer key" differs from all candidate "consumer keys". The designated "producer computations" are precisely the first evaluation of coordinate $(\mathsf{i}, q, d, v-1)$ and its ordinary signing-time recomputations. When $v = 0$, the hidden value is the independently sampled secret $X_0^{\mathsf{i},q,d}$, and the designated "consumer computations" are defined in the same way for coordinate $(\mathsf{i}, q, d, v)$.

*Step 2: Fix additional information without fixing the hidden value.* The adversarial view alone does not specify the honest records needed for the subsequent counting argument. We therefore refine the conditioning to a redacted execution history $\mathcal{R}_{t,(\mathsf{i},q,d,v)}$. For this, we start with the complete view $\mathcal{V}^+_t$ and additionally retain the other independent non-oracle random choices, the algorithmic order and roles of the queries before $\vartheta_t$, the signing progress, and their input-output assignments, subject to the following omissions:

- If $v = 0$, remove $X_0^{\mathsf{i},q,d}$ and its internal copies. If $v > 0$, remove the "producer response" $Y^-$, its endpoint $X_v^{\mathsf{i},q,d}$, and their internal copies, including the responses returned by ordinary signing-time lookups of this "producer record". Retain the "producer key" and input.

– Replace the "consumer key" by one common symbol at its initial honest evaluation and at every ordinary honest lookup belonging to the same Winternitz coordinate. Apply this replacement also to the corresponding table index, stored key copies, and lookup references. Retain the input $IV$ and the response $Y^+$.
– Do not additionally retain equality or freshness flags, record-sharing information, or bad-event bookkeeping that depends on these removed fields. In particular, do not retain a total-record counter or numerical clock annotations that would recover such information. The algorithmic query order and roles are retained, but additional proof-only bookkeeping is not.

Note that the omissions apply only to the additional internal information, and nothing in $\mathcal{V}_t^+$ is removed.

The *retained records*, i.e., ideal-cipher input-output pairs specified by the retained queries, form a partial permutation table for each key. The special "producer" and "consumer records" are not included in these tables unless they are also specified by a retained query outside their designated honest computations. This construction makes $\mathcal{R}_{t,(\mathrm{i},q,d,v)}$ a refinement of $\mathcal{V}_t^+$, while leaving the special assignments to be reconstructed for each candidate $x$.

*Step 3: Separate histories on which event* $\mathsf{E}$ *is impossible.* Fix a value $\mathsf{tr}$ of $\mathcal{R}_{t,(\mathrm{i},q,d,v)}$ having positive probability. Membership of the coordinate in $\mathcal{HS}$ is determined by the retained signing progress. If $(\mathrm{i},q,d,v) \notin \mathcal{HS}$, then $\mathsf{E}$ is impossible.

Suppose henceforth that $(\mathrm{i},q,d,v) \in \mathcal{HS}$ still holds. When $v > 0$, the predecessor coordinate $(\mathrm{i},q,d,v-1) \in \mathcal{HS}$ also holds. If a retained query outside the designated "producer computations" already specifies a record of the form $(W^-; IV, \star)$, then $\mathsf{E}$ is again impossible. Indeed, an adversarial access to this record causes $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ for the predecessor coordinate; sharing it with the first evaluation of another collected challenge causes clause (i) of $\mathsf{BadHonest}$. This also covers a "producer record" already present in the adversarial view. Therefore, when $v > 0$, we may restrict attention to *histories containing no retained record of the form* $(W^-; IV, \star)$.

*Step 4: Count candidates excluded by retained records.* A $(\mathrm{i},q,d,v)$ *candidate* is an $s$-bit value $x \in \{0,1\}^s$ for the hidden secret $X_v^{\mathrm{i},q,d}$. Such a candidate $x$ should be excluded, if setting $X_v^{\mathrm{i},q,d} = x$ would cause contradiction with $\mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}$. Concretely, such a contradiction occurs if at least one of the following holds:

(i) a retained record has key $W_{(\mathrm{i},q,d,v)}(x)$ and input $IV$;
(ii) a retained record has key $W_{(\mathrm{i},q,d,v)}(x)$, input different from $IV$, and output $Y^+$;
(iii) when $v > 0$, the unique producer response satisfying $IV \boxplus Y^- = x$ is already assigned to another input under $W^-$.

(i) there is a record of the form $(W_{(\mathrm{i},q,d,v)}(x); IV, \star)$ in $\mathsf{tr}$;
(ii) there is a record of the form $(W_{(\mathrm{i},q,d,v)}(x); X', Y^+)$ in $\mathsf{tr}$ such that $X' \neq IV$;
(iii) when $v > 0$, there is a record of the form $(W^-; X', x \boxminus IV)$ in $\mathsf{tr}$ such that $X' \neq IV$.

The second and third conditions contradict injectivity of the corresponding permutations $\mathbf{E}_{W_{(\mathrm{i},q,d,v)}(x)}$ and $\mathbf{E}_{W^-}$. Under the first condition, a stored response different from $Y^+$ is incompatible with the consumer assignment. If the stored response is $Y^+$, the consumer record has been accessed outside its designated honest computations. An adversarial access causes $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, whereas sharing this record with another collected challenge causes clause (i) of $\mathsf{BadHonest}$ at the later first evaluation. Consequently, for every excluded candidate,

$$\Pr\Big[\mathsf{E} \cap \{X_v^{\mathrm{i},q,d} = x\} \;\Big|\; \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\Big] = 0.$$

In other words, an excluded candidate may describe an execution on which a bad event has already occurred.

Let $h$ be the number of excluded candidates, and select one retained "witness" record for each exclusion. These "witnesses" can be chosen distinct: different consumer candidates have different keys; each producer output determines only one endpoint $x$; and the producer key differs from every consumer key. For a nonexcluded $x$, let $r_x$ be the number of retained records under $W_{(\mathrm{i},q,d,v)}(x)$. Their inputs differ from $IV$, and their outputs differ from $Y^+$. These record families are pairwise disjoint. They also do not contain any witness record, whose key is either the producer key or the consumer key of an excluded candidate. Hence

$$h + \sum_{x \text{ nonexcluded}} r_x \leq \text{number of retained records} \leq T + \widehat{B}. \tag{33}$$

*Step 5: Compute the joint probability of the history and a candidate value.* For a nonexcluded candidate $x$, conditioned only on the $r_x$ retained records under the key $W_{(\mathrm{i},q,d,v)}(x)$, the input $IV$ is unassigned and $Y^+$ is an unused output. This means

$$\Pr\Big[\mathbf{E}_{W_{(\mathrm{i},q,d,v)}(x)}(IV) = Y^+ \;\Big|\; \mathbf{E}_{W_{(\mathrm{i},q,d,v)}(x)}(Z_j) = Y_j \text{ for } j = 1, \ldots, r_x\Big] = \frac{1}{2^s - r_x}.$$

We next prove that the probabilities calculated above for generating the hidden value are the same for all nonexcluded candidates $x$. If $v > 0$, the producer key and input are fixed, and its input is unassigned in the retained table by Step 3. Each nonexcluded $x$ requires one available producer response. If there are $m$ retained assignments under $W^-$, appending that response multiplies the table probability by $1/(2^s - m)$, independently of $x$. If $v = 0$, setting the direct secret to $x$ instead has probability $2^{-s}$, again independently of $x$.

We next verify that every nonexcluded candidate $x$ is consistent with the fixed retained history $\mathsf{tr}$ (and thus can appear). For this, for each such $x$, we reconstruct a valid execution with $X_v^{\mathrm{i},q,d} = x$ using the retained random choices and query information. Before the production or first use of the selected value, replacing this value changes no other chain or tree computation. At the producer query, insert the response required by $x$; at the consumer query, use $W_{(\mathrm{i},q,d,v)}(x)$ and the fixed response $Y^+$. The next Winternitz value is always $IV \boxplus Y^+$, so all subsequent values on this chain remain unchanged. When signing recomputes the chain, the designated producer and consumer lookups return these same assigned responses; they impose no new random assignment. Since the selected coordinate remains hidden, signing does not output its value. The retained signing progress and public responses therefore remain unchanged. Other concrete queries, including the adversarial queries chosen from the unchanged adversarial view, retain their parameters and responses. The omissions in Step 2 ensure that no additional key copy, equality flag, or proof-only update distinguishes these reconstructions. Thus every nonexcluded candidate gives the same retained history, subject only to the two assignments just described. Some such reconstructions may violate $\mathsf{G}_t$; they are not discarded from this calculation.

For completeness, a fixed set of $m$ consistent assignments under one uniform permutation has probability

$$\frac{(2^s - m)!}{(2^s)!}.$$

Multiplying these quantities over the retained per-key tables gives a probability independent of the candidate. Appending the producer assignment, or sampling the direct

secret, contributes the common probability calculated above. Ordinary lookups contribute no further sampling probability. Only the consumer assignment contributes the candidate-dependent term $1/(2^s - r_x)$. Consequently, let $A_{\mathsf{tr}} \geq 0$ be the product of the common probabilities calculated above: the probability of the retained random choices and per-key table entries, multiplied by the probability of the required producer response when $v > 0$, or of sampling the direct secret when $v = 0$. This product depends on the fixed history $\mathsf{tr}$ but not on the nonexcluded candidate $x$. Then, for every nonexcluded candidate $x$, we have

$$\Pr\Big[X_v^{\mathrm{i},q,d} = x \ \wedge \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\Big] = \frac{A_{\mathsf{tr}}}{2^s - r_x}. \tag{34}$$

Let $C_0 := A_{\mathsf{tr}} / \Pr[\mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}]$, then Bayes' rule implies that

$$\Pr\Big[X_v^{\mathrm{i},q,d} = x \ \Big| \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\Big] = \frac{C_0}{2^s - r_x} \geq \frac{C_0}{2^s} \qquad \text{for every nonexcluded } x.$$

There are $2^s - h$ nonexcluded candidates, and their posterior probabilities sum to at most one. Therefore

$$\big(2^s - h\big) \cdot \frac{C_0}{2^s} \leq \sum_{x \text{ nonexcluded}} \Pr\Big[X_v^{\mathrm{i},q,d} = x \ \Big| \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\Big] \leq 1,$$

which means

$$C_0 \leq \frac{2^s}{2^s - h}.$$

*Step 6: Normalize and bound the specified guess.* Consider the target $\widehat{x}$. If $\widehat{x}$ is excluded, the probability of $\mathsf{E}$ is zero. Otherwise, $\mathsf{E}$ implies $X_v^{\mathrm{i},q,d} = \widehat{x}$, and hence

$$\begin{aligned}
\Pr\big[\mathsf{E} \ \big| \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\big] &\leq \Pr\Big[X_v^{\mathrm{i},q,d} = \widehat{x} \ \Big| \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\Big] \\
&\leq \frac{C_0}{2^s - r_{\widehat{x}}} \\
&\leq \frac{2^s}{(2^s - h)(2^s - r_{\widehat{x}})} \\
&\leq \frac{1}{2^s - h - r_{\widehat{x}}} \leq \frac{1}{2^s - T - \widehat{B}}.
\end{aligned}$$

Here $h + r_{\widehat{x}} \leq T + \widehat{B} < 2^s$ by Eq. (33) and assumption of Theorem 3. Combining this with the zero-probability histories in Step 3 proves, for every fixed history of positive probability,

$$\Pr\big[\mathsf{E} \ \big| \ \mathcal{R}_{t,(\mathrm{i},q,d,v)} = \mathsf{tr}\big] \leq \frac{1}{2^s - T - \widehat{B}}. \tag{35}$$

*Step 7: Return to the adversarial view.* By construction, the refined history retains the complete $\mathcal{V}_t^+$. The law of iterated expectations (Sect. C.1) therefore gives

$$\Pr[\mathsf{E} \mid \mathcal{V}_t^+] = \mathbb{E}\big[\Pr\big[\mathsf{E} \mid \mathcal{R}_{t,(\mathrm{i},q,d,v)}\big] \ \big| \ \mathcal{V}_t^+\big] \leq \frac{1}{2^s - T - \widehat{B}},$$

which is Eq. (32). □

**Completing the analysis of BadHonest.** Throughout this subsection, fix an arbitrary identifier vector $\mathbf{I} = (I^1, \ldots, I^u)$, and write $\Pr_{\mathbf{I}}$ for probability in the resulting conditional experiment. Use $c = 1, \ldots, \gamma_{\mathrm{ch}}$ for the positions in the chronological list $\mathbf{ct}_1, \ldots, \mathbf{ct}_{\gamma_{\mathrm{ch}}}$ defined above, where $\gamma_{\mathrm{ch}} \le \widehat{\Gamma}$. For notational convenience, extend the index range through $\widehat{\Gamma}$ with dummy indices whose associated events are empty.

Fix an actual index $c$, and let $\mathcal{Q}_{\mathbf{ct}_c-}$ be the set of records present immediately before $\theta_{\mathbf{ct}_c}$. The "redacted" history $\mathcal{F}^{\circ}_{\mathbf{ct}_c-}$ fixes the number of records in $\mathcal{Q}_{\mathbf{ct}_c-}$ and the key of each such record.

*Probability That a First Honest Ideal-Cipher Query Is Non-Fresh.* Let $\mathsf{E}_c$ be the event that BadHonest occurs for the first time at the first honest ideal-cipher query of the challenge call at $\mathbf{ct}_c$, and that this first occurrence is caused by clause (i). If $\mathsf{E}_c$ occurs, then $\tau_{\mathrm{HB}} \ge \theta_{\mathbf{ct}_c}$ and the first-block key matches an earlier record key.

Since the map $a \mapsto K_{\mathbf{ct}_c}(a)$ is injective, each earlier record key corresponds to at most one candidate "anchor" value. For each such candidate, Eq. (31) bounds the conditional probability that the anchor equals that candidate and no earlier BadHonest has occurred. A conditional union bound therefore gives

$$\Pr_{\mathbf{I}}\big[\mathsf{E}_c \mid \mathcal{F}^{\circ}_{\mathbf{ct}_c-}\big] \le \frac{|\mathcal{Q}_{\mathbf{ct}_c-}|}{2^s - T - \widehat{B}} \le \frac{T + \widehat{B}}{2^s - T - \widehat{B}}. \tag{36}$$

Taking expectations and a union bound over the at most $\widehat{\Gamma}$ challenge calls yields

$$\begin{aligned}&\Pr_{\mathbf{I}}\big[\text{BadHonest first occurs via clause (i) at a first honest ideal-cipher query}\big]\\ &\le \frac{\widehat{\Gamma}(T + \widehat{B})}{2^s - T - \widehat{B}}.\end{aligned} \tag{37}$$

*Non-first honest ideal-cipher queries must be fresh.* Suppose, toward a contradiction, that the first occurrence of BadHonest is caused by clause (i) at a non-first honest ideal-cipher query on a challenge path. Let $(W_j; S_{j-1}, Y_j)$ and $(W_{j+1}; S_j, Y_{j+1})$, $S_j = S_{j-1} \boxplus Y_j$, be the records corresponding to the preceding and current honest ideal-cipher queries

The preceding honest ideal-cipher query is fresh; otherwise that query would already have caused clause (i). If the two underlying records are the same, equality of their inputs gives $S_j = S_{j-1}$, so the record accessed by the preceding query has endpoint equal to its source and already causes clause (ii). If they are distinct, the record accessed by the current query is already present when that query is made. Since the two honest ideal-cipher queries are consecutive within one atomic honest evaluation, that distinct record was already present when the preceding query materialized its record. The endpoint $S_j$ produced by the fresh preceding honest ideal-cipher query is therefore the source of an earlier distinct record, so clause (iii) already occurs. Both cases contradict the assumption that the current honest ideal-cipher query is the first to cause BadHonest. Consequently, clause (i) cannot cause the first bad occurrence at a non-first honest ideal-cipher query; before the first occurrence of BadHonest, every such honest ideal-cipher query is fresh.

*Endpoint conditions of fresh honest ideal-cipher queries.* Consider a fresh honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ on the branch on which it is the first query to cause clause (ii) or clause (iii), and stop immediately before $Y$ is sampled. Its key and source, as well as all earlier records, are fixed. The response is uniform among at least $2^s -$

$(T+\widehat{B})$ unused outputs of $\mathbf{E}_W$. Clause (ii) prescribes at most two endpoint targets, $IV$ and $X$. Clause (iii) prescribes the source or endpoint of one of at most $T+\widehat{B}$ earlier records and hence at most $2(T+\widehat{B})$ further targets. The conditional probability that this fresh honest ideal-cipher query first causes either clause is therefore at most $\frac{2+2(T+\widehat{B})}{2^s-(T+\widehat{B})}$. There are at most $\widehat{B}$ honest ideal-cipher queries underlying all challenge hash-calls. Combining this charge with Eq. (37) gives

$$\Pr_{\mathbf{I}}\big[\mathsf{BadHonest}\big] \le \frac{\widehat{\Gamma}(T+\widehat{B})}{2^s-T-\widehat{B}} + \frac{\widehat{B}\big(2+2(T+\widehat{B})\big)}{2^s-(T+\widehat{B})} \le \frac{\widehat{\Gamma}(T+\widehat{B})+2\widehat{B}(1+T+\widehat{B})}{2^s-T-\widehat{B}}.$$

The same estimate holds for every fixed identifier vector. Averaging over all identifier vectors gives

$$\Pr\big[\mathsf{BadHonest}\big] = \sum_{\mathbf{I}} \Pr[\mathbf{I}]\ \Pr_{\mathbf{I}}[\mathsf{BadHonest}] \le \frac{\widehat{\Gamma}(T+\widehat{B})+2\widehat{B}(1+T+\widehat{B})}{2^s-T-\widehat{B}}. \tag{38}$$

### C.4 Probability of Event $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ in $\mathbf{Exp}^{\textbf{m-unforge}}_{\mathbf{E},\mathbf{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$

As established in Sect. C.2, for every coordinate 4-tuple $(\mathrm{i},q,d,v) \in \mathcal{HS}$, the corresponding Winternitz-chain challenge $X_{v+1}^{\mathrm{i},q,d} \leftarrow \mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(I^{\mathrm{i}}\|[q]_{32}\|[d]_{16}\|[v]_8\|X_v^{\mathrm{i},q,d})$ can be evaluated using a single honest ideal-cipher query $Y_v^{\mathrm{i},q,d} \leftarrow \mathbf{E}_{W_v^{\mathrm{i},q,d}}(IV)$ via $X_{v+1}^{\mathrm{i},q,d} \leftarrow IV \boxplus Y_v^{\mathrm{i},q,d}$, where $W_v^{\mathrm{i},q,d} = \mathsf{lpad}(I^{\mathrm{i}}\|[q]_{32}\|[d]_{16}\|[v]_8\|X_v^{\mathrm{i},q,d})$. By this, in the setting of $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, the event $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ means that there exists a coordinate 4-tuple $(\mathrm{i},q,d,v)$ that is still in the set $\mathcal{HS}$, but the corresponding record $(W_v^{\mathrm{i},q,d}; IV, Y_v^{\mathrm{i},q,d})$ has been in the *adversarial ideal-cipher history*.

**Lemma 3.** *It holds that*

$$\Pr\left[\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}} \wedge \neg\mathsf{BadHonest}\right] \le \frac{\widehat{\Gamma}T}{2^s-T-\widehat{B}}. \tag{39}$$

*Proof.* Fix an arbitrary identifier vector. Under the augmented-history convention of Sect. B, certain records materialized during initialization are placed directly in the adversarial history, while signing queries may later transfer additional records into that history. Neither mechanism can first place the record of a still-hidden coordinate in the adversarial history outside $\mathsf{BadHonest}$. If such a record coincides with the record of another coordinate that remains hidden, the two honest challenge occurrences share one record, so the later occurrence already causes clause (i) of $\mathsf{BadHonest}$. Moreover, a signing update removes every disclosed coordinate from $\mathcal{HS}$ before transferring the corresponding records. Thus it remains to consider adaptive adversarial ideal-cipher queries.

Condition immediately before such a query is processed, after its key $W$ has been fixed. If $W$ is not the suffix padding of a correctly formatted Winternitz input, it matches no hidden coordinate. Otherwise injectivity of $\mathsf{lpad}$ and the fixed-length encoding determine a unique tuple $(II,q,d,v,\widehat{x})$ satisfying

$$W = \mathsf{lpad}\big(II\|[q]_{32}\|[d]_{16}\|[v]_8\|\widehat{x}\big).$$

Among all challenge occurrences there are at most $\widehat{\Gamma}$ coordinates having these formatted fields. For each such coordinate $(\mathrm{i},q,d,v)$, equality with its hidden key requires

$X_v^{\mathrm{i},q,d} = \widehat{x}$. Equation (32) bounds the conditional subprobability of this equality, together with the absence of an earlier $\mathsf{BadHonest}$ or $\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$, by $1/(2^s - T - \widehat{B})$. A union bound over these at most $\widehat{\Gamma}$ coordinates and then over the at most $T$ adversarial ideal-cipher queries proves the claim. The argument applies to both query directions; ignoring the additional requirement that the forward input, or the inverse response, equal $IV$ only enlarges the event. □

### C.5 Probability of Event $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ in $\mathrm{Exp}^{\textbf{m-unforge}}_{\mathbf{E},\mathbf{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},\boldsymbol{m},\boldsymbol{u}}$

**Additional Bad Events.** We introduce two additional bad events.

- $\mathsf{BadFwd}$ occurs if a fresh adversarial ideal-cipher query in the forward direction $Y \leftarrow \mathbf{E}_W(X)$ satisfies at least one of the following:
  (i) $X \boxplus Y = IV$;
  (ii) there is a record $(W'; X', Y')$ in $\mathcal{Q}$, already accessed by an honest ideal-cipher query for a challenge, such that $X \boxplus Y \in \{X', X' \boxplus Y'\}$.
- $\mathsf{BadBwd}$ occurs if a fresh adversarial ideal-cipher query in the backward direction $X \leftarrow \mathbf{E}_W^{-1}(Y)$ satisfies at least one of the following:
  (i) $X = IV$ or $X \boxplus Y = IV$;
  (ii) there is a record $(W'; X', Y')$ in $\mathcal{Q}$, already accessed by an honest ideal-cipher query for a challenge, such that

$$X \in \{X', X' \boxplus Y'\} \quad \text{or} \quad X \boxplus Y \in \{X', X' \boxplus Y'\}.$$

For a fresh adversarial ideal-cipher query in the forward direction, condition after $W, X$ have been fixed and before $Y$ is sampled. On the nontrivial branch, $Y$ is uniform over at least $2^s - (T + \widehat{B})$ unused outputs. Clause (i) specifies one answer and clause (ii) specifies at most two answers for each of at most $\widehat{B}$ records accessed by honest ideal-cipher queries for challenges. Therefore

$$\Pr[\mathsf{BadFwd}] \leq \frac{T + 2\widehat{B}T}{2^s - T - \widehat{B}}. \tag{40}$$

For a fresh adversarial ideal-cipher query in the backward direction, condition after $W, Y$ have been fixed and before $X$ is sampled. Clause (i) specifies at most two answers, while clause (ii) specifies at most four answers per record accessed by an honest ideal-cipher query for a challenge. Hence

$$\Pr[\mathsf{BadBwd}] \leq \frac{2T + 4\widehat{B}T}{2^s - T - \widehat{B}}. \tag{41}$$

**Probability of $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ for $\mathbf{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$.** The following deterministic argument uses injective length-binding suffix padding, the augmented-transcript convention of Sect. B, and the atomicity of every honest challenge evaluation with respect to adversarial ideal-cipher queries. Chronology refers to the first-materialization time of a record, not to a later occurrence of that record on a hashing path.

