Weak-Key Subspace Trails and Applications to AES

Lorenzo Grassi[1], Gregor Leander[2], Christian Rechberger[1], Cihangir Tezcan[3] and Friedrich Wiemer[2]

[1] IAIK, Graz University of Technology, Austria
firstname.lastname@iaik.tugraz.at
[2] Horst Görtz Institute for IT-Security, Ruhr-Universität Bochum, Germany
firstname.lastname@rub.de
[3] Informatics Institute, Department of Cyber Security, CYDES Laboratory, and Department of Mathematics, Middle East Technical University, Ankara, Turkey
cihangir@metu.edu.tr

Abstract. Invariant subspaces (Crypto'11) and subspace trails (FSE'17) are two related recent cryptanalytic approaches that led to new results on, e. g. PRINTCipher and AES. We extend the invariant subspace approach to allow for different subspaces in every round, something that so far only the subspace trail approach and a generalization for invariant subspace and invariant set attacks (Asiacrypt'18) were able to do. For an easier detection, we provide an algorithm which finds these weak-key subspace trails.

Using this framework, we perform an extensive analysis of weak-key distinguishers (in the single-key setting) for AES with several key schedule variants. Among others, we show that for the new key-schedule proposed at ToSC/FSE'18 – which is faster than the standard key schedule and ensures a higher number of active S-Boxes – it is possible to set up an invariant subspace distinguisher for any number of rounds.

Finally, we describe a property for *full* AES-128 and AES-256 in the chosen-key setting with complexity 2^{64} without requiring related keys. These chosen-key distinguishers are set up by exploiting the multiple-of-n property introduced at Eurocrypt'17, adapted to the case of AES instantiated with weak-keys.

Keywords: AES · Key Schedule · Weak-Keys · Invariant Subspaces · Chosen-Key Distinguisher

Contents

1 Introduction

Block ciphers are certainly among the most important cryptographic primitives. Their design and analysis are well advanced, and with today's knowledge designing a secure block cipher is a problem that is largely considered solved. Especially with the AES we have at hand a very well analyzed and studied cipher that, after more than 20 years of investigation still withstands all cryptanalytic attacks. However, new results on the AES still appear regularly, especially within the last couple of years (e. g. [Bar+18; Gra+17; Gra18; Røn+17]). While those papers do not pose any practical threat to the AES, they do give new insights into the internals of what is arguably the cipher that is responsible for the largest fraction of encrypted data worldwide.

Clearly, security of symmetric crypto is always security against specific attacks. The number of available attacks has increased significantly ever since the introduction of differential [BS90] and linear [Mat94] cryptanalysis in the early 1990. Besides the numerous variations of linear and differential attacks, e. g. truncated differentials [Knu95], impossible differentials [Bih+99; Knu98], zero-correlation attacks [BR14], and multidimensional linear cryptanalysis [Her+09] to name only a few, it turned out that in many cases combining two attack vectors might lead to new, more powerful attacks. The most prominent example is the combination of linear and differential cryptanalysis into differential-linear cryptanalysis [LH94].

Another important aspect is that the attacker model is regularly changing. With the introduction of statistical attacks, especially linear and differential cryptanalysis, the attacker was suddenly assumed to be able to retrieve, or even choose, large amounts of plaintext/ciphertext pairs. Later, in the related-key setting, the attacker became even more powerful and was assumed to be able to choose not only plaintexts but also ask for the encryption of chosen messages under a key that is related to the unknown secret key. Finally, in the open-key model, the attacker either knows the key or has the ability to choose the key herself.

While the practical impact of such models is often debatable, they actually might become meaningful when the block cipher is used as a building block for other primitives, in particular for the construction of hash-functions[1]. Moreover, even if those considerations do not pose practical attacks, they still provide very useful insights and observations that strengthen our understanding of block ciphers in general, and on the AES in particular.

Our work builds upon the above in the sense that we do combine previously separate attacks to derive new results on the AES, both in the secret-key as well as in the open-key model.

1.1 Our contribution

Weak-key subspace trail cryptanalysis. In this work we start by recalling the basic set-up of subspace trail cryptanalysis (see [Gra+16; Gra+17; Lea+18]) and invariant subspace attacks (see [Lea+11; Lea+15]) in Section 2. Our main focus is to point out the important differences of these two attacks. As we will explain, those concepts are not generalizations of each other but rather orthogonal attack vectors. From this point of view, a natural step is to combine both approaches into a new, more powerful, attack. This is in line with what was done previously with other attacks as mentioned above.

