Leakage-Resilient and Misuse-Resistant Authenticated Encryption

Francesco Berti, François Koeune, Olivier Pereira,
Thomas Peters, François-Xavier Standaert.

ICTEAM/ELEN/Crypto Group, Université catholique de Louvain, Belgium.
emails: {francesco.berti,francois.koeune,thomas.peters,olivier.pereira,fstandae}@uclouvain.be

Abstract. Leakage-resilience and misuse-resistance are two important properties for the deployment of authenticated encryption schemes. They aim at mitigating the impact of implementation flaws due to side-channel leakages and misused randomness. In this paper, we discuss their interactions and incompatibilities. For this purpose, we first show that a generic composition of existing leakage-resilient MAC and encryption schemes leads to a misuse-resistant authenticated encryption mode (without leakage). Next we show that this misuse-resistance does not hold in case the adversary can additionally observe leakages, and that misuse-resistance with leakage may be impossible to achieve with simple primitives such as hash functions and block ciphers. As a result, we formalize a new security notion of ciphertext integrity with misuse and leakage, which seems the best that can be achieved in a symmetric cryptographic setting, and describe first efficient constructions satisfying it.

1 Introduction

State-of-the-art. At CCS 2015, Pereira et al. introduced Leakage-Resilient (LR) authentication and encryption schemes, which they proved to be secure in a pragmatic model combining the minimal use of an (expensive) leak-free component with much more efficient (less protected) implementations [27]. Such a model nicely matches the reality of modern embedded devices, where physical security against side-channel attacks is now a necessary condition for deployment, while cost constraints require to limit the overheads of the countermeasures against such attacks. Concretely, the leak-free component will typically be implemented by a block cipher (e.g., the AES Rijndael) protected with a combination of hardware and algorithmic techniques, e.g., noise addition [17], masking [31] and shuffling [37]. The latter ones usually increase the "code size × cycle count" metric (for software implementations) or the "throughput / area" metric (for hardware ones) by factors ranging from hundreds to thousands, hence motivating their minimal use.[1] In practice, this good tradeoff between security and performance is achieved by requiring only a single execution of the leak-free component, independently of the length of the message to be encrypted or authenticated. For long messages, the majority of the computational work can then be performed by weakly protected block-cipher implementations.

Besides, the recent literature also suggests an increasing interest for combined primitives such as Authenticated Encryption (AE), which typically aims to prevent flaws in the interaction between secret-key encryption and secret-key authentication, as exhibited, e.g., in [2,8,26]. In this context, a desirable security notion is Misuse-Resistance (MR), which guarantees that the encryption scheme only provides minimum advantage to the adversary in case the nonce or IV (which is needed for semantic security) is weak or even controlled by the adversary [34]. Informally, the only thing an adversary will be able to detect is whether the same message is encrypted with the same nonce/IV twice. So, to some extent, misuse-resistance can also be viewed as an important property to prevent implementation flaws.

Eventually, we note that leakage-resilience also becomes a desirable feature for implementation in high(er)-end devices, as suggested by recent works on timing attacks against OpenSSL [1,14], or power and electromagnetic analyses of powerful ARM cores running at high frequencies [4,16].

[1] See Table 4 in [27] for an illustration of these overheads.

Our contributions. Based on this state-of-the-art, it appears as an important challenge to design (jointly) leakage-resilient and misuse-resistant encryption schemes. Our results in this direction are in six parts.

First, we show how to generically construct a misuse-resistant authenticated encryption scheme by combining an IV-based Message Authentication Code and an IV-based encryption scheme [24]. The resulting "Double IV" (DIV) composition differs from the SIV composition due to [33,34] (and generalized in [24,25]) since it encrypts the IV. While this may seem useless in front of an adversary controlling the IV, we argue that it leads to interesting opportunities to improve security in the presence of leakage (see Sections 5.3, 7.2 and 8 for the details). Since these properties are ensured by the CCS 2015 MAC and encryption schemes, such building blocks can be used to design a misuse-resistant authenticated encryption scheme (without leakage), next denoted as PSV-AE.

Second, we show that as soon as a leakage oracle is added to the adversary's capabilities, the misuse-resistance of PSV-AE falls down. More precisely, we show that there is a realistic standard Differential Power Analysis (DPA) attack [18] targeting the ephemeral key(s) of PSV-AE which enables forgery of valid ciphertexts with a few queries.[2] The attack essentially exploits the fact that the leakage-resilience of PSV-AE heavily relies on the randomness of its IVs, which can be forced to constant thanks to misuse.

Third, we introduce a new construction for authenticated encryption, that we denote as the DTE scheme (for Digest, Tag and Encrypt), which allows preventing this DPA forgery attack. We formalize the security property that this new construction achieves as Ciphertext Integrity with Misuse and Leakage (CIML), which is a natural extension of the ciphertext integrity given in [6].

Fourth, we argue that (standard) misuse-resistance with leakage may be impossible to achieve from standard symmetric cryptographic primitives only. For this purpose, we put forward a (more theoretical) Simple Power Analysis (SPA) attack against the DTE scheme, which also targets an ephemeral key forced to a constant thanks to misuse, and allows distinguishing actual ciphertexts from random ones.[3] Interestingly, this distinguishing attack can be viewed as an amplification of the impossibility result discussed in [27]. Namely, since a single bit of leakage on the plaintexts (which seems impossible to avoid without unrealistic hardware assumptions) trivially breaks the semantic security game, it is natural that the leakage of an ephemeral key (due to misuse) leads to even more serious issues. Intuitively, this observation once more highlights the separation between unpredictability-based and indistinguishability-based security in the presence of leakage, first mentioned in [23].

Fifth, since combining (standard) misuse-resistance and leakage-resilience appears to be impossible, we investigate the gains that can be obtained if we also drop the requirement of (standard) misuse-resistance without leakage – so that we reach similar security guarantees with and without leakage. As a result, we introduce another authenticated encryption scheme, that we denote as the DCE scheme (for Digest, Commit and Encrypt) which reduces the number of leak-free block cipher executions from two (in DTE) to one, at the cost of moving to the random oracle model instead of relying on standard assumptions.

Eventually, we show the leakage-resilient CPA security of our new constructions in a model borrowed from [27].

These results are summarized in Table 1.

[2] Informally, standard DPAs are side-channel attacks taking advantage of the leakage of multiple (different) inputs.

[3] Informally, SPAs are side-channel attacks taking advantage of the leakage of a single input, possibly measured multiple times to reduce the measurement noise, e.g., by exploiting powerful (yet less practical) algebraic/analytical techniques [13,36].

Table 1. Summary of our constructions. LR-CPA = leakage resilient chosen plaintext attack security; (~~LR~~) MR = misuse resistance in the absence of leakage; CIML = ciphertext integrity with misuse and leakage, LF executions counts the number of executions of the leak free component that are required for an encryption, and the models are either the standard one or the random oracle model.

	LR-CPA	(~~LR~~) MR	CIML	Model	LF executions
PSV-AE	✓	✓	✗	std.	2
DTE	✓	✓	✓	std.	2
DCE	✓	✗	✓	RO	1

We conclude the paper by discussing the remaining challenge of protecting an authenticated encryption scheme where the leakage of the decryption algorithm can also be exploited by the adversary (which was left out of the analysis in [27], motivated by applications such as smart cards where one low-cost prover has to be protected against side-channel attacks).

2 Background

We use calligraphic fonts for sets and denote as a (q, t)-bounded algorithm a probabilistic algorithm that can make at most q queries to the oracles he is granted access to and can perform computation bounded by running time t.

2.1 Definitions

We first need the following definition of collision-resistant hash function.

Definition 1. *A $(0, t, \epsilon_{\mathsf{cr}})$-collision resistant hash function $\mathsf{H} : \mathcal{S} \times \mathcal{M} \to \mathcal{B}$ is a function that is such that, for every $(0, t)$-bounded adversary $\mathcal{A}$, the probability that $\mathcal{A}(s)$ outputs a pair of distinct messages $(m_0, m_1) \in \mathcal{M}^2$ such that $\mathsf{H}^s(m_0) = \mathsf{H}^s(m_1)$ is bounded by $\varepsilon_{\mathsf{cr}}$, where $s \leftarrow \mathcal{S}$ is selected uniformly at random.*

We next need the following definition of range-oriented preimage resistance.

Definition 2. *A $(1, t, \epsilon_{\mathsf{pr}})$-range-oriented preimage resistant hash function $\mathsf{H} : \mathcal{S} \times \mathcal{M} \to \mathcal{B}$ is a function that is such that, for every $(0, t)$-bounded adversary $\mathcal{A}$, the probability that $\mathcal{A}(s, y)$ outputs a message $m \in \mathcal{M}$ such that $\mathsf{H}^s(m) = y$ is bounded by $\varepsilon_{\mathsf{pr}}$, where $s \leftarrow \mathcal{S}$, $y \leftarrow \mathcal{B}$ are selected uniformly at random.*

Note that the usual notion of preimage resistance samples a random $m_0 \leftarrow \mathcal{M}$ over the *domain* of H^s and then sets $y = H^s(m_0)$. Definition 2 uniformly samples $y \leftarrow \mathcal{B}$ over the *range* of H^s, which was introduced in [3].

In the following, we assume that the key s is not private, and refer to the hash function simply as H for simplicity, the key s being implicit.

We also need the following definition of pseudorandom function.

Definition 3. *A function $\mathsf{F} : \mathcal{K} \times \mathcal{B} \to \mathcal{T}$ is a $(q, t, \epsilon_{\mathsf{F}})$-*pseudorandom function *(PRF) if for all (q, t)-bounded adversaries $\mathcal{A}$ provided with oracle access to the function, the advantage*

$$\left| \Pr\left[\mathcal{A}^{\mathsf{F}_k(.)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\mathsf{f}(.)} \Rightarrow 1\right] \right|$$

is upper-bounded by ε_{F}, where k and f are chosen uniformly at random from their domains, namely $\mathcal{K}$ and the set of functions matching the signature of F.

In order to capture authenticity, we additionally introduce the notion of IV-based MAC. We use this variant of the standard definition of MAC because it actually corresponds to the construction of leakage-resilient MAC in [27]. We will naturally say that $\mathsf{ivM} = (\mathcal{K}, \mathsf{Mac}, \mathsf{Vrfy})$ is an IV-based MAC if there is a probabilistic algorithm $\mathsf{MAC} : \mathcal{K} \times \mathcal{M} \to \mathcal{T}$ which on inputs $k \in \mathcal{K}$ and $m \in \mathcal{M}$ picks a random $IV \in \mathcal{IV}$ and outputs IV and $\tau \leftarrow \mathsf{Mac}_k(IV, m)$.

Definition 4. *An IV-based MAC is a tuple* $\mathsf{ivM} = (\mathcal{K}, \mathsf{Mac}, \mathsf{Vrfy})$ *such that:*
- $\mathsf{Mac} : \mathcal{K} \times \mathcal{IV} \times \mathcal{M} \to \mathcal{T}$ *takes a key, an IV, and a message and outputs a tag.*
- $\mathsf{Vrfy} : \mathcal{K} \times \mathcal{IV} \times \mathcal{M} \times \mathcal{T} \to \{\top \cup \bot\}$ *and outputs* $\top$ *only if* τ *is a valid tag for IV, message* m *and key* k*.*

More precisely, $\forall k \in \mathcal{K}$, $\forall IV \in \mathcal{IV}$, $\forall m \in \mathcal{M}$: $\mathsf{Vrfy}_k(IV, m, \mathsf{Mac}_k(IV, m)) = \top$.

While the traditional property required from MACs is unforgeability, our constructions will rely on a stronger property of the Mac function. Namely, we will require Mac to be a pseudorandom function for any (potentially repeated) adversarially chosen IV.

Definition 5. ivM *is* $(q, t, \epsilon_{\mathsf{cip}})$ chosen-IV pseudorandom *if for all* (q, t)*-bounded adversary* $\mathcal{A}$*, the* cip *advantage*

$$\mathbf{Adv}^{\mathsf{cip}}_{\mathsf{ivM},\mathcal{A}} := \left| \Pr\left[\mathcal{A}^{\mathcal{O}^{\mathsf{cip}}_{\mathsf{real}}} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\mathcal{O}^{\mathsf{cip}}_{\mathsf{rand}}} \Rightarrow 1\right] \right|$$

is upper-bounded by ϵ_{cip}*. Here,* $\mathcal{O}^{\mathsf{cip}}_{\mathsf{real}}$ *is an oracle initialized on* $k \leftarrow \mathcal{K}$ *which on input* (IV, m) *outputs* $\tau \leftarrow \mathsf{Mac}_k(IV, m)$*, and where* $\mathcal{O}^{\mathsf{cip}}_{\mathsf{rand}}$ *is an oracle which on new input* (IV, m) *outputs a random* $\tau \leftarrow \mathcal{T}$ *and on re-used input outputs the corresponding previous outcome.*

Note that this security property of ivM does not introduce a significantly new object. If $\mathsf{Mac}'_k(m_1 \| m_2)$ is a (usual) pseudorandom MAC with message space $\mathcal{M}^2$, then $\mathsf{Mac}_k(IV, m) := \mathsf{Mac}'_k(IV \| m)$ easily leads to a chosen-IV pseudorandom IV-based MAC. Besides, this property is directly fulfilled by the CCS 2015 leakage-resilient MAC.

Besides, our authenticated encryption schemes will be based on IV-based encryption schemes, which we define following Rogaway and Shrimpton [33].

Definition 6. *An IV-based encryption scheme is a tuple* $\mathsf{ivE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *such that:*
- $\mathsf{Enc} : \mathcal{K} \times \mathcal{IV} \times \mathcal{M} \to \mathcal{C}$ *maps a key selected from* $\mathcal{K}$*, an IV selected from* $\mathcal{IV}$ *and a message from* $\mathcal{M}$ *to a ciphertext from* $\mathcal{C}$*.*
- $\mathsf{Dec} : \mathcal{K} \times \mathcal{IV} \times \mathcal{C} \to \mathcal{M}$ *provides the decryption of a pair containing an IV and a ciphertext.*

We will use $\mathsf{ENC} : \mathcal{K} \times \mathcal{M} \to \mathcal{IV} \times \mathcal{C}$ for the probabilistic function that picks a uniformly random IV and returns $(IV, \mathsf{Enc}(k, IV, m)) \leftarrow \mathsf{ENC}_k(m)$.

To capture message secrecy, we use the security definition of Namprempre et al. [24] and consider a distinguishing game in which the adversary tries to determine whether he is facing an encryption oracle or a random function.

Definition 7. *An IV-based encryption scheme* $\mathsf{ivE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *is* $(q, t, \epsilon_{\mathsf{IV\text{-}sec}})$-IV-sec *secure if for any* $k \leftarrow \mathcal{K}$ *and for every* (q, t)*-adversary* $\mathcal{A}$*, the advantage*

$$\mathbf{Adv}^{\mathsf{IV\text{-}sec}}_{\mathsf{ivE},\mathcal{A}} := \left| \Pr\left[\mathcal{A}^{\mathsf{ENC}_k(\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot)} \Rightarrow 1\right] \right|$$

is upper-bounded by $\epsilon_{\mathsf{IV\text{-}sec}}$*, where* $\$(m)$ *picks a random* $IV \leftarrow \mathcal{IV}$ *and outputs* (IV, σ)*, where* σ *is a random bit string of length* $|\mathsf{Enc}_k(IV, m)|$*.*

Resistance against misuse then captures the security in front of an adversary controlling the generation of the randomness used for encryption. In the case of authenticated encryption, the adversary is also granted access to a decryption oracle. We consider a definition of *misuse-resistant authenticated encryption* equivalent to the one appearing in [33].