**Lemma 4.** *It holds that*

$$\Pr[\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}] \leq \Pr[\mathsf{BadHonest}] + \Pr[\mathsf{BadFwd}] + \Pr[\mathsf{BadBwd}]. \tag{42}$$

*Proof.* Suppose that $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$ occurs, and let the following two paths be the earliest paths causing $\mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$:

$$\begin{aligned}\left(IV \xrightarrow{\mathsf{prefix}_i\|M_i} Z\right): &\quad S_0 = IV \xrightarrow{W_1} S_1 \xrightarrow{W_2} \cdots \xrightarrow{W_\ell} S_\ell = Z,\\ \left(IV \xrightarrow{\mathsf{prefix}_i\|M'} Z\right): &\quad \widehat{S}_0 = IV \xrightarrow{\widehat{W}_1} \widehat{S}_1 \xrightarrow{\widehat{W}_2} \cdots \xrightarrow{\widehat{W}_{\ell^\star}} \widehat{S}_{\ell^\star} = Z.\end{aligned}$$

In the above, every displayed arrow is a labeled edge in the transcript graph and corresponds to a record in the transcript—this holds because the adversary $\mathscr{A}$ has been normalized.

Assume toward a contradiction that none of $\mathsf{BadHonest}$, $\mathsf{BadFwd}$, and $\mathsf{BadBwd}$ occurs. This means the internal states $S_0, S_1, \ldots, S_\ell$ are pairwise distinct: $\neg\mathsf{BadHonest}$ (i) ensures that every honest ideal-cipher query on the first evaluation of this selected challenge path is fresh, and $\neg\mathsf{BadHonest}$ (ii) and (iii) then ensures the distinctness.

Consider deleting the maximal common initial sequence of labeled directed-edge occurrences from $\left(IV \xrightarrow{\mathsf{prefix}_i\|M_i} Z\right)$ and $\left(IV \xrightarrow{\mathsf{prefix}_i\|M'} Z\right)$. Since the padding $\mathsf{Ipad}$ is injective and $\mathsf{prefix}_i\|M_i \neq \mathsf{prefix}_i\|M'$, their padded block strings are different. This means we can obtain the following reduced path for some $j_1$, $1 \le j_1 \le \min\{\ell+1, \ell^\star\}$:

$$S_{j_1-1} \xrightarrow{\widehat{W}_{j_1}} \cdots \xrightarrow{\widehat{W}_{\ell^\star}} \widehat{S}_{\ell^\star} = Z. \tag{43}$$

Otherwise, $\left(IV \xrightarrow{\mathsf{prefix}_i\|M'} Z\right)$ would be a strict prefix of $\left(IV \xrightarrow{\mathsf{prefix}_i\|M_i} Z\right)$, so $Z$ would occur both at an internal state and at $S_\ell$, contrary to pairwise distinctness.

Considering $\widehat{S}_{\ell^\star} = Z = S_\ell$, let $j_2^\star$ be the smallest index in $\{j_1, ..., \ell^\star\}$ such that $\widehat{S}_{j_2^\star} \in \{S_0, S_1, ..., S_\ell\}$. Since $S_0, \ldots, S_\ell$ are pairwise distinct, there is a unique $j_2 \in \{0, \ldots, \ell\}$ such that $\widehat{S}_{j_2^\star} = S_{j_2}$. Let $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$ be the record in the reduced path

$$S_{j_1-1} \xrightarrow{\widehat{W}_{j_1}} \cdots \xrightarrow{\widehat{W}_{j_2^\star}} \widehat{S}_{j_2^\star} \tag{44}$$

with $\widehat{S}_{j_2^\star-1} \boxplus \widehat{Y}_{j_2^\star} = \widehat{S}_{j_2^\star}$. It cannot be $\widehat{S}_{j_2^\star} = S_0 = IV$: otherwise, the materialization of the record $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$ indicates $\mathsf{BadHonest}$ (ii) (when it is first materialized by a fresh honest ideal-cipher query), $\mathsf{BadFwd}$ (i) (when it is materialized by a fresh adversarial ideal-cipher query in the forward direction) or $\mathsf{BadBwd}$ (i) (when it is materialized by a fresh adversarial ideal-cipher query in the backward direction). This means $j_2 \in \{1, ..., \ell\}$ and there exists a record $(W_{j_2}; S_{j_2-1}, Y_{j_2})$ in $\left(IV \xrightarrow{\mathsf{prefix}_i\|M_i} Z\right)$ with $S_{j_2-1} \boxplus Y_{j_2} = S_{j_2}$.

We first show that $(W_{j_2}; S_{j_2-1}, Y_{j_2})$ and $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$ are distinct records. Suppose, toward a contradiction, that they are the same. If $j_2^\star > j_1$, then, immediately before traversing $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$ on $\left(IV \xrightarrow{\mathsf{prefix}_i\|M'} Z\right)$, we have

$$\widehat{S}_{j_2^\star-1} = S_{j_2-1} \in \{S_0, \ldots, S_\ell\}.$$

This means the aforementioned index $j_2^\star$ was not the smallest, a contradiction.

On the other hand, if $j_2^\star = j_1$, then

$$\widehat{S}_{j_1-1} = S_{j_1-1} = S_{j_2-1}.$$

If $j_1 = \ell + 1$, this equality is impossible because $j_2 \le \ell$ and the challenge states $S_0, ..., S_\ell$ are pairwise distinct. Otherwise, pairwise distinctness gives $j_2 = j_1$. Since

$(W_{j_2}; S_{j_2-1}, Y_{j_2}) = (\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$, the two paths then traverse the same labeled record immediately after their deleted common initial sequence. That common initial sequence could therefore be extended by one more occurrence, contradicting its maximality. Hence, it has to be $(W_{j_2}; S_{j_2-1}, Y_{j_2}) \neq (\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$.

At this stage, if $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$ is materialized after $(W_{j_2}; S_{j_2-1}, Y_{j_2})$, then it indicates $\mathsf{BadHonest}$ (iii), $\mathsf{BadFwd}$ (ii), or $\mathsf{BadBwd}$ (ii) (since clause (i) of $\mathsf{BadHonest}$ does not occur, the record $(W_{j_2}; S_{j_2-1}, Y_{j_2})$ was first materialized by the $j_2$-th honest ideal-cipher query in $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_i \| M_i)$). If $(W_{j_2}; S_{j_2-1}, Y_{j_2})$ is materialized after $(\widehat{W}_{j_2^\star}; \widehat{S}_{j_2^\star-1}, \widehat{Y}_{j_2^\star})$, then:

- If $(W_{j_2}; S_{j_2-1}, Y_{j_2})$ is not first materialized by the $j_2$-th honest ideal-cipher query in $\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_i \| M_i)$, then $\mathsf{BadHonest}$ (i) occurs;
- Otherwise, $S_{j_2-1} \boxplus Y_{j_2} = \widehat{S}_{j_2^\star-1} \boxplus \widehat{Y}_{j_2^\star}$ indicates $\mathsf{BadHonest}$ (iii).

These complete the proof. □

### C.6 Concluding on $\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}}$

Lemmas 1 and 4 give the event inclusion

$$\begin{aligned}
&\left\{\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\right\} \\
&\subseteq \mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}} \vee \mathsf{BadFPMSpr}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}} \\
&\subseteq \mathsf{BadHonest} \vee (\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}} \wedge \neg\mathsf{BadHonest}) \vee \mathsf{BadFwd} \vee \mathsf{BadBwd}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\Pr\left[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\right] \\
&\quad \leq \Pr[\mathsf{BadHonest}] \\
&\qquad + \Pr\left[\mathsf{BadGuess}_{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}} \wedge \neg\mathsf{BadHonest}\right] + \Pr[\mathsf{BadFwd}] + \Pr[\mathsf{BadBwd}].
\end{aligned}$$

Substituting Eqs. (38), (39), (40), and (41) proves Eq. (24):

$$\begin{aligned}
\Pr\left[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{sMD}^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\right] &\leq \frac{2\widehat{\Gamma}T + \widehat{\Gamma}\widehat{B} + 8\widehat{B}T + 2\widehat{B}^2 + 2\widehat{B} + 3T}{2^s - T - \widehat{B}} \\
&\leq \frac{10\widehat{B}T + 3\widehat{B}^2 + 2\widehat{B} + 3T}{2^s - T - \widehat{B}} \\
&\leq \frac{26\widehat{B}T + 12\widehat{B}^2}{2^s}. \qquad (45)
\end{aligned}$$

The second inequality uses $\widehat{\Gamma} \leq \widehat{B}$. For the last one, the theorem assumption gives $2^s - T - \widehat{B} \geq 2^{s-1}$, while $\widehat{B} \geq 1$ gives $10\widehat{B}T + 3\widehat{B}^2 + 2\widehat{B} + 3T \leq 13\widehat{B}T + 6\widehat{B}^2$.

## D Detailed Analysis for $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$

This section applies the framework in Sect. B to prove security for $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$. For convenience, we recall Theorem 2 below.

**Theorem 4 (Multi-user unforgeability of $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$).** *During the experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$, *assume that: (i) the adversary* $\mathscr{A}$ *makes at most* $T$ *adversarial ideal-cipher queries to the ideal-cipher* $\mathbf{E}$ *(including the auxiliary queries for our proof); (ii)* $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ *internally makes at most* $\widehat{\Gamma}$ *calls to* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}$, *which results in at most* $\widehat{B}$ *Davies-Meyer calls; (iii) for every hash-call* $\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(U)$, *the padded input* $\mathsf{lpad}(U)$ *has at most* $\ell_{\max}$ $r$*-bit blocks. Put* $Q := T + \widehat{B}$, *and further assume* $Q \le 2^{n-1}$. *Then, for every integer threshold* $\mu \in \{1, ..., \widehat{\Gamma}\}$, *the construction* $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ *has*

$$
\begin{aligned}
&\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u} \Rightarrow 1\big] \\
&\le \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{8\mu T + 2\mu\widehat{B} + 4\widehat{B}\widehat{\Gamma}}{2^n} + \frac{16n^2T + 22\widehat{B}T + 8\widehat{B}^2}{2^s} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n},\ \frac{2\mu(T+\widehat{B})^3}{2^{n+s}}\right\} + \min\left\{\frac{2\widehat{\Gamma}T}{2^n},\ \frac{2\mu T(T+\widehat{B})^2}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}T}{2^n},\ \frac{1}{2^n} + \frac{(128+96n)\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n},\ \frac{1}{2^n} + \frac{12\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2^{2s+n}} + \frac{4(n+1)\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\},
\end{aligned}
\tag{46}
$$

To prove Theorem 4, we first elaborate on notations and useful properties of the challenge calls in Sect. D.1 and D.1 respectively. Then, we: (i) introduce and analyze some helper bad events in Sect. D.3; (ii) establish some helpful combinatorial properties of the ideal-cipher query structures, including number of certain paths in Sect. D.4 and collision-degree in D.5; (iii) introduce the event $\mathsf{BadHonest}$ in Sect. D.6; (iv) bound the probabilities of the event $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ in Sect. D.7; (v) bound the probabilities of the events $\mathsf{PreChSpr}$ and $\mathsf{PostChSpr}$, which represent two flavors of $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$, in Sect. D.8 and D.9 respectively, and combine them to yield probability bound for $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$. We finally complete the proof in Sect. D.11.

## D.1 Additional Notations

**Induced Transcript Graphs.** We use the same ideal-cipher-record and transcript convention as in Sect. C.3.

Let $\Omega$ denote the underlying probability space of $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$. An outcome $\omega \in \Omega$ specifies a complete realization of the ideal-cipher together with all other random coins used in the experiment. Once $\omega$ is fixed, the entire execution trace, including all adaptive ideal-cipher queries, is determined.

Fix $\omega \in \Omega$, and consider the corresponding execution $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}(\omega)$: all objects defined below are with respect to this fixed execution, and we omit their dependence on $\omega$ from the notation. Let $Q^\star(\omega)$ be the number of such fresh ideal-cipher queries in $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}(\omega)$, where $Q^\star(\omega) \le Q$. We henceforth write $Q^\star$ for $Q^\star(\omega)$.

For $i = 1, ..., Q^\star$, let $\mathcal{Q}_i = \big((W_1; X_1, Y_1), ..., (W_i; X_i, Y_i)\big)$ be the sub-transcript consisting of the first $i$ records, and let $\mathcal{Q}_0 = \emptyset$. For $i \in \{0, ..., Q^\star\}$, we define a directed labeled multigraph $\mathcal{G}_i = (\mathcal{V}_i, \mathcal{E}_i)$ over the $i$ th sub-transcript $\mathcal{Q}_i$, where $\mathcal{V}_i = \{0,1\}^s$ and

$$\mathcal{E}_i = \begin{cases} \emptyset & \text{if } i = 0 \\ \Big\{\big(X_j \xrightarrow{W_j} X_j \boxplus Y_j\big) : 1 \le j \le i\Big\} & \text{if } 1 \le i \le Q^\star \end{cases}$$

Moreover, let $\mathsf{Hist}_{i-}$ denote the complete execution history immediately before the $i$th fresh ideal-cipher query is made, and let $\mathsf{Hist}_{\mathrm{end}}$ denote the complete history at the end of the experiment.

For notational convenience, for $Q^\star < i \le Q$, we set

$$\mathcal{Q}_i := \mathcal{Q}_{Q^\star}, \qquad \mathcal{G}_i := \mathcal{G}_{Q^\star}, \qquad \text{and} \qquad \mathsf{Hist}_{i-} := \mathsf{Hist}_{\mathrm{end}}.$$

Thus $\mathcal{G}_Q = \mathcal{G}_{Q^\star}$ in every execution. In particular, at every dummy index $i > Q^\star$, both $\mathcal{G}_{i-1}$ and the dummy increment introduced below are $\mathsf{Hist}_{i-}$-measurable.

In the adaptive-exposure arguments below, notation of the form $\mathcal{F}_{t-}$ likewise denotes the complete execution history immediately before the fresh coordinate in slot $t$ is sampled. Thus, all ideal-cipher query parameters and all previously exposed records are $\mathcal{F}_{t-}$-measurable, whereas the current fresh coordinate is not yet exposed. Conditional probability bounds established for every fixed history can therefore be averaged using Eq. (28). Likewise, whenever a random variable $Y$ is determined by the pre-exposure history, it may be taken outside the inner conditional expectation according to Eq. (26). These applications require no independence between the current exposure and the preceding transcript.

Consider the $i$ th graph $\mathcal{G}_i$. Adjacent edges in $\mathcal{G}_i$ form *paths*. Concretely, a sequence of $\ell$ records

$$(W_{i_1}; X_{i_1}, Y_{i_1}),\ (W_{i_2}; X_{i_2}, Y_{i_2}), ..., (W_{i_\ell}; X_{i_\ell}, Y_{i_\ell})$$

satisfying $X_{i_{j+1}} = X_{i_j} \boxplus Y_{i_j}$ for all $j = 1, ..., \ell - 1$ form a *length-$\ell$ path* from the vertex $X_{i_1}$ to the vertex $X_{i_\ell} \boxplus Y_{i_\ell}$. Note that the indices $i_1, ..., i_\ell$ need not be distinct. We denote this path by $\big(X_{i_1} \xrightarrow{\mathbf{W}} X_{i_\ell} \boxplus Y_{i_\ell}\big)$, where $\mathbf{W} = W_{i_1} \| ... \| W_{i_\ell}$.

For every $S \in \{0,1\}^s$, denote by $d^-(S)$ its indegree in $\mathcal{G}_i$, and define the *excess indegree of* $\mathcal{G}_i$ as

$$\kappa(\mathcal{G}_i) = \sum_{S \in \{0,1\}^s} \max\big\{d^-_{\mathcal{G}_i}(S) - 1, 0\big\}. \tag{47}$$

### D.2 Properties of the Challenge Calls, and Event $\mathsf{BadID}_\mu$

We use the five challenge types and the time-dependent set $\mathcal{HS}$ exactly as defined in Sect. B. The general coordinate–prefix mapping from Sect. C.2 continues to apply. We define and analyze the identifier-multiplicity event $\mathsf{BadID}_\mu$ below.

As mentioned in Sect. 2.2, while we are using a general parameter tuple $(\iota, n, s, r)$, for the "block layout" of challenge hash-calls we simply follow the LMS_SHA256_M24_H$h$:

– For each coordinate 4-tuple $(\mathfrak{i}, q, d, v)$, define

$$W_v^{\mathfrak{i},q,d} := \mathsf{Ipad}\big(I^{\mathfrak{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X_v^{\mathfrak{i},q,d}\big).$$

The unpadded input has length $128+32+16+8+192 = 376$ bits. Since $376+65 = 441 \le 512$, it follows that $|W_v^{\mathfrak{i},q,d}| = r = 512$, and the challenge call is evaluated via a single ideal-cipher call $X_{v+1}^{\mathfrak{i},q,d} \leftarrow \mathsf{left}_n\big(\mathbf{E}_{W_v^{\mathfrak{i},q,d}}(IV) \oplus IV\big)$.

– For each coordinate triple $(\mathsf{i}, \rho, 8282)$, define

$$W^{\mathsf{i},\rho}_{8282} := \mathsf{lpad}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8282]_{16} \| pk^{\mathsf{i}}_{\rho - 2^{h^{\mathsf{i}}}}\big).$$

The unpadded input has length $128 + 32 + 16 + 192 = 368$ bits, and the 8282 challenge is evaluated via a single ideal-cipher-call $T[\rho]^{\mathsf{i}} \leftarrow \mathsf{left}_n\big(\mathbf{E}_{W^{\mathsf{i},\rho}_{8282}}(IV) \oplus IV\big)$.

– For each coordinate triple $(\mathsf{i}, \rho, 8383)$, the unpadded input has length $176 + 2 \cdot 192 = 560$ bits. Its padded block length is $\lceil (560 + 65)/512 \rceil = 2$. The first padded block is

$$\overline{U}^{\mathsf{i},\rho}_{8383}[1] := I^{\mathsf{i}} \| [\rho]_{32} \| [8383]_{16} \| T[2\rho]^{\mathsf{i}} \| \mathsf{left}_{144}\big(T[2\rho+1]^{\mathsf{i}}\big),$$

and the first ideal-cipher call is $\mathbf{E}_{\overline{U}^{\mathsf{i},\rho}_{8383}[1]}(IV)$.

– For each coordinate triple $(\mathsf{i}, q, 8080)$, the unpadded input has length $176 + 192p^{\mathsf{i}}$ bits. This is the longest key-generation hash evaluation, and its first padded block is

$$\overline{U}^{\mathsf{i},q}_{8080}[1] := I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| X^{\mathsf{i},q,0}_{e^{\mathsf{i}}} \| \mathsf{left}_{144}\big(X^{\mathsf{i},q,1}_{e^{\mathsf{i}}}\big),$$

and the first ideal-cipher call is $\mathbf{E}_{\overline{U}^{\mathsf{i},q}_{8080}[1]}(IV)$.

Distinct user indices $\mathsf{i}, \mathsf{i}'$ may give rise to identical prefix values when $I^{\mathsf{i}} = I^{\mathsf{i}'}$. Thus, let $\mathsf{BadID}_\mu$ occur if some identifier value is assigned to more than $\mu$ users, i.e.,

$$\max_{II \in \{0,1\}^\iota} \big|\{\mathsf{i} \in \{1, \ldots, u\} : I^{\mathsf{i}} = II\}\big| > \mu. \tag{48}$$

**Lemma 5 (LMS prefix multiplicity).** *In the experiment* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\mathrm{LMS}^{\mathsf{hP}},\mathscr{A},m,u}$*, conditioned on* $\neg\mathsf{BadID}_\mu$*, every concrete prefix value occurs at most* $\mu$ *times. Moreover,*

$$\Pr[\mathsf{BadID}_\mu] \le \frac{u^{\mu+1}}{(\mu+1)!2^{\iota\mu}}. \tag{49}$$

*Proof.* If $\mathsf{prefix}_{\mathbf{ct}} = \mathsf{prefix}_{\mathbf{ct}'}$ for two distinct coordinate tuples, their user indices have the same identifier. Therefore, on $\neg\mathsf{BadID}_\mu$, every concrete prefix value has multiplicity at most $\mu$. Since $I^1, \ldots, I^u$ are sampled independently and uniformly from $\{0,1\}^\iota$, a union bound over sets of $\mu + 1$ users gives

$$\Pr[\mathsf{BadID}_\mu] \le \binom{u}{\mu+1} \frac{1}{2^{\iota\mu}} \le \frac{u^{\mu+1}}{(\mu+1)!2^{\iota\mu}},$$

as claimed. □

### D.3 Additional Bad Events

**Events BadFwd and BadBwd on adversarial ideal-cipher queries.** We introduce two additional bad events as follows.

– $\mathsf{BadFwd}$: During $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}^{\mathsf{DM}^{\mathbf{E}}}_n},\mathscr{A},m,u}$, there occurs a fresh forward adversarial ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ such that either of the following is fulfilled:
- $X \boxplus Y = IV$;
- there exists a record $(W'; X', Y')$ in $\mathcal{Q}$accessed by an honest ideal-cipher query for a challenge such that $X \boxplus Y = X'$ or $X \boxplus Y = X' \boxplus Y'$.

– BadBwd: During $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$, there occurs a fresh backward adversarial ideal-cipher query $X \leftarrow \mathbf{E}_W^{-1}(Y)$ such that either of the following is fulfilled:
  - $X = IV$ or $X \boxplus Y = IV$;
  - there exists a record $(W'; X', Y')$ in $\mathcal{Q}$accessed by an honest ideal-cipher query for a challenge such that $X \in \{X', X' \boxplus Y'\}$ or $X \boxplus Y \in \{X', X' \boxplus Y'\}$.

For each fresh forward adversarial ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$, the answer $Y$ is uniform over at least $2^s - T - \widehat{B}$ possibilities. Hence (i) the probability that its Davies-Meyer endpoint satisfies $X \boxplus Y = IV$ is at most $1/(2^s - T - \widehat{B})$; and (ii) the probability that $X \boxplus Y$ equals $X'$ or $X' \boxplus Y'$ for $(W'; X', Y')$ accessed by one of the at most $\widehat{B}$ honest ideal-cipher queries is at most $2\widehat{B}/(2^s - T - \widehat{B})$. Summing over the at most $T$ adversarial ideal-cipher queries yields

$$\Pr\big[\mathsf{BadFwd}\big] \le \frac{T + 2\widehat{B}T}{2^s - T - \widehat{B}}. \tag{50}$$

Similarly by symmetry, for each fresh backward adversarial ideal-cipher query $X \leftarrow \mathbf{E}_W^{-1}(Y)$, the answer $X$ is uniform in at least $2^s - T - \widehat{B}$ possibilities, and it can be seen

$$\Pr\big[\mathsf{BadBwd}\big] \le \frac{2T + 4\widehat{B}T}{2^s - T - \widehat{B}}. \tag{51}$$

**Event BadHonest.** We introduce a bad event BadHonest. It occurs if, during the first evaluation of a challenge hash-call collected as in Sect. B in $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},m,u}$, there exists an underlying honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ for which at least one of the following conditions holds:

(i) Before $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ makes the honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$, the record $(W; X, Y)$ has been in $\mathcal{Q}$; equivalently, the query is an ordinary lookup;
(ii) The query $Y \leftarrow \mathbf{E}_W(X)$ has $X \boxplus Y = IV$ or $X \boxplus Y = X$;
(iii) There has been another distinct record $(W'; X', Y')$ in $\mathcal{Q}$ with $X \boxplus Y = X'$ or $X \boxplus Y = X' \boxplus Y'$.