As invariant subspace attacks are weak-key attacks by nature, the new attack originating from the combination of invariant subspace attacks and subspace trail cryptanalysis is a weak-key attack as well. Here, weak-key refers to the fact that the attacks do not work for any key, but rather only for a fraction of all keys. Consequently, in Section 2 we coin the

[1] As concrete examples, in Appendix H we present collisions for Matyas-Meyer-Oseas, Davies–Meyer and Miyaguchi-Preneel compressing modes instantiated with 7-round AES-256.

Table 1: Summary of concrete new results and observations on AES with various key schedules in the single-key setting (no related-keys). The respective sections have more detailed comparison.

Property	Key Schedule	Rounds	Remarks	Reference
Trunc. Diff.	trivial	any	2^{64} keys	folklore, § 4.1
Trunc. Diff.	FSE 2018 [Kho+17]	any	2^{64} keys	§ 4.2
Trunc. Diff.	AES-128/192/256	3	all keys	folklore
	AES-128/192/256	5	2^{32} / 2^{64} / 2^{128} keys	§ 5.1
	AES-256	6/7/8	2^{96} / 2^{64} / 2^{32} keys	§ 5.1
Multiple-of-n	AES-128/192/256	5	all keys	[Gra+17]
	AES-128/192/256	6	2^{32} / 2^{64} / 2^{128} keys	§ 5.2
	AES-256	7/8/9	2^{96} / 2^{64} / 2^{32} keys	§ 5.2
Multiple-of-n	AES-128	10 (full)	CKD*	§ 6.5
Multiple-of-n	AES-192	11	CKD*	App. G.2
Multiple-of-n	AES-256	14 (full)	CKD*	App. G.3
Compr. Collision	AES-256	7	cost 2^{32}	App. H

* Chosen-Key Distinguisher

new strategy *weak-key subspace trail cryptanalysis*. To be able to detect these trails, we provide an algorithmic way, based on previous search algorithms for invariant subspaces.

Previously, invariant subspace attacks were only applied to ciphers with very simple key schedule algorithms. As a result, ciphers where the round keys differed not only by round constants seemed secure against this type of attacks. In particular, up to now, it seemed impossible to apply invariant subspace attacks on the AES.

With our new combination of invariant subspace attacks and subspace trail cryptanalysis, we overcome this inherently difficult problem. As a showcase of the increased possibilities of our attack, and as the most important example anyway, in Sections 4 and 5 we present several new observations on the AES. An overview of these properties is given in Table 1. Using as starting point the invariant subspace found by our algorithm and presented in Section 4, we show that (almost) all the secret-key distinguishers for round-reduced AES currently present in the literature can be set up for a higher number of rounds of AES if the whitening key is a weak-key.

As a side-result, in Section 4.2 we show that the recently proposed alternative AES key schedule [Kho+17] actually leads to a cipher that can be broken in our setting.

Chosen-key distinguisher for (full) AES-128 and AES-256 in the single-key setting. Building up on those results we are able to show a non-random property for full AES-128 and AES-256 in the chosen-key setting that seems difficult to produce generically. This improves all the chosen-key distinguishers for AES in the single-key setting. In particular, in Section 6 we exhibit *a chosen-key distinguisher with complexity* 2^{64} *for full AES-128* in the single-key model, valid for 2^{32} keys.[2].

For these results we combine two weak-key subspace trails in an inside-out manner and, instead of a simple truncated differential property at the plaintexts and ciphertexts, we use a variant of the "multiple-of-n" property recently shown for AES in [Gra+17].

[2] A 10-round known-key distinguisher for AES has been proposed by Gilbert [Gil14] at Asiacrypt 2014. Echoing Grassi and Rechberger [GR17], in Section 6.1.2 we argue why such distinguisher can be considered artificial. Briefly, the property of this distinguisher does not involve *directly* the plaintexts/ciphertexts, but their encryption/decryption after one round.

Open problems. Finally, we point out that our work leaves many open questions for future research. In particular, the analysis of [Bei+17] that resulted in efficient ways to prove the resistance of a given cipher to invariant subspace attacks is not applicable to the weak-key subspace trail cryptanalysis. Thus, an important and highly non-trivial task is to leverage this analysis to the case of weak-key subspace trails.