Definition 8. *An* authenticated encryption scheme *is a tuple* $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *such that:*
- $\mathsf{Enc} : \mathcal{K} \times \mathcal{R} \times \mathcal{M} \to \mathcal{C}$ *maps a key selected from* $\mathcal{K}$*, randomness selected from* $\mathcal{R}$ *and a message from* $\mathcal{M}$ *to a ciphertext in* $\mathcal{C}$*.*
- $\mathsf{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{M} \cup \{\perp\}$ *provides the decryption of a ciphertext, and can return the special symbol* $\perp$ *if decryption fails.*

The associated probabilistic algorithm first picks a random coin $r \in \mathcal{R}$ and returns $c = \mathsf{Enc}_k(r, m) := \mathsf{Enc}(k, r, m)$. We stress that Dec_k only needs c to recover m, which is the main difference between our definition and previous IV-based schemes for which an IV additionally needs to be provided. By contrast, in our case, the encrypted randomness is part of the ciphertext. As mentioned in introduction, this is motivated by our improved leakage-resilience goal.

Definition 9. *An authenticated encryption scheme* $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *offers* (q, t, ε) *misuse-resistance if, for every* (q, t)*-bounded adversary* $\mathcal{A}$*, the advantage*

$$\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE},\mathcal{A}} := \left| \Pr\left[\mathcal{A}^{\mathsf{Enc}_k(\cdot,\cdot),\mathsf{Dec}_k(\cdot)} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\$(\cdot,\cdot),\perp(\cdot)} \Rightarrow 1\right] \right|$$

is upper-bounded by ε*, where* $\$(r, m)$ *outputs* c *selected as a random bit string of length* $\mathsf{Enc}_k(r, m)$ *and the oracle* $\perp(c)$ *outputs* $\perp$ *except if* c *was output by the* $\$(\cdot, \cdot)$ *oracle earlier, in which case it returns the associated* m*.*

Note that for conciseness we ignore the specific treatment of associated data in our constructions, which is orthogonal to the discussions on misuse-resistance and leakage-resistance that motivate our results and could be carried out using standard techniques (see [28] for a recent example).

Eventually, whenever instantiating our building blocks in the paper, we will consider $\mathcal{K} = \mathcal{T} = \mathcal{R} = \mathcal{B} = \mathcal{IV} = \{0,1\}^n$ using n as a security parameter, and $\mathcal{M} = \{0,1\}^{n\ell}$, that is, a message is made of on ℓ blocks of n bits.

2.2 Building blocks

Our starting points are the block-cipher based leakage-resilient MAC and encryption schemes from CCS 2015. In the next sections, we will explore their composition into an authenticated encryption scheme, point out limitations in this composition, and propose improved solutions.

The CCS 2015 leakage-resilient MAC is represented in Figure 1. For readability, we use the color code red for long term secrets, orange for ephemeral secrets and green for publicly released values. It is based on two block-ciphers F and F^*, both treated as PRF's, but with the distinction that F is assumed to be cheap and efficiently implemented but leaking, while F^* is assumed to be an expensive and leak-free component.

CCS 2015 leakage-resilient MAC (PSV-MAC)
$\mathsf{Mac}_k(IV, m)$ where $m = m_1 \| \cdots \| m_\ell$ – $k_0 \leftarrow \mathsf{F}^*_k(IV)$ – $k_i \leftarrow \mathsf{F}_{k_{i-1}}(m_i)$, $\forall i \in [1, \ell]$ – return $\tau \leftarrow k_\ell$
$\mathsf{Vrfy}_k(IV, m, \tau)$ proceeds in the natural way.

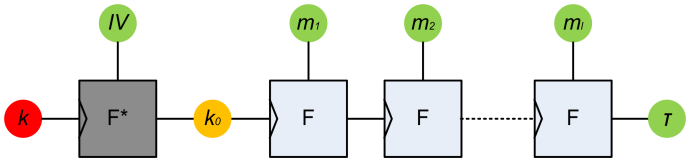

Fig. 1. PSV-MAC leakage-resilient MAC [27].

The CCS 2015 leakage-resilient encryption scheme in Figure 2 is based on the same components as the previous leakage-resilient MAC.

CCS 2015 leakage-resilient encryption (PSV-ENC)
$\mathsf{Enc}_k(IV, m)$, where $m = m_1 \| \cdots \| m_\ell$ – $k_0 \leftarrow \mathsf{F}^*_k(IV)$ – $\forall i \in [1, \ell] : k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $y_i \leftarrow \mathsf{F}_{k_{i-1}}(p_B)$, $c_i \leftarrow y_i \oplus m_i$, where p_A, p_B are public constants – return $C = c_1 \| c_2 \| \cdots \| c_\ell$
$\mathsf{Dec}_k(IV, C)$ proceeds in the natural way

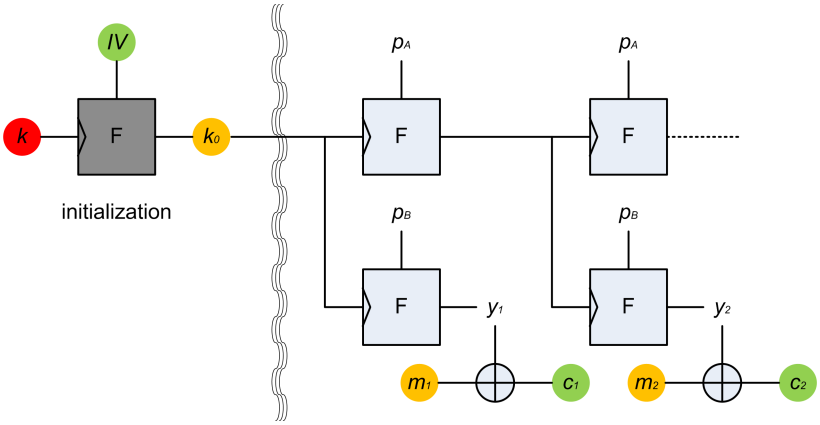

Fig. 2. PSV-ENC leakage-resilient encryption [27].

3 Generic misuse-resistance

We now show how an IV-based MAC can be composed with an IV-based encryption scheme to get an authenticated encryption scheme. This composition is named DIV referring to *Double-IV*.

3.1 The DIV Composition

Let $\mathsf{ivM} = (\mathcal{K}, \mathsf{Mac}, \mathsf{Vrfy})$ be an IV-based MAC with IV-space $\mathcal{IV}$, message space $\mathcal{M}$ and tag space $\mathcal{T}$, and let $\mathsf{ivE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ be an IV-based encryption scheme with message space $\mathcal{IV} \times \mathcal{M}$, IV-space $\mathcal{T}$ and ciphertext space $\mathcal{C}$. Then, $\mathsf{AE}_{\mathsf{DIV}} = (\mathcal{K}^2, \mathsf{DIV.Enc}, \mathsf{DIV.Dec})$ is an authenticated encryption resulting from the DIV composition if:

– (k_M, k_E) is the key selected from $\mathcal{K}^2$
– $\mathsf{DIV.Enc}_{k_M,k_E}(IV, m)$, given $IV \in \mathcal{IV}$ and $m \in \mathcal{M}$, returns $\tau \leftarrow \mathsf{Mac}_{k_M}(IV, m)$ and $c \leftarrow \mathsf{Enc}_{k_E}(\tau, (IV, m))$.
– $\mathsf{DIV.Dec}_{k_M,k_E}(\tau, c)$ returns $(IV, m) \leftarrow \mathsf{Dec}_{k_E}(\tau, c)$ if $\mathsf{Vrfy}_{k_M}(IV, m, \tau)$ succeeds. The error symbol $\perp$ is returned otherwise.

We have $\mathcal{R}_{\mathsf{DIV}} = \mathcal{IV}$, $\mathcal{M}_{\mathsf{DIV}} = \mathcal{M}$, $\mathcal{C}_{\mathsf{DIV}} = \mathcal{T} \times \mathcal{C}$ and the correctness of $\mathsf{AE}_{\mathsf{DIV}}$ follows from the correctness of ivM and ivE in their respective sense. We will show that the authenticated encryption $\mathsf{AE}_{\mathsf{DIV}}$ is misuse-resistant as long as (1) ivM is chosen-IV pseudorandom (2) ivE is IV-sec-secure.

Note that we saw in Section 2.1 that chosen-IV pseudorandom IV-based MAC derives easily from usual pseudorandom MAC. In fact, this security notion is unavoidable for our purpose: if the authenticated encryption $\mathsf{AE}_{\mathsf{DIV}}$ is misuse-resistant then the underlying ivM must be chosen-IV pseudorandom as well (which directly follows from the definitions).

Before moving to the security analysis of the DIV composition, we introduce a strong flavor of ciphertext integrity which simplifies the presentation of the proof.

3.2 Strong Authentication

Namprempre et al. introduced a nonce-based variant of ciphertext integrity [25, Appendix A], called authenticity (Auth), to ease their security analysis. In the same spirit, we introduce an even stronger notion: strong authenticity (SA). While Auth lets $\mathcal{A}$ choose non-repeating nonces in the INT-CTXT experiment, SA lets $\mathcal{A}$ completely free in its choice of random coins. It can be viewed as a misuse-resistant variation of ciphertext integrity.

Definition 10. *Let* $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *be an authenticated encryption scheme. Then,* AE *satisfies the notion of* $(q, t, \varepsilon_{\mathsf{sa}})$-strong authenticity *if, for each* (q, t)*-bounded adversary* $\mathcal{A}$*, the* sa *advantage*

$$\mathbf{Adv}^{\mathsf{sa}}_{\mathsf{AE},\mathcal{A}} := \left| \Pr\left[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}} \Rightarrow 1\right] - \Pr\left[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}} \Rightarrow 1\right] \right|$$

is upper-bounded by $\varepsilon_{\mathsf{sa}}$*, where* $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}$ *is an oracle which on input a coin* $r \in \mathcal{R}$ *and a message* $m \in \mathcal{M}$ *outputs a ciphertext* $c \leftarrow \mathsf{Enc}_k(r, m)$*,* $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}$ *is an oracle which on input* $c \in \mathcal{C}$ *outputs* $\mathsf{Dec}_k(c)$*, and* $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}$ *is an oracle which on input* $c \in \mathcal{C}$ *outputs* $\perp$ *except if* c *is an output of* $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}$ *on a past query* (r, m)*, in which case it returns the corresponding* $m \in \mathcal{M}$*.*

Clearly, a misuse-resistant authenticated encryption scheme satisfies this indistinguishability-based notion. The next section shows that a combination of weaker primitives is in fact enough to get misuse-resistant AE. So the SA notion is only useful as an intermediate step which appears in the hybrid argument of Section 3.3 through the following result.

Lemma 1. *Let* $\mathsf{AE}^{\dagger}_{\mathsf{DIV}} = (\mathcal{K}^2, \mathsf{DIV.Enc}^{\dagger}, \mathsf{DIV.Dec}^{\dagger})$ *be the scheme obtained from* $\mathsf{AE}_{\mathsf{DIV}}$ *by replacing the* Mac *algorithm by a truly random function* $\mathsf{f} : \mathcal{IV} \times \mathcal{M} \to \mathcal{T}$*. Namely,* $\mathsf{DIV.Enc}^{\dagger}_{k_M,k_E}(IV, m)$ *outputs* $\tau \leftarrow \mathsf{f}(IV, m)$ *and* $c \leftarrow \mathsf{Enc}_{k_E}(\tau, (IV, m))$ *using* ivE *(and* $\mathsf{DIV.Dec}^{\dagger}_{k_M,k_E}$ *also tests whether* $\tau = \mathsf{f}(IV, m)$ *holds). Then, for any* (q, ∞)*-bounded adversary* A*,* $\mathbf{Adv}^{\mathsf{sa}}_{\mathsf{AE}^{\dagger}_{\mathsf{DIV}},\mathcal{A}}(n) \leq q/|\mathcal{T}|$*.*

Proof. Let $\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{real}}$, $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{real}}$ and $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}$ denote the oracles defined in the definition of strong authenticity in the case of $\mathsf{AE}^{\dagger}_{\mathsf{DIV}}$. To see why $\mathcal{A}$'s advantage against the strong authenticity is negligible let us recall how $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{real}}$ and $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}$ differ from each other. In $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{real}}$, a valid ciphertext (τ, c) is an encryption of some (IV, m) under Enc_{k_E} and for which $\mathsf{f}(IV, m) = r$, where f is a truly random function. In

$\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{fake}}$, there is no valid ciphertext not computed by $\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{real}}$. Therefore, this advantage is bounded by the probability of being able to make a decryption query on a fresh but valid ciphertext $(\tau^\star, c^\star)$ with respect to $\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{real}}$. Even though it could not be efficiently computable from $\mathcal{A}$'s perspective let $(IV^\star, m^\star) = \mathsf{Dec}_{k_E}(\tau^\star, c^\star)$. We claim that $(IV^\star, m^\star)$ was never submitted to $\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{real}}$ otherwise $(\tau^\star, c^\star)$ would have been returned as an output of $\mathsf{DIV.Enc}^\dagger_k$ which is deterministic, where $k = (k_M, k_E)$. Consequently, we are sure that $(IV^\star, m^\star)$ is also fresh and that $\tau^\star = \mathsf{f}(IV^\star, m^\star)$ holds from the validity of the ciphertext. But that means that in a way or another, $\mathcal{A}$ is able to predict a new output of f which is completely independent of $\mathcal{A}$'s view, so that the probability for that event to occur is $q_d/|\mathcal{T}|$, where q_d corresponds to the number of queries to the decryption oracle. □

3.3 Security Analysis

Our security proof follows standard hybrid arguments and extends those of [25] from nonce-based security to misuse-resistance. We do not consider any leakage-resilient security so far.

Theorem 1. *Let* ivM *be a chosen-IV pseudorandom IV-based MAC and let* ivE *be a secure IV-based encryption scheme. Then* $\mathsf{AE_{DIV}}$ *is a misuse-resistant authenticated encryption scheme. More precisely, let* $\mathcal{A}$ *be a* (q,t)*-bounded adversary against* $\mathsf{AE_{DIV}}$*, then we build a* (q,t_1)*-bounded adversary* $\mathcal{B}_1(\mathcal{A})$ *against* ivM *and a* (q,t_2)*-bounded adversary* $\mathcal{B}_2(\mathcal{A})$ *against* ivE *such that* $\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE_{DIV}},\mathcal{A}}$ *is upper-bounded by*

$$\mathbf{Adv}^{\mathsf{cip}}_{\mathsf{ivM},\mathcal{B}_1(\mathcal{A})} + \mathbf{Adv}^{\mathsf{sa}}_{\mathsf{AE}^\dagger_{\mathsf{DIV}},\mathcal{A}} + \mathbf{Adv}^{\mathsf{IV\text{-}sec}}_{\mathsf{ivE},\mathcal{B}_2(\mathcal{A})},$$

where $\mathsf{AE}^\dagger_{\mathsf{DIV}}$ *is the scheme described in Lemma 1. Moreover, the running times satisfy* $t_1 \leq t + q \cdot t_e$ *and* $t_2 \leq t + q \cdot t'$*, where* t_e *is the maximum time needed to perform encryption or decryption in* ivE *and* t' *is the time to look for an entry in a table of size at most* q*.*

Proof. Let $\mathcal{A}$ be a (q,t)-bounded adversary against the misuse-resistance of $\mathsf{AE_{DIV}}$. By definition, if we set $k = (k_M, k_E) \leftarrow \mathcal{K}^2$, the advantage $\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE_{DIV}},\mathcal{A}}$ is given by $\left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}} \Rightarrow 1]\right|$, where $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}$ runs $\mathsf{DIV.Enc}_k$ on inputs (IV, m) and $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}$ runs $\mathsf{DIV.Dec}_k$ on input ciphertexts (τ, c) for the left-hand side probability, and where $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}$ returns random $(\tau, c) \leftarrow \mathcal{C}_{\mathsf{DIV}}$ on inputs (IV, m) and where $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}$ returns $\perp$ on input ciphertexts (τ, c) not generated from $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}$ and m if (τ, c) was the answer of an encryption query (IV, m) for the right-hand side probability.