The analysis of BadHonest is a bit complicated, and is deferred to Sect. D.6. For convenience, define

$$\mathsf{Bad} := \mathsf{BadID}_\mu \vee \mathsf{BadHonest} \vee \mathsf{BadFwd} \vee \mathsf{BadBwd}. \tag{52}$$

**Second preimage due to challenge calls.** Let $\gamma_{\mathrm{ch}} \le \widehat{\Gamma}$ be the number of challenge hash-calls that actually occur in the fixed execution, list them in chronological order, and let $\mathcal{T}_{\mathrm{ch}}$ be the set of their coordinate tuples, listed in the same order as $\mathbf{ct}_1, \ldots, \mathbf{ct}_{\gamma_{\mathrm{ch}}}$. Write the $i$th call as $Z_i \leftarrow \mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}(\mathsf{prefix}_i \| M_i)$, where $\mathsf{prefix}_i = \mathsf{prefix}_{\mathbf{ct}_i}$, for $1 \le i \le \gamma_{\mathrm{ch}}$. Processing this call yields a challenge path

$$\big(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i \| M_i)} Z_i \| R_i\big)$$

for some $R_i \in \{0,1\}^{s-n}$. Let $(W_{\tau(i)}; X_{\tau(i)}, Y_{\tau(i)})$ be its terminal record. Thus

$$\tau : \{1, \ldots, \gamma_{\mathrm{ch}}\} \longrightarrow \{1, \ldots, Q\}, \qquad X_{\tau(i)} \boxplus Y_{\tau(i)} = Z_i \| R_i.$$

The map need not be injective in the unrestricted experiment; its injectivity in the stopped transcript is proved when it is used below.

Every second-preimage witness $P\|M'$ considered in this section must belong to the complete hash-input domain specified in Sect. 2.2. In particular, its bit length is less than $2^{64}$ and its padded block length is at most $\ell_{\max}$; for an 8181 prefix, $M'$ must parse as an $n$-bit randomizer followed by a message in the domain of SIGN. This convention applies to $\mathsf{BadHonSpr}_n$, $\mathsf{PreChSpr}$, $\mathsf{PostChSpr}$, and $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$.

With the above notations, we introduce an event $\mathsf{BadHonSpr}_n$, which occurs if some records accessed by honest ideal-cipher queries underlying a challenge hash-call form a fixed-prefix second-preimage path for that challenge hash-call. Concretely, $\mathsf{BadHonSpr}_n$ occurs if during processing a challenge hash-call, $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ makes a fresh honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ such that:

- Before $Y \leftarrow \mathbf{E}_W(X)$ is made, there has been another "challenge path" $\big(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i\|M_i)} Z_i\|R_i\big)$ in $\mathcal{G}$, and
- After the record $(W; X, Y)$ is materialized in $\mathcal{Q}$, there appears a fixed-prefix second preimage for $\big(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i\|M_i)} Z_i\|R_i\big)$. Concretely, there appears a path $\big(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i\|M')} Z_i\|R'\big)$ in $\mathcal{G}$ for some $M' \neq M_i$ and $R' \in \{0,1\}^{s-n}$.

Run the experiment with the following stopping rule: abort immediately when $\mathsf{Bad}$ first occurs, and otherwise stop at the first fresh honest ideal-cipher query that causes $\mathsf{BadHonSpr}_n$. Let the record it materializes be $q = (W; X, Y)$. Immediately before the response $Y$ is sampled, condition only on the history exposed up to that time. All earlier challenge paths and their outputs are then fixed. Choose a witnessing second-preimage path that is absent before $q$ is materialized and present afterwards. This path must contain $q$.

Consider the last occurrence of $q$ on this witnessing path. If this occurrence is nonterminal, the record immediately following it was already present before $q$ was materialized and has source $X \boxplus Y$, which causes clause (iii) of $\mathsf{BadHonest}$. Moreover, $q$ cannot occur twice on the witnessing path: after a non-last occurrence, either the next occurrence is again $q$, in which case $X \boxplus Y = X$ and clause (ii) of $\mathsf{BadHonest}$ occurs, or the next record was already present and has source $X \boxplus Y$, which again causes clause (iii). Thus, unless materializing $q$ causes $\mathsf{BadHonest}$, it occurs exactly once on the witnessing path and is its terminal record. Consequently, $\mathsf{left}_n(X \boxplus Y) = Z_i$ for one of the at most $\widehat{\Gamma}$ challenge outputs already present at the stopping time.

Conditioned on the stopped history, $W$ and $X$ and all these target values are fixed, whereas, because the query materializing $q$ is a fresh ideal-cipher query, $Y$ is uniform over at least $2^s - Q$ unused outputs of the permutation $\mathbf{E}_W$. For each fixed target $z \in \{0,1\}^n$, at most $2^{s-n}$ choices of $Y$ satisfy $\mathsf{left}_n(X \boxplus Y) = z$. Charging the first occurrence only and summing over the at most $\widehat{B}$fresh honest ideal-cipher queries for challenges gives

$$\Pr\big[\mathsf{BadHonSpr}_n \wedge \neg\mathsf{Bad}\big] \leq \frac{\widehat{B}\widehat{\Gamma}2^{s-n}}{2^s - Q} \leq \frac{\widehat{B}\widehat{\Gamma}}{2^n - Q}. \tag{53}$$

**Pre/Post-challenge second-preimage events.** We introduce two second preimage-related events, i.e., *post-challenge second preimage event* $\mathsf{PostChSpr}$ and *pre-challenge second preimage event* $\mathsf{PreChSpr}$:

– PostChSpr: After any collected challenge path $\big(IV \xrightarrow{\mathsf{lpad}(P\|M)} Z\|R\big)$ is created during the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}^{\mathsf{DM}^{\mathbf{E}}}_n},\mathscr{A},m,u}$, a subsequent fresh adversarial ideal-cipher query creates a second preimage path $\big(IV \xrightarrow{\mathsf{lpad}(P\|M')} Z\|R'\big)$ with $R' \in \{0,1\}^{s-n}$ and $M' \neq M$;
– PreChSpr: When any collected challenge path $\big(IV \xrightarrow{\mathsf{lpad}(P\|M)} Z\|R\big)$ is newly created during the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}^{\mathsf{DM}^{\mathbf{E}}}_n},\mathscr{A},m,u}$, there already exists a corresponding second preimage path $\big(IV \xrightarrow{\mathsf{lpad}(P\|M')} Z\|R'\big)$ for some $R' \in \{0,1\}^{s-n}$ and $M' \neq M$.

As will be proven in Lemma 13, BadFPMSpr implies at least one of

$$\mathsf{BadHonest},\quad \mathsf{BadFwd},\quad \mathsf{BadBwd},\quad \mathsf{BadHonSpr}_n,\quad \mathsf{PreChSpr},\quad \mathsf{PostChSpr}.$$

### D.4 Number of Certain Types of Paths in $\mathcal{G}_i$

In the fixed execution, let $\alpha_P$ be the number of distinct prefix values among the actually occurring challenge calls $\mathsf{prefix}_1, \ldots, \mathsf{prefix}_{\gamma_{\mathrm{ch}}}$ (and put $\alpha_P := 0$ when $\gamma_{\mathrm{ch}} = 0$), and enumerate them in order of their first occurrence as $P_1, \ldots, P_{\alpha_P}$. Thus $\alpha_P \le \gamma_{\mathrm{ch}} \le \widehat{\Gamma}$. As established in Sect. D.1, these values are prefix-free and are shorter than one block. A vertex $S$ is *reachable from $IV$ via $P_k$* in $\mathcal{G}_i$ if $\mathcal{G}_i$ contains a path $\big(IV \xrightarrow{P_k\|\star} S\big)$. For $S^{\mathsf{tar}} \in \{0,1\}^s$, a vertex $S \neq S^{\mathsf{tar}}$ has a *limited path to $S^{\mathsf{tar}}$ in $\mathcal{G}_i$* if $\mathcal{G}_i$ contains a non-empty path from $S$ to $S^{\mathsf{tar}}$ of length at most $\ell_{\max} - 1$.

**Lemma 6 (Number of paths in $\mathcal{G}_i$).** *Fix $0 \le i \le Q^\star$ and assume that no record in $\mathcal{Q}_i$ has Davies-Meyer endpoint $IV$. Then:*

*(i) For every vertex $S$,*

$$\#\Big\{k : \big(IV \xrightarrow{P_k\|\star} S\big) \text{ occurs in } \mathcal{G}_i\Big\} \le \kappa(\mathcal{G}_i) + 1.$$

*(ii) Let $N_{k,i}$ be the number of distinct vertices reachable from $IV$ via $P_k$ in $\mathcal{G}_i$. Then*

$$\sum_{k=1}^{\alpha_P} N_{k,i} \le \big(\kappa(\mathcal{G}_i) + 1\big) i.$$

*(iii) For every $S^{\mathsf{tar}} \in \{0,1\}^s$, at most*

$$\big(\kappa(\mathcal{G}_i) + 1\big)(\ell_{\max} - 1)$$

*distinct vertices $S \neq S^{\mathsf{tar}}$ have a limited path to $S^{\mathsf{tar}}$ in $\mathcal{G}_i$. If $S = S^{\mathsf{tar}}$ is additionally admitted only when there is a non-empty path from $S^{\mathsf{tar}}$ back to $S^{\mathsf{tar}}$ of the prescribed length, the bound is increased by at most one.*

*Proof.* For clause (i), fix $S$, write $m(S)$ for the number on the left-hand side of that clause, and, for each of these $m(S)$ prefix values, choose one witnessing path. If $S = IV$, then $m(S) = 0$ because no edge enters $IV$, so assume $S \neq IV$. Truncate every chosen path immediately after its *first* arrival at $S$, and let $\mathcal{H}$ be the union of the resulting

truncated paths. This first-arrival truncation is essential: it makes $S$ have outdegree zero in $\mathcal{H}$, even when the original witness contains a cycle through $S$.

The first records of the truncated paths are distinct. Indeed, all paths start at $IV$, and equality of two first block labels would make two distinct prefix-free values prefixes of the same block. Hence $IV$ has outdegree at least $m(S)$ in $\mathcal{H}$. Every vertex other than $IV$ and $S$ has outdegree at least one. Consequently,

$$|E(\mathcal{H})| \geq m(S) + |V(\mathcal{H})| - 2.$$

No edge of $\mathcal{G}_i$ enters $IV$ by assumption, while every other vertex of $\mathcal{H}$ has indegree at least one. Therefore

$$|E(\mathcal{H})| = |V(\mathcal{H})| - 1 + \sum_{\widetilde{S} \in V(\mathcal{H})} \max\{d^-_{\mathcal{H}}(\widetilde{S}) - 1, 0\} \leq |V(\mathcal{H})| - 1 + \kappa(\mathcal{G}_i).$$

Comparison proves $m(S) \leq \kappa(\mathcal{G}_i) + 1$.

For clause (ii), every reachable vertex is the endpoint of an edge of $\mathcal{G}_i$; it cannot be $IV$. There are at most $i$ distinct such vertices. Double counting the pairs $(k, S)$ and applying clause (i) gives

$$\sum_{k=1}^{\alpha_P} N_{k,i} = \sum_{S} m(S) \leq \big(\kappa(\mathcal{G}_i) + 1\big) i.$$

For clause (iii), put $L := \ell_{\max} - 1$ and let

$$\mathcal{S}_{\mathsf{tar},i} := \big\{ S \neq S^{\mathsf{tar}} : 1 \leq \mathrm{dist}_{\mathcal{G}_i}(S, S^{\mathsf{tar}}) \leq L \big\}.$$

Construct a subgraph $\mathcal{A}_{\mathsf{tar}}$ with vertex set $\mathcal{S}_{\mathsf{tar},i} \cup \{S^{\mathsf{tar}}\}$. For every $S \in \mathcal{S}_{\mathsf{tar},i}$, choose one edge $S \to S'$ satisfying

$$\mathrm{dist}_{\mathcal{G}_i}(S', S^{\mathsf{tar}}) = \mathrm{dist}_{\mathcal{G}_i}(S, S^{\mathsf{tar}}) - 1,$$

using a fixed lexicographic rule to break ties. Distance strictly decreases along every selected edge, so $\mathcal{A}_{\mathsf{tar}}$ is a directed tree toward $S^{\mathsf{tar}}$: every non-root vertex has outdegree one, $S^{\mathsf{tar}}$ has outdegree zero, and $|E(\mathcal{A}_{\mathsf{tar}})| = |V(\mathcal{A}_{\mathsf{tar}})| - 1$.

Let $n_d$ be the number of vertices of indegree $d$ in $\mathcal{A}_{\mathsf{tar}}$. The tree identity gives

$$n_0 = 1 + \sum_{\widetilde{S} \in V(\mathcal{A}_{\mathsf{tar}})} \max\{d^-_{\mathcal{A}_{\mathsf{tar}}}(\widetilde{S}) - 1, 0\} \leq 1 + \kappa(\mathcal{G}_i).$$

Every non-root vertex lies on a selected path from an indegree-zero vertex to $S^{\mathsf{tar}}$, and each such path has at most $L$ non-root vertices. Thus

$$|\mathcal{S}_{\mathsf{tar},i}| \leq n_0 L \leq \big(\kappa(\mathcal{G}_i) + 1\big)(\ell_{\max} - 1).$$

Admitting $S^{\mathsf{tar}}$ itself as the start of a non-empty cycle adds only the single vertex $S^{\mathsf{tar}}$. This proves the last assertion and the lemma. □

**Lemma 7 (Bounds on $\kappa(\mathcal{G}_Q)$).** *Assume that $Q < 2^s$. Then, in* $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMDDM}^{\mathbf{E}}_n},\mathscr{A},m,u}$,

$$\Pr\left[\kappa(\mathcal{G}_Q) > \frac{\mathrm{e}Q(Q-1)}{2(2^s - Q)} + n\right] \leq \frac{1}{2^n}. \tag{54}$$

*Proof.* For every fresh ideal-cipher query with index $j \in \{1, \ldots, Q^\star\}$, define

$$I_j := \kappa(\mathcal{G}_j) - \kappa(\mathcal{G}_{j-1}). \tag{55}$$

It thus holds $\kappa(\mathcal{G}_i) = \sum_{j=1}^{i} I_j$ for all $i \in \{0, ..., Q\}$.

To work with the deterministic upper bound $Q$, if $Q^\star < Q$, then, for every $Q^\star < j \le Q$, we set

$$\mathcal{G}_j := \mathcal{G}_{Q^\star} \qquad \text{and} \qquad I_j := 0.$$

Note that these are merely "dummy" auxiliary variables.

Consider the record $(W_j; X_j, Y_j)$ materialized by the $j$th fresh ideal-cipher query. When this record is materialized in $\mathcal{Q}$, only the indegree of the vertex $X_j \boxplus Y_j$ changes. Therefore,

$$I_j = \mathbf{1}[X_j \boxplus Y_j \text{ is already the endpoint of some edge in } \mathcal{G}_{j-1}] \in \{0, 1\}. \tag{56}$$

Namely, $I_j$ *is* the increment of the excess indegree. Conditioned on the history $\mathsf{Hist}_{j-}$ before the $j$ th fresh ideal-cipher query is made, the graph $\mathcal{G}_{j-1}$ contains at most $j-1$ distinct edge-endpoints.

When the record $(W_j; X_j, Y_j)$ is materialized, $X_j \boxplus Y_j$ is uniform in at least $2^s - T - \widehat{B}$ possibilities. By this,

$$\mathbb{E}\big[I_j \mid \mathsf{Hist}_{j-}\big] = \Pr\big[I_j = 1 \mid \mathsf{Hist}_{j-}\big] \le \frac{j-1}{2^s - T - \widehat{B}}. \tag{57}$$

For the "dummy" query indices $j \in \{Q^\star + 1, ..., T + \widehat{B}\}$, the same inequality holds trivially due to $I_j = 0$. By the linearity of expectation, for every $0 \le i \le Q$ we have

$$\mathbb{E}\big[\kappa(\mathcal{G}_i)\big] \le \sum_{j=1}^{i} \frac{j-1}{2^s - T - \widehat{B}} \le \sum_{j=1}^{Q} \frac{j-1}{2^s - T - \widehat{B}} = \frac{Q(Q-1)}{2(2^s - Q)}. \tag{58}$$

We next establish the required probabilistic bound. Since $I_j \in \{0, 1\}$, we have

$$\mathrm{e}^{I_j} = 1 + (\mathrm{e} - 1) I_j. \tag{59}$$

Therefore, by Eq. (57) and the inequality $1 + x \le \mathrm{e}^x$,

$$\begin{aligned} \mathbb{E}\Big[\mathrm{e}^{I_j} \mid \mathsf{Hist}_{j-}\Big] &= 1 + (\mathrm{e} - 1)\mathbb{E}\big[I_j \mid \mathsf{Hist}_{j-}\big] \\ &\le 1 + \frac{(\mathrm{e}-1)(j-1)}{2^s - T - \widehat{B}} \le \mathrm{e}^{\frac{(\mathrm{e}-1)(j-1)}{2^s - T - \widehat{B}}}. \end{aligned} \tag{60}$$

**Moreover, $\kappa(\mathcal{G}_{j-1})$ is $\mathsf{Hist}_{j-}$-measurable and**

$$\kappa(\mathcal{G}_j) = \kappa(\mathcal{G}_{j-1}) + I_j.$$

Thus, the tower property gives

$$\begin{aligned} \mathbb{E}\Big[\mathrm{e}^{\kappa(\mathcal{G}_j)}\Big] &= \mathbb{E}\Big[\mathrm{e}^{\kappa(\mathcal{G}_{j-1})}\mathbb{E}\big[\mathrm{e}^{I_j} \mid \mathsf{Hist}_{j-}\big]\Big] \\ &\le \mathrm{e}^{\frac{(\mathrm{e}-1)(j-1)}{2^s - T - \widehat{B}}}\, \mathbb{E}\Big[\mathrm{e}^{\kappa(\mathcal{G}_{j-1})}\Big]. \end{aligned} \tag{61}$$

Iterating this inequality from $j = 1$ to $Q$ and using $\kappa(\mathcal{G}_0) = 0$, we obtain

$$\begin{aligned}\mathbb{E}\Big[\mathrm{e}^{\kappa(\mathcal{G}_Q)}\Big] &\le \mathrm{e}^{\sum_{j=1}^{Q} \frac{(\mathrm{e}-1)(j-1)}{2^s - T - \widehat{B}}} \\ &= \mathrm{e}^{\frac{(\mathrm{e}-1)Q(Q-1)}{2(2^s-Q)}}. \end{aligned} \tag{62}$$

We now apply Markov's inequality to the non-negative random variable $\mathrm{e}^{\kappa(\mathcal{G}_Q)}$: for every $u \ge 0$,

$$\begin{aligned}\Pr\left[\kappa(\mathcal{G}_Q) > \frac{\mathrm{e}Q(Q-1)}{2(2^s-Q)} + u\right] &= \Pr\left[\mathrm{e}^{\kappa(\mathcal{G}_Q)} > \mathrm{e}^{\frac{\mathrm{e}Q(Q-1)}{2(2^s-Q)}+u}\right] \\ &\le \mathrm{e}^{-\frac{\mathrm{e}Q(Q-1)}{2(2^s-Q)}-u}\mathbb{E}\Big[\mathrm{e}^{\kappa(\mathcal{G}_Q)}\Big] \\ &\le \mathrm{e}^{-\frac{Q(Q-1)}{2(2^s-Q)}-u} \le \mathrm{e}^{-u}. \end{aligned} \tag{63}$$

Setting $u := n$ yields

$$\Pr\left[\kappa(\mathcal{G}_Q) > \frac{\mathrm{e}Q(Q-1)}{2(2^s-Q)} + n\right] \le \mathrm{e}^{-n} \le \frac{1}{2^n},$$

as claimed. □

### D.5 Degree of Certain Types of Collisions

We use a stopped transcript rather than condition on the eventual non-occurrence of BadHonest. Couple the original experiment with an experiment that stops immediately before the first honest ideal-cipher query that would make BadHonest true. When this decision depends on the response to a fresh honest ideal-cipher query, sample the candidate response in the coupling, but do not append the offending record to the stopped transcript. Thus the indicators below are stopped subprobabilities; no ideal-cipher response is conditioned on a future good event.

Enumerate the retained records materialized by fresh ideal-cipher queries as

$$q_1 = (W_1; X_1, Y_1), \ldots, q_{q^\circ} = (W_{q^\circ}; X_{q^\circ}, Y_{q^\circ}), \qquad q^\circ \le Q,$$

and put $S_j^{\mathsf{rec}} := X_j \boxplus Y_j$. Let $\mathcal{C}^\circ$ be the set of logical challenge calls completed in this stopped transcript and put $\gamma^\circ := |\mathcal{C}^\circ| \le \widehat{\Gamma}$. For $i \in \mathcal{C}^\circ$, let $\tau(i)$ be the retained index of the record materialized by its terminal honest ideal-cipher query, and put

$$Z_i := \mathsf{left}_n(S_{\tau(i)}^{\mathsf{rec}}).$$

The map $i \mapsto \tau(i)$ is injective. Indeed, the terminal honest ideal-cipher query of one logical challenge is fixed by its atomic evaluation. If it were an ordinary lookup of a record already materialized by another logical challenge, clause (i) of BadHonest would stop the experiment before the later challenge were completed. Thus terminal *logical* incidences, rather than merely distinct terminal values, are counted correctly even when several $Z_i$ happen to be equal.