1.2 Related work

Regarding weak-key cryptanalysis, the most famous example of weak-keys is given for the DES. The block cipher DES has a few specific keys termed "weak-keys" and "semi-weak-keys" [MS87]. These are keys that cause the encryption mode of DES to act identically to the decryption mode of DES (albeit potentially that of a different key). Several other examples can be found in the literature, e. g. for Blowfish [KM07; Vau96], PRESENT [Ohk09], or Piccolo [WW16]. Typical "weak-key" attacks (so called as these attacks work only when a key of a special form is used, thus a "weak-key") are the already mentioned invariant subspace attack [Lea+11; Lea+15], the invariant set (or nonlinear invariant) attack [Tod+16], see also the recent work by Beyne [Bey18] for a generalization of invariant subspace and invariant set attacks. As an example for the risk of weak-keys, we mention the case of the stream cipher RC4 [Flu+01], where RC4's weak initialization vectors allow an attacker to mount a known-plaintext attack, which has been (widely) used to compromise the security of WEP.

Weak-keys are much more often a problem where the adversary has some control over what keys are used, such as when a block cipher is used in a mode of operation intended to construct a secure cryptographic hash function. For example, the Davies-Meyer construction or the Miyaguchi-Preneel one can transform a secure block cipher into a secure compression function. In a hash setting, block cipher security models such as the known-key model (or the chosen-key model) makes sense since in practice the attacker has full access and control over the internal computations.

The idea of known-key distinguishers was introduced by Knudsen and Rijmen in [KR07] for their analysis of AES and a class of Feistel ciphers. They examined the security of these block ciphers in a model where the adversary knows the key. To succeed, the adversary has to discover some property of the attacked cipher that e. g. holds with a probability higher than for an ideal cipher, or is generally believed to be hard to exhibit generically. The idea of chosen-key distinguishers was popularized in the attack on the full-round AES-256 [BK09; Bir+09] in a related-key setting. This time the adversary is assumed to have a full control over the key. A chosen-key attack was shown on 9-round reduced AES-128 in [Fou+13] in the related-key setting, and on 8-round AES-128 in [Der+12] in the single-key setting. Both the known-key and chosen-key distinguishers are collectively known as *open-key distinguishers*.

An attack in these (open-key) models depicts a structural flaw of the cipher, while it should be desired to work with a primitive that does not have any flaw, even in the most generous security model for the attacker. A classical example is the devastating effect on the compression function security of weak-keys for a block cipher [Wei+12], which are usually considered as a minor flaw for a block cipher if the set of these weak-keys is small.

2 Weak-key (invariant) subspace trails

Before coming to the weak-key variant of subspace trails, let us recapitulate the differences between invariant subspaces and subspace trails.

2.1 Subspace trails

Subspace trails have been first defined in [Gra+16] and later used to attack reduced round versions of AES [Gra+17] and PRINCE [GR16] as well as on Simpira [Røn16]. We refer to [Gra+16] for more details about the concept of subspace trails. Our treatment here is however meant to be self-contained.

We recall the definition of a subspace trail next. For this, let F denote a round function of a key-alternating block cipher, and let $U \oplus a$ denote a coset of a vector space U. By U^c we denote the complementary subspace of U.

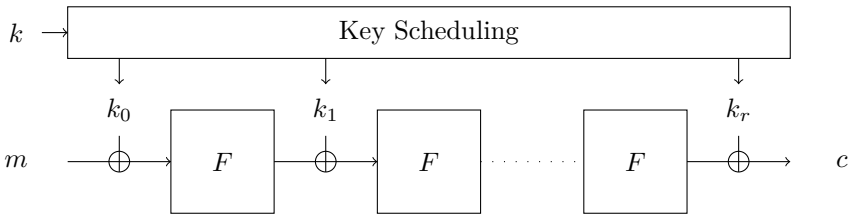
Definition 1 (Subspace trails)**.** Let $(U_1, U_2, \ldots, U_{r+1})$ denote a set of $r+1$ subspaces with $\dim(U_i) \leqslant \dim(U_{i+1})$. If for each $i = 1, \ldots, r$ and for each a_i, there exists (unique) $a_{i+1} \in U_{i+1}^c$ such that

$$F(U_i \oplus a_i) \subseteq U_{i+1} \oplus a_{i+1},$$

then $(U_1, U_2, \ldots, U_{r+1})$ is a *subspace trail* of length r for the function F.

If all the previous relations hold with equality, the trail is called a *constant-dimensional subspace trail*.