The first step of the proof is to rely on the chosen-IV pseudorandomness of ivM to replace the real Mac_{k_M} in the computation of $\mathsf{DIV.Enc}_k(IV, m)$ and $\mathsf{DIV.Dec}_k(\tau, c)$ by a truly random function $\mathsf{f} \leftarrow \mathcal{F}$ where $\mathcal{F}$ is the set of all functions from $\mathcal{IV} \times \mathcal{M}$ to $\mathcal{T}$. This modification results in replacing the real execution of $\mathsf{AE_{DIV}}$ by the real execution of $\mathsf{AE}^\dagger_{\mathsf{DIV}}$. To avoid confusion with the oracles of $\mathsf{AE_{DIV}}$, $\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{real}}$ and $\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{real}}$ stand for the real oracles of $\mathsf{AE}^\dagger_{\mathsf{DIV}}$. Therefore, $\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE_{DIV}},\mathcal{A}}$ is upper-bounded by

$$\begin{aligned}
&\left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{real}}} \Rightarrow 1]\right| \\
&\qquad + \left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{real}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{fake}}} \Rightarrow 1]\right| \\
&\qquad\qquad + \left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{fake}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}} \Rightarrow 1]\right| ,
\end{aligned}$$

where the last term vanishes since $\mathcal{O}^{\mathsf{Enc}^\dagger}_{\mathsf{rand}} = \mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}$ and $\mathcal{O}^{\mathsf{Dec}^\dagger}_{\mathsf{fake}} = \mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}$ are independent of the description of $\mathsf{AE_{DIV}}$ and $\mathsf{AE}^\dagger_{\mathsf{DIV}}$. This development leads to

$$\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE_{DIV}},\mathcal{A}} \leq \mathbf{Adv}^{\mathsf{cip}}_{\mathsf{ivM},\mathcal{B}_1(\mathcal{A})} + \mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE}^\dagger_{\mathsf{DIV}},\mathcal{B}'_2(\mathcal{A})}$$

for some challenger $\mathcal{B}_1$ against the chosen-IV pseudorandomness of ivM and some challenger $\mathcal{B}'_2$ against the misuse-resistance of $\mathsf{AE}^{\dagger}_{\mathsf{DIV}}$ which still need to be described. Given an oracle access to either Mac_{k_M} or f, algorithm $\mathcal{B}_1$ emulates encryption and decryption as follows: picks a key $k_E \leftarrow \mathcal{K}$ of $\mathsf{ivE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ such that on an encryption query (IV, m), $\mathcal{B}_1$ sends (IV, m) to its own cip oracle which is either $\mathcal{O}^{\mathsf{cip}}_{\mathsf{real}}$ or $\mathcal{O}^{\mathsf{cip}}_{\mathsf{rand}}$ and gets back some $\tau \in \mathcal{T}$ from which it computes $c = \mathsf{Enc}_{k_E}(\tau, (IV, m))$ and returns (τ, c); on decryption query (τ, c), $\mathcal{B}_1$ first computes $(IV, m) = \mathsf{Dec}_{k_E}(\tau, c)$ and makes a deterministic-tag query on (IV, m), then from the answer τ', $\mathcal{B}_1$ outputs m if there is a match with τ and outputs $\perp$ otherwise. If q is the cumulated number of encryption and decryption queries made by $\mathcal{A}$, then $\mathcal{B}_1$ makes at most q queries to its cip oracle and runs in time $t_2 \leq t + q \cdot t_e$, where t_e is the maximum running time of Enc_{k_E} and Dec_{k_E}. This shows that we indeed have $\mathbf{Adv}^{\mathsf{cip}}_{\mathsf{ivM},\mathcal{B}_1(\mathcal{A})}$ which is a negligible function by assumption. As a result, it remains to show that the second term of the above development is negligible. To do so, we now upper-bound $\mathbf{Adv}^{\mathsf{mr}}_{\mathsf{AE}^{\dagger}_{\mathsf{DIV}},\mathcal{A}}$ by

$$\begin{aligned}&\left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{real}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}} \Rightarrow 1]\right| \\ &+ \left|\Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}} \Rightarrow 1]\right|\end{aligned}$$

and we argue that $\mathbf{Adv}^{\mathsf{sa}}_{\mathsf{AE}^{\dagger}_{\mathsf{DIV}},\mathcal{A}}$ and $\mathbf{Adv}^{\mathsf{IV\text{-}sec}}_{\mathsf{ivE},\mathcal{B}_2(\mathcal{A})}$ upper-bound this expression term-by-term for some explicit challenger $\mathcal{B}_2$ against the $\mathsf{IV\text{-}sec}$ security of ivE.

By definition, the first term is indeed the advantage of $\mathcal{A}$ against the strong authenticity of $\mathsf{AE}^{\dagger}_{\mathsf{DIV}}$. Lemma 1 shows that this advantage is smaller than $q/|\mathcal{T}|$ which is negligible since ivM and/or ivE are secure. To conclude the proof we need to show the last term is bounded by the advantage of some $\mathcal{B}_2(\mathcal{A})$ against ivE.

Since $\mathcal{O}^{\mathsf{Dec}^{\dagger}}_{\mathsf{fake}}$ gives no information on whether $\mathcal{A}$ is exchanging with $\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{real}}$ or $\mathcal{O}^{\mathsf{Enc}^{\dagger}}_{\mathsf{rand}}$, and since answer (τ, c) to encryption query (IV, m) has a value $\tau \in \mathcal{T}$ independent of (IV, m), the only relevant part in both views is whether $c = \mathsf{Enc}_{k_E}(\tau, (IV, m))$ or $c = \mathsf{f}'(\tau, IV, m)$ for another random function $\mathsf{f}' : \mathcal{T} \times \mathcal{IV} \times \mathcal{M} \to \mathcal{C}$. It is now straightforward to build a challenger $\mathcal{B}_2$ against the security of ivE. In more details, when $\mathcal{A}$ makes an encryption query (IV, m), $\mathcal{B}_1$ queries its own oracle on the message (IV, m) and gets back (τ, c), where τ is a random IV of ivE. Then, $\mathcal{B}_1$ stores (IV, m, τ, c) in a new entry of an initially empty table. When $\mathcal{A}$ makes a decryption query on some (τ, c), $\mathcal{B}_2$ first looks in the table for an entry of the form (IV, m, τ, c), for some (IV, m). If such entry lies in the table $\mathcal{B}_2$ returns the corresponding (IV, m) to $\mathcal{A}$, and $\perp$ otherwise. Obviously, $\mathcal{B}_2$ makes at most the same amount of queries as $\mathcal{A}$ since on $\mathcal{A}$'s decryption query $\mathcal{B}_2$ does not make any query. If $\mathcal{B}_2$'s running time is t_2 and if t' is the maximum time needed to store and look for an entry in a table of size q, we find $t_2 \leq t + q \cdot t'$, which concludes the proof of the theorem. □

4 The PSV-AE authenticated encryption

In this section, we apply the DIV composition to PSV-MAC (Figure 1) and PSV-ENC (Figure 2) to get PSV-AE. Theorem 1 directly implies that PSV-AE is misuse-resistant without leakage. We then illustrate the limitations of this combination by trying to improve PSV-AE in two ways: first we look at its single key variant, next we try look at its misuse-resistance with leakage. However, none of these attempts succeeds and we exhibit attacks in both cases, which therefore motivates our new constructions in the next sections.

4.1 Specification

We instantiate PSV-MAC with ℓ blocks and PSV-ENC with $\ell + 1$ blocks to get PSV-AE with ℓ blocks by applying the DIV composition. We split PSV-AE into two parts: the first part generates an authenticated ephemeral key k_0, as shown in Figure 3.

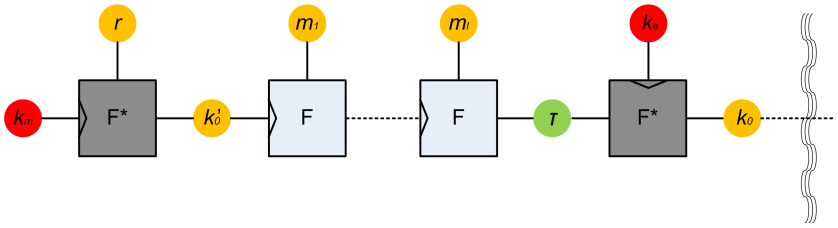

Fig. 3. PSV-AE leakage-resilient AE (part I).

Given k_0, the second part generates, using the public constant p_A and p_B, a pseudorandom vector $(y_0, y_1, \ldots, y_\ell)$ of $\ell + 1$ blocks in order to XOR $(r, m_1, \ldots, m_\ell)$ with it, as shown in Figure 4. The full specification is given in the PSV-AE box.

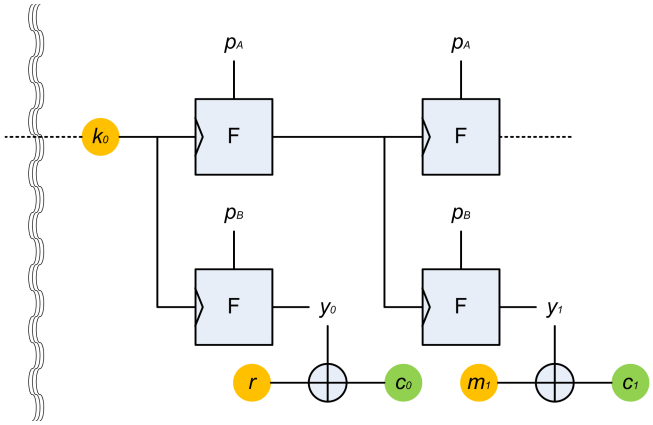

Fig. 4. PSV-AE leakage-resilient AE (part II).

4.2 Misuse-resistance without leakage

The security of PSV-AE is stated in the next theorem.

Theorem 2. *Let assume that the functions* F* *and* F *are pseudorandom, then PSV-AE is a misuse-resistant authenticated encryption scheme.*

It is already established that PSV-ENC is a chosen-IV pseudorandom IV-based MAC and PSV-ENC is a secure IV-based encryption if the underlying functions F are taken within a PRF familly. Therefore, the result follows directly from Theorem 1. Note that this result is qualitative and proven for the authenticated encryption of fixed-length messages. We ignored the treatment of quantitative bounds and variable-length messages because the following sections will anyway show that PSV-AE is not a good candidate for misuse-resistance with leakage. Quantitative bounds will be given for our improved constructions, for which variable-length security is directly obtained by the use of a hash function.

PSV-AE: PSV-ENC $\circ_{\mathsf{DIV}}$ PSV-MAC

$\mathsf{Enc}_k(m)$:
- Parse $m = (m_1, \ldots, m_\ell)$
- Pick $r \xleftarrow{\$} \{0,1\}^n$
- $k'_0 \leftarrow \mathsf{F}^*_{k_M}(r)$
- $k'_i \leftarrow \mathsf{F}_{k'_{i-1}}(m_i) \quad \forall i = 1, \ldots, \ell$
- $\tau \leftarrow k'_l$
- $k_0 \leftarrow \mathsf{F}^*_{k_E}(\tau)$
- $c_0 \leftarrow \mathsf{F}_{k_0}(p_B) \oplus r$ // $y_i := \mathsf{F}_{k_i}(p_B)$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $c_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus m_i \quad \forall i = 1, \ldots, \ell$
- return $C \leftarrow (\tau, c_0, c_1, c_2, \ldots, c_l)$

$\mathsf{Dec}_k(C)$:
- Parse $C = (\tau, c_0, c_1, c_2, \ldots, c_l)$
- $k_0 \leftarrow \mathsf{F}^*_{k_E}(\tau)$
- $r \leftarrow \mathsf{F}_{k_0}(p_B) \oplus c_0$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $m_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus c_i \quad \forall i = 1, \ldots, \ell$
- $k'_0 \leftarrow \mathsf{F}^*_{k_M}(r)$
- $k'_i \leftarrow \mathsf{F}_{k'_{i-1}}(m_i) \quad \forall i = 1, \ldots, \ell$
- if $k'_l = \tau$ return $(m_1, ..., m_l)$, else return $\perp$.

4.3 Insecurity of single-key variant

One natural improvement of the PSV-AE would be to use a single long-term key k, i.e. $k_M = k_E$, for its two leak-free components. We show here that misuse resistance falls down in this case due to the following attack (when $\ell = 2$). (1) The adversary $\mathcal{A}$ requests an encryption on (r, m_1, m_2), where r is any chosen value of $\{0,1\}^n$. $\mathcal{A}$ receives back (τ, c_0, c_1, c_2) (2) $\mathcal{A}$ requests a second encryption on (τ, p_A, p_B) and gets $(\tau', c'_0, c'_1, c'_2)$. These queries allow $\mathcal{A}$ to distinguish the evaluation of the real encryption scheme from the function which outputs random elements. Indeed, $\mathcal{A}$ simply has to test whether $c_1 \oplus \tau' = m_1$ holds: if $\mathcal{A}$ is facing the real encryption, this always passes the test, and if $\mathcal{A}$ receives random answer, this test fails with overwhelming probability. Misuse resistance no more holds.

4.4 Misuse with leakage

We now consider the case where the adversary can not only control the random coins r of Figures 3 and 4 but also observe the leakages due to the computation of these schemes. In this context, we show that the leakage security argument used in [27] directly falls down, leading to an even more serious (forgery) attack. For this purpose, let us start by considering the variant of PSV-AE with a single long-term key (of which misuse resistance has just been broken) and assume F is instantiated with a block cipher such as the AES. In this case, by keeping the r constant, the adversary can ensure that the ephemeral keys k'_0 and k_0 actually become long-term secrets, which can directly be turned into a forgery attack. More precisely, the adversary can proceed as follows: (1) Fix the random coin r. (2) Recover k'_0 via standard DPA by encrypting several (different) ℓ-block messages m. (3) Pick up a new $(\ell - 1)$-block message m' that has not been encrypted yet. (4) For $i = 1$ to $\ell - 1$, compute $k'_i = F_{k'_{i-1}}(m_i)$. (5) Compute $m_\ell = F^{-1}_{k'_{\ell-1}}(r)$. (6) Encrypt m' with k'_0.