For $z, \delta \in \{0,1\}^n$, define

$$L_z := \#\big\{j \in \{1, \ldots, q^\circ\} : \mathsf{left}_n(S_j^{\mathsf{rec}}) = z\big\}, \tag{64}$$

$$B_\delta := \#\Big\{(i,j) \in \mathcal{C}^\circ \times \{1, \ldots, q^\circ\} : j \ne \tau(i),\ Z_i \boxminus \mathsf{left}_n(S_j^{\mathsf{rec}}) = \delta\Big\}. \tag{65}$$

Put

$$\begin{aligned}\mathsf{R}_n &:= \frac{\mathrm{e}Q}{2^n - Q} + 2n,\\ \mathsf{U}_n &:= \frac{\mathrm{e}\widehat{\Gamma}Q}{2^n - Q} + 2n,\\ \mathsf{L}_n &:= 2\mathsf{R}_n\mathsf{U}_n,\end{aligned} \tag{66}$$

and define the auxiliary stopped-transcript event

$$\mathsf{BadBucket}_n^{<\mathrm{HB}} := \left\{ \max_{z\in\{0,1\}^n} L_z > \mathsf{R}_n \right\} \vee \left\{ \max_{\delta\in\{0,1\}^n} B_\delta > \mathsf{L}_n \right\}. \tag{67}$$

**Lemma 8 (Stopped projected-difference buckets).** *Assume $n \geq 2$ and $Q < 2^n$. Then*

$$\Pr\left[\mathsf{BadBucket}_n^{<\mathrm{HB}}\right] \leq \frac{1}{2^n}. \tag{68}$$

*Consequently, except with probability $2^{-n}$, every $\delta \in \{0,1\}^n$ has the deterministic rank-one candidate bound*

$$B_\delta \leq 2\left(\frac{\mathrm{e}Q}{2^n - Q} + 2n\right)\left(\frac{\mathrm{e}\widehat{\Gamma}Q}{2^n - Q} + 2n\right).$$

*The same assertion holds for any transcript stopped earlier than* BadHonest*, since deleting records and logical challenge incidences can only decrease the two loads.*

*Proof.* Expose the execution in at most $Q$ fresh-response slots and pad the unused slots with dummies. Let $\mathcal{F}_{k-}$ contain the complete history before the candidate response in slot $k$, including the ideal-cipher query direction, key, input or output, the stopping status, and whether the current honest ideal-cipher query is terminal for a logical challenge. For a dummy or post-stop slot, let $\mathcal{F}_{k-}$ instead be the terminal history of the stopped experiment and assign $S_k^{\mathsf{rec}}$ an arbitrary fixed dummy value. Let $\rho_k$ indicate that the candidate record in slot $k$ is actually retained before the BadHonest stop; set $\rho_k = 0$ for dummy and post-stop slots. If $S_k^{\mathsf{rec}}$ denotes the candidate Davies-Meyer endpoint, then, for every $z \in \{0,1\}^n$,

$$\mathbb{E}[\rho_k\,\mathbf{1}[\mathsf{left}_n(S_k^{\mathsf{rec}}) = z] \mid \mathcal{F}_{k-}] \leq \frac{2^{s-n}}{2^s - Q} \leq \frac{1}{2^n - Q}. \tag{69}$$

The first inequality holds in both ideal-cipher query directions. Multiplication by $\rho_k$ only removes responses that stop the experiment and therefore cannot increase this conditional point mass.

For a fixed $z$, apply the adaptive Bernoulli exponential-moment bound to the $Q$ indicators in Eq. (69). Their total predictable mean is at most $Q/(2^n - Q)$. For every $u \geq 0$ this gives

$$\Pr\left[L_z > \frac{\mathrm{e}Q}{2^n - Q} + u\right] \leq \mathrm{e}^{-u}.$$

Taking $u = 2n$ and a union bound over all $2^n$ values of $z$ yields

$$\Pr\left[\max_z L_z > \mathsf{R}_n\right] \leq \frac{1}{2^{n+1}}. \tag{70}$$

We next construct a predictable budget for the directed pairs. Just before a real slot $k$, let $\mathcal{P}_k$ contain

(i) $(i,k)$ for every logical challenge $i$ whose terminal record was retained before slot $k$; and
(ii) $(i,j)$ for every earlier retained record $j$ when the current fresh-response attempt corresponds to the terminal honest ideal-cipher query of logical challenge $i$.

For a dummy or post-stop slot put $\mathcal{P}_k := \varnothing$. Put $m_k := |\mathcal{P}_k|$. Both $\mathcal{P}_k$ and $m_k$ are $\mathcal{F}_{k-}$-measurable: all earlier terminal incidences are fixed, and the role of the current ideal-cipher query in the atomic challenge evaluation is fixed before its response is sampled.

Every directed incidence pair consisting of one logical challenge terminal attempt and one distinct record exposure is assigned to exactly one set $\mathcal{P}_k$, namely the set belonging to the later of its two exposure times. There are at most $\widehat{\Gamma}$ logical terminal attempts and at most $Q$ record slots. This proves, trajectory by trajectory, the predictable pair-budget identity and bound

$$\sum_{k=1}^{Q} m_k \le \widehat{\Gamma} Q. \tag{71}$$

This argument also covers a terminal attempt whose candidate response would trigger the stop: its candidate pairs are charged once, although none is retained in $B_\delta$.

For fixed $\delta$, let $I_{\delta,k}$ indicate that slot $k$ is retained and that at least one pair in $\mathcal{P}_k$ satisfies the projected-difference equality defining $B_\delta$. Immediately before the candidate response, each such equality fixes one $n$-bit projection of $S_k^{\mathsf{rec}}$. A union bound, Eq. (69), and the measurability of $m_k$ give

$$\mathbb{E}[I_{\delta,k} \mid \mathcal{F}_{k-}] \le \frac{m_k}{2^n - Q}. \tag{72}$$

For completeness, the required adaptive exponential-moment step remains valid for these random predictable budgets. Indeed,

$$\exp\left( \sum_{r=1}^{k} I_{\delta,r} - (\mathrm{e}-1) \sum_{r=1}^{k} \frac{m_r}{2^n - Q} \right)$$

is a supermartingale. Combining this with Eq. (71) and applying Markov's inequality shows that, for every $u \ge 0$,

$$\Pr\left[ \sum_{k=1}^{Q} I_{\delta,k} > \frac{\mathrm{e}\widehat{\Gamma} Q}{2^n - Q} + u \right] \le \mathrm{e}^{-u}.$$

Taking $u = 2n$ and a union bound over all $\delta$ gives

$$\Pr\left[ \max_{\delta \in \{0,1\}^n} \sum_{k=1}^{Q} I_{\delta,k} > \mathsf{U}_n \right] \le \frac{1}{2^{n+1}}. \tag{73}$$

It remains to pass from the number of successful exposure slots to the number of logical pairs. Let $b_{\delta,k}$ be the number of actual pairs counted in $B_\delta$ whose later retained exposure is $k$. If $q_k$ is not a challenge terminal, every such pair is $(i,k)$ and its target must satisfy $Z_i = \delta \boxplus \mathsf{left}_n(S_k^{\mathsf{rec}})$. Injectivity of $\tau$ implies that the number of logical targets having any fixed value is at most $\max_z L_z$. Hence

$$b_{\delta,k} \le I_{\delta,k} \max_z L_z.$$

If $q_k$ is the terminal record of a logical challenge, injectivity says it is the terminal of exactly one such challenge. The new pairs split into the two classes $(i,k)$ and $(i_k,j)$; each class contains at most $\max_z L_z$ hitting pairs. Therefore, in all cases,

$$b_{\delta,k} \le 2I_{\delta,k} \max_z L_z.$$

Summing over $k$ gives the deterministic inequality

$$B_\delta \le 2\left(\max_z L_z\right) \sum_{k=1}^{Q} I_{\delta,k}. \tag{74}$$

On the simultaneous complement of the events in Eqs. (70) and (73), this is at most $2\mathsf{R}_n\mathsf{U}_n = \mathsf{L}_n$ for every $\delta$. The two failure probabilities sum to $2^{-n}$, proving the lemma. □

### D.6 Probability of BadHonest

Concretely, this section bounds $\Pr\big[\mathsf{BadHonest} \mid \neg\mathsf{BadID}_\mu\big]$. We first establish the conditional anchor-value estimate for an arbitrary fixed identifier vector. Only after that estimate has been proved do we impose $\neg\mathsf{BadID}_\mu$ for the prefix-multiplicity count.

**"Anchor" values entering first challenge blocks.** For every $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, let $\mathsf{anch}_{\mathbf{ct}}$ denote the complete $n$-bit value occupying the first variable position of the first padded block of the challenge call at coordinate $\mathbf{ct}$. Concretely,

$$\mathsf{anch}_{\mathbf{ct}} := \begin{cases} X_v^{\mathrm{i},q,d}, & \text{if } \mathbf{ct} = (\mathrm{i},q,d,v), \\ X_{e^{\mathrm{i}}}^{\mathrm{i},q,0}, & \text{if } \mathbf{ct} = (\mathrm{i},q,8080), \\ R^{\mathrm{i},q}, & \text{if } \mathbf{ct} = (\mathrm{i},q,8181), \\ pk^{\mathrm{i}}_{\rho-2^{h^{\mathrm{i}}}}, & \text{if } \mathbf{ct} = (\mathrm{i},\rho,8282), \\ T[2\rho]^{\mathrm{i}}, & \text{if } \mathbf{ct} = (\mathrm{i},\rho,8383). \end{cases} \tag{75}$$

The formatting rules and injectivity of $\mathsf{lpad}$ imply that, after all the other bits of the first block have been fixed, every candidate $a \in \{0,1\}^n$ determines a distinct complete first-block key; denote it by $K_{\mathbf{ct}}(a)$. Unlike Sect. C.3, the "anchor" values here are of $n$ bits.

Following Sect. C.3, we equip each execution with a *global clock* that assigns distinct time indices to successive actions (including ideal-cipher queries, random sampling, and updates to auxiliary variables or query histories) and allows us to refer unambiguously to the execution history immediately before or after any action. Let $\tau_{\mathrm{HB}}$ and $\tau_{\mathrm{G}}$ be the first occurrence times of $\mathsf{BadHonest}$ and $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ on this clock, with value $\infty$ if the corresponding event never occurs.

For each $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, let $\theta_{\mathbf{ct}}$ be the time immediately before the first honest ideal-cipher query of the challenge call at coordinate $\mathbf{ct}$. Let $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ denote the execution history at time $\theta_{\mathbf{ct}}$, with the anchor value $\mathsf{anch}_{\mathbf{ct}}$, the query response producing it (if any), and all copies or metadata that would reveal these values removed. Concretely, if $\mathsf{anch}_{\mathbf{ct}}$ is a direct secret or randomizer, $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ omits its value; if $\mathsf{anch}_{\mathbf{ct}}$ is derived from another "producer ideal-cipher query", $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ omits the "producer query's" response whose Davies-Meyer endpoint is $\mathsf{anch}_{\mathbf{ct}}$. Subject to these omissions, $\mathcal{F}^{\circ}_{\mathbf{ct}-}$ retains the schedule, identifiers, and all other query keys, inputs, public data, and assigned table

entries available no later than $\theta_{\mathbf{ct}}$. In the derived-anchor case, it also retains the (already fixed) "producer query's" key and input.

For the $t$ th adversarial ideal-cipher query, let $\vartheta_t$ be the time after its direction, key, and input or output have been fixed but before the query is processed. Let $\mathcal{V}_t^+$ be the complete adversarial view at this time: it includes the fixed query parameters, all preceding public keys, signing queries and responses, and the adversarial ideal-cipher history, but excludes records that have neither been accessed by an adversarial ideal-cipher query nor been transferred (due to signing) to the adversarial ideal-cipher history.

With the above notations, the following lemma is the $n$-bit variant of Lemma 2.

**Lemma 9 (Conditional bounds for $n$-bit-state anchor values).** *Assume $T+\widehat{B} \le 2^n$. Fix an arbitrary identifier vector $(I^1, ..., I^u)$.*

*(i) For every $\mathbf{ct} \in \mathcal{T}_{\mathrm{ch}}$, condition on an arbitrary fixed value of the redacted history $\mathcal{F}^{\circ}_{\mathbf{ct}-}$. Then, for every $\mathcal{F}^{\circ}_{\mathbf{ct}-}$-measurable "target value" $\widehat{a} \in \{0,1\}^n$,*

$$\Pr\big[\tau_{\mathrm{HB}} \ge \theta_{\mathbf{ct}} \ \wedge \ \mathsf{anch}_{\mathbf{ct}} = \widehat{a} \ \big|\ \mathcal{F}^{\circ}_{\mathbf{ct}-}\big] \le \frac{1}{2^n - Q}. \tag{76}$$

*(ii) Consider an adversarial ideal-cipher query selected adaptively from the information available immediately before it is processed. Let $t$ be its possibly random index among the adversarial ideal-cipher queries, and let $\vartheta_t$ denote the time immediately before it is processed. Here, $\mathcal{V}_t^+$ denotes the complete view available to the adversary at time $\vartheta_t$. Further, let the Winternitz coordinate $(\mathrm{i}, q, d, v)$ and $\widehat{x} \in \{0,1\}^n$ be $\mathcal{V}_t^+$-measurable. Put*

$$\mathsf{G}_t := \{\tau_{\mathrm{HB}} \ge \vartheta_t\} \cap \{\tau_{\mathrm{G}} \ge \vartheta_t\}.$$

*Then*

$$\Pr\Big[\mathsf{G}_t \ \wedge \ \big\{(\mathrm{i}, q, d, v) \in \mathcal{HS} \text{ at } \vartheta_t\big\} \ \wedge \ \big\{X_v^{\mathrm{i},q,d} = \widehat{x}\big\} \ \Big|\ \mathcal{V}_t^+\Big] \le \frac{1}{2^n - Q}. \tag{77}$$

The proof follows that of Lemma 2, with a modified counting argument for the full states sharing the same $n$ leftmost bits. Concretely, put $C := 2^{s-n}$. Each derived anchor candidate $x$ corresponds to exactly $C$ producer responses before accounting for retained records. If $r_-$ producer outputs are retained, let $b_x$ count those whose Davies-Meyer endpoints have $n$-bit projection $x$. Then $\sum_x b_x = r_-$, and $C - b_x$ producer responses remain available for candidate $x$. Thus Clause (i) follows from

$$\frac{C - b_{\widehat{a}}}{2^s - r_-} \le \frac{1}{2^n - Q}.$$

For Clause (ii), use the same redaction and reconstruction argument, but exclude candidates only through the consumer-record conditions. Let $h$ count these excluded candidates, and define $r_x$ as in the full-state proof. The disjoint retained record families satisfy

$$h + r_- + \sum_{x \text{ nonexcluded}} r_x \le Q.$$

The joint-probability identity in Eq. (34) now has right-hand side

$$A_{\mathrm{tr}} \frac{C - b_x}{2^s - r_x},$$

where $A_{\mathrm{tr}}$ is independent of $x$. The additional term $C - b_x$ counts the available producer responses and cannot in general be absorbed into this common constant. Normalization, using the record budget above, bounds the conditional probability of the target joint event by

$$\frac{1}{(1 - r_{\widehat{x}}/2^s)(2^n - h - r_-/C)} \le \frac{1}{2^n - Q}.$$

For directly sampled anchors, use their uniform prior $2^{-n}$ and take $r_- = b_x = 0$. The treatment of ordinary signing-time lookups and the final averaging over refined histories are unchanged. As in the full-state proof, $\mathsf{G}_t$ remains part of the joint event rather than an additional condition.

**Completing the analysis of BadHonest.** For the remainder of this subsection, fix an arbitrary identifier vector satisfying $\neg\mathsf{BadID}_\mu$. The same upper bound obtained below holds for every such vector and therefore yields the desired conditional probability bound.

*Probability That a First Honest Ideal-Cipher Query Is Non-Fresh.* For $c \in \{1, \ldots, \gamma_{\mathrm{ch}}\}$, the coordinate of the $c$ th challenge call is $\mathbf{ct}_c$, and its fixed formatted prefix is $\mathsf{prefix}_{\mathbf{ct}_c}$. Let $N_c$ be the number of records already present at time $\theta_{\mathbf{ct}_c}$ whose ideal-cipher key has the unique format $\mathsf{prefix}_{\mathbf{ct}_c}\|\star$. The redacted execution history fixes this set. For every one of these records, equality with the complete first-block key fixes at most one value of $\mathsf{anch}_{\mathbf{ct}_c}$. Eq. (76) and a union bound therefore charge the event that $\mathsf{BadHonest}$ first occurs at this first honest ideal-cipher query and that this first occurrence is caused by clause (i) by at most

$$\frac{\mathbb{E}[N_c]}{2^n - Q}.$$

The formatted challenge prefixes are prefix-free. Consequently a record key has at most one compatible concrete formatted prefix.

For a concrete prefix $P$, put

$$m(P) := \#\{c \in \{1, \ldots, \gamma_{\mathrm{ch}}\} : \mathsf{prefix}_{\mathbf{ct}_c} = P\}.$$

Lemma 5 implies that, under the fixed good identifier vector, $m(P) \le \mu$. If $P(r)$ is the unique compatible prefix of a record $r$ (and we put $m(P(r)) := 0$ when none exists), then, trajectory by trajectory,

$$\sum_{c=1}^{\gamma_{\mathrm{ch}}} N_c = \sum_{r \in \mathcal{Q}} \sum_{\substack{1 \le c \le \gamma_{\mathrm{ch}}: \\ r \text{ exists before } \theta_{\mathbf{ct}_c}}} \mathbf{1}[\mathsf{prefix}_{\mathbf{ct}_c} = P(r)] \le \sum_{r \in \mathcal{Q}} m(P(r)) \le \mu|\mathcal{Q}| \le \mu Q. \quad (78)$$

Thus the multiplicity argument is pathwise even though both the earlier record set and the later challenge schedule may be adaptive. Summing the stopped charges gives

$$\Pr\big[\mathsf{BadHonest} \text{ first occurs via clause (i) at a first honest ideal-cipher query of a challenge} \mid \neg\mathsf{BadID}_\mu\big] \le \frac{\mu Q}{2^n - Q}. \quad (79)$$

*Non-first honest ideal-cipher queries must be fresh.* Consider two consecutive records on an honestly evaluated challenge path,

$$(W_i; S_{i-1}, Y_i) \quad \text{and} \quad (W_{i+1}; S_i, Y_{i+1}), \qquad S_i = S_{i-1} \boxplus Y_i.$$

Suppose that the honest ideal-cipher query accessing the second record is an ordinary lookup. On the first-bad-event branch under consideration, the honest ideal-cipher query accessing the first record is fresh; otherwise that query has already caused clause (i). If the two records are the same, then $S_i = S_{i-1}$, and the fresh honest ideal-cipher query materializing the first record causes clause (ii). If they are distinct, atomicity implies that the second record was already present when the first record was materialized. The endpoint $S_i$ of the first record then equals the source of another existing record, so the fresh honest ideal-cipher query materializing the first record causes clause (iii) of $\mathsf{BadHonest}$. Consequently, before the first occurrence of $\mathsf{BadHonest}$ every non-first honest ideal-cipher query in a challenge evaluation is fresh; no separate address-reuse charge is needed for it.

*Endpoint conditions of fresh honest ideal-cipher queries.* Consider any fresh honest ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$ and stop immediately before its response is sampled. The key, source, and all earlier records are now fixed. The response is uniform among at least $2^s - Q$ unused outputs of $\mathbf{E}_W$. Clause (ii) prescribes at most two endpoint targets, namely $IV$ and $X$. Clause (iii) prescribes the source or endpoint of one of at most $Q$ earlier physical records and hence at most $2Q$ further targets. Thus the stopped conditional probability that this fresh honest ideal-cipher query first causes clause (ii) or clause (iii) is at most

$$\frac{2 + 2Q}{2^s - Q}.$$

There are at most $\widehat{B}$ honest ideal-cipher queries underlying all challenge hash-calls. Combining this observation with Eq. (79) yields

$$\Pr\left[\mathsf{BadHonest} \mid \neg\mathsf{BadID}_\mu\right] \le \frac{\mu(T + \widehat{B})}{2^n - Q} + \frac{\widehat{B}\big(2 + 2(T + \widehat{B})\big)}{2^s - T - \widehat{B}}. \tag{80}$$

## D.7 Probability of Event $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ in $\mathbf{Exp}^{\mathbf{m\text{-}unforge}}_{\mathbf{E},\mathbf{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}},\mathscr{A},\boldsymbol{m},\boldsymbol{u}}$

**Concrete meaning of $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$.** For a Winternitz coordinate $(\mathrm{i}, q, d, v)$, recall the notation $W_{(\mathrm{i},q,d,v)}(x) = \mathsf{lpad}\big(I^{\mathrm{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| x\big)$ from Sect. D.6. Then, $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ occurs precisely when there is a coordinate $(\mathsf{i}, q, d, v) \in \mathcal{HS}$ while the record

$$\big(W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d}); IV, \mathbf{E}_{W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d})}(IV)\big)$$

has entered the adversarial ideal-cipher history. Note that querying the same ideal-cipher key at an input different from $IV$ does not by itself put the ideal-cipher query record $\big(W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d}); IV, \mathbf{E}_{W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d})}(IV)\big)$ in that history.

### Probability of $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$.

**Lemma 10.** *It holds*

$$\Pr\left[\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}} \wedge \neg\mathsf{BadID}_\mu \wedge \neg\mathsf{BadHonest}\right] \leq \frac{\mu T}{2^n - Q}. \tag{81}$$

*Proof.* Fix an arbitrary good identifier vector. We distinguish the ways in which the record $\left(W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d}); IV, \mathbf{E}_{W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d})}(IV)\right)$ can enter the adversarial history while $(\mathrm{i},q,d,v) \in \mathcal{HS}$.

*Records transferred during initialization.* The general framework transfers the records underlying the 8080, 8282, and 8383 challenge calls to the adversarial history during initialization. If one of these transferred records is the same record as $r_{(\mathrm{i},q,d,v)}$ for a Winternitz coordinate $(\mathrm{i},q,d,v)$ that remains in $\mathcal{HS}$, then the group-2 challenge occurrence and the Winternitz challenge occurrence use the same record. Whichever of the two initialization challenge occurrences is processed later makes an honest ideal-cipher query that is an ordinary lookup of that record and causes clause (i) of $\mathsf{BadHonest}$. Initialization-time transfers therefore make no contribution together with $\neg\mathsf{BadHonest}$.

*Adaptive adversarial ideal-cipher queries.* Consider the $t$th forward or backward adversarial ideal-cipher query and condition on the view $\mathcal{V}_t^+$ at global time $\vartheta_t$, after its parameters have been fixed but before the query is processed. Let $W$ be its ideal-cipher key. If $W$ is not the suffix padding of a correctly formatted Winternitz input, the query cannot enter $r_{(\mathrm{i},q,d,v)}$ for a hidden coordinate $(\mathrm{i},q,d,v)$.

Otherwise, injectivity of $\mathsf{lpad}$ and the fixed-length coordinate encoding determine a unique tuple

$$(II, q, d, v, \widehat{x})$$

such that

$$W = \mathsf{lpad}\big(II \| [q]_{32} \| [d]_{16} \| [v]_8 \| \widehat{x}\big).$$

Under $\neg\mathsf{BadID}_\mu$, at most $\mu$ user indices $\mathrm{i}$ satisfy $I^{\mathrm{i}} = II$. Hence there are at most $\mu$ currently hidden coordinates $(\mathrm{i},q,d,v)$ for which $W = W_{(\mathrm{i},q,d,v)}(\widehat{x})$. For every such coordinate, equality of the query key and the hidden key $W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d})$ requires

$$X_v^{\mathrm{i},q,d} = \widehat{x}.$$

Eq. (77), applied at this adversarial-view stopping time, bounds the conditional subprobability of this equality together with the absence of an earlier $\mathsf{BadHonest}$ or $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$ by $1/(2^n - Q)$. A union bound over the at most $\mu$ candidate users therefore charges at most $\mu/(2^n - Q)$ to the first bad guess made by this query.

This necessary key-equality condition applies in both directions of adversarial ideal-cipher queries. A forward adversarial ideal-cipher query can enter $r_{(\mathrm{i},q,d,v)}$ only when its input is $IV$. A backward adversarial ideal-cipher query can enter $r_{(\mathrm{i},q,d,v)}$ only when its inverse response is $IV$. Ignoring either additional requirement only enlarges the event being bounded. The argument is valid for adaptive adversarial ideal-cipher queries because $W$, $\widehat{x}$, and the at most $\mu$ coordinate candidates determined by the formatted key and the good identifier vector are all $\mathcal{V}_t^+$-measurable before the adversarial ideal-cipher query response is exposed. Membership of each candidate in $\mathcal{HS}$ is handled by the indicator in Eq. (77); it need not itself be visible to the adversary. Summing over the at most $T$ adversarial ideal-cipher queries gives

$$\frac{\mu T}{2^n - Q}.$$

*Records and values disclosed by signing.* As specified in Sect. B, the signing update first removes from $\mathcal{HS}$ the coordinates made computable by the response and only then transfers their records to the adversarial history. Thus, when a signing response discloses a Winternitz state, every coordinate whose own record thereby becomes computable is removed from $\mathcal{HS}$; its record cannot newly witness $\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}$. If a record disclosed for one coordinate were the same record as $r_{(\mathrm{i},q,d,v)}$ for another coordinate $(\mathrm{i},q,d,v)$ that remains in $\mathcal{HS}$, then the later of the two initialization challenge occurrences would have made an honest ideal-cipher query that is an ordinary lookup of that record and would have caused clause (i) of $\mathsf{BadHonest}$.