One important observation is the following. Consider a key-alternating cipher E_k using F as a round function and where the round keys are xored in between the rounds, as depicted below:



In this case, a subspace trail for F will extend to a subspace trail for E_k for any choice of round keys. This is a simple consequence as

$$F(U_i \oplus a_i) \subseteq U_{i+1} \oplus a_{i+1} \text{ implies } F_{k^i}(U_i \oplus a_i) \equiv F(U_i \oplus a_i) \oplus k^i \subseteq U_{i+1} \oplus a'_{i+1}$$

for a suitable $a'_{i+1} = a_i \oplus k^i$ (where k^i is the i-th subkey). In other words, the key addition changes only the coset of the subspace U_{i+1}, while it does not affect the subspace itself. Thus, not only do subspace trails work for all keys, they are also completely independent of the key schedule. Here, invariant subspace attacks behave very differently.

2.2 Invariant subspace attacks

Invariant subspace attacks, which can be seen as a general way of capturing symmetries, have been first introduced in [Lea+11] in an attack on PRINTCipher. Later, those attacks have been applied to several other (mainly lightweight) primitives, e. g. in [Lea+15], where a generic tool to detect them has been proposed.

As above, denoting by $F_k(\cdot) = F(\cdot) \oplus k$ the round function of an iterated block-cipher, let $U \subset \mathbb{F}_2^n$ be a subspace. Then, U is called an invariant subspace if there exist constants $a, b \in \mathbb{F}_2^n$ such that $F_k(U \oplus a) = U \oplus b$. In order to extend the invariant subspace $U \oplus a_i \mapsto U \oplus a_{i+1}$ to the whole cipher, we need all round keys to be in specific cosets[3] of U namely, $k_i \in U \oplus (a_{i+1} \oplus b_i)$ (where $F(U \oplus a_i) = U \oplus b_i$):

$$F_k(U \oplus a_i) = F(U \oplus a_i) \oplus k = U \oplus b_i \oplus \underbrace{k}_{\in U \oplus (a_{i+1} \oplus b_i)} = U \oplus a_{i+1}.$$

[3]It is not necessary that $a_i = a_{i+1}$ for all i in order to set up an invariant subspace attack.

Definition 2 (Invariant subspace trail)**.** Let K_{weak} be a set of keys and $k \in K_{\text{weak}}$, with $k \equiv (k^0, k^1, \ldots, k^r)$ where k^j is the j-th round key. For each $k \in K_{\text{weak}}$, the subspace U generates an *invariant subspace trail* of length r for the function $F_k(\cdot) \equiv F(\cdot) \oplus k$ if for each $i = 1, \ldots, r$ there exists a non-empty set $A_i \subseteq U^c$ for which the following property holds:

$$\forall a_i \in A_i : \quad \exists! a_{i+1} \in A_{i+1} \text{ s.t. } F_{k^i}(U \oplus a_i) \equiv F(U \oplus a_i) \oplus k^i = U \oplus a_{i+1}.$$

All keys in the set K_{weak} are *weak-keys*.

In the following, we combine both concepts into weak-key subspace trails.

2.3 Weak-key subspace trails

When comparing subspace trail and invariant subspace attacks, two obvious but important differences can be observed. First, subspace trails are clearly much more general as they allow different spaces in the domain and co-domain of F. Second, subspace trails are by far more restrictive, as not only one coset of the subspace has to be mapped to one coset of (a potentially different) subspace, but rather all cosets have to be mapped to cosets. For subspace trails, the later fact is the main reason for allowing arbitrary round keys.

The main idea for weak-key subspace trails is to stick to the property of invariant subspace attacks where only few (even just one) cosets of a subspace are mapped to other cosets of a subspace. However, borrowing from subspace trails, we allow those subspaces to be different for each round. As this will again restrict the choice of round keys that will keep this property invariant to a class of weak-keys we call this combination *weak-key subspace trails* (or simply, weak subspace trails). The formal definition is the following.

Definition 3 (Weak-key subspace trails)**.** Let K_{weak} be a set of keys and $k \in K_{\text{weak}}$ with $k \equiv (k^0, k^1, \ldots, k^r)$ where k^j is the j-th round key. Further let $(U_1, U_2, \ldots, U_{r+1})$ denote a set of $r+1$ subspaces with $\dim(U_i) \leqslant \dim(U_{i+1})$. For each $k \in K_{\text{weak}}$, $(U_1, U_2, \ldots, U_{r+1})$ is a *weak-key subspace trail* (WKST) of length r for the function $F_k(\cdot) \equiv F(\cdot) \oplus k$ if for each $i = 1, \ldots, r$ there exists a non-empty set $A_i \subseteq U_i^c$ for which the following property holds:

$$\forall a_i \in A_i : \quad \exists! a_{i+1} \in A_{i+1} \text{ s.t. } F_{k^i}(U_i \oplus a_i) \equiv F(U_i \oplus a_i) \oplus k^i \subseteq U_{i+1} \oplus a_{i+1}.$$

All keys in the set K_{weak} are *weak-keys*. If all the previous relations hold with equality, the trail is called a *weak-key constant-dimensional subspace trail*.