Intuitively, the attack essentially works by adapting the last message block m_ℓ so that the tag produced corresponds to the chosen coin r, which ensures that the two ephemeral keys k'_0 and k_0 are identical which consequently allows forging valid ciphertexts (without the target chip).

Now just observe that for the actual PSV-AE scheme with two long-term keys, the forgery of a valid tag (for the first part of the authenticated encryption scheme) remains unchanged. And the

forgery of valid ciphertexts could simply take advantage of the fact that the encryption mode of Figure 4 emulates a one-time pad so that if the ephemeral key k'_0 is fixed (despite unknown), it is directly possible to forge the ciphertexts corresponding to any linear combination of previously observed messages.[4] As explained in Appendix A, the alternative leakage-resilient MAC proposed in [27] could be broken by similar (though slightly more elaborate) attacks.

5 Digest, Tag and Encrypt (DTE)

In this section we investigate whether we can get better efficiency and security than the previous composition. For this purpose, we build a scheme with a single key which enjoys a form of leakage-resilience.

5.1 Specifications

We apply the DIV composition – with the exception that k_M and k_E are replaced by a unique key k – to another IV-based MAC combined with PSV-ENC, resulting in DTE. This MAC uses a hash function as a sub-ingredient, hence the name DTE for digest, tag, and encrypt.

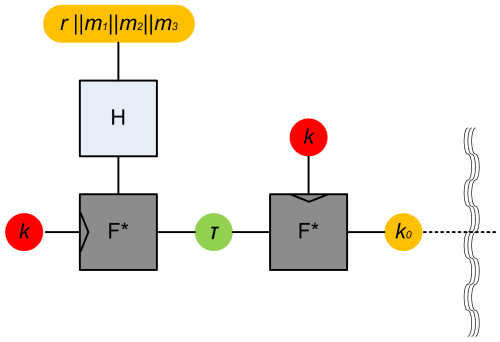

Fig. 5. DTE leakage-resilient AE (part I). Part II is identical to Fig. 4.

The full description of DTE is given below. The values p_A and p_B are public constants in $\{0,1\}^n$.

5.2 Misuse-resistance without leakage

It would be possible to prove the misuse resistance of DTE by applying Theorem 1 and showing that the single-key variant is indistinguishable from the two-key variant. However, to get a better bound, we analyze the misuse-resistance of DTE from scratch.

Theorem 3. *Let* $\mathsf{H} : \{0,1\}^n \times \{0,1\}^\star \to \{0,1\}^n$ *be a* $(0, t_1, \varepsilon_{cr})$*-collision resistant and* $(1, t_1, \varepsilon_{pr})$*-range-oriented preimage resistant hash function. Let* $\mathsf{F}^\star : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *be a* $(2q, t_1, \varepsilon_{\mathsf{F}^\star})$*-pseudorandom function and* $\mathsf{F} : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *be a* $(2, t_2, \varepsilon_{\mathsf{F}})$*-pseudorandom function. Then the* DTE *authenticated encryption scheme which encrypts* ℓ*-block messages is* (q, t, ε)*-misuse resistant as long as* $t \leq \min\{t_1 - q(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}}), t_2 - q_e(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}})\}$ *with* $0 \leq q_e + q_d \leq q$*, where* q_e *(resp.* q_d*) is the number of encryption (resp. decryption) queries, where* t_{H} *and* t_{F} *are the time needed to evaluate* H *and* F*, and we have*

$$\varepsilon \leq \varepsilon_{\mathsf{F}^\star} + \varepsilon_{cr} + 2q \cdot \varepsilon_{pr} + q(\ell+1) \cdot \varepsilon_{\mathsf{F}} + \left(q_d + q_e^2 + q_e^2(\ell+1)^2\right) \cdot 2^{-n}.$$

[4] Note that recovering the k_0 corresponding to the selected $\tau = r$ thanks to SPA is also feasible (see Section 6).

DTE

$\mathsf{Enc}_k(m)$:
- Parse $m = (m_1, m_2, \ldots, m_\ell)$
- $r \xleftarrow{\$} \{0,1\}^n$
- $h \leftarrow \mathsf{H}(r||m)$ // digest
- $\tau \leftarrow \mathsf{F}^*_k(h)$ // tag
- $k_0 \leftarrow \mathsf{F}^*_k(\tau)$ // ...and encrypt
- $c_0 \leftarrow \mathsf{F}_{k_0}(p_B) \oplus r$ // $y_i := \mathsf{F}_{k_i}(p_B)$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $c_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus m_i \quad \forall i = 1, \ldots, \ell$
- return $C \leftarrow (\tau, c_0, c_1, c_2, \ldots, c_\ell)$

$\mathsf{Dec}_k(C)$:
- Parse $C = (\tau, c_0, c_1, c_2, \ldots, c_\ell)$
- $k_0 \leftarrow \mathsf{F}^*_k(\tau)$
- $r \leftarrow \mathsf{F}_{k_0}(p_B) \oplus c_0$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $m_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus c_i \quad \forall i = 1, \ldots, \ell$
- $h \leftarrow \mathsf{H}(r||m)$
- if $\tau = \mathsf{F}^*_k(h)$ return $(m_1, ..., m_\ell)$, return $\perp$.

The guideline of the proof follows the same principle as in the proof of Theorem 1 for the generic construction. First we start by arguing that all decryption queries can be answered by $\perp$ and then answers to encryption queries are gradually replaced by random outputs, block by block.

The easiest transition relies on the pseudorandmoness of $\mathsf{F}^\star$, which is replaced by a truly random function f. Therefrom, we can move to show the invalidity of the first fresh decryption query $C = (\tau, c)$, where $c = (c_0, c_1, \ldots, c_\ell)$. Since (τ, c) is fresh, we will see that the decrypted tuple $(r, m = (m_1, \ldots, m_\ell))$ is fresh. Thereby, the collision resistance ensures that $h = \mathsf{H}(r\|m)$ is not the output of any previous evaluation of H during the encryption queries. If h never appeared until the first decryption query, then $\mathsf{f}(h) \neq \tau$ except by pure chance. However, we must also consider the event by which $h = \tau'$, where τ' is the returned tag associated to some previous encryption query.[5] Hence the need of the range-oriented preimage resistance of H since, as an output of f, τ_i is random over $\{0,1\}^n$.

In order to ease the process of the proof we rely on an equivalent definition of range-oriented preimage resistance where an adversary can ask as many targets $y \xleftarrow{\$} \{0,1\}^n$ as wanted to return one preimage. A straightforward hybrid argument shows that any efficient (q,t)-adversary receiving at most q targets can be reduced to an adversary receiving at most one target with a security loss factor q, in the same time.

Remark 1. A stronger flavor of range-oriented preimage resistance was introduced by Rogaway and Shrimpton [32] under the name of *everywhere preimage resistance*. In their definition, the hardness of computing preimage must hold for all the possible targets $y \in \{0,1\}^n$ and thus for all the possible distributions over $\{0,1\}^n$ where y is sampled. Like standard preimage resistance, Definition 2 focuses on the hardness over a single distribution.

Proof. By the definition of misuse resistance and using the same notation as in the proof of Theorem 1, for any efficient adversary $\mathcal{A}$ against DTE, we show that the following advantage is negligible:

$$\left| \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}, \mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}} \Rightarrow 1] - \Pr[\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}, \mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}} \Rightarrow 1] \right|.$$

Let $\mathcal{A}$ be a (q, t, ε)-adversary against the misuse resistance of DTE, meaning that the above distance is bounded by ε when $\mathcal{A}$ runs in time t and makes at most $q \geq q_e + q_d$ queries, where q_e is the

[5] Intuitively, this event would reveal information to the adversary: since $\mathsf{f}(\tau') = k'_0$ and is revealed to the adversary, we could not state that $\mathsf{f}(h)$ is independent from all other observed values.

number of encryption queries made by $\mathcal{A}$ to $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}$ or $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}$, and where q_d is the number of decryption queries made by $\mathcal{A}$ to $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}$ or $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}$ respectively. Without loss of generality, we assume that any answer to some encryption query is not later sent as a decryption query and that any answer to some decryption query is not later sent as an encryption query. Indeed, if it were not the case we could simply send back the input of the previous query as the answer to the later query. Moreover, we also assume that a same query is never made twice for encryption and decryption.

We will use a sequence of hybrid games, beginning with the real game, named game 0, where $\mathcal{A}$ interacts with $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}}$ and $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}$ and ending with random-and-invalid game, named game 3, where $\mathcal{A}$ interacts with $\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}}$ and $\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}$. Then, using the adversary $\mathcal{A}$ we show that any transition can be reduced to either an efficient distinguisher against the PRF $\mathsf{F}^\star$ or F, or to an efficient algorithm that either outputs a collision or a range-oriented preimage of H. We name E_i the event whereby $\mathcal{A}$ outputs 1 at the end of the game i. We then start from $\Pr[E_0] = \Pr[\,\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{real}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{real}}} \Rightarrow 1]$ and we end with $\Pr[E_3] = \Pr[\,\mathcal{A}^{\mathcal{O}^{\mathsf{Enc}}_{\mathsf{rand}},\mathcal{O}^{\mathsf{Dec}}_{\mathsf{fake}}} \Rightarrow 1]$.

We define game 1 as game 0, except that, in the computation of the encryption and decryption, the function $\mathsf{F}^\star$ is replaced by a random function f, namely we assume that f is chosen uniformly at random within all the possible functions from $\{0,1\}^n$ to $\{0,1\}^n$. In order to show that $|\Pr[E_0] - \Pr[E_1]|$ is negligible we build a challenger $\mathcal{B}_1$, which on input 1^n and given an oracle access to either $\mathsf{F}^\star_k$ or f, has to distinguish which functions the oracle evaluates. The challenger $\mathcal{B}_1$ picks $p_A, p_B \leftarrow \{0,1\}^n$ and H at random and emulates the encryption and decryption oracles interacting with $\mathcal{A}$ as follows. On each query made by $\mathcal{A}$, the challenger runs all the steps described by DTE, except for the computation of the tags τ and the ephemeral key k_0: $\mathcal{B}_1$ calls its own oracle on h to get τ and then it calls it again on τ to get k_0. When $\mathcal{A}$ outputs its guess bit $\mathcal{B}_1$ simply returns that bit as its own guess. Obviously, depending on whether $\mathcal{B}_1$ is given oracle access to either $\mathsf{F}^\star$ or f, $\mathcal{A}$ is playing game 0 or game 1. Therefore, any difference between $\Pr[E_0]$ and $\Pr[E_1]$ leads to the same difference in distinguishing $\mathsf{F}^\star$ from f, making $\mathcal{B}_1$ a $(2q, t + q(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}}))$-adversary against the PRF $\mathsf{F}^\star$, since two evaluations of the function are needed in each encryption and each decryption emulation, and where t_{H} and t_{F} are the time needed to evaluate H and F. Moreover, by assumption we have $t + q(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}}) \leq t_1$ and $\mathsf{F}^\star$ is $(2q, t_1, \varepsilon_{\mathsf{F}^\star})$-pseudorandom so that $|\Pr[E_0] - \Pr[E_1]| \leq \varepsilon_{\mathsf{F}^\star}$.

In game 1 we consider two events F_1 and F_2 depending on some efficiently checkable properties related to the encryption and decryption queries. To each encryption query (r_i, m_i) for some $m_i = (m_1^i, \ldots, m_\ell^i)$ we associate the digest $h_i = \mathsf{H}(r_i \| m_i)$ and the tag $\tau_i = \mathsf{f}(h_i)$. To each decryption query $C_j = (\tau_j, c_j)$ for some $c_j = (c_0^j, c_1^j, \ldots, c_\ell^j)$ we associate the digest $h_j = \mathsf{H}(r_j \| m_j)$ and the tag τ_j, where r_j and $m_j = (m_1^j, \ldots, m_\ell^j)$ are computed during the (emulation of the) decryption, but we consider this association only if (τ_j, c_j) is considered valid in the game, namely if $\mathsf{f}(h_j) = \tau_j$. First, we define F_1 as the event that at least two *associated* digests are equal. Second, we define F_2 as the event that some *associated* digest is equal to some *associated* tag. If we let F_3 be the complement of $F_1 \cup F_2$, we have $\Pr[E_1] \leq \Pr[F_1] + \Pr[F_2] + \Pr[E_1 | F_3]$.

We use the collision resistance of H to bound $\Pr[F_1]$. If F_1 occurs it happens that $\mathsf{H}(r\|m) = \mathsf{H}(r'\|m')$ whereas (r, m) and (r', m') come from (answer to) distinct queries. If some of this tuple, say (r, m), is the answer to some decryption query, say (τ, c), it must be the case that a re-encryption of (r, m) gives back (τ, c), since the random coin is fixed and $\tau = \mathsf{f}(\mathsf{H}(r\|m))$. Actually, the (emulated) encryption algorithm is deterministic given the coin and the message and we avoid that $(\tau, c) = (\tau', c')$ happens in our analysis. Therefore, we must have $(r, m) \neq (r', m')$ in all the cases. As a consequence, in F_1, it is easy to build an adversary having the same running time and answering queries in the same way as $\mathcal{B}_1$ and which is an adversary against the $(0, t_1, \varepsilon_{cr})$-collision resistant hash function H. By assumption we have $t + q(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}}) \leq t_1$ and then $\Pr[F_1] \leq \varepsilon_{cr}$.

We use the range-oriented preimage resistance of H to bound $\Pr[F_2]$. Actually, we have $\Pr[F_2] = \Pr[F_2']$ where F_2' is the analogue of F_2 in game 1'. The difference between game 1 and game 1' is purely conceptual. In game 1', instead of picking a random function f that is directly evaluated, the outputs are now implicitly computed by the challenger $\mathcal{B}_1'$ against the range-oriented preimage resistance of H. It means that each time $\mathcal{B}_1'$ needs a new evaluation of the random function f to emulate encryption and decryption as $\mathcal{B}_1$ does, $\mathcal{B}_1'$ simply requests a new random target in $\{0,1\}^n$. If F_2' occurs, it happens that some associated digest $h = \mathsf{H}(r\|m)$ is equal to some associated tag τ' which is among the at most $2q$ targets. Since $\mathsf{H}(r\|m) = \tau'$, $\mathcal{B}_1'$ is at most a $(2q, t + q(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}}))$-adversary against the multiple target range-oriented preimage resistance which is $(2q, t_1, 2q \times \varepsilon_{pr})$-secure if H is a $(1, t_1, \varepsilon_{pr})$-range-oriented preimage resistant hash function (with one target) and we must have $\Pr[F_2'] \leq 2q\varepsilon_{pr}$. (Note that it is equivalent that $\mathcal{B}_1'$ receives all the targets at the beginning of game 1'.)

We go on with bounding $\Pr[E_1|F_3]$. We define game 2 as the conditional game "game 1 if F_3", except that to each decryption query we return $\perp$ (we do not modify the emulation of the encryption). Clearly, $|\Pr[E_1|F_3] - \Pr[E_2]| = \Pr[F_4]$ where F_4 occurs if some of the q_d ciphertexts sent as decryption requests were valid, as the new game will deem them invalid. If F_4 occurs, it must hold that for some $h = \mathsf{H}(r, m)$ computed from a decryption query (τ, c) we have $\mathsf{f}(h) = \tau$. However, since F_3 also occurs, h was never an input of f before the challenger checks the validity of the ciphertext and then $\mathsf{f}(h)$ remains completely independent of the adversary's view. Therefore, $\mathsf{f}(h) \neq \tau$ except by pure chance and we thus get $\Pr[F_4] \leq q_d/2^n$. To ensure freshness of the tags associated to encryption queries in the remaining part of the proof we define game 2' as game 2, except that we abort the game if a collision on these tags occurs. We then have $|\Pr[E_2] - \Pr[E_2']| \leq q_e(q_e+1)/2^{n+1}$.