Finally, disclosure of an $n$-bit value equal to $X_v^{\mathrm{i},q,d}$ under a different ideal-cipher key does not itself put $r_{(\mathrm{i},q,d,v)}$ in the adversarial history. If the adversary subsequently uses that value to construct $W_{(\mathrm{i},q,d,v)}(X_v^{\mathrm{i},q,d})$, the resulting adversarial ideal-cipher query is already covered by the adaptive-query analysis above. This is exactly why $\mathcal{V}_t^+$ in Lemma 9 includes all preceding signing responses.

The three cases exhaust the ways in which $r_{(\mathrm{i},q,d,v)}$ can enter the adversarial history while $(\mathrm{i},q,d,v) \in \mathcal{HS}$. We have proved the claimed bound for every fixed good identifier vector. Averaging over these vectors and multiplying by $\Pr[\neg\mathsf{BadID}_\mu] \le 1$ proves Eq. (81). □

### D.8 Probability of PreChSpr

We use the experiment that is coupled to the original experiment and stops immediately before the first occurrence of $\mathsf{Bad}$. Its retained transcript will be called the *stopped transcript*. Any execution contributing to $\mathsf{PreChSpr} \wedge \neg\mathsf{Bad}$ in the original experiment is also an execution in which $\mathsf{PreChSpr}$ occurs in this stopped experiment. All histories below are therefore stopped histories; in particular, no response to a fresh ideal-cipher query is conditioned on the future non-occurrence of a bad event.

**Lemma 11 (Pre-challenge second preimage).** *If $Q < 2^n$, then*

$$\Pr\left[\mathsf{PreChSpr} \wedge \neg\mathsf{Bad}\right] \le \frac{Q}{2^n - Q} \min\left\{\widehat{\Gamma},\, \mu\left(1 + \frac{Q(Q-1)}{2(2^s - Q)}\right)\right\}. \tag{82}$$

*If $Q \ge 2^n$, the right-hand side is replaced by the trivial bound 1. Thus no expression having denominator $2^n - Q$ is used outside the nontrivial branch $Q < 2^n$.*

*Proof.* Expose at most $\widehat{\Gamma}$ *potential terminal-attempt slots* in chronological order, padding unused positions with dummies. A real slot is installed when the execution, before the stop, is about to sample the response of a fresh honest ideal-cipher query that is terminal for a logical challenge. For every real slot $a$, let $\mathcal{F}_{a-}$ be this complete pre-response history, let $i(a)$ be the corresponding logical challenge occurrence, and define $\varphi(a)$ as the unique index satisfying

$$\mathsf{prefix}_{i(a)} = P_{\varphi(a)}.$$

The challenge role, $i(a)$, $\varphi(a)$, the key, and the source are all $\mathcal{F}_{a-}$-measurable. For a dummy slot, take $\mathcal{F}_{a-}$ to be the terminal history. If the next terminal honest ideal-cipher query for a challenge would be an ordinary lookup, clause (i) of $\mathsf{BadHonest}$ stops the coupling before a fresh-response slot is installed.

For a real slot $a$, let $r(a)$ be the candidate index of the record to be materialized by the fresh ideal-cipher query and define the predictable set

$$\mathcal{A}_{a-} := \left\{S : IV \xrightarrow{P_{\varphi(a)}\|\star} S \text{ in } \mathcal{G}_{r(a)-1}\right\}, \qquad N_{a-} := |\mathcal{A}_{a-}|.$$

For a dummy slot put $N_{a-} := 0$, $\rho_a := 0$, and $E_a := 0$, and assign its unused $r(a)$ and $S^{\mathsf{rec}}_{r(a)}$ symbols arbitrary fixed dummy values. For a real slot, let $\rho_a$ indicate that the candidate response is retained by the stopped experiment and completes the challenge, and let $E_a$ be the event that this retained response causes $\mathsf{PreChSpr}$. Although $\rho_a$ may depend on the sampled response, it only deletes responses and is never placed in the conditioning sigma-field.

If $E_a = 1$, the witnessing colliding path is complete before the challenge terminal record is retained. If $S$ is its full terminal state, then $S \in \mathcal{A}_{a-}$ and necessarily

$$\mathsf{left}_n(S^{\mathsf{rec}}_{r(a)}) \in \big\{\mathsf{left}_n(S) : S \in \mathcal{A}_{a-}\big\}.$$

We count all states in $\mathcal{A}_{a-}$, including those reached by a path having the wrong padding or length. Before the response is sampled, the target set is fixed, while the response to the fresh ideal-cipher query is uniform over at least $2^s - Q$ unused permutation values. Therefore, using that multiplication by $\rho_a$ can only decrease a pre-response subprobability and that $Q < 2^n$,

$$\begin{aligned}\mathbb{E}[\mathbf{1}[E_a] \mid \mathcal{F}_{a-}] &\le \mathbb{E}\big[\rho_a\, \mathbf{1}\big[\mathsf{left}_n(S^{\mathsf{rec}}_{r(a)}) \in \mathsf{left}_n(\mathcal{A}_{a-})\big] \;\big|\; \mathcal{F}_{a-}\big] \\ &\le \frac{N_{a-}2^{s-n}}{2^s - Q} \le \frac{N_{a-}}{2^n - Q}. \end{aligned} \tag{83}$$

Here $\mathsf{left}_n(\mathcal{A})$ denotes $\{\mathsf{left}_n(S) : S \in \mathcal{A}\}$.

The first occurrence of $\mathsf{PreChSpr}$ in the stopped experiment implies $E_a = 1$ for some potential slot. Summing the predictable charges over the deterministic collection of at most $\widehat{\Gamma}$ slots and applying the tower property gives

$$\Pr_{\mathrm{stop}}[\mathsf{PreChSpr}] \le \frac{1}{2^n - Q}\mathbb{E}\left[\sum_a N_{a-}\right]. \tag{84}$$

Pathwise, the sum in Eq. (84) has two bounds. The trivial one is $\widehat{\Gamma}Q$. For the refined one, monotonicity of reachability gives $N_{a-} = N_{\varphi(a),r(a)-1} \le N_{\varphi(a),Q}$ at every real slot, while dummy slots contribute zero. Since the stopped ledger contains no occurrence of $\mathsf{BadID}_\mu$, each concrete prefix value is used by at most $\mu$ collected challenge occurrences. Hence, using clause (ii) of Lemma 6,

$$\sum_a N_{a-} \le \mu \sum_{k=1}^{\alpha_P} N_{k,Q} \le \mu\big(\kappa(\mathcal{G}_Q) + 1\big)Q. \tag{85}$$

The stopped graph is obtained by deleting a suffix of exposures, so the expectation estimate in Eq. (58) continues to apply. Therefore

$$\mathbb{E}\left[\sum_a N_{a-}\right] \le Q \min\left\{\widehat{\Gamma},\, \mu\left(1 + \frac{Q(Q-1)}{2(2^s - Q)}\right)\right\}.$$

Combining this inequality with Eq. (84), and then capping a probability by one, proves Eq. (82). The case $Q \ge 2^n$ is bounded directly by one. □

### D.9 Probability of PostChSpr

We use the same stopped experiment as above. There are at most $T$ positions of adversarial ideal-cipher queries at which a completed challenge path is already available.

Query positions that are not used can be filled by dummy positions. We use this single deterministic cap for both ideal-cipher query directions; this remains valid even when the adversary chooses the direction adaptively and avoids replacing branch-dependent query counts by random "resource bounds."

Put

$$\begin{aligned}\mathsf{K}_s &:= \frac{\mathrm{e}Q(Q-1)}{2(2^s-Q)} + n,\\ \mathsf{M}_s &:= 1 + \big(\mathsf{K}_s + 1\big)(\ell_{\max} - 1).\end{aligned} \tag{86}$$

The leading 1 in $\mathsf{M}_s$ admits the starting state equal to the terminal state through a nonempty cycle.

**Lemma 12 (Post-challenge second preimage).** *Assume $Q < 2^n$, and define*

$$\begin{aligned}\mathsf{P}_{\mathrm{f}} &:= \min\left\{\frac{\widehat{\Gamma}T}{2^n-Q},\ \frac{\mu T}{2^n-Q}\left(1+\frac{Q(Q-1)}{2(2^s-Q)}\right)\right\},\\ \mathsf{P}_{\mathrm{b}} &:= \min\left\{\frac{\widehat{\Gamma}T}{2^n-Q},\ \frac{1}{2^n}+\frac{T\mathsf{L}_n}{2^s-Q}\right\},\\ \mathsf{P}_{\mathrm{int}} &:= \min\left\{\frac{\widehat{\Gamma}Q}{2^n-Q},\ \frac{1}{2^n}+\frac{\mathsf{M}_s\widehat{\Gamma}TQ}{(2^s-Q)(2^n-Q)}\right\}.\end{aligned}$$

*Then*

$$\Pr\left[\mathsf{PostChSpr}\wedge\neg\mathsf{Bad}\right] \le \mathsf{P}_{\mathrm{f}} + \mathsf{P}_{\mathrm{b}} + \mathsf{P}_{\mathrm{int}}. \tag{87}$$

*Consequently, in the original experiment,*

$$\Pr[\mathsf{PostChSpr}] \le \Pr[\mathsf{Bad}] + \min\{1, \mathsf{P}_{\mathrm{f}} + \mathsf{P}_{\mathrm{b}} + \mathsf{P}_{\mathrm{int}}\} \tag{88}$$

*Proof.* It suffices to work in the stopped experiment. Expose at most $T$ *potential post-challenge slots for adversarial ideal-cipher queries* and pad unused positions with dummies. A real slot $c$ is installed after the adversary has fixed the query direction, key, and input or output of a fresh adversarial ideal-cipher query, but before its response is sampled. Let $\mathcal{F}_{c-}$ be this pre-response history. For a real slot, put $r(c)-1$ equal to the number of records materialized by fresh ideal-cipher queries and retained in that history; if the candidate is retained, its record index is $r(c)$. This index is therefore $\mathcal{F}_{c-}$-measurable. Let $\rho_c$ indicate that the candidate record is retained by the stopped experiment. The direction and all adversarially supplied parameters are $\mathcal{F}_{c-}$-measurable; $\rho_c$ may depend on the response but only deletes outcomes. For a dummy slot, let $\mathcal{F}_{c-}$ be the terminal history of the stopped experiment, set $\rho_c := 0$ and $E_c := \varnothing$, and assign all unused candidate-record symbols arbitrary fixed values.

For a real slot, let $E_c$ be the event that the retained record at slot $c$ is the first record whose materialization causes $\mathsf{PostChSpr}$. On $E_c$, choose one newly completed second-preimage path by a fixed lexicographic rule. If the same record occurs more than once on that path, distinguish its first and last record occurrences. The three alternatives below refer to this last occurrence; let $\mathsf{Int}_c$ denote the event that it is internal. Every displayed pre-response charge is understood with the factor $\rho_c\mathbf{1}[E_c]$ on its left. Hence summing over the deterministic potential slots and applying the tower property is valid; no query index is selected after seeing its response. This convention also covers fixed points, cycles, and repeated uses of one record.

*The last occurrence of c is terminal: forward adversarial ideal-cipher query.* First suppose that $c = (W; X, Y)$ is materialized by the forward adversarial ideal-cipher query $Y \leftarrow \mathbf{E}_W(X)$. If the first occurrence of $c$ on the chosen path is not its first record, then, before $c$ is materialized, the old part of the path already reaches $X$ via its collected prefix value. By clause (i) of Lemma 6, at most $\kappa(\mathcal{G}_{r(c)-1}) + 1$ distinct collected prefix values can do so. If the first occurrence of $c$ is the first record, then $X = IV$ and the first-block key $W$ fixes at most one collected prefix value, because the collected prefix values are prefix-free and shorter than one block.

Under the stopped ledger, every concrete prefix value is associated with at most $\mu$ collected challenge targets. Thus, in either situation, the endpoint projection produced by the fresh ideal-cipher query must hit at most $\mu(\kappa(\mathcal{G}_{r(c)-1}) + 1)$ fixed targets. The direct target-count cap is $\widehat{\Gamma}$. Therefore the stopped conditional charge is at most

$$\frac{\min\{\widehat{\Gamma}, \mu(\kappa(\mathcal{G}_{r(c)-1}) + 1)\}}{2^n - Q}.$$

Summing over the at most $T$ positions of adversarial ideal-cipher queries and using Eq. (58) yields

$$\Pr[\text{terminal-forward branch}] \le \mathsf{P}_{\mathrm{f}}. \tag{89}$$

*The last occurrence of c is terminal: backward adversarial ideal-cipher query.* Now suppose that $c$ is materialized by the backward adversarial ideal-cipher query $X \leftarrow \mathbf{E}_W^{-1}(Y)$. A direct projection union bound over all existing challenge targets gives $\widehat{\Gamma}/(2^n - Q)$ per adversarial ideal-cipher query. For the refined bound, the first occurrence of $c$ cannot be the first record of the path, since this would require $X = IV$ and would trigger $\mathsf{BadBwd}$. Hence $X = S^{\mathsf{in}}$ for the endpoint $S^{\mathsf{in}}$ of an old record $q_j$ on a prefix path. If $a$ is the target challenge incidence, terminal closure requires

$$\mathsf{left}_n(Y) = Z_a \boxminus \mathsf{left}_n(S^{\mathsf{in}}). \tag{90}$$

The old record $q_j$ cannot be the terminal record of challenge $a$: otherwise $X$ equals a full challenge endpoint and the backward adversarial ideal-cipher query triggers $\mathsf{BadBwd}$. Immediately before the inverse response is sampled, define the history-measurable pair set

$$\mathcal{C}_{c-}(Y) := \big\{(a, j) : j \ne \tau(a),\ Z_a \boxminus \mathsf{left}_n(S_j^{\mathsf{rec}}) = \mathsf{left}_n(Y)\big\},$$

where only challenge incidences and records already retained before slot $c$ are used. It is a prefix of the stopped projected-difference bucket $B_{\mathsf{left}_n(Y)}$. If $|\mathcal{C}_{c-}(Y)| > \mathsf{L}_n$, the bucket failure has already been determined by the current history and is charged once to $\mathsf{BadBucket}_n^{<\mathrm{HB}}$. Otherwise there are at most $\mathsf{L}_n$ candidate full states $S^{\mathsf{in}} = S_j^{\mathsf{rec}}$. Conditional on the pre-query history, the inverse response $X$ is uniform over at least $2^s - Q$ full-state inputs. Hence every prescribed full-state candidate costs $1/(2^s - Q)$, rather than $1/(2^n - Q)$. Multiplication by the retention and first-cause indicators only decreases this subprobability. Lemma 8 bounds the once-only bucket charge by $2^{-n}$; summing the remaining predictable charges over the at most $T$ adversarial positions gives

$$\Pr[\text{terminal-backward branch}] \le \min\left\{\frac{\widehat{\Gamma} T}{2^n - Q},\ \frac{1}{2^n} + \frac{T\mathsf{L}_n}{2^s - Q}\right\} = \mathsf{P}_{\mathrm{b}}. \tag{91}$$

*The last occurrence of c is internal.* Let $S^{\mathsf{out}} := \mathrm{dst}(c)$, let $S^{\mathsf{tar}}$ be the full terminal state of the chosen second-preimage path, and let $a$ be its already completed challenge target, so $\mathsf{left}_n(S^{\mathsf{tar}}) = Z_a$. Because this is the last occurrence of $c$, the nonempty tail

$$S^{\mathsf{out}} \xrightarrow{\mathsf{tail}} S^{\mathsf{tar}}$$

consists entirely of records present before $c$. The chosen second-preimage input is admissible, so its entire padded path has at most $\ell_{\max}$ record occurrences. Removing the prefix through the last occurrence of $c$ leaves at most $\ell_{\max} - 1$ record occurrences in this nonempty tail. Let $u$ denote the retained index of the *old tail terminal record*; thus $S^{\mathsf{rec}}_u = S^{\mathsf{tar}}$. The index $u$ is not the index of the later closing record $c$.

We claim that $u \neq \tau(a)$. Otherwise the old tail and the honest challenge path share their terminal record and hence their full terminal state $S^{\mathsf{tar}}$. Write the honest states as $IV = S^{\mathsf{hon}}_0, S^{\mathsf{hon}}_1, \ldots, S^{\mathsf{hon}}_m = S^{\mathsf{tar}}$. If $S^{\mathsf{out}}$ is one of these states (including $IV$), materializing $c$ returns to an honest challenge state and causes BadFwd or BadBwd. Otherwise follow the old tail until its first entry into $\{S^{\mathsf{hon}}_0, \ldots, S^{\mathsf{hon}}_m\}$; such an entry exists at $S^{\mathsf{tar}}$. Let $q$ be the record closing this first entry and let $S^{\mathsf{hit}}$ be its endpoint. If $S^{\mathsf{hit}} = IV$, then materialization of $q$ by an adversarial ideal-cipher query causes clause (i) of BadFwd or BadBwd, while materialization by an honest ideal-cipher query causes clause (ii) of BadHonest. Hence $S^{\mathsf{hit}} \neq IV$. Let $q_{\mathsf{hon}}$ be the corresponding record on the honest challenge path. The two records cannot be materialized simultaneously: if they were the same record, the tail would already be at the honest source of $q_{\mathsf{hon}}$, giving an earlier entry. If $q$ is materialized first, the later honest ideal-cipher query either is an ordinary lookup at its address or produces its endpoint, causing clause (i) or (iii) of BadHonest. If $q_{\mathsf{hon}}$ is materialized first, a later adversarial ideal-cipher query materializing $q$ causes BadFwd or BadBwd, while a later honest ideal-cipher query for a challenge that materializes $q$ causes clause (iii) of BadHonest. This includes the possibilities of re-entry at $IV$ and of a record belonging to a later honest challenge. Every case contradicts the stopped ledger, proving the claim.

Accordingly, every internal witness first requires a nontrivial terminal–target contact

$$(a, u), \qquad u \neq \tau(a), \qquad \mathsf{left}_n(S^{\mathsf{rec}}_u) = Z_a.$$

Let

$$J := \#\big\{(a, u) : u \neq \tau(a),\ \mathsf{left}_n(S^{\mathsf{rec}}_u) = Z_a\big\} \leq B_0. \tag{92}$$

Here $B_0$ is the bucket in the transcript stopped only before BadHonest; the present Bad-stopped transcript is its prefix, so the inequality, rather than equality, is the generally valid relation. The predictable directed-pair budget Eq. (71), together with the stopped point-mass bound Eq. (69), gives

$$\mathbb{E}[J] \leq \frac{\widehat{\Gamma} Q}{2^n - Q}. \tag{93}$$

No independence between $S^{\mathsf{rec}}_u$ and $Z_a$ is asserted here: the equality is charged when the later of their two endpoints generated by fresh ideal-cipher queries is exposed. Since an internal witness implies $J \geq 1$, Markov's inequality immediately gives

$$\Pr[\text{internal branch}] \leq \frac{\widehat{\Gamma} Q}{2^n - Q}. \tag{94}$$

For the two-time refined bound, introduce a second, online stopping rule. At every fresh-response slot, sample the candidate response but do not retain it, and stop, if retaining it would make $\kappa$ exceed $\mathsf{K}_s$ for the first time. Since $\kappa$ is nondecreasing under record materializations, occurrence of this first-crossing stop implies $\kappa(\mathcal{G}_Q) > \mathsf{K}_s$ in the coupled original transcript. Thus Eq. (54) charges the stop by at most $2^{-n}$. At every history retained before this online stop, the current graph has $\kappa \le \mathsf{K}_s$; no future good event has been placed in the conditioning sigma-field. The additional retention indicator can only delete candidate responses. Write $\widehat{\rho}_c \le \rho_c$ for the indicator that slot $c$ is retained by both stopping rules.

At a history retained by both online stopping rules, and before the candidate response in a potential post-challenge slot $c$ is exposed, let $J_{<c}$ be the number of contacts already installed, and let $\mathcal{S}_{c-}$ be the union, over those contacts $(a,u)$, of all states having an old nonempty path of length at most $\ell_{\max}-1$ to $S_u^{\mathsf{rec}}$. Include $S_u^{\mathsf{rec}}$ itself only when such a nonempty cycle exists. The set $\mathcal{S}_{c-}$ is completely determined by the history before $c$. Clause (iii) of Lemma 6 and the deterministic stop threshold give

$$|\mathcal{S}_{c-}| \le \mathsf{M}_s J_{<c}.$$

If $c$ is forward, the full-state endpoint generated by this fresh ideal-cipher query must belong to $\mathcal{S}_{c-}$. If $c$ is backward with adversarially fixed output $Y$, the condition $X \boxplus Y \in \mathcal{S}_{c-}$ is equivalent to $X \in \mathcal{S}_{c-} \boxminus Y$, where $\mathcal{S}_{c-} \boxminus Y := \{S \boxminus Y : S \in \mathcal{S}_{c-}\}$; translation in $\mathbb{G}_s$ preserves cardinality. Thus both directions of adversarial ideal-cipher queries satisfy the same stopped conditional charge

$$\mathbb{E}[\widehat{\rho}_c\, \mathbf{1}[E_c \wedge \mathsf{Int}_c] \mid \mathcal{F}_{c-}] \le \frac{\mathsf{M}_s J_{<c}}{2^s - Q}. \tag{95}$$

This is the required separation of the two exposure times: $u$ installs the old terminal–target contact, whereas the later query materializing the record $c$ makes the full-state closing hit.