Usually, the set $A_i \subseteq U_i^c$ reduces to a single element a_i, that is $A_i \equiv \{a_i\}$. Moreover, we can easily see that Definition 3 is a generalization of both Definitions 1 and 2:

- if K_{weak} is equal to the whole set of keys and if $A_i = U_i^c$, then it corresponds to subspace trails;
- if $U_i = U_{i+1}$ for all i, then it corresponds to invariant subspace trails.

Security Problem. Clearly, a WKST allows greater freedom for an attacker. In comparison to invariant subspace attacks, WKSTs have the potential of being better applicable to block ciphers with a non trivial key schedule. At the same time, with respect to subspace trails it is not necessary for WKSTs to hold for all possible keys.

Interestingly, proving resistance against invariant subspace (or more generally invariant sets) in the case of identical round keys (up to the addition of round constants) is well understood, see [Bei+17]. However, the situation changes completely when considering WKSTs and/or ciphers with a non-trivial key schedule. In those situations, the analysis

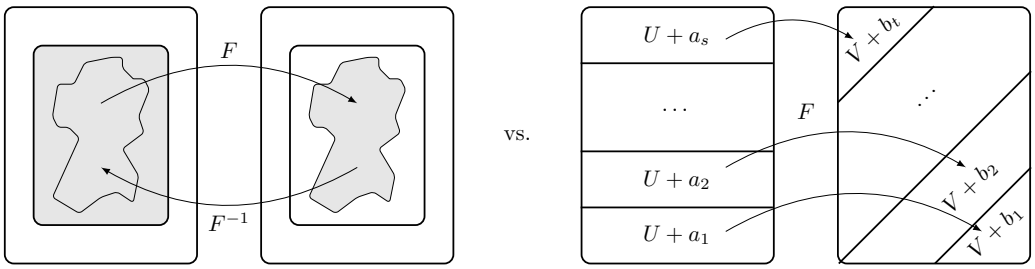


Figure 1: To find invariant subspaces (left part), we iteratively compute the image of the current subspace and map the span of it backwards through the inverse function, until the process stabilizes. For subspace trails (right part), *all* cosets of the starting subspace get mapped to a coset of the ending subspace. This implies that the derivative of the round function is in the ending subspace.

of [Bei+17] is no longer applicable and we do not have a generic approach to argue the resistance against WKSTs.

It follows that the concept of WKSTs opens up many new opportunities and raises many new, probably highly non-trivial questions on how to protect against it. We focus on demonstrating the new opportunities by investigating the AES. How to (generically) protect against those attacks is left as an open question for future research.

2.4 Algorithmic detection of weak-key subspace trails

Before discussing the actual weak subspace trails, let us take a look at how we can find these algorithmically. To begin with, we recapitulate how the algorithms for invariant subspaces [Lea+15] and subspace trails work [Lea+18].

First, Fig. 1 (left part) sketches the idea for invariant subspaces. Given a round function $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$, the algorithm guesses a starting offset a for the affine subspace $U \oplus a$ and then maps $U \oplus a$ forwards and back through F and F^{-1}, every-time computing the span of the image. If the subspace stabilizes, we have found an invariant subspace.

Second, Fig. 1 (right part) illustrates the main idea for subspace trails. The important difference to invariant subspaces is that *every* coset of the starting subspace U is mapped to some coset of the ending subspace V. The implication of this is, see [Lea+18, Lemma 1], the images of the derivatives $\Delta_u F(\cdot) := F(\cdot \oplus u) \oplus F(\cdot)$ of the round function F span a subspace of V. In other words, if $U \xrightarrow{F} V$ is a subspace trail, then

$$U \xrightarrow{F} \operatorname{span}\left(\bigcup_{u \in U} \operatorname{Im}(\Delta_u(F))\right) \subseteq V.$$

We cannot exploit this fact for WKSTs, though. Instead we base the algorithm on the idea for invariant subspaces.