In game 2' we reach a game where decryption gives no information to the adversary. We will show in game 3 how all the q_e $(\ell+2)$-block ciphertexts of game 2' can be indistinguishably replaced by q_e random $\ell+2$ blocks of $\{0,1\}^n$. Actually, all the tags τ are already distinct and random from game 2'. Therefore, for each encryption query (r_i, m_i), where $m_i = (m_1^i, \ldots, m_\ell^i)$, for $1 \leq i \leq q_e$, it is enough to show that from the ephemeral keys $(k_0^i, k_1^i \ldots, k_\ell^i)$ the tuple $(y_0^i, y_1^i, \ldots, y_\ell^i)$ computed in the emulation of the encryption of game 2' can be replaced by a uniformly random tuple. Indeed, we have $c_i = (c_0^i, c_1^i, \ldots, c_\ell^i) = (m_1^i, \ldots, m_\ell^i) \oplus (y_0^i, y_1^i, \ldots, y_\ell^i)$ and $C_i = (\tau_i, c_i)$. For that purpose, we consider $q_e(\ell+1)$ hybrid games $G_{i,j}$, for $1 \leq i \leq q_e$ and $0 \leq j \leq \ell$, such that each block pair (k_{v+1}^u, y_v^u) with $(u < i)$ or $(u = i \wedge v \leq j)$ are random[6] and $k_{\ell+1}^i$ never appeared yet in the game[7] whereas all the remaining pairs are computed as in game 2' resuming from the random but fresh ephemeral key k_{j+1}^i if $(j < \ell)$ or k_0^{i+1} if $(j = \ell \wedge i < q_e)$. We move from one hybrid to the other with respect to the lexicographic order of the indexes (i, j). At the end of these transitions we reach game 3 since it is equal to game $G_{q_e,\ell}$.

We rename game 2' as game $G_{0,\ell}$ in order to show that each transition from game $G_{i,\ell}$ to game $G_{i+1,0}$, for $0 \leq i < q_e$, can be reduced to the pseudorandomness of F. In game $G_{i,\ell}$ and game $G_{i+1,0}$, the i first encryption queries are answered with random ciphertexts and $f(\tau_{i+1}) = k_0^{i+1}$ is already random. But, in game $G_{i,\ell}$ we have $\mathsf{F}_{k_0^{i+1}}(p_A) = k_1^{i+1}$ and $\mathsf{F}_{k_0^{i+1}}(p_B) = y_0^{i+1}$ whereas in game $G_{i+1,0}$ the ephemeral key k_1^{i+1} and y_0^{i+1} are random and is k_1^{i+1}. The remaining parts of the encryption from k_1^{i+1} are the same in both games and as in game 2' as well as the answers to all the next encryption queries. Let $\mathcal{B}_{i+1,0}''$ be a challenger against the $(2, t_2, \varepsilon_{\mathsf{F}})$-pseudorandom F. To emulate ciphertexts, $\mathcal{B}_{i+1,0}''$ answers with random outputs to the i first encryption queries. Since τ_{i+1} never appears until this time, k_0^{i+1} has never been generated yet. Therefore, instead of computing k_0^{i+1},

[6] Note that we introduce a dummy ephemeral key $k_{\ell+1}^i$ which is not used in encryption.

[7] Otherwise, the next y_{j+1}^i could not be indistinguishably replaced by an independent value since $\mathsf{F}_{k_{\ell+1}^i}(p_B)$ would already be defined in the adversary's view.

$\mathcal{B}''_{i+1,0}$ calls its PRF oracle on p_A and p_B to get k_1^{i+1} and y_0^{i+1} and aborts if k_1^{i+1} matches a previous ephemeral key. From these outputs, $\mathcal{B}''_{i+1,0}$ is efficiently able to end the emulation as specified in game 2'. Obviously, we have $|\Pr[E_{i,\ell}] - \Pr[E_{i+1,0}]| \leq \varepsilon_{\mathsf{F}} + [i(\ell+1)+1]/2^n$ since $\mathcal{B}''_{i+1,0}$ runs in time at most $t + q_e(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}})$ which is less than t_2 by assumption and where $[i(\ell+1)+1]/2^n$ is the probability that that challenger aborts. (We stress that $\mathcal{B}''_{i+1,0}$ actually computes at most $q_e - i$ hashes/digests and $2\ell(q_e - i) - 2$ evaluations of F).

We now move on to transition from game $G_{i,j}$ to game $G_{i,j+1}$, for each $1 \leq i \leq q_e$ and each $0 \leq j < \ell$, whose indistinguishability is also implied by the pseudorandomness of F. In game $G_{i,j}$ and game $G_{i,j+1}$ the $i-1$ first encryption queries are answered with random ciphertexts and in the answer to the $i-th$ encryption query $(k_0^i, \ldots, k_j^i, k_{j+1}^i)$ and $(y_0^i, \ldots, y_j^i)$ are already random. However, in game $G_{i,j}$ we have $\mathsf{F}_{k_{j+1}^i}(p_A) = k_{j+2}^i$ and $\mathsf{F}_{k_{j+1}^i}(p_B) = y_{j+1}^i$ whereas in game $G_{i,j+1}$ the ephemeral key k_{j+2}^i and y_{j+1}^i are random and k_{j+2}^i is fresh. The remaining parts of the encryption are the same in both games and as in game 2'. Let $\mathcal{B}''_{i,j+1}$ be a challenger against the $(2, t_2, \varepsilon_{\mathsf{F}})$-pseudorandom F. To emulate ciphertexts, $\mathcal{B}''_{i,j+1}$ answers with random outputs to the $i-1$ first encryption queries. For the $i-th$ encryption query, $\mathcal{B}''_{i,j+1}$ picks $y_0^i, \ldots, y_j^i \leftarrow \{0,1\}^n$ at random and, instead of picking a random k_{j+1}^i, it calls its PRF oracle on p_A and p_B to get k_{j+2}^i and y_{j+1}^i and aborts if k_{j+2}^i is not fresh. From these outputs, $\mathcal{B}''_{i,j}$ can end the emulation as specified in game 2'. Obviously, we have $|\Pr[E_{i,j}] - \Pr[E_{i,j+1}]| \leq \varepsilon_{\mathsf{F}} + [(i-1)(\ell+1) + (j+2)]/2^n$ since $\mathcal{B}''_{i,j+1}$ runs in time at most $t + q_e(t_{\mathsf{H}} + 2\ell t_{\mathsf{F}})$ which is less than t_2 by assumption and where $[(i-1)(\ell+1)+(j+2)]/2^n$ is the probability that the challenger aborts. (We stress that $\mathcal{B}''_{i,j+1}$ actually computes at most $q_e - i$ hashes/digests and $(q_e - i)2\ell + 2(\ell - j) - 2$ evaluations of F).

Putting all the probabilities together we find,

$$\begin{aligned}|\Pr[E_0] - \Pr[E_3]| \leq |\Pr[E_0] - \Pr[E_1]| + \Pr[F_1] + \Pr[F_2'] \\ + \Pr[F_4] + |\Pr[E_2] - \Pr[E_2']| + |\Pr[E_2'] - \Pr[E_3]|,\end{aligned}$$

where the last term $|\Pr[E_2'] - \Pr[E_3]|$ is bounded by

$$\begin{aligned}&\sum_{i=1}^{q_e} \Big(|\Pr[E_{i-1,\ell}] - \Pr[E_{i,0}]| + \sum_{j=0}^{\ell-1} |\Pr[E_{i,j}] - \Pr[E_{i,j+1}]| \Big) \\ &\quad \leq\ q_e(\ell+1) \cdot \varepsilon_{\mathsf{F}} \ +\ \frac{1}{2^n} \cdot \sum_{i=0}^{q_e-1} \sum_{j=0}^{\ell} i(\ell+1) + (j+1)\end{aligned}$$

so that $\varepsilon = |\Pr[E_0] - \Pr[E_3]|$ is bounded by

$$\begin{aligned}\varepsilon_{\mathsf{F}^\star} + \varepsilon_{cr} + 2q \cdot \varepsilon_{pr} + q_d \cdot 2^{-n} + q_e(q_e+1) \cdot 2^{-n-1} \\ + q_e(\ell+1) \cdot \varepsilon_{\mathsf{F}} + q_e(\ell+1)[q_e(\ell+1)+1] \cdot 2^{-n-1}\ ,\end{aligned}$$

which concludes the proof. □

5.3 Ciphertext integrity with misuse & leakage

We now generalize the definition of *ciphertext-integrity* given by Bellare and Namprempre [6] to capture both the ability for an adversary to generate the random coins and to learn more information from a leakage function of the encryption algorithm. In this respect, a significant issue that appears when trying to deal with side-channel information in a formal way is the problem of defining and

quantifying, or at least upper-bounding, the amount of information that can be disclosed through side-channel leakage. We will indeed have to tackle this problem in Section 7. However, in the current section, we are able to provide a positive result (i.e., a provably secure construction) in a conservative context where no assumptions are needed to limit the amount of information leaked to the adversary (beyond our leak-free execution). We first define this *unbounded leakage* model.

Definition 11. *We define the unbounded leakage model as a model in which, when queried, oracles return, in addition to the usual output values, a function* L *yielding all ephemeral keys and random coins generated during the computation of the oracle's answer.*

So this definition is in line with our assumptions (in Section 2.2) regarding the availability of two distinct pseudorandom functions: a cheap, leaking F, and an expensive but leak-free component F^*. In other words, our model assumes that the only values that will not be revealed through leakage are the long-term secrets used exclusively in combination with F^*. *All* other processed data and intermediary values are supposed to be totally disclosed through leakage, which is modeled by the fact that they can be reconstructed by the attacker based on the output of L by the oracle. As an example, in the unbounded leakage model, our DTE construction yields the leakage function $\mathsf{L}(r, m; k) := k_0$ (r is not explicitly needed as part of the output as it can be reconstructed from k_0).

Considering an authenticated encryption $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$, we define the CIML experiment, in which the adversary tries to generate a fresh valid ciphertext having access to unbounded leakage during encryption queries in addition to the encryption and decryption oracle. Note again that as in [27], the adversary is not given the leakage during decryption queries (see the conclusions section for a complementary discussion on this issue).

CIML experiment	
Initialization: $k \xleftarrow{\$} \mathcal{K}$ $\mathcal{S} \leftarrow \emptyset$ Finalization: $C \leftarrow \mathcal{A}^{\mathsf{EncL}_k(\cdot,\cdot),\mathsf{Dec}_k(\cdot)}$ If $C \in \mathcal{S}$, return $b = 0$ If $\perp = \mathsf{Dec}_k(C)$, return $b = 0$ return $b = 1$	Oracle $\mathsf{EncL}_k(r, m)$: $C = \mathsf{Enc}_k(r, m)$ $\mathcal{S} \leftarrow \mathcal{S} \cup \{C\}$ return $(C, \mathsf{L}(r, m; k))$ Oracle $\mathcal{O}^{\mathsf{Dec}_k(C)}$: return $\mathsf{Dec}_k(C)$

Definition 12. *An authenticated encryption* AE *provides* (q, t, ϵ)-ciphertext integrity with coin misuse and unbounded leakage on encryption *if for all* (q, t)*-bounded adversaries* $\mathcal{A}$*, we have:*

$$\Pr\left[\mathsf{CIML}_{\mathsf{AE},\mathcal{A}} \Rightarrow 1\right] \leq \epsilon.$$

As usual, q is an upper bound on the total number of queries made to oracles.

We now prove that DTE meets this definition.

Theorem 4. *Let* $\mathsf{H} : \{0,1\}^n \times \{0,1\}^\star \to \{0,1\}^n$ *be a* $(0, t', \varepsilon_{cr})$*-collision resistant and* $(1, t', \varepsilon_{pr})$*-range-oriented preimage resistant hash function. Let* $\mathsf{F}^* : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ *be a* $(2q + 2, t', \varepsilon_{\mathsf{F}^*})$*-pseudorandom function. Then* DTE *provides* (q, t, ε)*-ciphertext integrity with coin misuse and unbounded leakage on encryption as long as* $t \leq t' - (q+1)(t_{\mathsf{H}} + 2lt_{\mathsf{F}})$ *where* t_{H} *and* t_{F} *are the time needed to evaluate* H *and* F*, and we have*

$$\varepsilon \leq \varepsilon_{\mathsf{F}^\star} + \varepsilon_{cr} + 2q \cdot \varepsilon_{pr} + (q+1) \cdot 2^{-n}.$$

In the unbounded leakage model applied to DTE the leakage function L returns the ephemeral key k_0 computed during the encryption of some (r, m). Given k_0 the adversary is able to derive all the ephemeral keys $(k_0, k_1, \ldots, k_\ell)$ used during encryption queries. For instance, the adversary is able to re-compute $(y_0, y_1, \ldots, y_\ell)$ as in the encryption algorithm using F and check whether $(c_0, c_1, \ldots, c_\ell) = (r, m_1, \ldots, m_\ell) \oplus (y_0, y_1, \ldots, y_\ell)$ holds. Theorem 4 shows that whether F is pseudorandom or not has no impact on the success probability $\Pr[\mathsf{CIML}_{\mathsf{DTE},\mathcal{A}}] = \varepsilon$.

Proof. Let $\mathcal{A}$ be a (q, t)-CIML adversary against DTE making $q_e + q_d \leq q$ queries, where q_e is the number of encryption queries and q_d the number of decryption queries. We say that the final output ciphertext $(\tau^\dagger, c^\dagger)$ is the $(q+1)$-th query of the game. Without loss of generality we assume that any answer to some encryption query is never sent as a decryption query and conversely. We also assume that the final output is not an answer to some encryption query, otherwise the adversary looses anyway.

Since we are in the same condition as in the proof of misuse resistant, we name by $\bar{E}_i$ the event where the winning condition of CIML is satisfied which can be viewed as the analogue of E_i with an additional decryption query: the $(q+1)$-th query which is the last of the game. We thus have to focus on proving that the $(q+1)$-th query is also invalid even when all the ephemeral key k_0's associated to the encryption queries only are given in E_i.

Let see what happens in E_1 where F^* was replaced by a random function f if $\mathsf{f}(\tau) = k_0$ was given to the adversary, where $\tau = \mathsf{f}(\mathsf{H}(r\|m))$ for the encryption query (r, m). Obviously, k_0 gives nothing more since in E_1 the encryption algorithm from k_0 is run honestly as in E_0. We then get an adversary against F^* in $\bar{E}_1$ making at most $2(q+1)$ queries since we must count the $(q+1)$-th query and running in time bounded by $t + (q+1)(t_{\mathsf{H}} + 2lt_{\mathsf{F}}) \leq t'$. Nevertheless, we assume F^* to be $(2q+2, t', \varepsilon_{\mathsf{F}^*})$-pseudorandom and we find $|\Pr[\bar{E}_0] - \Pr[\bar{E}_1]| \leq \varepsilon_{\mathsf{F}^*}$.