There are at most $T$ positions of closing adversarial ideal-cipher queries. Pathwise, $J_{<c} \le J$, so the tower property and a union bound yield

$$\sum_c \mathbb{E}[J_{<c}] \le T\mathbb{E}[J].$$

Using Eq. (93) in Eq. (95) therefore gives

$$\Pr[\text{internal branch}] \le \min\left\{ \frac{\widehat{\Gamma} Q}{2^n - Q},\ \frac{1}{2^n} + \frac{\mathsf{M}_s \widehat{\Gamma} T Q}{(2^s - Q)(2^n - Q)} \right\} = \mathsf{P}_{\mathrm{int}}. \tag{96}$$

The terminal-forward, terminal-backward, and internal alternatives are mutually exclusive for the chosen last occurrence of $c$. Summing their bounds and capping by one proves Eq. (87). The original-experiment statement follows by splitting on $\mathsf{Bad}$. For $Q \ge 2^n$ we use only the trivial probability bound, as claimed. □

### D.10 Probability of Event $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM^E}}}$ in $\mathbf{Exp}^{\mathbf{m\text{-}unforge}}_{\mathbf{E},\mathbf{LMS}^{\mathsf{ChopMD}_n^{\mathsf{DM^E}}},\mathscr{A},\boldsymbol{m},\boldsymbol{u}}$

**Lemma 13.** *If none of the events* $\mathsf{BadHonest}$, $\mathsf{BadFwd}$, $\mathsf{BadBwd}$, $\mathsf{BadHonSpr}_n$, $\mathsf{PreChSpr}$, *and* $\mathsf{PostChSpr}$ *occurs, then* $\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM^E}}}$ *cannot occur. Moreover, partitioning*

*the experiment according to the first event in the stopped ledger gives*

$$\begin{aligned}
&\Pr\left[\mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathsf{DM}^{\mathbf{E}}}}\right] \\
\le\ &\Pr\left[\mathsf{BadID}_\mu\right] + \Pr\left[\mathsf{BadHonest} \wedge \neg\mathsf{BadID}_\mu\right] \\
&+ \Pr\left[\mathsf{BadFwd} \wedge \neg\mathsf{BadID}_\mu \wedge \neg\mathsf{BadHonest}\right] \\
&+ \Pr\left[\mathsf{BadBwd} \wedge \neg\mathsf{BadID}_\mu \wedge \neg\mathsf{BadHonest} \wedge \neg\mathsf{BadFwd}\right] \\
&+ \Pr\left[\mathsf{BadHonSpr}_n \wedge \neg\mathsf{Bad}\right] \\
&+ \Pr\left[\mathsf{PreChSpr} \wedge \neg\mathsf{Bad} \wedge \neg\mathsf{BadHonSpr}_n\right] \\
&+ \Pr\left[\mathsf{PostChSpr} \wedge \neg\mathsf{Bad} \wedge \neg\mathsf{BadHonSpr}_n \wedge \neg\mathsf{PreChSpr}\right]. \qquad (97)
\end{aligned}$$

*Proof.* Assume that $\mathsf{BadFPMSpr}$ occurs while none of the six events in the statement occurs. Let the involved challenge path and a witnessing second-preimage path be

$$\mathsf{H} = \left(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i \| M_i)} Z_i \| R_i\right)$$

and

$$\mathsf{Alt} = \left(IV \xrightarrow{\mathsf{lpad}(\mathsf{prefix}_i \| M')} Z_i \| R'\right), \qquad M' \neq M_i.$$

Let $k$ be the smallest record index such that both paths are present immediately after the record $(W_k; X_k, Y_k)$ is materialized.

First suppose that $k \neq \tau(i)$. Outside $\mathsf{BadHonest}$, every honest ideal-cipher query made during the first evaluation of a challenge call is fresh, so $\mathsf{H}$ is completed exactly when its terminal record is materialized. Hence $k > \tau(i)$. If the record that completes $\mathsf{Alt}$ is materialized by a fresh adversarial ideal-cipher query, $\mathsf{PostChSpr}$ occurs. If it is materialized by an honest ideal-cipher query while processing a later collected challenge call, $\mathsf{BadHonSpr}_n$ occurs. Thus this case is impossible.

It remains that $k = \tau(i)$. If $\mathsf{Alt}$ is already complete before the record $(W_k; X_k, Y_k)$ is materialized, then $\mathsf{PreChSpr}$ occurs. Otherwise, $\mathsf{Alt}$ contains this newly materialized record. Outside $\mathsf{BadHonest}$, that record occurs exactly once on $\mathsf{Alt}$ and is terminal. Indeed, if its last occurrence were nonterminal, the first record of the nonempty old tail following it would already have source $X_k \boxplus Y_k$, causing clause (iii) of $\mathsf{BadHonest}$. If it occurred more than once, then after a non-last occurrence either the next occurrence would be the same record, giving $X_k \boxplus Y_k = X_k$ and clause (ii), or the first record in the old segment leading to the next occurrence would already have source $X_k \boxplus Y_k$, giving clause (iii).

Consequently, immediately before the record $(W_k; X_k, Y_k)$ is materialized, there are two distinct labeled paths

$$\mathsf{H}_0 = \left(IV \xrightarrow{\mathbf{W}_\mathsf{H}} X_k\right), \qquad \mathsf{Alt}_0 = \left(IV \xrightarrow{\mathbf{W}_\mathsf{Alt}} X_k\right),$$

where

$$\mathbf{W}_\mathsf{H} \| W_k = \mathsf{lpad}(\mathsf{prefix}_i \| M_i), \qquad \mathbf{W}_\mathsf{Alt} \| W_k = \mathsf{lpad}(\mathsf{prefix}_i \| M').$$

The two labeled block sequences are distinct because suffix padding is injective and $M' \neq M_i$.

Write the states of the honest prefix as

$$S_0 = IV, S_1, \ldots, S_\ell = X_k.$$

Outside $\mathsf{BadHonest}$, these states are pairwise distinct: the endpoint of a fresh honest ideal-cipher query can equal neither $IV$ nor its own source by clause (ii), and can equal neither the source nor the endpoint of an earlier distinct record by clause (iii).

Delete the maximal common initial sequence of labeled directed-edge occurrences from $\mathsf{H}_0$ and $\mathsf{Alt}_0$. An edge of $\mathsf{Alt}_0$ must remain. Otherwise $\mathsf{Alt}_0$ would be a strict prefix of $\mathsf{H}_0$ ending at $X_k$, so $X_k$ would occur both internally and at the end of $\mathsf{H}_0$, contrary to the pairwise distinctness just established. Starting with the first remaining alternative edge, traverse at least one remaining occurrence and then follow $\mathsf{Alt}_0$ until it first re-enters

$$\mathcal{S}_{\mathsf{H}} := \{S_0, S_1, \ldots, S_\ell\}.$$

Such a re-entry exists because $\mathsf{Alt}_0$ ends at $X_k = S_\ell$. Let $q = (W; X, Y)$ be the record occurrence closing this first re-entry, and put

$$S^{\mathsf{hit}} := X \boxplus Y \in \mathcal{S}_{\mathsf{H}}.$$

By construction, the occurrence $q$ is not among the edge occurrences deleted with the common initial sequence.

First suppose that $S^{\mathsf{hit}} = IV$. If the record underlying $q$ is first materialized by a forward adversarial ideal-cipher query, clause (i) of $\mathsf{BadFwd}$ occurs; if it is first materialized by a backward adversarial ideal-cipher query, clause (i) of $\mathsf{BadBwd}$ occurs. If an honest ideal-cipher query for a challenge accesses this record, then either the query is an ordinary lookup, causing clause (i) of $\mathsf{BadHonest}$, or the endpoint of the fresh ideal-cipher query is $IV$, causing clause (ii) of $\mathsf{BadHonest}$. A later ordinary lookup of the same record cannot evade this classification. Hence $S^{\mathsf{hit}} = IV$ is impossible.

It remains that $S^{\mathsf{hit}} = S_j$ for some $j \in \{1, \ldots, \ell\}$. Let

$$c_{\mathsf{hit}} = (W_j; S_{j-1}, Y_j)$$

be the unique record occurrence accessed by an honest ideal-cipher query entering $S^{\mathsf{hit}}$ on $\mathsf{H}_0$; uniqueness follows from the pairwise distinctness of the honest states. We first show that the record underlying $q$ is different from the record underlying $c_{\mathsf{hit}}$. Suppose otherwise. Immediately before traversing $q$, $\mathsf{Alt}_0$ is then at the honest source $S_{j-1}$. If $q$ is not the first remaining alternative edge after the deleted common prefix, this source is an earlier re-entry into $\mathcal{S}_{\mathsf{H}}$, contradicting the choice of $q$. If $q$ is the first remaining alternative edge, then $S_{j-1}$ is the state at which the deleted common prefix ends. The deleted prefix cannot already exhaust $\mathsf{H}_0$, since that would give $S_{j-1} = S_\ell$ with $j - 1 < \ell$, contrary to pairwise distinctness. Hence $\mathsf{H}_0$ has a next edge. Pairwise distinctness shows that this next honest edge is precisely $c_{\mathsf{hit}}$; because $q$ and $c_{\mathsf{hit}}$ are assumed to be the same record, the two paths therefore have the same next labeled directed edge. This contradicts the maximality of the deleted common initial sequence. Thus $q$ and $c_{\mathsf{hit}}$ are distinct records.

The global transcript clock is atomic, so these two distinct records have different first-materialization times. In what follows, chronology always refers to these first-materialization times, rather than to a later occurrence of an already existing record on a path.

Suppose first that the record underlying $q$ is materialized before $c_{\mathsf{hit}}$. When the honest ideal-cipher query accessing $c_{\mathsf{hit}}$ is made, either it is an ordinary lookup, causing clause (i) of $\mathsf{BadHonest}$, or the endpoint of this fresh ideal-cipher query $S^{\mathsf{hit}}$ equals the endpoint of the earlier record $q$, causing clause (iii). Thus this order is impossible.

Suppose next that the record underlying $q$ is materialized after $c_{\mathsf{hit}}$. If it is first materialized by an honest ideal-cipher query for a challenge, its endpoint $S^{\mathsf{hit}}$ equals the endpoint of the earlier record $c_{\mathsf{hit}}$, causing clause (iii) of $\mathsf{BadHonest}$. If it is first materialized by an adversarial ideal-cipher query, then the earlier record $c_{\mathsf{hit}}$ is already in $\mathcal{Q}$. Materialization by a forward adversarial ideal-cipher query has endpoint $X \boxplus Y =$

$S^{\mathsf{hit}}$ and causes clause (ii) of $\mathsf{BadFwd}$. Materialization by a backward adversarial ideal-cipher query has endpoint $X \boxplus Y = S^{\mathsf{hit}}$ and causes clause (ii) of $\mathsf{BadBwd}$. This exhausts all possible first-materialization orders and gives the required contradiction.

We have proved the deterministic implication. For the probability statement, first split according to $\mathsf{BadID}_\mu$, and on its complement split successively according to $\mathsf{BadHonest}$, $\mathsf{BadFwd}$, $\mathsf{BadBwd}$, $\mathsf{BadHonSpr}_n$, $\mathsf{PreChSpr}$, and $\mathsf{PostChSpr}$. Since $\mathsf{Bad} = \mathsf{BadID}_\mu \vee \mathsf{BadHonest} \vee \mathsf{BadFwd} \vee \mathsf{BadBwd}$, this disjoint stopped-event decomposition is exactly Eq. (97). □

## D.11 Concluding on $\mathrm{LMS}^{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}$

*Proof (Proof of Theorem 4).* Lemma 1 first gives

$$\Pr[\mathsf{Forge}] \le \Pr\left[\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}} \vee \mathsf{BadFPMSpr}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}}\right].$$

Use the stopped decomposition of Lemma 13. An occurrence of $\mathsf{BadGuess}$ on a trajectory already containing $\mathsf{BadID}_\mu$ or $\mathsf{BadHonest}$ is charged to the first of those two events; otherwise it is charged to the stopped intersection bounded by Eq. (81). Dropping conditions on $\mathsf{BadFwd}$ and $\mathsf{BadBwd}$ only enlarges their probabilities. We therefore obtain

$$\begin{aligned}\Pr[\mathsf{Forge}] \le\ & \Pr[\mathsf{BadID}_\mu] + \Pr[\mathsf{BadHonest} \wedge \neg\mathsf{BadID}_\mu] + \Pr[\mathsf{BadFwd}] + \Pr[\mathsf{BadBwd}] \\ & + \Pr[\mathsf{BadGuess}_{\mathsf{ChopMD}_n^{\mathrm{DM}^{\mathbf{E}}}} \wedge \neg\mathsf{BadID}_\mu \wedge \neg\mathsf{BadHonest}] \\ & + \Pr[\mathsf{BadHonSpr}_n \wedge \neg\mathsf{Bad}] + \Pr[\mathsf{PreChSpr} \wedge \neg\mathsf{Bad}] + \Pr[\mathsf{PostChSpr} \wedge \neg\mathsf{Bad}].\end{aligned}$$

The first term is bounded by Eq. (49). For the second term,

$$\begin{aligned}\Pr[\mathsf{BadHonest} \wedge \neg\mathsf{BadID}_\mu] =\ & \Pr[\neg\mathsf{BadID}_\mu]\ \Pr[\mathsf{BadHonest} \mid \neg\mathsf{BadID}_\mu] \\ \le\ & \Pr[\mathsf{BadHonest} \mid \neg\mathsf{BadID}_\mu],\end{aligned}$$

and the last conditional probability is bounded by Eq. (80). The remaining intersections are bounded directly by Eqs. (81), (50), (51), (53), (82), and (87).

Substituting these bounds, putting $Q = T + \widehat{B}$, and using $\mu \le \widehat{\Gamma}$, $\widehat{B} \ge 1$, $\mathrm{e} < 3$, $2\mathrm{e}^2 < 16$, $2^n - Q \ge 2^{n-1}$, $2^n - Q \ge 2^{n-1}$, $2^s - Q \ge 2^{s-1}$, together with $\min\{x, y+z\} \le$

$y + \min\{x, z\}$ for $x, y, z \geq 0$, $\frac{\widehat{\Gamma}T(T+\widehat{B})^2}{2^{s+2n}} \leq \frac{\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}$ gives

$$
\begin{aligned}
&\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{E},\mathrm{LMS}^{\mathsf{ChopMD}^{\mathsf{DM}^{\mathbf{E}}}_n},\mathscr{A},m,u} \Rightarrow 1\big] \\
&\leq \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{2\mu T + \mu\widehat{B} + \widehat{B}\widehat{\Gamma}}{2^n - T - \widehat{B}} + \frac{2\widehat{B} + 3T + 8\widehat{B}T + 2\widehat{B}^2}{2^s - T - \widehat{B}} \\
&\quad + \min\left\{\frac{\widehat{\Gamma}(T+\widehat{B})}{2^n - T - \widehat{B}},\ \frac{\mu(T+\widehat{B})}{2^n - T - \widehat{B}} + \frac{\mu(T+\widehat{B})^3}{2(2^n - T - \widehat{B})(2^s - T - \widehat{B})}\right\} \\
&\quad + \min\left\{\frac{\widehat{\Gamma}T}{2^n - T - \widehat{B}},\ \frac{\mu T}{2^n - T - \widehat{B}} + \frac{\mu T(T+\widehat{B})^2}{2(2^s - T - \widehat{B})(2^n - T - \widehat{B})}\right\} \\
&\quad + \min\left\{\frac{\widehat{\Gamma}T}{2^n - T - \widehat{B}},\ \frac{1}{2^n} + \frac{16\widehat{\Gamma}T(T+\widehat{B})^2}{(2^n - T - \widehat{B})^2(2^s - T - \widehat{B})} + \frac{24n\widehat{\Gamma}T(T+\widehat{B})}{(2^n - T - \widehat{B})(2^s - T - \widehat{B})} + \frac{8n^2T}{2^s - T - \widehat{B}}\right\} \\
&\quad + \min\left\{\frac{\widehat{\Gamma}Q}{2^n - T - \widehat{B}},\ \frac{1}{2^n} + \frac{3\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2(2^s - T - \widehat{B})^2(2^n - T - \widehat{B})} + \frac{(n+1)\ell_{\max}\widehat{\Gamma}TQ}{(2^s - T - \widehat{B})(2^n - T - \widehat{B})}\right\} \\
&\leq \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{8\mu T + 2\mu\widehat{B} + 4\widehat{B}\widehat{\Gamma}}{2^n} + \frac{16n^2T + 22\widehat{B}T + 8\widehat{B}^2}{2^s} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n},\ \frac{2\mu(T+\widehat{B})^3}{2^{n+s}}\right\} + \min\left\{\frac{2\widehat{\Gamma}T}{2^n},\ \frac{2\mu T(T+\widehat{B})^2}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}T}{2^n},\ \frac{1}{2^n} + \frac{(128+96n)\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\} \\
&\quad + \min\left\{\frac{2\widehat{\Gamma}(T+\widehat{B})}{2^n},\ \frac{1}{2^n} + \frac{12\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})^3}{2^{2s+n}} + \frac{4(n+1)\ell_{\max}\widehat{\Gamma}T(T+\widehat{B})}{2^{s+n}}\right\}.
\end{aligned}
$$

This is Eq. (46) and completes the proof of Theorem 4.

# E Multi-user Security of LMS$^{\mathbf{RO}}$ with Random User Identifiers

This appendix consider the LMS model LMS$^{\mathbf{RO}}$ based on a random oracle $\mathbf{RO} : \{0,1\}^* \to \{0,1\}^n$, and LMS$^{\mathbf{RO}}$ has an additional identifier length parameter $\iota$. We adapt Katz's multi-user unforgeability proof for LMS$^{\mathbf{RO}}$ [33] from the setting of distinct user identifiers to our random-user-identifier setting.

Throughout this section all users employ one common standardized LMS/LM-OTS parameter set $(h, w)$. Put $e := 2^w - 1$, and let $p$ be the corresponding number of Winternitz chains. For every user $\mathsf{i} \in \{1, \ldots, u\}$, every LM-OTS leaf index $q \in \{0, \ldots, 2^h - 1\}$, every chain index $d \in \{0, \ldots, p-1\}$, and every chain position $v \in \{0, \ldots, e\}$, write

$$
\begin{aligned}
&X_0^{\mathsf{i},q,d} := x_d^{\mathsf{i},q}, \\
&X_{v+1}^{\mathsf{i},q,d} := \mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X_v^{\mathsf{i},q,d}\big), \qquad 0 \leq v < e.
\end{aligned} \tag{98}
$$

The corresponding LM-OTS public key is

$$
pk_{\mathrm{ots}}^{\mathsf{i},q} := \mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| X_e^{\mathsf{i},q,0} \| \cdots \| X_e^{\mathsf{i},q,p-1}\big). \tag{99}
$$

We first consider an auxiliary multi-instance LM-OTS forgery experiment. The adversary in this experiment is given the collection of LM-OTS public keys $pk_{\mathrm{ots}}^{\mathsf{i},q}$ and may request a signature on at most one message of its choice for each pair $(\mathsf{i}, q)$. We call the returned signature the reference signature for that pair.

For the reference signature associated with $(\mathrm{i}, q)$, write its message as $M^{\mathrm{i},q}$, its randomizer as $R^{\mathrm{i},q}$, and its message digest as

$$D^{\mathrm{i},q} := \mathbf{RO}\big(I^{\mathrm{i}} \| [q]_{32} \| [8181]_{16} \| R^{\mathrm{i},q} \| M^{\mathrm{i},q}\big). \tag{100}$$

Let $V_0^{\mathrm{i},q}, \ldots, V_{p-1}^{\mathrm{i},q}$ be the Winternitz digits obtained from $D^{\mathrm{i},q} \| \operatorname{checksum}(D^{\mathrm{i},q})$. Thus the $d$-th component of the reference signature is $X^{\mathrm{i},q,d}_{V_d^{\mathrm{i},q}}$.

A forgery specifies an exact target pair $(\mathrm{i}, q)$; equivalently, in the LMS experiment the value $q$ is the leaf index carried by the forged LMS signature. We write $\mathsf{Forge}_{\mathrm{ots}}$ for the event that the adversary outputs a valid fresh LM-OTS forgery for some exact target pair $(\mathrm{i}, q)$.

We normalize the adversary in this experiment as follows. Before outputting a purported LM-OTS forgery, it makes every random-oracle call needed to materialize the forged 8181 digest, the remaining Winternitz-chain steps, and the final 8080 LM-OTS public-key computation. These normalization queries are included in the deterministic query budget $T$ used below.

*Identifier multiplicity.* We use the same identifier-multiplicity event as in the chopped-output ideal-cipher proof; cf. Eq. (48):

$$\mathsf{BadID}_\mu := \left\{ \max_{II \in \{0,1\}^\iota} \big|\{\mathrm{i} \in \{1, \ldots, u\} : I^{\mathrm{i}} = II\}\big| > \mu \right\}. \tag{101}$$

Indeed, for any fixed set of $\mu + 1$ users, their identifiers are all equal with probability $2^{-\iota\mu}$. A union bound over the $\binom{u}{\mu+1}$ possible sets gives

$$\Pr[\mathsf{BadID}_\mu] \le \frac{u^{\mu+1}}{(\mu+1)!\, 2^{\iota\mu}}. \tag{102}$$

In the remainder of the proof we first fix an arbitrary identifier vector $(I^1, \ldots, I^u)$ satisfying $\neg\mathsf{BadID}_\mu$. Hence every concrete identifier value occurs for at most $\mu$ users.

*Syntactic lazy-sampling experiment.* As in Katz's proof, it is convenient to use an equivalent lazy-sampling experiment. All honest Winternitz-chain records and the LM-OTS public-key records are materialized by the single random oracle $\mathbf{RO}$. For a future reference signing query on $(\mathrm{i}, q)$, we may by deferred decisions imagine that $R^{\mathrm{i},q}$ and $D^{\mathrm{i},q}$ have already been sampled; if the exact 8181 input has previously been materialized, $D^{\mathrm{i},q}$ is set to its already defined answer. This is distribution preserving.

When two distinct users have different identifiers, all their formatted LMS prefixes are different. If they have the same identifier, corresponding coordinates may use the same formatted prefix. This is the only source of prefix multiplicity and is the reason for conditioning on $\neg\mathsf{BadID}_\mu$.

Whenever two complete random-oracle inputs have distinct formatted prefixes, they are distinct inputs even if their remaining fields coincide. The random-function values at such distinct inputs are independent uniform outputs under lazy sampling. An accidental equality between these outputs is nevertheless allowed: it is not deleted from either output's support. When such an equality is relevant to a corresponding pair of same-identifier honest coordinates, it is charged to $\mathsf{HonColl}_{\mathbf{RO}}$ below.

*Exposure-relevant honest equality.* For distinct users $\mathfrak{i} \neq \mathfrak{i}'$ with $I^{\mathfrak{i}} = I^{\mathfrak{i}'}$, call the following coordinates corresponding when their local labels agree:

(i) the Winternitz states $X_v^{\mathfrak{i},q,d}$ and $X_v^{\mathfrak{i}',q,d}$ for $0 \le q < 2^h$, $0 \le d < p$, and $0 \le v \le e$;
(ii) the reference randomizers $R^{\mathfrak{i},q}$ and $R^{\mathfrak{i}',q}$;
(iii) the reference digests $D^{\mathfrak{i},q}$ and $D^{\mathfrak{i}',q}$;
(iv) the LM-OTS public keys $pk_{\mathrm{ots}}^{\mathfrak{i},q}$ and $pk_{\mathrm{ots}}^{\mathfrak{i}',q}$.

Let

$$H_{\mathrm{ots}} := 2^h\big(p(e+1)+3\big), \tag{103}$$

which is a deterministic upper bound on the number of corresponding labels for one unordered user pair in this auxiliary experiment.

Define $\mathsf{HonColl}_{\mathbf{RO}}$ to occur if, before any adversarial collision event defined below has occurred, two same-identifier users acquire equality at a pair of corresponding coordinates for the first time. Comparisons are made in topological order of the two honest dependency DAGs. At a derived random-oracle coordinate, if the two complete oracle inputs are equal, we trace backwards to the first equal parent coordinate; hence the first equality at a derived coordinate is always an equality of outputs at distinct complete oracle inputs.

**Lemma 14 (Same-identifier honest equality).** *It holds*

$$\Pr[\mathsf{HonColl}_{\mathbf{RO}} \wedge \neg \mathsf{BadID}_{\mu}] \le \binom{u}{2} \frac{H_{\mathrm{ots}}}{2^{\iota+n}}. \tag{104}$$

*Proof.* Fix an unordered pair $\{\mathfrak{i}, \mathfrak{i}'\}$. Their identifiers agree with probability $2^{-\iota}$. Condition on such an equality and expose the two corresponding honest dependency DAGs in topological order, stopping at their first corresponding equality or at the first adversarial collision event.