Goal and details of the algorithm. Given a round function $R : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and a key schedule $K_i : \mathbb{F}_2^m \to \mathbb{F}_2^n$ for $0 \leq i \leq r$ rounds, the goal is to find two subspaces $U, V \subset \mathbb{F}_2^n$ and a subset $S \subseteq \mathbb{F}_2^m$, s. t. every message m chosen from U and every key $k \in S$ get mapped to a ciphertext $c = E_k(m) \in V$, where the encryption uses the round function R and key schedule K_i for the i-th round key. Thus, all master keys in S are weak-keys.

As a starting point, we assume that the zero message $m = 0$ is in our starting subspace U_0. This is anyway always the case, as we assume all U_i's to be subspaces. Additionally,

Algorithm 1 Compute an initial Weak-key subspace trail

Precondition: A round function $R : \mathbb{F}_2^n \to \mathbb{F}_2^n$ and a key schedule $K_i : \mathbb{F}_2^n \to \mathbb{F}_2^n$ for $0 \leq i \leq r$ rounds. An upper bound max_rnd on the number of rounds to cover.

Postcondition: A weak-key subspace trail $U_0 \to \cdots \to U_l$ over l rounds for a set S of weak-keys.

```
1  function WKST(R, K_i, max_rnd)
2      S ← {0}
3      L ← [U_0 = {0, K_0(0)}]
4      while for the last element U_i in L: dim(U_i) < n do
5          U_{i+1} ← ∅
6          for enough x ∈_R U_i do
7              U_{i+1} ← U_{i+1} ∪ {R(x)}
8          U_{i+1} ← span(U_{i+1} ∪ {K_{i+1}(k) | ∀k ∈ S})
9          append U_{i+1} to L
10         if len(L) ≥ max_rnd then
11             return L
12     return (L, S)
```

we require that a certain key k^{weak} – chosen by the user – is weak, thus in S. Since $k^{\text{weak}} = 0 \in S$ is very often the case if invariant subspace attacks apply, we assume $k^{\text{weak}} = 0$ in the following. In particular, we have the following conditions:

$$\begin{aligned} 0 \in U_i, \qquad & R(U_i) \subseteq U_{i+1}, \\ K_i(S) \subseteq U_i, \qquad & R(K_i(S)) \subseteq U_{i+1}. \end{aligned} \tag{1}$$

Exploiting these conditions and starting at the above mentioned point, we can simply compute the WKST forwards. We may want to check if the resulting trail is invariant, for that we can simply compute the trail backwards at some point. For the complete pseudocode[4] see Algorithm 1 – Sage code that implements this algorithm can be found in the Appendix I (see Listing 1).

The runtime of our algorithm depends on the while and for loop. The first loop iterates over the subspaces in our trail and is thus bound by the length of the WKST. For the later loop, we have to iterate over "enough x". Following the same argument as in [Lea+18] tells us that sampling $n + 100$ random inputs is enough to compute the following subspace with overwhelming probability.

3 Subspace trail properties of the AES

We start with a brief revision of the AES, as its details are important for the remainder of the work, and known subspace trails.

The Advanced Encryption Standard [DR02] is a *Substitution-Permutation network* that supports key sizes of 128, 192 and 256 bits. The 128-bit plaintext initializes the internal state as a 4×4 matrix of bytes as values in the finite field $\mathbb{F}_{256}$, defined using the irreducible polynomial $x^8 + x^4 + x^3 + x + 1$. Depending on the version of AES, N_r rounds are applied to the state: $N_r = 10$ for AES-128, $N_r = 12$ for AES-192 and $N_r = 14$ for AES-256. An AES round applies four operations to the state matrix:

- *SubBytes* (SB) – applying the same 8-bit to 8-bit invertible S-Box 16 times in parallel on each byte of the state (it provides non-linearity in the cipher);

[4] Only for simplicity, the update process for the set S is not included in the algorithm. More weak-keys can be found by computing backward from the U_i's, see Eq. (1) and the Sage implementation.

- *ShiftRows* (SR) – cyclic shift of each row to the left;
- *MixColumns* (MC) – multiplication of each column by a constant 4×4 invertible matrix M_{MC} (MC and SR provide diffusion in the cipher);
- *AddRoundKey* (ARK) – XORing the state with a 128-bit subkey.

One round of AES can be described as $R(x) = K \oplus \text{MC} \circ \text{SR} \circ \text{SB}(x)$. In the first round an additional AddRoundKey operation (using a whitening key) is applied, and in the last round the MixColumns operation is omitted.