Likewise with E_1, we consider the partition $\bar{E}_1 \cap (\bar{F}_1 \cup \bar{F}_2)$ and $\bar{E}_1 \cap \bar{F}_3$, where $\bar{F}_1$ is the analogue of F_1 meaning that collision on *associated* digests occurs, where $\bar{F}_2$ is an extended version of F_2 where some *associated* digest $\mathsf{H}(r, m) = h$ is equal to some *associated* τ' or to some *associated* k_0'' (which simply has the form $\mathsf{f}(\tau'')$ for some associated τ''), and where $\bar{F}_3$ is the complement of $\bar{F}_1 \cup \bar{F}_2$. We stress that the fact that the k_0's associated to encryption queries leak does not affect the emulations made in F_1, F_2 and F_3 since we remain in the same game. It is now straightforwards that $\Pr[\bar{F}_1] \leq \varepsilon_{cr}$ since we get an adversary against the $(0, t', \varepsilon_{cr})$-collision resistance of H running in the time bounded by t'. Moreover, since in F_2' we already put targets of the range-oriented preimage resistance of H in place of all the associated tags *and* the associated ephemeral key k_0's we also have an adversary here (built from $\mathcal{A}$), for $\bar{F}_2$, asking/receiving at most $(2q+2)$ targets and running in time bounded by t'. By assumption on H, we must have $\Pr[\bar{F}_2] \leq \varepsilon_{cr}$ and we are thus left with bounding $\Pr[\bar{E}_1 | \bar{F}_3]$.

We are ready for the last transition from $\bar{E}_1 | \bar{F}_3$ to $\bar{E}_2$ where we reach the game where all the decryption queries including the $(q+1)$-th one are answered by $\perp$. It is straightforward to show that $|\Pr[\bar{E}_1 | \bar{F}_3] - \Pr[\bar{E}_2]| \leq (q+1)/2^n$, which concludes the proof. □

6 On the impossibility of misuse-resistance with leakage

The previous section showed that it is possible to combine some form of misuse-resistance and leakage-resilience, as formalized by the (concretely achievable) notion of ciphertext integrity with misuse and leakage. Yet, this security notion is admittedly weaker than a combination of standard misuse resistance (such as discussed in Sections 3, 4.2, 5.2) and leakage-resilience. In this section, we argue that such a stronger security notion may be impossible to achieve from standard block ciphers and hash functions only.

For this purpose, we focus on the second part of the previous authenticated encryption schemes, represented in Figure 4. In particular, we observe that by fixing the randomness r, we again have the problem that the ephemeral key k_0 will be fixed. So it is theoretically possible to recover this ephemeral key via SPA. At this point is is worth insisting that such an SPA would require to obtain noise-free measurements of the encryptions of p_A and p_B under k_0 and to perform some kind of algebraic/analytical attack as mentioned in Footnote 3. So it is admittedly more challenging to perform than the DPA of Section 4.4 which takes advantage of the fact that the key $\tilde{k}'_0$ can encrypt an arbitrary number of messages. On the other hand, and from a theoretical point-of-view, it is also impossible to argue why this attack should not be covered by our threat model. Based on this fact, it is clear that full misuse-resistance with leakage is not be achievable with the types of constructions considered in this work. Indeed, recovering the ephemeral key k_0 of our encryption schemes trivially allows the adversary to distinguish his ciphertexts from random. To a large extent, we believe this is a fundamental limitation, since IV-misuse will always transform ephemeral keys into long-term ones. As mentioned in introduction, this observation can be seen as one more illustration of the separation between unpredictability-based and indistinguishability-based security in the presence of leakage [23]. Overall, this discussion confirms that ciphertext integrity may be the best security notion we can achieve for authenticated encryption based on standard symmetric cryptographic primitives, when an adversary can jointly exploit IV misuse and leakage.

7 Digest, Commit and Encrypt (DCE)

Motivated by the previous observation, we now present a construction that drops the requirement of misuse-resistance without leakage, and only guarantees ciphertext integrity when randomness misuse is granted to the adversary (with and without leakage). This construction has the advantage of only requiring one execution of the leak-free function, but at the expense of relying on the random oracle model for its proof of ciphertext integrity (yet not for its leakage-resilient CPA security, as will be shown in Section 8).

We note that the use of a random oracle assumption when analyzing implementation weaknesses is admittedly questionable (since the random oracle abstraction excludes leakage). However, and as discussed in [10,38,39], it sometimes comes in handy to argue about the security of natural constructions of which the leakage-resilience seems hard to reach in the standard model. In view of the practical interest of the next DCE construction, we therefore include a proof in this model in our treatment and suggest the further investigation of DCE instances as an interesting scope for further research. We note that our proof does not make use of the programmability of the random oracle, which is a common source of gaps in the soundness of schemes that are proven to be secure in this model but are insecure for any instantiation of the random oracle.

7.1 Specifications

The construction named DCE is based on Figure 6 below which is then plugged to Figure 4. The full specifications are described in the box where H is a hash function and p_A and p_B are constants from $\{0,1\}^n$. The key k is picked randomly from $\mathcal{K}$, as usual.

7.2 Ciphertext integrity with misuse & leakage

Theorem 5. *Let* $\mathsf{H} : \{0,1\}^n \times \{0,1\}^\star \longmapsto \{0,1\}^n$ *be modeled as a random oracle. Let* $\mathsf{F}^* : \{0,1\}^n \times \{0,1\}^n \longmapsto \{0,1\}^n$ *be* $(q+1, t', \varepsilon_{\mathsf{F}^*})$*-pseudorandom. Then,* DCE *provides* (q, t', ε)*-ciphertext-integrity*

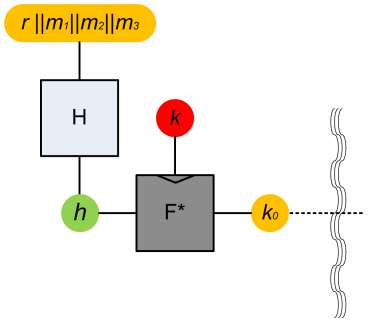

Fig. 6. DCE leakage-resilient AE (part I). Part II is identical to Fig. 4.

DCE

$\mathsf{Enc}_k(m)$:
- parse $m = (m_1, m_2, \ldots, m_l)$
- $r \xleftarrow{\$} \{0,1\}^n$
- $\mathsf{h} \leftarrow \mathsf{H}(r\|m)$
- $k_0 \leftarrow \mathsf{F}^*_k(\mathsf{h})$
- $c_0 \leftarrow \mathsf{F}_{k_0}(p_B) \oplus r$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $c_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus m_i \quad (\forall i = 1, \ldots, \ell)$
- return $C = (\mathsf{h}, c_0, c_1, c_2, \ldots, c_l)$

$\mathsf{Dec}_k(C)$:
- parse $C = (\mathsf{h}, c_0, c_1, c_2, \ldots, c_l)$
- $k_0 \leftarrow \mathsf{F}^*_k(\mathsf{h})$
- $r \leftarrow \mathsf{F}_{k_0}(p_B) \oplus c_0$
- $k_i \leftarrow \mathsf{F}_{k_{i-1}}(p_A)$, $m_i \leftarrow \mathsf{F}_{k_i}(p_B) \oplus c_i \quad (\forall i = 1, \ldots, \ell)$
- if $\mathsf{h} = \mathsf{H}(r\|m)$ return $m = (m_1, ..., m_l)$, else return $\perp$.

with coin misuse and unbounded leakage during encryption for l-block messages, where $t \leq t' - (q+1)(t_{\mathsf{H}} + 2lt_{\mathsf{F}})$, where t_{H} and t_{F} are the time needed to evaluate H and F, and we have

$$\varepsilon \leq \varepsilon_{\mathsf{F}^*} + 4(q+1)^2/2^n + (q+1)/2^n.$$

Proof. Let $\mathcal{A}$ be a (q,t)-CIML adversary against DCE making $q_e + q_d \leq q$ queries, where q_e is the number of encryption queries and q_d the number of decryption queries. We have to bound the probability $\Pr[\mathsf{CIML}_{\mathsf{DCE},\mathcal{A}} = 1]$. Without loss of generality we assume that any answer to some encryption query is never sent as a decryption query and conversely. We also assume that the final output is not an answer to some encryption query, otherwise the adversary looses anyway.

The proof is in the spirit of the proof of Theorem 4 except that $\mathcal{A}$ cannot compute H itself: it must query the random oracle to get h. However, since h is random here, the distribution of $\mathsf{F}^*_k(\mathsf{H}(r\|m))$ in DCE is similar to the distribution of $\mathsf{F}^*_k \circ \mathsf{F}^*_k(\mathsf{H}(r\|m))$ in DTE by relying on the pseudorandomness of F^*. Then, all the ephemeral keys k_0 *associated* to encryption queries are random (See the proof of Theorem 4).

Let us assume that the final output ciphertext $(\tau^\dagger, c^\dagger)$ is the $(q+1)$-th query of the game. Then we only need to replace $q+1$ outputs of F^*_k by random values (instead of computing k_0's). By reusing the argument detailed in the proof of Theorem 4, we obtain that the $(q+1, t', \varepsilon_{\mathsf{F}^*})$-pseudorandomness of F^* is sufficient to bound the gap resulting from this transition by $\varepsilon_{\mathsf{F}^*}$: we can easily build an adversary running in time $t + (q+1)2lt_{\mathsf{F}} \leq t'$, since all the h's are already random.

The probability that some collision occurs among all the h's and the k_0's is bounded by $4(q+1)^2/2^n$. Therefore, assuming that no collision happens, if a decryption query (h, c) is valid it must be the case that $\mathsf{H}(r\|m)$ returned by the random oracle where r and m are computed during decryption matches h which has a probability bounded by $1/2^n$ for each query. Thus all the ciphertexts of the encryption queries including the $(q+1)$-th one are invalid except with probability $(q+1)/2^n$. □

8 Leakage-resilient CPA security

The ciphertext integrity properties discussed in the previous sections do not imply anything about the confidentiality of the messages that are encrypted with the DTE and DCE schemes.

This section shows the leakage-resilient CPA security of these schemes, which is measured by the probability that an adversary distinguishes between playing the $\mathsf{PrivK}^{\mathsf{lmcpa},0}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}$ and $\mathsf{PrivK}^{\mathsf{lmcpa},1}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}$ games, defined below and borrowed from PSV [27]. This is essentially the traditional CPA game, with the addition that the adversary can access a leakage oracle L that can give him leakages from the attacked circuit on chosen inputs, and that the challenger provides leakages for any computation it performs, including the test query at step 3.

$\mathsf{PrivK}^{\mathsf{lmcpa},\mathsf{b}}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}$, with $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$, is the output of the following experiment:

1. Select $k \xleftarrow{\$} \mathcal{K}$
2. $\mathcal{A}^{\mathsf{L}}$ gets access to a leaking encryption oracle that, when queried on a message m of arbitrary block length, returns $\mathsf{Enc}_k(m)$ together with the leakage resulting from the encryption process.
3. $\mathcal{A}^{\mathsf{L}}$ submits two messages m_0 and m_1 of identical block length, to which he is replied with $\mathsf{Enc}_k(m_b)$ and the corresponding leakage.
4. $\mathcal{A}^{\mathsf{L}}$ can keep accessing the leaking encryption oracle.
5. $\mathcal{A}^{\mathsf{L}}$ outputs a bit b'.

The $\mathsf{PrivK}^{\mathsf{leav},\mathsf{b}}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}$ game [27], modeling leakage-resilient eavesdropper security, is defined just in the same way, except that the encryption oracles from steps 2 and 4 disappear.

Definition 13. *An authenticated encryption scheme* $\mathsf{AE} = (\mathcal{K}, \mathsf{Enc}, \mathsf{Dec})$ *with leakage function* L *is* (q, t, ϵ) lmcpa*-secure (resp.* leav*-secure) if, for every* (q, t)*-bounded adversary* $\mathcal{A}^{\mathsf{L}}$*, the advantage* $|\mathsf{PrivK}^{\mathsf{lmcpa},0}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}} - \mathsf{PrivK}^{\mathsf{lmcpa},1}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}|$ *(resp.* $|\mathsf{PrivK}^{\mathsf{leav},0}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}} - \mathsf{PrivK}^{\mathsf{leav},1}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}|$*) is bounded by* ϵ*.*

8.1 Background on the LMCPA security of the PSV-ENC scheme

Observing that the encryption part of all our schemes essentially follows the PSV-ENC scheme, we can hope to import the results of the previous analyzes of that scheme.

The security of an implementation of the PSV-ENC scheme relies on the assumption that the block cipher implementation that it uses has 2-simulatable leakages. (The unbounded leakage used in the previous sections does not make sense here anymore, since an unbounded leakage would trivially allow the adversary to win any confidentiality-related game.)

The notion of simulatable leakages is based on the q-sim-game below, from which q-simulatable leakages are defined. This game essentially measures the capability of a simulator to produce leakages that look consistent with given inputs and outputs of a block cipher, without knowing the key used in the computation.

Game q-$\mathsf{sim}(\mathcal{A}, \mathsf{F}, \mathsf{L}, \mathcal{S}, b)$ [35, Section 2.1].		
The challenger selects two random keys $k, k^* \xleftarrow{\$} \mathcal{K}$. The output of the game is a bit b' computed by $\mathcal{A}^{\mathsf{L}}$ based on the challenger responses to a total of at most q adversarial queries of the following type:		
Query	Response if $b = 0$	Response if $b = 1$
$\mathsf{Enc}(x)$	$\mathsf{F}_k(x)$, $\mathsf{L}(k, x)$	$\mathsf{F}_k(x)$, $\mathcal{S}^{\mathsf{L}}(k^*, x, \mathsf{F}_k(x))$
and one query of the following type:		
Query	Response if $b = 0$	Response if $b = 1$
$\mathsf{Gen}(z, x)$	$\mathcal{S}^{\mathsf{L}}(z, x, k)$	$\mathcal{S}^{\mathsf{L}}(z, x, k^*)$

Definition 14. [q-simulatable leakages [35, Defn. 1]] *Let F be a PRF having leakage function L. Then F is said to have $(q_{\mathcal{S}}, t_{\mathcal{S}}, q_{\mathcal{A}}, t_{\mathcal{A}}, \epsilon_{q\text{-}sim})$ q-simulatable leakages if there is a $(q_{\mathcal{S}}, t_{\mathcal{S}})$-bounded simulator $\mathcal{S}^{\mathsf{L}}$ such that, for every $(q_{\mathcal{A}}, t_{\mathcal{A}})$-bounded adversary $\mathcal{A}^{\mathsf{L}}$, we have:*

$$|\Pr[q\text{-}\mathsf{sim}(\mathcal{A}, \mathsf{F}, \mathsf{L}, \mathcal{S}^{\mathsf{L}}, 1) = 1] - \Pr[q\text{-}\mathsf{sim}(\mathcal{A}, \mathsf{F}, \mathsf{L}, \mathcal{S}^{\mathsf{L}}, 0) = 1]| \leq \epsilon_{q\text{-}sim}.$$

Based on this definition, the eavesdropper security of PSV-ENC can be summarized as follows.