At a chain seed or a reference randomizer, the later sampled direct coin hits the already fixed peer value with probability $2^{-n}$. At every other corresponding coordinate, before the first equality the two complete random-oracle inputs are distinct; otherwise an equal parent coordinate would have occurred earlier. Nor can the later input already have been materialized at the corresponding coordinate of a third same-identifier user: equality of those complete inputs traces back to an earlier equality of corresponding parent coordinates and hence to an earlier occurrence of $\mathsf{HonColl}_{\mathbf{RO}}$. Honest records belonging to a different identifier, local coordinate, or domain have a different formatted prefix. If an adversarial record is the first materializer, the stopped experiment has already terminated in an adversarial collision event and this trial is not charged to $\mathsf{HonColl}_{\mathbf{RO}}$. In every remaining case the later honest input therefore opens a fresh random-oracle record; its uniform answer hits the already fixed peer value with probability $2^{-n}$.

A union bound over at most $H_{\mathrm{ots}}$ corresponding labels gives $H_{\mathrm{ots}}/2^n$ for this pair conditioned on identifier equality. Averaging over the identifier equality and summing over the $\binom{u}{2}$ unordered pairs proves the lemma.

*Katz collision events.* Fix a target pair $(\mathsf{i}, q)$. We use the following events, which are the direct analogues of Katz's $\mathsf{Coll}_1$, $\mathsf{Coll}_2^*$ and $\mathsf{Coll}_3^*$ events, but with the exact target value $q$ fixed.

The counters introduced below are transcript-dependent random variables, whereas $T$ is a deterministic upper bound fixed before the experiment on the total number of direct adversarial random-oracle calls, including the normalization calls above. All bounds involving a local counter are consequently stated with its expectation. Equivalently, one may stop immediately before the first collision event, charge each eligible query by its conditional hit probability given the pre-query history, and then average these conditional bounds. The number of charged queries is at most the corresponding full-transcript counter.

- $\mathsf{Coll}_{1,\mathsf{i},q}$ occurs if the adversary queries

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| Y_0 \| \cdots \| Y_{p-1}\big)$$

  with $(Y_0, \ldots, Y_{p-1}) \neq (X_e^{\mathsf{i},q,0}, \ldots, X_e^{\mathsf{i},q,p-1})$ and receives the answer $pk_{\mathrm{ots}}^{\mathsf{i},q}$. Let $T_{1,\mathsf{i},q}$ be the random number of direct adversarial queries of the form

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [8080]_{16} \| \star\big).$$

- $\mathsf{Coll}_{2,\mathsf{i},q}$ occurs if, before the reference signing query for $(\mathsf{i}, q)$, the adversary queries an input of the form

$$I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \| R^{\mathsf{i},q} \| \star\,.$$

  Let $T_{2,\mathsf{i},q}$ be the random number of direct adversarial queries of the form

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \| \star\big).$$

- $\mathsf{Coll}_{2,\mathsf{i},q}^*$ occurs if $\mathsf{Coll}_{2,\mathsf{i},q}$ occurs, or if either of the following happens:
  (i) before the reference signing query, a direct adversarial query under the prefix $I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16}$ receives the answer $D^{\mathsf{i},q}$;
  (ii) after the reference signature on $M^{\mathsf{i},q}$ has been returned, a direct adversarial query

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [8181]_{16} \| R \| M\big), \qquad M \neq M^{\mathsf{i},q},$$

  receives the answer $D^{\mathsf{i},q}$.
- For $d \in \{0, \ldots, p-1\}$ and $v \in \{0, \ldots, e-1\}$, $\mathsf{Coll}_{3,\mathsf{i},q,d,v}$ occurs if the adversary queries

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X_v^{\mathsf{i},q,d}\big)$$

  either before the reference signing query for $(\mathsf{i}, q)$, or after that query while $v < V_d^{\mathsf{i},q}$. Let $T_{3,\mathsf{i},q,d,v}$ be the random number of direct adversarial queries of the form

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| \star\big),$$

  and put $T_{3,\mathsf{i},q,d,-1} := 0$.
- $\mathsf{Coll}_{3,\mathsf{i},q,d,v}^*$ occurs if either $\mathsf{Coll}_{3,\mathsf{i},q,d,v}$ occurs, or the adversary queries

$$\mathbf{RO}\big(I^{\mathsf{i}} \| [q]_{32} \| [d]_{16} \| [v]_8 \| X\big), \qquad X \neq X_v^{\mathsf{i},q,d},$$

  and receives the answer $X_{v+1}^{\mathsf{i},q,d}$.

Put

$$\mathsf{AdvColl}_{\mathbf{RO}} := \bigvee_{\mathrm{i},q} \left( \mathsf{Coll}_{1,\mathrm{i},q} \vee \mathsf{Coll}^*_{2,\mathrm{i},q} \vee \bigvee_{d,v} \mathsf{Coll}^*_{3,\mathrm{i},q,d,v} \right). \tag{105}$$

For the probability analysis we stop at the first occurrence of $\mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}}$.

The following estimates adapt Katz's Lemmas 2–7 [33]. Relative to Katz's distinct-identifier setting, we additionally stop at $\mathsf{HonColl}_{\mathbf{RO}}$: if a relevant record was already materialized by a corresponding honest computation of another user with the same identifier, equality with the target is charged to that event. The query counters below may depend on the transcript, so we condition immediately before each fresh answer, multiply its conditional charge by the corresponding stopped-query indicator, and then take expectations.

**Lemma 15 (Forgery implies a stopped collision event).** *Conditioned on* $\neg\mathsf{BadID}_{\mu}$,

$$\mathsf{Forge}_{\mathrm{ots}} \subseteq \mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}}. \tag{106}$$

*Proof.* On the complement of $\mathsf{HonColl}_{\mathbf{RO}}$, the proof of Katz's Lemma 1 [33] applies to the exact target pair $(\mathrm{i}, q)$. Briefly, absence of $\mathsf{Coll}_{1,\mathrm{i},q}$ forces the terminal values reconstructed from the forgery to equal the honest Winternitz terminals. Absence of $\mathsf{Coll}^*_{2,\mathrm{i},q}$ forces the forged digest to differ from the reference digest, so the checksum property gives some $d$ with $V_d < V_d^{\mathrm{i},q}$. The first contact of that forged chain with the honest target chain then gives $\mathsf{Coll}^*_{3,\mathrm{i},q,d,v}$ for some $v$. The only additional possibility created by repeated identifiers is that the relevant record was materialized by the corresponding honest computation of another same-identifier user, which is precisely $\mathsf{HonColl}_{\mathbf{RO}}$. This proves the inclusion.

*Probability of* $\mathsf{Coll}_1$.

**Lemma 16.** *For every fixed good identifier vector and every* $(\mathrm{i}, q)$,

$$\Pr\left[\mathsf{Coll}_{1,\mathrm{i},q} \text{ is the first stopped collision event}\right] \le \frac{\mathbb{E}[T_{1,\mathrm{i},q}]}{2^n}. \tag{107}$$

*Proof.* This is Katz's Lemma 2 [33] under the preceding stopped convention. Before the first stopped event, an earlier adversarial materialization would already have caused the same event, while a corresponding same-identifier honest record is charged to $\mathsf{HonColl}_{\mathbf{RO}}$. Every remaining counted query therefore opens a fresh record and hits $pk_{\mathrm{ots}}^{\mathrm{i},q}$ with conditional probability $2^{-n}$. Summing the stopped-query indicators and taking expectations gives the claim.

*Probability of* $\mathsf{Coll}^*_2$.

**Lemma 17.** *For every fixed good identifier vector and every* $(\mathrm{i}, q)$,

$$\Pr\left[\mathsf{Coll}_{2,\mathrm{i},q} \text{ is the first stopped collision event}\right] \le \frac{\mathbb{E}[T_{2,\mathrm{i},q}]}{2^n}. \tag{108}$$

*Proof.* This adapts Katz's Lemma 3 [33]. Before the reference signature is issued, $R^{\mathrm{i},q}$ is hidden and uniform; each counted query fixes one candidate for it. A corresponding same-identifier equality is charged to $\mathsf{HonColl}_{\mathbf{RO}}$, and the stopped-indicator calculation for all remaining candidates gives $\mathbb{E}[T_{2,\mathrm{i},q}]/2^n$.

**Lemma 18.** *For every fixed good identifier vector and every* $(\mathrm{i}, q)$*,*

$$\Pr\left[\mathsf{Coll}^{*}_{2,\mathrm{i},q} \text{ is the first stopped collision event}\right] \le \frac{2\mathbb{E}[T_{2,\mathrm{i},q}]}{2^n}. \tag{109}$$

*Proof.* As in Katz's Lemma 4 [33], Lemma 17 bounds the first clause. For the additional clauses, an earlier adversarial materialization would already have caused the same stopped event, while a corresponding same-identifier record is charged to $\mathsf{HonColl}_{\mathbf{RO}}$. Otherwise the fresh answer hits the fixed, or independently hidden, value $D^{\mathrm{i},q}$ with conditional probability $2^{-n}$. The same stopped-indicator calculation supplies the second $\mathbb{E}[T_{2,\mathrm{i},q}]/2^n$ contribution.

*Probability of* $\mathsf{Coll}^{*}_{3}$*.*

**Lemma 19.** *For every fixed good identifier vector and every* $(\mathrm{i}, q, d, v)$*, if* $T + \mu - 1 < 2^{n-1}$*, then*

$$\Pr\left[\mathsf{Coll}_{3,\mathrm{i},q,d,v} \text{ is the first stopped collision event}\right] \le \frac{2\mathbb{E}[T_{3,\mathrm{i},q,d,v}]}{2^n}. \tag{110}$$

*Proof.* This is the random-identifier refinement of Katz's Lemma 5 [33]. Before the first stopped collision event, the hidden state $X_v^{\mathrm{i},q,d}$ has not been disclosed. Condition on the complete pre-query history of the current adversarial call. Earlier adversarial queries under the preceding chain prefix can exclude at most $T$ candidate values: if one of their answers were $X_v^{\mathrm{i},q,d}$, an earlier $\mathsf{Coll}^{*}_{3,\mathrm{i},q,d,v-1}$ would have occurred. On $\neg\mathsf{BadID}_{\mu}$, the corresponding states of the at most $\mu-1$ other users with the same identifier exclude at most $\mu - 1$ further values, because equality with one of them would be $\mathsf{HonColl}_{\mathbf{RO}}$.

Outputs assigned at distinct complete random-oracle inputs remain independent uniform values. In particular, an accidental equality between outputs having distinct formatted prefixes is allowed and is not removed from the support merely because the prefixes differ. Only the explicit stopped inequalities just described exclude candidates. Consequently the conditional point probability of $X_v^{\mathrm{i},q,d}$ is at most

$$\frac{1}{2^n - T - (\mu - 1)} < \frac{2}{2^n}.$$

For each of the at most $T$ adversarial calls, multiply this bound by the pre-query-history-measurable indicator that the call is counted by $T_{3,\mathrm{i},q,d,v}$ and that no stopped event occurred earlier. Summing and taking expectations proves Eq. (110).

**Lemma 20.** *For every fixed good identifier vector and every* $(\mathrm{i}, q, d, v)$*, if* $T + \mu - 1 < 2^{n-1}$*, then*

$$\Pr\left[\mathsf{Coll}^{*}_{3,\mathrm{i},q,d,v} \text{ is the first stopped collision event}\right] \le \frac{3\mathbb{E}[T_{3,\mathrm{i},q,d,v}]}{2^n}. \tag{111}$$

*Proof.* Combine Lemma 19 with the fresh-successor argument of Katz's Lemma 6 [33]. For its additional clause, an earlier adversarial materialization would already have caused that event, while a corresponding same-identifier honest record is charged to $\mathsf{HonColl}_{\mathbf{RO}}$. Otherwise a fresh answer hits $X_{v+1}^{\mathrm{i},q,d}$ with conditional probability $2^{-n}$. Summing the stopped indicators adds $\mathbb{E}[T_{3,\mathrm{i},q,d,v}]/2^n$ to the preceding bound.

**Lemma 21.** *For every fixed good identifier vector and every* $(\mathrm{i}, q, d)$*, if* $T + \mu - 1 < 2^{n-1}$*, then*

$$\Pr\left[\bigvee_{v=0}^{e-1} \mathsf{Coll}^{*}_{3,\mathrm{i},q,d,v} \text{ is the first stopped collision event}\right] \le 3\sum_{v=0}^{e-1} \frac{\mathbb{E}[T_{3,\mathrm{i},q,d,v}]}{2^n}. \quad (112)$$

*Proof.* Apply a first-occurrence union bound over $v$ and Lemma 20; its deterministic hypothesis holds simultaneously for every $v$.

*Multiplicity-aware query accounting.*

**Lemma 22.** *For every fixed identifier vector satisfying* $\neg\mathsf{BadID}_\mu$*, the following inequality holds on every execution trace:*

$$\sum_{\mathrm{i}=1}^{u}\sum_{q=0}^{2^h-1} T_{1,\mathrm{i},q} + \sum_{\mathrm{i}=1}^{u}\sum_{q=0}^{2^h-1} T_{2,\mathrm{i},q} + \sum_{\mathrm{i}=1}^{u}\sum_{q=0}^{2^h-1}\sum_{d=0}^{p-1}\sum_{v=0}^{e-1} T_{3,\mathrm{i},q,d,v} \le \mu T, \quad (113)$$

*Here* $T$ *is the deterministic upper bound on the total number of direct adversarial random-oracle calls. In particular, taking expectations in Eq.* (113) *gives*

$$\sum_{\mathrm{i},q} \mathbb{E}[T_{1,\mathrm{i},q}] + \sum_{\mathrm{i},q} \mathbb{E}[T_{2,\mathrm{i},q}] + \sum_{\mathrm{i},q,d,v} \mathbb{E}[T_{3,\mathrm{i},q,d,v}] \le \mu T. \quad (114)$$

*Proof.* The 8080, 8181, and Winternitz-chain formatted domains are disjoint. Parse one direct adversarial query according to its concrete identifier value $II$ and its explicit local coordinates $q$ and, for a chain query, $d, v$. On $\neg\mathsf{BadID}_\mu$, at most $\mu$ user indices satisfy $I^{\mathrm{i}} = II$. For each such user the remaining local coordinates identify at most one counter on the left-hand side. Hence one adversarial query increments at most $\mu$ counters. Summing over the $T$ queries proves the pathwise inequality; monotonicity and linearity of expectation give its expected version.

**Theorem 5 (Multiplicity-aware Katz bound for the LM-OTS layer).** *For every integer* $\mu \ge 1$*, a normalized auxiliary adversary making at most the deterministic budget* $T$ *of direct random-oracle calls satisfies*

$$\Pr[\mathsf{Forge}_{\mathrm{ots}}] \le \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{3\mu T}{2^n} + \binom{u}{2}\frac{H_{\mathrm{ots}}}{2^{\iota+n}}. \quad (115)$$

*Proof.* Decompose according to $\mathsf{BadID}_\mu$. On $\neg\mathsf{BadID}_\mu$, Lemma 15 gives

$$\mathsf{Forge}_{\mathrm{ots}} \subseteq \mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}}.$$

Therefore

$$\begin{aligned}\Pr[\mathsf{Forge}_{\mathrm{ots}}] \le\ & \Pr[\mathsf{BadID}_\mu] + \Pr[\mathsf{HonColl}_{\mathbf{RO}} \wedge \neg\mathsf{BadID}_\mu] \\ & + \Pr[\mathsf{AdvColl}_{\mathbf{RO}} \wedge \neg\mathsf{BadID}_\mu \wedge \neg\mathsf{HonColl}_{\mathbf{RO}}]. \end{aligned} \quad (116)$$

The first term is bounded by Eq. (102), and the second by Lemma 14. For the third term, fix an arbitrary good identifier vector and use the disjoint first-stopped-event decomposition. Lemmas 16, 18, and 21 yield

$$\begin{aligned}
&\Pr[\mathsf{AdvColl}_{\mathbf{RO}} \text{ occurs before } \mathsf{HonColl}_{\mathbf{RO}}] \\
&\le \sum_{\mathsf{i},q} \frac{\mathbb{E}[T_{1,\mathsf{i},q}]}{2^n} + 2\sum_{\mathsf{i},q} \frac{\mathbb{E}[T_{2,\mathsf{i},q}]}{2^n} + 3\sum_{\mathsf{i},q,d,v} \frac{\mathbb{E}[T_{3,\mathsf{i},q,d,v}]}{2^n} \\
&\le 3\left(\sum_{\mathsf{i},q} \mathbb{E}[T_{1,\mathsf{i},q}] + \sum_{\mathsf{i},q} \mathbb{E}[T_{2,\mathsf{i},q}] + \sum_{\mathsf{i},q,d,v} \mathbb{E}[T_{3,\mathsf{i},q,d,v}]\right) 2^{-n} \\
&\le \frac{3\mu T}{2^n},
\end{aligned}$$

where the final inequality is Eq. (114). The same estimate holds for every good identifier vector and can therefore be averaged over them.

If $T = 0$, the normalized adversary cannot complete the verification computations specified above without exceeding its budget, so the forgery probability is zero. We may therefore assume $T \ge 1$.

It remains only to separate the trivial range from the range used in Lemma 21. If $T + \mu - 1 \ge 2^{n-1}$, then, since $(T-1)(\mu-1) \ge 0$, we have $\mu T \ge T + \mu - 1$, and hence $3\mu T/2^n \ge 3/2$. In this case the claimed theorem is already trivial. Otherwise, the one deterministic hypothesis of Lemma 21 holds for every relevant triple. Combining the three bounds proves Eq. (115).

*From the LM-OTS layer to complete LMS.* We now adapt Sect. 3.4 of Katz [33]. The only additional issue in our setting is that two users can have the same identifier. A random-oracle query in the 8282 or 8383 domain can then be compatible with corresponding Merkle nodes of several users. On $\neg\mathsf{BadID}_\mu$, their number is at most $\mu$; accidental equalities between their corresponding honest nodes are handled separately below.

Throughout this full-LMS part, $T$ denotes the direct-call budget of the original normalized LMS adversary, including the calls used to materialize verification of its final output. Honest key-generation records and the proof-only materialization of the honest trees are not adversarial calls and are not included in this budget.

For every user $\mathsf{i}$, define the complete honest LMS tree by

$$T^{\mathsf{i}}[\rho] := \mathbf{RO}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8282]_{16} \| pk_{\mathrm{ots}}^{\mathsf{i},\rho-2^h}\big), \qquad 2^h \le \rho < 2^{h+1}, \tag{117}$$

$$T^{\mathsf{i}}[\rho] := \mathbf{RO}\big(I^{\mathsf{i}} \| [\rho]_{32} \| [8383]_{16} \| T^{\mathsf{i}}[2\rho] \| T^{\mathsf{i}}[2\rho+1]\big), \qquad 1 \le \rho < 2^h. \tag{118}$$

The public root of user $\mathsf{i}$ is $T^{\mathsf{i}}[1]$. Put

$$H_{\mathrm{tree}} := 2^{h+1} - 1, \qquad H_{\mathrm{LMS}} := H_{\mathrm{ots}} + H_{\mathrm{tree}} = 2^h\big(p(e+1)+5\big) - 1. \tag{119}$$

Thus $H_{\mathrm{tree}}$ is the number of 8282 and 8383 outputs in one complete honest tree, and $H_{\mathrm{LMS}}$ bounds all corresponding honest coordinates used in this appendix for one unordered pair of users.

*Corresponding honest Merkle nodes.* Define $\mathsf{HonTreeColl}_{\mathbf{RO}}$ to occur if, before $\mathsf{HonColl}_{\mathbf{RO}}$ or any adversarial collision event considered below, two distinct users $\mathsf{i} \neq \mathsf{i}'$ with $I^{\mathsf{i}} = I^{\mathsf{i}'}$ acquire

$$T^{\mathsf{i}}[\rho] = T^{\mathsf{i}'}[\rho]$$

for the first time at a corresponding tree position $\rho$. Tree positions are exposed from the leaves toward the root. If the complete inputs at a corresponding 8282 node agree, then the corresponding LM-OTS public keys were already equal and $\mathsf{HonColl}_{\mathbf{RO}}$ occurred earlier. If the complete inputs at a corresponding 8383 node agree, then both corresponding children were already equal and a lower occurrence of $\mathsf{HonTreeColl}_{\mathbf{RO}}$ occurred earlier. Consequently, at the first stopped tree equality the two complete random-oracle inputs are distinct.

**Lemma 23 (Same-identifier honest Merkle equality).** *It holds*

$$\Pr\left[\mathsf{HonTreeColl}_{\mathbf{RO}} \wedge \neg\mathsf{BadID}_{\mu} \wedge \neg\mathsf{HonColl}_{\mathbf{RO}}\right] \leq \binom{u}{2}\frac{H_{\mathrm{tree}}}{2^{\iota+n}}. \tag{120}$$

*Proof.* Repeat the proof of Lemma 14, exposing the two Merkle DAGs in bottom-up order. At the first charged 8282 equality, equal complete inputs would imply an earlier equality of the corresponding LM-OTS public keys; at an 8383 node they would imply an earlier equality of corresponding children. Thus the later input is fresh on the stopped good branch and hits its peer value with probability $2^{-n}$. There are at most $H_{\mathrm{tree}}$ charged positions per equal-identifier user pair. Averaging over identifier equality and summing over unordered user pairs gives the claimed bound.

*Adversarial Merkle collisions.* For every user $\mathsf{i}$ and tree position $\rho$, define $\mathsf{TreeColl}_{\mathsf{i},\rho}$ as follows.

- If $2^h \leq \rho < 2^{h+1}$, it occurs when a direct adversarial query

$$\mathbf{RO}\big(I^{\mathsf{i}}\|[\rho]_{32}\|[8282]_{16}\|PK\big), \qquad PK \neq pk_{\mathrm{ots}}^{\mathsf{i},\rho-2^h},$$

  receives the response $T^{\mathsf{i}}[\rho]$.
- If $1 \leq \rho < 2^h$, it occurs when a direct adversarial query

$$\mathbf{RO}\big(I^{\mathsf{i}}\|[\rho]_{32}\|[8383]_{16}\|L\|R\big), \qquad (L,R) \neq \big(T^{\mathsf{i}}[2\rho], T^{\mathsf{i}}[2\rho+1]\big),$$

  receives the response $T^{\mathsf{i}}[\rho]$.

Let $T_{\mathrm{tree},\mathsf{i},\rho}$ be the random number of direct adversarial queries having the corresponding fixed prefix $I^{\mathsf{i}}\|[\rho]_{32}\|[8282]_{16}$ or $I^{\mathsf{i}}\|[\rho]_{32}\|[8383]_{16}$, respectively, and put

$$\mathsf{AdvTreeColl}_{\mathbf{RO}} := \bigvee_{\mathsf{i}=1}^{u}\bigvee_{\rho=1}^{2^{h+1}-1}\mathsf{TreeColl}_{\mathsf{i},\rho}. \tag{121}$$

The direct queries here include the calls used to materialize verification of the adversary's final LMS output.