Key schedule AES-128. The key schedule of AES-128 takes the user key and transforms it into 11 subkeys of 128 bits each. The subkey array is denoted by $W[0, \ldots, 43]$, where each word of $W[\cdot]$ consists of 4 bytes and where the first 4 words of $W[\cdot]$ are loaded with the user secret key. The remaining words of $W[\cdot]$ are updated according to the following rule:

$$W[i][j] = \begin{cases} W[i][j-4] \oplus \text{SB}(W[i+1][j-1]) \oplus R[i][j/4] & \text{if } j \bmod 4 = 0 \\ W[i][j-1] \oplus W[i][j-4] & \text{otherwise} \end{cases}$$

where $i = 0, 1, 2, 3$, $j = 4, \ldots, 43$ and $R[\cdot]$ is an array of predetermined constants.[5]

Key schedule AES-192. The key schedule of AES-192 is similar to the one given for AES-128. In this case, the subkey array is denoted by $W[0, \ldots, 51]$, where here the first 6 words of $W[\cdot]$ are loaded with the user secret key. The remaining words of $W[\cdot]$ are updated according to the following rule:

$$W[i][j] = \begin{cases} W[i][j-6] \oplus \text{SB}(W[i+1][j-1]) \oplus R[i][j/6] & \text{if } j \bmod 6 = 0 \\ W[i][j-1] \oplus W[i][j-6] & \text{otherwise} \end{cases}$$

where $i = 0, 1, 2, 3$, $j = 6, \ldots, 51$ and $R[\cdot]$ is an array of predetermined constants.

Key schedule AES-256. The case AES-256 is a little different from the previous cases. In this case, the subkey array is denoted by $W[0, \ldots, 59]$, where here the first 8 words of $W[\cdot]$ are loaded with the user secret key. The remaining words of $W[\cdot]$ are updated according to the following rule:

$$W[i][j] = \begin{cases} W[i][j-8] \oplus \text{SB}(W[i+1][j-1]) \oplus R[i][j/8] & \text{if } j \bmod 8 = 0 \\ W[i][j-8] \oplus \text{SB}(W[i][j-1]) & \text{if } j \bmod 8 = 4 \\ W[i][j-1] \oplus W[i][j-8] & \text{otherwise} \end{cases}$$

where $i = 0, 1, 2, 3$, $j = 8, \ldots, 59$ and $R[\cdot]$ is an array of predetermined constants.

The notation used in the paper. Let x denote a plaintext, a ciphertext, an intermediate state or a key. Then $x_{i,j}$ or $x_{i+4\times j}$ with $i, j \in \{0, \ldots, 3\}$ denotes the byte in the row i and in the column j. We denote by k^r the key of the r-th round. If only one key is used, then we denote it by k to simplify the notation. Finally, we denote by R one round of AES, while we denote r rounds of AES by R^r. We sometimes use the notation R_K instead of R to highlight the round key K. As last thing, in the paper we often use the term "partial collision" (or "collision") when two texts belong to the same coset of a given subspace X.

[5] The round constants are defined in $GF(2^8)[X]$ as $R[0][1] = X$, $R[0][r] = X \cdot R[0][r-1]$ if $r \geq 2$ and $R[i][\cdot] = 0$ if $i \neq 0$. For the following, let $R[r] \equiv R[0][r]$.

3.1 Subspace trails of AES

In this section, we recall the main concepts of the subspace trails of AES presented in [Gra+16] – we refer to [Gra+16] for more details and examples.

For the following, we only work with vectors and vector spaces over $\mathbb{F}_{2^8}^{4\times 4}$, and we denote by $\{e_{0,0}, \ldots, e_{3,3}\}$ the unit vectors of $\mathbb{F}_{2^8}^{4\times 4}$ (e. g. $e_{i,j}$ has a single 1 in row i and column j). We also recall that given a subspace X, the cosets $X \oplus a$ and $X \oplus b$ (where $a \neq b$) are *equal* ($X \oplus a \equiv X \oplus b$) if and only if $a \oplus b \in X$.

Definition 4. The *column spaces* $\mathcal{C}_i$ are defined as $\mathcal{C}_i = \langle e_{0,i}, e_{1,i}, e_{2,i}, e_{3,i} \rangle$.