Theorem 6 ([27], Thm 3.). *Let F be a $(q, t, \epsilon_{\mathsf{F}})$-PRF whose implementation has running time t_{F} and a leakage function L_{F} with $(q_{\mathcal{S}}, t_{\mathcal{S}}, q, t, \epsilon_{2\text{-}sim})$ 2-simulatable leakages.*

The advantage of every $(q - q_r, t - t_r)$-bounded $\mathcal{A}^{\mathsf{L}_{\mathsf{F}}}$ playing the $\mathsf{PrivK}^{\mathsf{leav,b}}_{\mathsf{PSV-ENC}}$ game is bounded by $\epsilon^{\mathsf{eav}}_{\mathsf{PSV-ENC}} = \ell(\mathbf{Adv}_s + 4(\epsilon_{\mathsf{F}} + \epsilon_{2\text{-}sim}))$ where $\mathbf{Adv}_s$ is a bound on the eavesdropper advantage of a $(q - q_{r'}, t - t_{r'})$-bounded adversary trying to distinguish the encryptions of two single-block messages encrypted with the PSV-ENC scheme, q_r, $q_{r'}$ are $\mathcal{O}(\ell q_{\mathcal{S}})$ and $t_r, t_{r'}$ are $\mathcal{O}(\ell(t_{\mathcal{S}} + t_{\mathsf{F}}))$.

This result relates the eavesdropper security of the PSV-ENC scheme to the security that is offered in front of an adversary who can only get a single encryption of a single block messages, which is expected to be much simpler to evaluate (see discussion in [27]). Note that, in our analysis below, we will not need to use any result about the CPA security of PSV-ENC.

8.2 Bounding hash function leakages

The security of the PSV-ENC scheme is going to be helpful for the encryption part of the DTE and DCE modes, but the first parts of our modes also include the evaluation of a hash function running on the message to be encrypted, which may in turn leak information about the message and help win the $\mathsf{PrivK}^{\mathsf{lmcpa,b}}_{\mathcal{A}^{\mathsf{L}},\mathsf{AE}}$ game: if the implementation of the hash function just leaks its input in full, we can obviously not hope for any confidentiality. We therefore turn to the definition of our security assumption about the hash function implementation, before analyzing the DCE and DTE schemes.

We need a bound on the distinguishing probability of an adversary who would see the leakages resulting from hashing something containing a message m_0 and those resulting from hashing something containing m_1. Simply assuming the indistinguishability of leakages on adversarially chosen m_0 and m_1 would be way too strong from a physical point of view: if an adversary knows m_0 and m_1, he can obtain leakages computed on these two values directly from the hash function implementation, and compare those leakages with the leakage returned by the challenger. Recognizing if a leakage matches another leakage seen before is typically an easy task.

Here, the adversary faces a more difficult problem, since he is not able to predict what message is hashed when he gets leakages to distinguish. More precisely, the adversary may be able to choose 2 messages m_0 and m_1, but then must to decide the value of b when he gets $\mathsf{H}(r\|m_b), \mathsf{L}_{\mathsf{H}}(r\|m_b)$ in return, where r is a fresh unknown random value and $\mathsf{L}_{\mathsf{H}}(x)$ is the leakage resulting from evaluating the hash function on x. A simple comparison is now impossible since the value r is unknown.

Definition 15. *A hash function* $\mathsf{H} : \mathcal{R} \times \mathcal{M} \to \mathcal{B}$ *with leakage function* L *is* (q, t, ϵ)*-leakage resilient if, for every* (q, t)*-bounded adversary* $\mathcal{A}^{\mathsf{L}}$*, the advantage* $\left|\mathsf{Hash}^0_{\mathcal{A}^{\mathsf{L}},\mathsf{H}} - \mathsf{Hash}^1_{\mathcal{A}^{\mathsf{L}},\mathsf{H}}\right|$ *is bounded by* ϵ*, where* $\mathsf{Hash}^b_{\mathcal{A}^{\mathsf{L}},\mathsf{H}}$ *is defined as the probability that* $\mathcal{A}^{\mathsf{L}}$ *outputs 1 when, after a query* $(m_0, m_1) \in \mathcal{M}^2$*, he is returned with the pair* $(\mathsf{H}(r\|m_b), \mathsf{L}(r\|m_b))$ *with* $r \xleftarrow{\$} \mathcal{R}$*.*

The observation that we made above about the ease of recognizing leakages has an important impact on our designs. If we inspect modes that are related to the DCE and DTE modes, like the SIV mode [34] for instance, we see that SIV can be expected to be very hard to implement securely in front of an adversary with leakage access: SIV uses an IV that is computed only from a nonce made public and the message, which offers a convenient oracle to the adversary for matching leakages. The DCE and DTE schemes address this difficulty by making sure that, while playing the CPA game, the adversary never fully knows the inputs of the hash function that is being evaluated: we always add secret randomness in the first place. As an implementation note, we expect that the leakage-resilience of a hash function (in the sense defined above) will be higher if, when hashing $(r\|m)$, the block containing the randomness r is processed before the blocks containing the message. It guarantees that the adversary only sees leakages about a state that he cannot fully predict.

Remark 2. Admittedly, the following results should be understood similarly to the ones in [27], where it was argued that semantic security is impossible to achieve even if the leakage of an encryption would be as low as a single bit. So informally, what we show next is that the execution of our leakage-resilient authentication scheme for many messages does not significantly degrade the security compared to the situation with a single message. Concretely though, it always remains that manipulating the message leaks some information that can be exploited via SPA, because of the initial hashing of Figures 5 and 6 and the stream encryption of Figure 4. In this respect, we note that further improvements should possible. For example, one could replace the hash function of Figures 5 and 6 by the re-keying MAC of Figure 1, keep its intermediate PRF computation results, and then use these intermediates (rather than the message blocks) when encrypting in Figure 4 (so that the message is only manipulated once during the authenticated encryption). This would also break some possible correlations between the message and ciphertext blocks (e.g., if the message has a special structure) due to the fact that the ciphertext blocks are just obtained by XORing the key stream in Figure 4.[8] Again, such a construction would not change the impossibility result regarding semantic security, but further minimize the amount of leakage available due to message manipulation. We leave its investigation as an interesting scope for further research.

8.3 LMCPA Security of the DCE and DTE schemes

We start by focusing on the LMCPA security of the DCE scheme. The leakage function $\mathsf{L}(k, r, m)$ for DCE is defined by the pair $(\mathsf{L}_{\mathsf{H}}(r, m), \mathsf{L}_{\mathsf{PSV}}(k_0, r\|m))$, where k_0 is naturally defined as $\mathsf{F}_k(\mathsf{H}(r\|m))$. The L_{H} component of this leakage contains the leakage occurring during the evaluation of the hash function in DCE encryption, and the $\mathsf{L}_{\mathsf{PSV}}$ component contains the leakage of the encryption part of the DCE as depicted in Fig. 4, which we refer to as the "PSV-encryption component" of DCE. The $\mathsf{L}_{\mathsf{PSV}}$ function itself returns leakages that are made of individual leakages by each PRF and XOR operation, as defined in [27], but this is irrelevant for our analysis.

Theorem 7. *Let* $\mathsf{H} : \mathcal{R} \times \mathcal{M} \to \mathcal{B}$ *be a* $(0, t, \epsilon_{cr})$*-collision resistant and* $(q, t, \epsilon_{\mathsf{L}_{\mathsf{H}}})$*-leakage resilient hash function. Let* F *be a* $(q, t, \epsilon_{\mathsf{F}})$*-pseudorandom function. Let* DCE *be implemented with a PSV-encryption component that is* $(q, t, \epsilon^{\mathsf{leav}}_{\mathsf{PSV}-\mathsf{ENC}})$*-*leav*secure.*

[8] Note that the latter can also be avoided by replacing this XOR by a block cipher.

Then, DCE *with the leakage function* L *described above is* $(q', t', \epsilon_{\mathsf{lmcpa}})$*-secure. Here:* $q' \leq q - q_e - 1$ *where* q_e *is the number of encryption queries made by the* (q', t')*-bounded LMCPA adversary;* $t' \leq t_1 - t_c - t_{sc}$*, where* t_c *is the running time needed to run the LMCPA challenger in front of a* (q, t')*-bounded adversary,* t_{sc} *is the time needed to determine whether a list of* q_e *hash values contains a collision; and* $\epsilon_{\mathsf{lmcpa}} \leq 2\frac{q_e^2}{|\mathcal{R}|} + 2\epsilon_{cr} + 4\epsilon_{\mathsf{F}} + \epsilon_{\mathsf{L_H}} + \epsilon^{\mathsf{leav}}_{\mathsf{PSV-ENC}}$.

Proof. We start by defining Game 0 as the $\mathsf{PrivK}^{\mathsf{lmcpa},0}_{\mathcal{A}^{\mathsf{L}},\mathsf{DCE}}$ game.

Game 1 is equal to Game 0, except that we abort if, when processing the queries of $\mathcal{A}^{\mathsf{L}}$, the same randomness r is picked twice. The probability of this event is bounded by $q_e^2/|\mathcal{R}|$.

Game 2 is equal to Game 1, except that we abort if, when processing the queries of $\mathcal{A}^{\mathsf{L}}$, a collision happens on the hash function, that is, it the adversary provides messages m and m' such that, when performing their encryption, it happens that $\mathsf{H}(r|m) = \mathsf{H}(r'|m')$ (note that $r \neq r'$, because of the failure condition of Game 1). The gap between Game 2 and Game 1 is bounded by ϵ_{cr}: a collision resistance adversary can run $\mathcal{A}^{\mathsf{L}}$ and its LMCPA challenger (in time t_c, and using $q_e + 1$ leakage queries), and search for a collision (in time t_{sc}), placing us within the bounds of the hash function security.

Game 3 is equal to Game 2 except that, for all queries, the challenger replaces the computation of the key $k_0 = \mathsf{F}_k(\mathsf{h})$ with the selection of a random key $k_0 \xleftarrow{\$} \mathcal{B}$ (we assume that this does not increase its running time). Since the previous failure conditions guarantee that h is always fresh, the gap between Game 3 from Game 2 is bounded by ϵ_{F}: a PRF adversary can run $\mathcal{A}^{\mathsf{L}}$ and its LMCPA challenger (within $(q_e + 1, t_c)$ bounds), except that it queries the PRF challenger with all the h values that it computes.

Game 4 is equal to Game 3 except that, during the test query of the LMCPA game, the computation of $H(r\|m_0)$ (and the corresponding leakage) is replaced by the computation of $\mathsf{H}(r\|m_1)$. Here the probability of distinguishing Game 4 from Game 3 is bounded by ϵ_{LH}: an adversary against the leakage resilience of H can run $\mathcal{A}^{\mathsf{L}}$ and its LMCPA challenger (as tweaked in Game 3, and within $(q_e + 1, t_c)$ bounds), except that it hands the computation of h to the leakage resilient hash function challenger during the test query.

Game 5 is equal to Game 4 except that, during the test query of the LMCPA game, the selection of a random k_0 (from Game 3) is replaced by the selection of a random h^* and the computation of $k_0 = \mathsf{F}_k(\mathsf{h}^*)$. The gap between Game 5 from Game 4 is bounded by ϵ_{F}: a PRF adversary can run $\mathcal{A}^{\mathsf{L}}$ and its LMCPA challenger (within $(q_e + 1, t_c)$ bounds), except that it queries the PRF challenger with the h^* value that it computes.

To sum up, at this stage, $\mathcal{A}^{\mathsf{L}}$ sees:

- During an encryption query: the expected hash and leakage, and an encryption component encrypting that hash and leakage, but with a randomly chosen k_0 (hence independent of the long-term key k).
- During the test query: the hash and leakage of $(r\|m_1)$, followed by a PSV encryption of $(r\|m_0)$ with key k.

The presence of this isolated PSV encryption makes it possible to use the leakage resilient eavesdropper security of that scheme.

Game 6 is equal to Game 5 except that, during the test query of the LMCPA game, we encrypt $(r\|m_1)$ instead of $(r\|m_0)$. The gap between Game 6 and Game 5 is bounded by $\epsilon^{\mathsf{eav}}_{\mathsf{PSV-ENC}}$, since we can build an EAV adversary running $\mathcal{A}^{\mathsf{L}}$ and the LMCPA challenger (within $(q_e + 1, t_c)$ bounds), except that it hands the two messages $(r\|m_0)$ and $(r\|m_1)$ to the leavchallenger and returns the corresponding ciphertext to $\mathcal{A}^{\mathsf{L}}$.

Game 7 now hops to the $\mathsf{PrivK}^{\mathsf{lmcpa},1}_{\mathcal{A}^{\mathsf{L}},\mathsf{DCE}}$ game by undoing most of the hops that we made before, introducing the same gaps again, but keeping m_1 in place:

- We go back to a uniformly random k_0 by undoing the Game 4-5 transform.
- We go back to the selection of random k_0's everywhere to the use of a PRF as in the Game 2-3 transform.
- We stop aborting if the same randomness r is picked twice or if a collision happens in the hash function, as in the Game 0-2 transforms.

To sum-up we observe that the total gap introduced by our sequence of games is bounded by $2\frac{q_e}{|\mathcal{R}|} + 2\epsilon_{cr} + 4\epsilon_{\mathsf{F}} + \epsilon_{\mathsf{L_H}} + \epsilon^{\mathsf{eav}}_{\mathsf{PSV-ENC}}$. Besides, none of our reductions requires more leakage function queries than those needed to run the LMCPA challenger, and time more than the one needed to run that challenger and look for a collision in the outputs of the evaluation of the hash function that result from answering the adversary's queries in the LMCPA game (in Game 2). □

The leakage-resilient CPA security of the DTE scheme can be shown in an almost identical way.

Theorem 8. *Let* $\mathsf{H} : \mathcal{R} \times \mathcal{M} \to \mathcal{B}$ *be a* $(0, t, \epsilon_{cr})$*-collision resistant and* $(q, t, \epsilon_{\mathsf{L_H}})$*-leakage resilient hash function. Let* F *be a* $(2q, t, \epsilon_{\mathsf{F}})$*-pseudorandom function. Let* DTE *be implemented with a PSV-encryption component that is* $(q, t, \epsilon^{\mathsf{leav}}_{\mathsf{PSV-ENC}})$*-*leav*secure.*

Then, DTE *with the leakage function* L *described above is* $(q', t', \epsilon_{\mathsf{lmcpa}})$*-secure. Here:* $q' \leq q - q_e - 1$ *where* q_e *is the number of encryption queries made by the* (q', t')*-bounded LMCPA adversary;* $t' \leq t_1 - t_c - t_{sc}$*, where* t_c *is the running time needed to run the LMCPA challenger in front of a* (q', t')*-bounded adversary,* t_{sc} *is the time needed to determine whether a list of* q_e *hash values contains a collision; and* $\epsilon_{\mathsf{lmcpa}} \leq 2\frac{q_e^2}{|\mathcal{R}|} + 4\frac{(q_e+1)^2}{|\mathcal{B}|} + 2\epsilon_{cr} + 4\epsilon_{\mathsf{F}} + \epsilon_{\mathsf{L_H}} + \epsilon^{\mathsf{leav}}_{\mathsf{PSV-ENC}}$.

The proof shares almost all features of the one for the DCE scheme, and the handling of adversarial queries is the same. The double use of F_k just loosens the bounds of Thm. 7 by constant factors, by increasing the probability of collisions and doubling the number of queries that are needed when replacing the evaluation of F with the selection of random values (which is included in the t_c bound on the challenger running time).