For the remainder of the full-LMS analysis, put

$$\mathsf{Stop}_{\mathrm{LMS}} := \mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{HonTreeColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}} \vee \mathsf{AdvTreeColl}_{\mathbf{RO}}.$$

The stopped experiment halts at the first time at which $\mathsf{Stop}_{\mathrm{LMS}}$ occurs. The earlier LM-OTS bounds remain valid under this refined stopping rule: relative to the LM-OTS analysis, the two Merkle events merely stop the transcript earlier, and therefore can only remove later occurrences that would otherwise have been charged to an LM-OTS collision event.

**Lemma 24 (Probability of an adversarial Merkle collision).** *For every fixed good identifier vector and every* $(\mathsf{i}, \rho)$*,*

$$\Pr\big[\mathsf{TreeColl}_{\mathsf{i},\rho} \text{ is the first stopped collision event}\big] \le \frac{\mathbb{E}[T_{\mathrm{tree},\mathsf{i},\rho}]}{2^n}. \tag{122}$$

*Proof.* This is the stopped, random-identifier analogue of the collision estimate in Katz's Lemma 9 [33]. For a counted query, an earlier adversarial materialization would already have caused the same event, while materialization at the corresponding node of another same-identifier user is charged to $\mathsf{HonTreeColl}_{\mathbf{RO}}$. Every remaining query opens a fresh record at an input distinct from the honest target input and hits $T^{\mathsf{i}}[\rho]$ with conditional probability $2^{-n}$. Summing the stopped-query indicators and taking expectations proves the lemma.

The multiplicity accounting from Lemma 22 extends without loss to the tree domains.

**Lemma 25 (Combined multiplicity-aware query accounting).** *For every fixed identifier vector satisfying* $\neg\mathsf{BadID}_\mu$*, every execution trace satisfies*

$$\sum_{\mathsf{i},q} T_{1,\mathsf{i},q} + \sum_{\mathsf{i},q} T_{2,\mathsf{i},q} + \sum_{\mathsf{i},q,d,v} T_{3,\mathsf{i},q,d,v} + \sum_{\mathsf{i}=1}^{u} \sum_{\rho=1}^{2^{h+1}-1} T_{\mathrm{tree},\mathsf{i},\rho} \le \mu T. \tag{123}$$

*Proof.* The proof is identical to that of Lemma 22. Domain separation places each direct query in at most one of the Winternitz-chain, 8080, 8181, 8282, and 8383 domains, and its identifier matches at most $\mu$ users on $\neg\mathsf{BadID}_\mu$.

*The full-LMS reduction.* Let $\mathsf{Forge}_{\mathrm{LMS}}$ denote the event that the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{RO},\mathrm{LMS}^{\mathbf{RO}},\mathscr{A},m,u}$ outputs 1, and let $\mathsf{Forge}^{=}_{\mathrm{LMS}}$ be its subevent on which the LM-OTS public key recovered from the forged signature equals the honest public key at the claimed leaf. Normalize the adversary so that, before output, it has made every random-oracle call needed to evaluate the forged LM-OTS signature, its 8282 leaf, and all 8383 nodes on its authentication path. All these direct calls are included in $T$.

The simulation underlying Katz's reduction remains distribution preserving when identifiers repeat. Extend the syntactic auxiliary experiment so that, after generating all $pk^{\mathsf{i},q}_{\mathrm{ots}}$, its honest initialization materializes Eqs. (117) and (118) and gives the resulting roots to the LMS adversary. These are honest table records, rather than direct adversarial queries counted by $T$. A signing query for $(\mathsf{i}, q)$ is answered with the corresponding LM-OTS signing oracle and the authentication path already present in that tree. Direct random-oracle queries are forwarded. The honest tree-construction records lie in the domain-separated 8282 and 8383 domains and therefore do not increase any LM-OTS counter. If a leaf has not previously been signed when it is selected by the forgery, the auxiliary adversary may make no signing query for that LM-OTS instance; equivalently, after seeing the forged message it may request a reference signature on a distinct dummy message before outputting its LM-OTS forgery. This dummy signature

is not returned to the LMS adversary and therefore does not alter its view. In either formulation the simulation establishes

$$\mathsf{Forge}^{=}_{\mathrm{LMS}} \subseteq \mathsf{Forge}_{\mathrm{ots}}.$$

**Lemma 26 (A full LMS forgery implies a stopped collision).** *Conditioned on* $\neg\mathsf{BadID}_{\mu}$,

$$\mathsf{Forge}_{\mathrm{LMS}} \subseteq \mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{HonTreeColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}} \vee \mathsf{AdvTreeColl}_{\mathbf{RO}}. \tag{124}$$

*Proof.* Let $pk^{\star}$ be the LM-OTS public key recovered from a valid fresh forgery for user $\mathsf{i}$ and leaf $q$. If $pk^{\star} = pk^{\mathsf{i},q}_{\mathrm{ots}}$, then Lemma 15 gives $\mathsf{HonColl}_{\mathbf{RO}} \vee \mathsf{AdvColl}_{\mathbf{RO}}$. Otherwise, apply the first-convergence argument of Katz's Lemma 9 [33] to the normalized authentication path. Either its 8282 leaf value already equals the honest leaf value, or the forged and honest paths first agree at a later 8383 node. In both cases a payload different from the honest payload maps to the honest node value, so $\mathsf{TreeColl}_{\mathsf{i},\rho}$ occurs for some $\rho$. This proves the inclusion.

**Theorem 6 (Multi-user unforgeability of LMS$^{\mathbf{RO}}$ with random identifiers).** *For every integer $\mu \geq 1$, every normalized adversary making at most the deterministic budget $T$ of direct random-oracle calls satisfies*

$$\Pr\bigl[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{RO},\mathrm{LMS}^{\mathbf{RO}},\mathscr{A},m,u}\bigr] \leq \frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{3\mu T}{2^{n}} + \binom{u}{2}\frac{H_{\mathrm{LMS}}}{2^{\iota+n}}. \tag{125}$$

*Here $H_{\mathrm{LMS}} = 2^{h}\bigl(p(e+1)+5\bigr)-1$. The allowed parameter ranges $h \leq 31$, $p \leq 2^{15}$, and $e = 2^{w}-1 \leq 2^{8}-1$ therefore give*

$$H_{\mathrm{LMS}} \leq 2^{31}\bigl(2^{15}\cdot 2^{8}+5\bigr)-1 = 2^{54}+5\cdot 2^{31}-1 < 2^{55}.$$

*Proof.* By Lemma 26 and the first-event decomposition associated with $\mathsf{Stop}_{\mathrm{LMS}}$,

$$\begin{aligned}\Pr\bigl[\mathsf{Forge}_{\mathrm{LMS}}\bigr] \leq{}& \Pr\bigl[\mathsf{BadID}_{\mu}\bigr] + \Pr\bigl[\mathsf{HonColl}_{\mathbf{RO}} \wedge \neg\mathsf{BadID}_{\mu}\bigr] \\ &+ \Pr\bigl[\mathsf{HonTreeColl}_{\mathbf{RO}} \wedge \neg\mathsf{BadID}_{\mu} \wedge \neg\mathsf{HonColl}_{\mathbf{RO}}\bigr] \\ &+ \Pr\bigl[(\mathsf{AdvColl}_{\mathbf{RO}} \vee \mathsf{AdvTreeColl}_{\mathbf{RO}}) \wedge \neg\mathsf{BadID}_{\mu} \\ &\qquad\qquad \wedge \neg\mathsf{HonColl}_{\mathbf{RO}} \wedge \neg\mathsf{HonTreeColl}_{\mathbf{RO}}\bigr].\end{aligned}$$

The identifier term is bounded by Eq. (102). Lemmas 14 and 23, together with Eq. (119), bound the two honest-equality contributions by

$$\binom{u}{2}\frac{H_{\mathrm{LMS}}}{2^{\iota+n}}.$$

Fix an arbitrary good identifier vector. If $T+\mu-1 < 2^{n-1}$, the stopped bounds in Lemmas 16, 18, 21, and 24 give

$$\begin{aligned}&\Pr\bigl[\mathsf{AdvColl}_{\mathbf{RO}} \vee \mathsf{AdvTreeColl}_{\mathbf{RO}} \text{ occurs before an honest stopped event}\bigr] \\ &\leq \frac{1}{2^{n}}\Biggl(\sum_{\mathsf{i},q}\mathbb{E}[T_{1,\mathsf{i},q}] + 2\sum_{\mathsf{i},q}\mathbb{E}[T_{2,\mathsf{i},q}] + 3\sum_{\mathsf{i},q,d,v}\mathbb{E}[T_{3,\mathsf{i},q,d,v}] + \sum_{\mathsf{i},\rho}\mathbb{E}[T_{\mathrm{tree},\mathsf{i},\rho}]\Biggr) \\ &\leq \frac{3\mu T}{2^{n}},\end{aligned}$$

where the last step follows by taking expectations in Eq. (123). This estimate is the same for every good identifier vector and hence gives the corresponding conditional bound. □

### E.1 Concrete Corollaries on LMS_SHAKE_M32_H$h$ and LMS_SHAKE_M24_H$h$

LMS_SHAKE_M32_H$h$ and LMS_SHAKE_M24_H$h$ instantiate the VIL random oracle in LMS with SHAKE256/256 and SHAKE256/192 respectively, both of which follow the permutation-based sponge construction [26] and are *indifferentiable* from VIL random oracles [5,38] in the random permutation model. The concrete security of LMS_SHAKE_M32_H$h$ and LMS_SHAKE_M24_H$h$ can be derived by combining Theorem 6 and the indifferentiability bounds in [5,38].

Concretely, let $\pi$ be a random permutation, and let $\mathsf{Sponge}[r, c, n]^{\pi}$ be a sponge construction based on $\pi$ and with rate $r$, capacity $c$ and output length $n$. The concrete meanings of $r$ and $c$ are not relevant here: it suffices to note that $c$ determines its security and $n$ determines its output length. Furthermore, assume that in the experiment $\mathbf{Exp}^{\text{m-unforge}}_{\pi,\text{LMS}^{\mathsf{Sponge}[r,c,n]^{\pi}},\mathscr{A},m,u}$ for the adversary $\mathscr{A}$, the random permutation $\pi$ receives at most $Q$ queries.

With these notations, Lefevre [38, page 230] showed that there exists an indifferentiability simulator $\mathcal{S}^{\mathbf{RO}}$ making at most $Q$ queries to $\mathbf{RO}$, such that we can replace the pair of primitives $(\pi, \mathsf{Sponge}[r, c, n]^{\pi})$ with $(\mathcal{S}^{\mathbf{RO}}, \mathbf{RO})$, and this replacement introduces a statistical distance of $\mathcal{O}\Big(\frac{Q^2}{2^c}\Big)$. We can then view the combination of $\mathscr{A}$ and $\mathcal{S}^{\mathbf{RO}}$ as a single adversary making at most $Q$ queries to $\mathbf{RO}$, and its success probability is bounded by Theorem 6. Gathering Eq. (125) and the gap $\mathcal{O}\Big(\frac{Q^2}{2^c}\Big)$ yields

$$\Pr\big[\mathbf{Exp}^{\text{m-unforge}}_{\pi,\text{LMS}^{\mathsf{Sponge}[r,c,n]^{\pi}},\mathscr{A},m,u} \Rightarrow 1\big] \leq O\Big(\frac{u^{\mu+1}}{(\mu+1)!\,2^{\iota\mu}} + \frac{\mu Q}{2^n} + \frac{Q^2}{2^c} + \binom{u}{2}\frac{H_{\text{LMS}}}{2^{\iota+n}}\Big).$$

Since $H_{\text{LMS}} < 2^{55}$, $\iota = 128$ and $n \in \{192, 256\}$, when $u \leq 2^{64}$, the last term is at most

$$\frac{2^{128+55-1}}{2^{128+192}} \ll 2^{-128},$$

which can be neglected. Then, the function SHAKE256/256 in LMS_SHAKE_M32_H$h$ has $c = 512$ and $n = 256$, indicating that the concrete security bound is at most

$$O\Big(\frac{u^{\mu+1}}{(\mu+1)!2^{128\mu}} + \frac{\mu Q}{2^{256}} + \frac{Q^2}{2^{512}}\Big),$$

whereas the function SHAKE256/192 in LMS_SHAKE_M24_H$h$ has $c = 512$ and $n = 192$, indicating that the concrete security bound is at most

$$O\Big(\frac{u^{\mu+1}}{(\mu+1)!2^{128\mu}} + \frac{\mu Q}{2^{192}} + \frac{Q^2}{2^{512}}\Big).$$

When $\mu$ is treated as a constant and the identifier-dependent terms are negligible, the query-dependent parts of these bounds reach constant order at $Q \approx 2^{256}$ and $Q \approx 2^{192}$, respectively.

## F Extension to the Hierarchical Signature System

This appendix informally records the security implications for HSS. The Hierarchical Signature System (HSS) consists of a hierarchy of LMS trees of depth $L$ [41, Sect. 6].

Its public key contains the public key of the root LMS component, whereas an HSS signature has the form

$$\Sigma_{\mathrm{HSS}} = (\Sigma_0, pk_1, \Sigma_1, pk_2, \ldots, \Sigma_{L-2}, pk_{L-1}, \Sigma_{L-1}).$$

For $0 \le i < L-1$, the complete LMS signature $\Sigma_i$ authenticates the child public key $pk_{i+1}$ under $pk_i$; the final LMS signature $\Sigma_{L-1}$ authenticates the application message under $pk_{L-1}$. The verifier accepts only if all these LMS verifications accept. A component key at a non-root level is replaced whenever it is exhausted, and its freshly and independently generated replacement is signed by the current parent LMS key. RFC 8554 requires every component LMS key pair to be generated independently and every replacement to use a new identifier [41, Sects. 6.1–6.2]. The discussion below inherits the independent-secret scope of the LMS theorems. For seeded NIST-profile component keys, the corresponding seed-expander distinguishing advantages must additionally be included.

The design intuition is that every HSS forgery contains an LMS forgery under one of its component keys, as formalized for the static setting by Fluhrer [20, Theorem 3]. Hence the full-state LMS bound in Theorem 3 has an informal HSS consequence when all component keys are generated at once during **Initialization**: the components are treated as one global multi-user LMS family, using the total budget for adversarial ideal-cipher queries and the total budget for honest ideal-cipher queries of that family. Full-fledged HSS instead generates replacement LMS trees, and therefore new challenge evaluations, online; formalizing this case requires either a dynamic variant of Fig. 4 or an explicit coupling to a pre-generated hidden bank of component keys, and is left for future work.

# G Query-Preprocessing Setting

## G.1 Multi-user Unforgeability with Preprocessing for LMS-like Schemes

In this appendix, we consider the multi-user unforgeability with preprocessing experiment in Fig. 7, which is an extension of Fig. 4. The defining feature of the experiment is that $\mathscr{A}_{\mathrm{pre}}$ receives oracle access immediately after $\mathbf{p}$ is sampled, before the user identifiers and key pairs are generated. It may compile the resulting transcript into an arbitrary state $\mathsf{st}_{\mathrm{pre}}$ at line (2), which $\mathscr{A}_{\mathrm{post}}$ uses after seeing the public keys. All calls made in both phases refer to the same sampled oracle; in particular, the oracle is not reset between preprocessing and key generation.

The values $m$ and $u$, together with the per-user public-parameter vector

$$\big((h^1, w^1), \ldots, (h^u, w^u)\big),$$

are fixed before the experiment begins. In particular, these parameters cannot depend on the preprocessing output $\mathsf{st}_{\mathrm{pre}}$. After preprocessing, the identifiers and the fresh random coins used by the key-generation algorithms are sampled independently of the random coins and output of $\mathscr{A}_{\mathrm{pre}}$, subject only to the fact that all algorithms access the same already sampled primitive $\mathbf{p}$.

**The** m-unforge-p **experiment** $\mathbf{Exp}^{\text{m-unforge-p}}_{\mathbf{p},\mathbf{Sign}^{\mathbf{p}},\mathscr{A},m,u}$**:**

1 **Initialization:**
   (1) Set $n \leftarrow 8m$, sample an ideal oracle $\mathbf{p} \xleftarrow{\$} \mathcal{PS}$;
   (2) Run $\mathsf{st}_{\text{pre}} \leftarrow \mathscr{A}^{\mathbf{p}}_{\text{pre}}$ (*pre-challenge processing*);
   (3) For $\mathsf{i} = 1, \ldots, u$, select the public parameters $(h^{\mathsf{i}}, w^{\mathsf{i}})$ independently of the ideal oracle, the user identifiers, and all key-generation randomness. Sample $I^{\mathsf{i}} \xleftarrow{\$} \{0,1\}^{\iota}$ and invoke $(pk_{\mathsf{i}}, sk_{\mathsf{i}}) \leftarrow \text{SIGN}^{\mathbf{p}}.\mathsf{Gen}(m, (h^{\mathsf{i}}, w^{\mathsf{i}}), I^{\mathsf{i}})$.

2 **Interaction:** $(\mathsf{i}^*, M^*, \Sigma^*) \leftarrow \mathscr{A}^{\mathsf{muSign}^{\mathbf{p}},\mathbf{p}}((I^1, pk_1), ..., (I^u, pk_u))$, where the oracle $\mathsf{muSign}^{\mathbf{p}}(\mathsf{i}, q, M^{\mathsf{i},q})$ returns $\text{SIGN}^{\mathbf{p}}.\mathsf{Sign}(sk_{\mathsf{i}}, q, M^{\mathsf{i},q}, I^{\mathsf{i}})$ when $1 \le \mathsf{i} \le u$, $0 \le q < 2^{h^{\mathsf{i}}}$ and $M^{\mathsf{i},q}$ lies in the message domain of SIGN, and $\perp$ otherwise.
   – **Adversarial restriction**: for each valid pair $(\mathsf{i}, q)$, $\mathscr{A}$ makes at most one query of the form $\mathsf{muSign}^{\mathbf{p}}(\mathsf{i}, q, \star)$.

3 **Finalization:** Parse $\Sigma^* = (q^*, \sigma^*, p^*_0, \ldots, p^*_{h^{\mathsf{i}^*}-1})$, output 1 if and only if: (i) $1 \le \mathsf{i}^* \le u$, (ii) $0 \le q^* < 2^{h^{\mathsf{i}^*}}$, (iii) $M^*$ lies in the message domain of SIGN, (iv) no signing query $\mathsf{muSign}^{\mathbf{p}}(\mathsf{i}^*, q^*, M^*)$ was made before, and (v) $\text{SIGN}^{\mathbf{p}}.\mathsf{Vrfy}(pk_{\mathsf{i}^*}, M^*, \Sigma^*) = 1$.

**Fig. 7:** Experiment for multi-user unforgeability of LMS-like signatures with (query)-preprocessing.

### G.2 m-unforge Security Tightly Implies m-unforge-p Security

Below is an immediate consequence of Ghoshal and Tessaro [25, Lemma 1].

**Corollary 1 (**m-unforge $\Rightarrow$ m-unforge-p**).** *Suppose that, whenever the game-side computation makes at most $\widehat{B}$ queries to* $\mathbf{p}$*, every adversary $\mathscr{B}$ making at most $T$ queries to* $\mathbf{p}$ *in the ordinary* m-unforge *experiment satisfies*

$$\Pr\left[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{SIGN}^{\mathbf{p}},\mathscr{B},m,u} \Rightarrow 1\right] \le \varepsilon(\widehat{B}, T).$$

*Then every preprocessing adversary $\mathscr{A} = (\mathscr{A}_{\text{pre}}, \mathscr{A}_{\text{post}})$ with the same game-side bound, making at most $T_{\text{pre}}$ and $T_{\text{post}}$ queries to* $\mathbf{p}$ *in the two phases, respectively, where $T_{\text{pre}} + T_{\text{post}} \le T$, satisfies*

$$\Pr\left[\mathbf{Exp}^{\text{m-unforge-p}}_{\mathbf{p},\text{SIGN}^{\mathbf{p}},\mathscr{A},m,u} \Rightarrow 1\right] \le \varepsilon(\widehat{B}, T).$$

*Proof.* Fix $\mathscr{A} = (\mathscr{A}_{\text{pre}}, \mathscr{A}_{\text{post}})$, and regard the ordinary m-unforge experiment as the oracle game $G$ in [25, Lemma 1]. Both the game and the adversary access the same sampled interface $\mathbf{p}$, so $G$ is compatible with the corresponding ideal primitive. The cited lemma gives an adversary $\mathscr{B}$ for $G$ such that

$$\Pr\left[\mathbf{Exp}^{\text{m-unforge-p}}_{\mathbf{p},\text{SIGN}^{\mathbf{p}},\mathscr{A},m,u} \Rightarrow 1\right] = \Pr\left[\mathbf{Exp}^{\text{m-unforge}}_{\mathbf{p},\text{SIGN}^{\mathbf{p}},\mathscr{B},m,u} \Rightarrow 1\right].$$

Concretely, $\mathscr{B}$ first runs $\mathscr{A}_{\text{pre}}$ with access only to $\mathbf{p}$ and retains its output state. It does not reveal to $\mathscr{A}_{\text{pre}}$ any public keys, initialization values, or game-dependent oracles supplied by $G$. It then runs $\mathscr{A}_{\text{post}}$ on that state and on the inputs supplied by $G$, forwarding its primitive and game-dependent queries. The public-parameter vector is

fixed before either execution, and is the same in both orderings. Conditional on $\mathbf{p}$, the user identifiers and the key-generation random coins are independent of the random coins and output of $\mathscr{A}_{\mathrm{pre}}$. Although the initialization of $G$ has already been performed when $\mathscr{B}$ is invoked, moving the computation of $\mathscr{A}_{\mathrm{pre}}$ past that initialization therefore does not change the joint distribution: $\mathbf{p}$ is one static ideal object sampled for the entire experiment, and $\mathscr{A}_{\mathrm{pre}}$ receives no output of the initialization. Thus the preprocessing and game-initialization computations may be interchanged while preserving their joint distribution.

The resulting adversary makes at most $T_{\mathrm{pre}} + T_{\mathrm{post}} \leq T$ primitive queries and exactly the same game-dependent queries as $\mathscr{A}_{\mathrm{post}}$. The game $G$ itself is unchanged, so its game-side primitive-query bound remains $\widehat{B}$. The assumed ordinary-security bound applied to $\mathscr{B}$ therefore gives the claimed inequality. □

Gathering this corollary with Theorems 3, 4 and 6 imply the same levels of security for the three LMS models in the query-preprocessing setting.

## AI Statement

The authors used OpenAI ChatGPT 5.5 and 5.6 as an auxiliary tool during the development of this work. Concretely, the overall attack and proof ideas were due to the authors. Then, ChatGPT crucially assisted in proving various lemmas for Theorem 2. Representative substantive prompts included:

> [Representative prompt 1: is the current proof in LMS-sMD-proof.tex correct?] No, the proof shall use a stopped anchor lemma ...
>
> [Representative prompt 2: if the number of collided edges in the transcript graph is $s(T+B)^2/2^s$, can we still prove an analogue of Lemma 6 in the paper Permutation-Based Hash from Non-Idealized Assumptions: Adding Feed-Forward to Sponge?] ... (this time GPT proposed to use excess indegree to express the result)

It also insisted on using the a general lemma (Lemma 2) for conditional distribution of critical values in the experiment. All resulting arguments, calculations, and attack strategies were independently checked, refined, and formalized by the authors. The authors take full responsibility for the correctness and presentation of all results in this paper.