Proof. We only detail the steps that differ from the proof of Thm. 7.

We split Game 3 into two steps, in order to be able to replace the tag τ and key k_0 values with random values. In the first step, we replace F_k with a random function f, bringing an ϵ_{F} gap as before. In the second step, we replace the evaluation of f by the selection of random values, which is only equivalent if f is never queried on the same value twice. This is actually the case, except with probability less than $4(q_e + 1)^2/|\mathcal{B}|$. Indeed: a collision between two hashes is precluded by Games 1 and 2; a collision between two τ values can only happen with probability bounded by $(q_e + 1)^2/|\mathcal{B}|$ (this upper-bounds the probability of a collision in the range of f invoked on distinct values); and a collision between a hash and a τ value is also bounded by by $(q_e + 1)^2/|\mathcal{B}|$ (the τ's a selected at random by f, and each of them will collide with one of the $q_e + 1$ distinct hashes with probability $(q_e + 1)/\mathcal{B}$.

In a similar way, we add a step in Game 7, in order to revert the transform above, bringing a second $2(q_e + 1)^2/|\mathcal{B}|$ gap. □

9 Conclusion and open problems

To conclude this paper, we first observe that as in [27], our analyses focused on the leakages occurring during authenticated encryption, so far excluding the possibility to target a decryption device. Interestingly, this limitation is very strong in the CCS 2015 leakage-resilient MAC and encryption

schemes because random IVs are strictly needed for leakage security, and a decryption oracle allows the adversary to control the IV. As discussed in Section 4.4, contradicting this requirement directly enables devastating forgery attacks based on a standard DPA. By contrast, our notion of ciphertext integrity with misuse and leakage aims to mitigate the impact of IV control. So it is natural to investigate whether it formally rules out any attack against the decryption oracle.

Unfortunately, and despite ciphertext integrity with misuse and leakage indeed rules out many realistic attacks against a decryption device, our schemes remain susceptible to strong attacks when the decryption leaks. Taking the example of DTE, we can for example show that it is possible to forge valid ciphertexts thanks to decryption leakages as follows: (1) Pick a random r and message m and compute $h = \mathsf{H}(r\|m)$. (2) Ask decryption of ciphertext $C^i = (\tau, c^i)$ with $\tau = h$ and a random c^i and recover k_0 thanks to leakage. (3) Ask decryption of ciphertext $C^j = (\tau', c^j)$ with $\tau' = k_0$ and a random c^j and recover k'_0 thanks to leakage. (4) From k'_0, compute the ciphertext c produced using the encryption part of DTE from the ephemeral key k'_0, the random r and the message m, so that $C = (k_0, c)$ is valid (and has decryption m). A completely similar attack can be done for DCE. As in Section 6, this attack is admittedly more challenging than the standard DPAs against the CCS 2015 building blocks. Yet, as in Section 6, it is also impossible to argue why such attacks should not be covered by our threat model. So we conclude our work by observing that the DTE and DCE authenticated and encryption schemes make a small step in making side-channel attacks targeting their decryption leakages less devastating in practice. And we leave the design of authenticated encryption modes that further minimize the attack surface against decryption devices as an interesting scope for further research.

Eventually, and quite independently to the previous challenges regarding how to design good modes of operation mitigating randomness misuse and leakage, the question of how to implement the leak-free component F^* is of course another important challenge. In this respect, we note that besides the protection of standard block ciphers with countermeasures such as noise addition, masking and shuffling, some alternatives are worth further investigation. One is simply to design block ciphers that are easier to protect against side-channel analysis (see for example [11,12,29] and follow-up works). Another option is to consider specialized constructions of leakage-resilient PRFs that overcome the hardness of protecting stateless primitives discussed in [5], by relying on other (non-standard) assumptions. For example, a CHES 2012 proposal in this direction assumes that the leakage of the different S-boxes in a block cipher are similar [21], and an ASIACRYPT 2016 paper investigates the use of unknown plaintexts for similar purposes [22]. A third possibility is to exploit ideas from the fresh-rekeying literature [20,19,7]. In particular, the recent construction from CRYPTO 2016 that mixes the efficiency advantages of a re-keying function enjoying (almost) key-homomorphism with the formal security guarantees of a wPRF [9] appears as a natural candidate to combine with the leakage-resilient authenticated encryption in this paper.

Acknowledgments. François-Xavier Standaert is a research associate and Thomas Peters is a postdoctoral researcher of the Belgian Fund for Scientific Research (F.R.S.-FNRS). This work has been funded in parts by the FNRS, the INNOVIRIS projects SCAUT and C-Cure, the ARC project NANOSEC and the ERC project 280141.

References

1. M. R. Albrecht and K. G. Paterson. Lucky microseconds: A timing attack on amazon's s2n implementation of TLS. In M. Fischlin and J. Coron, editors, *EUROCRYPT*, volume 9665 of *LNCS*, pages 622–643. Springer, 2016.
2. M. R. Albrecht, K. G. Paterson, and G. J. Watson. Plaintext recovery attacks against SSH. In *S&P*, pages 16–26. IEEE Computer Society, 2009.
3. E. Andreeva and M. Stam. The symbiosis between collision and preimage resistance. In L. Chen, editor, *IMACC*, volume 7089 of *LNCS*, pages 152–171. Springer, 2011.
4. J. Balasch, B. Gierlichs, O. Reparaz, and I. Verbauwhede. DPA, bitslicing and masking at 1 GHz. In Güneysu and Handschuh [15], pages 599–619.
5. S. Belaïd, V. Grosso, and F. Standaert. Masking and leakage-resilient primitives: One, the other(s) or both? *Cryptography and Communications*, 7(1):163–184, 2015.
6. M. Bellare and C. Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. *J. Cryptology*, 21(4):469–491, 2008.
7. C. Dobraunig, F. Koeune, S. Mangard, F. Mendel, and F. Standaert. Towards fresh and hybrid re-keying schemes with beyond birthday security. In N. Homma and M. Medwed, editors, *CARDIS*, volume 9514 of *LNCS*, pages 225–241. Springer, 2015.
8. T. Duong and J. Rizzo. Cryptography in the web: The case of cryptographic design flaws in ASP.NET. In *S&P*, pages 481–489. IEEE Computer Society, 2011.
9. S. Dziembowski, S. Faust, G. Herold, A. Journault, D. Masny, and F. Standaert. Towards sound fresh re-keying with hard (physical) learning problems. In M. Robshaw and J. Katz, editors, *CRYPTO*, volume 9815 of *LNCS*, pages 272–301. Springer, 2016.
10. S. Faust, K. Pietrzak, and J. Schipper. Practical leakage-resilient symmetric cryptography. In Prouff and Schaumont [30], pages 213–232.
11. B. Gérard, V. Grosso, M. Naya-Plasencia, and F. Standaert. Block ciphers that are easier to mask: How far can we go? In G. Bertoni and J. Coron, editors, *CHES*, volume 8086 of *LNCS*, pages 383–399. Springer, 2013.
12. V. Grosso, G. Leurent, F. Standaert, and K. Varici. LS-designs: Bitslice encryption for efficient masked software implementations. In C. Cid and C. Rechberger, editors, *FSE*, volume 8540 of *LNCS*, pages 18–37. Springer, 2014.
13. V. Grosso and F. Standaert. ASCA, SASCA and DPA with enumeration: Which one beats the other and when? In T. Iwata and J. H. Cheon, editors, *ASIACRYPT*, volume 9453 of *LNCS*, pages 291–312. Springer, 2015.
14. D. Gruss, R. Spreitzer, and S. Mangard. Cache template attacks: Automating attacks on inclusive last-level caches. In J. Jung and T. Holz, editors, *USENIX Security*, pages 897–912. USENIX Association, 2015.
15. T. Güneysu and H. Handschuh, editors. *CHES*, volume 9293 of *LNCS*. Springer, 2015.
16. J. Longo, E. D. Mulder, D. Page, and M. Tunstall. SoC it to EM: electromagnetic side-channel attacks on a complex system-on-chip. In Güneysu and Handschuh [15], pages 620–640.
17. S. Mangard. Hardware countermeasures against DPA ? A statistical analysis of their effectiveness. In T. Okamoto, editor, *CT-RSA*, volume 2964 of *LNCS*, pages 222–235. Springer, 2004.
18. S. Mangard, E. Oswald, and F. Standaert. One for all - all for one: unifying standard differential power analysis attacks. *IET Information Security*, 5(2):100–110, 2011.
19. M. Medwed, C. Petit, F. Regazzoni, M. Renauld, and F. Standaert. Fresh re-keying II: securing multiple parties against side-channel and fault attacks. In E. Prouff, editor, *CARDIS*, volume 7079 of *LNCS*, pages 115–132. Springer, 2011.
20. M. Medwed, F. Standaert, J. Großschädl, and F. Regazzoni. Fresh re-keying: Security against side-channel and fault attacks for low-cost devices. In D. J. Bernstein and T. Lange, editors, *AFRICACRYPT*, volume 6055 of *LNCS*, pages 279–296. Springer, 2010.
21. M. Medwed, F. Standaert, and A. Joux. Towards super-exponential side-channel security with efficient leakage-resilient PRFs. In Prouff and Schaumont [30], pages 193–212.
22. M. Medwed, F. Standaert, V. Nikov, and M. Feldhofer. Unknown-input attacks in the parallel setting: Improving the security and performances of the CHES 2012 leakage-resilient PRF. *ASIACRYPT 2016, to appear.*
23. S. Micali and L. Reyzin. Physically observable cryptography (extended abstract). In M. Naor, editor, *TCC*, volume 2951 of *LNCS*, pages 278–296. Springer, 2004.
24. C. Namprempre, P. Rogaway, and T. Shrimpton. Reconsidering generic composition. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT*, volume 8441 of *LNCS*, pages 257–274. Springer, 2014.
25. C. Namprempre, P. Rogaway, and T. Shrimpton. Reconsidering generic composition. *IACR Cryptology ePrint Archive*, 2014:206, 2014.
26. K. G. Paterson and N. J. AlFardan. Plaintext-recovery attacks against datagram TLS. In *NDSS*. The Internet Society, 2012.
27. O. Pereira, F. Standaert, and S. Vivek. Leakage-resilient authentication and encryption from symmetric cryptographic primitives. In I. Ray, N. Li, and C. Kruegel, editors, *ACM CCS*, pages 96–108. ACM, 2015.

28. T. Peyrin and Y. Seurin. Counter-in-tweak: Authenticated encryption modes for tweakable block ciphers. In M. Robshaw and J. Katz, editors, *CRYPTO*, volume 9814 of *LNCS*, pages 33–63. Springer, 2016.
29. G. Piret, T. Roche, and C. Carlet. PICARO - A block cipher allowing efficient higher-order side-channel resistance. In F. Bao, P. Samarati, and J. Zhou, editors, *ACNS*, volume 7341 of *LNCS*, pages 311–328. Springer, 2012.
30. E. Prouff and P. Schaumont, editors. *CHES*, volume 7428 of *LNCS*. Springer, 2012.
31. M. Rivain and E. Prouff. Provably secure higher-order masking of AES. In S. Mangard and F. Standaert, editors, *CHES*, volume 6225 of *LNCS*, pages 413–427. Springer, 2010.
32. P. Rogaway and T. Shrimpton. Cryptographic hash-function basics: Definitions, implications, and separations for preimage resistance, second-preimage resistance, and collision resistance. In B. K. Roy and W. Meier, editors, *FSE*, volume 3017 of *LNCS*, pages 371–388. Springer, 2004.
33. P. Rogaway and T. Shrimpton. Deterministic authenticated-encryption: A provable-security treatment of the key-wrap problem. *IACR Cryptology ePrint Archive*, 2006:221, 2006.
34. P. Rogaway and T. Shrimpton. A provable-security treatment of the key-wrap problem. In S. Vaudenay, editor, *EUROCRYPT*, volume 4004 of *LNCS*, pages 373–390. Springer, 2006.
35. F. Standaert, O. Pereira, and Y. Yu. Leakage-resilient symmetric cryptography under empirically verifiable assumptions. In R. Canetti and J. A. Garay, editors, *CRYPTO*, volume 8042 of *LNCS*, pages 335–352. Springer, 2013.
36. N. Veyrat-Charvillon, B. Gérard, and F. Standaert. Soft analytical side-channel attacks. In P. Sarkar and T. Iwata, editors, *ASIACRYPT*, volume 8873 of *LNCS*, pages 282–296. Springer, 2014.
37. N. Veyrat-Charvillon, M. Medwed, S. Kerckhof, and F. Standaert. Shuffling against side-channel attacks: A comprehensive study with cautionary note. In X. Wang and K. Sako, editors, *ASIACRYPT*, volume 7658 of *LNCS*, pages 740–757. Springer, 2012.
38. Y. Yu and F. Standaert. Practical leakage-resilient pseudorandom objects with minimum public randomness. In E. Dawson, editor, *CT-RSA*, volume 7779 of *LNCS*, pages 223–238. Springer, 2013.
39. Y. Yu, F. Standaert, O. Pereira, and M. Yung. Practical leakage-resilient pseudorandom generators. In E. Al-Shaer, A. D. Keromytis, and V. Shmatikov, editors, *ACM CCS*, pages 141–151, 2010.

A Alternative PSV constructions

Pereira et al. also proposed the hash then MAC construction informally pictured in Figure 7, leading to the authenticated encryption scheme outlined in Figure 8. It is easy to see that the attack of Section 4.4 applies nearly identically to this construction. Namely, and first considering the a single-key variant, the adversary will first recover k'_0 via a DPA attack. Then, he will choose a message m' and compute a tag τ. The only difference is that this time, he cannot force the second ephemeral key k_0 to be identical to k'_0 since it would require that $\tau = r$ which implies finding a preimage to the hash function. Yet, what he can easily do is to use again the leakage of his leaking device, by setting the random coin r at τ, and to perform a second DPA against the output of the first leak-free block cipher, which this time will leak k_0 and allow forging valid ciphertexts. As for the 2-key variant, an SPA attack against the second ephemeral key k_0 is the easiest option.

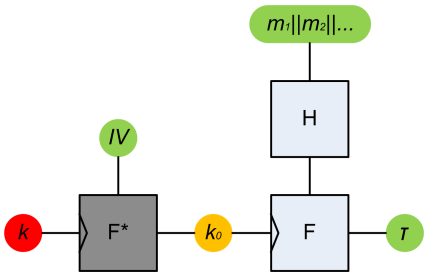

Fig. 7. PSV-MAC' leakage-resilient MAC [27].

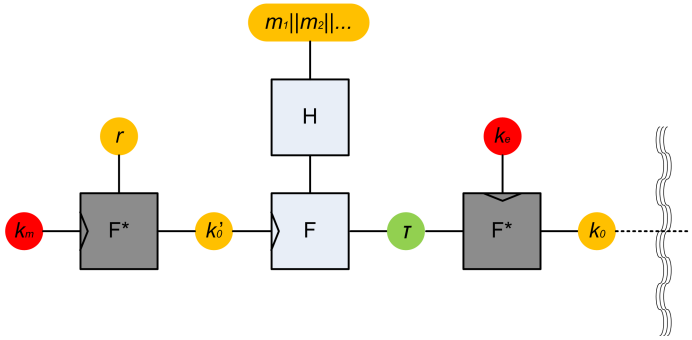

Fig. 8. PSV-AE' leakage-resilient AE (part I).