Definition 5. The *diagonal spaces* $\mathcal{D}_i$ and the *inverse-diagonal spaces* $\mathcal{ID}_i$ are respectively defined as $\mathcal{D}_i = \mathrm{SR}^{-1}(\mathcal{C}_i) \equiv \langle e_{0,i}, e_{1,i+1}, e_{2,i+2}, e_{3,i+3} \rangle$ and $\mathcal{ID}_i = \mathrm{SR}(\mathcal{C}_i) \equiv \langle e_{0,i}, e_{1,i-1}, e_{2,i-2}, e_{3,i-3} \rangle$, where the indexes are taken modulo 4.

Definition 6. The *i-th mixed spaces* $\mathcal{M}_i$ are defined as $\mathcal{M}_i = \mathrm{MC}(\mathcal{ID}_i)$.

Definition 7. For $I \subseteq \{0, 1, 2, 3\}$, let $\mathcal{C}_I$, $\mathcal{D}_I$, $\mathcal{ID}_I$ and $\mathcal{M}_I$ be defined as

$$\mathcal{C}_I = \bigoplus_{i \in I} \mathcal{C}_i, \qquad \mathcal{D}_I = \bigoplus_{i \in I} \mathcal{D}_i, \qquad \mathcal{ID}_I = \bigoplus_{i \in I} \mathcal{ID}_i, \qquad \mathcal{M}_I = \bigoplus_{i \in I} \mathcal{M}_i.$$

As shown in detail in [Gra+16]:

- for any coset $\mathcal{D}_I \oplus a$ there exists unique $b \in \mathcal{C}_I^{\perp}$ such that $R(\mathcal{D}_I \oplus a) = \mathcal{C}_I \oplus b$;
- for any coset $\mathcal{C}_I \oplus a$ there exists unique $b \in \mathcal{M}_I^{\perp}$ such that $R(\mathcal{C}_I \oplus a) = \mathcal{M}_I \oplus b$.

Theorem 1 ([Gra+16])**.** *For each I and for each $a \in \mathcal{D}_I^{\perp}$, there exists one and only one $b \in \mathcal{M}_I^{\perp}$ such that*

$$R^2(\mathcal{D}_I \oplus a) = \mathcal{M}_I \oplus b.$$

We refer to [Gra+16] for a complete proof of this theorem. Observe that b depends on a (the constant that defines the initial coset of $\mathcal{D}_I$) and on the secret key k.

Observe that if X is a generic subspace, $X \oplus a$ is a coset of X and x and y are two elements of the (same) coset $X \oplus a$, then $x \oplus y \in X$. It follows that:

Lemma 1. *For all x, y and for all $I \subseteq \{0, 1, 2, 3\}$:*

$$\Pr\left[R^2(x) \oplus R^2(y) \in \mathcal{M}_I \;\middle|\; x \oplus y \in \mathcal{D}_I\right] = 1.$$

As demonstrated in [Gra+16], we finally recall that for each $I, J \subseteq \{0, 1, 2, 3\}$, then : $\mathcal{M}_I \cap \mathcal{D}_J = \{0\}$ if and only if $|I| + |J| \leq 4$. It follows that

Theorem 2 ([Gra+16])**.** *Let $I, J \subseteq \{0, 1, 2, 3\}$ such that $|I| + |J| \leq 4$. For all $x \neq y$:*

$$\Pr\left[R^4(x) \oplus R^4(y) \in \mathcal{M}_I \;\middle|\; x \oplus y \in \mathcal{D}_J\right] = 0.$$

For completeness, we briefly describe the subspace trail notation using a more "classical" one. If two texts t^1 and t^2 are equal except for the bytes in the i-th diagonal[6] for each $i \in I$, then they belong in the same coset of $\mathcal{D}_I$. Two texts t^1 and t^2 belong in the same coset of $\mathcal{M}_I$ if the bytes of their difference $\mathrm{MC}^{-1}(t^1 \oplus t^2)$ in the i-th anti-diagonal for each $i \notin I$ are equal to zero. Similar considerations hold for the column space $\mathcal{C}_I$ and the inverse-diagonal space $\mathcal{ID}_I$.

[6] The i-th diagonal of a 4×4 matrix A is defined as the elements that lie on row r and column c such that $r - c = i \bmod 4$. The i-th anti-diagonal of a 4×4 matrix A is defined as the elements that lie on row r and column c such that $r + c = i \bmod 4$.

Generic subspace trail of length 1 for AES

For the follow-up, we introduce a generic subspace trail of length 1.

Definition 8. Let I be a subset of $\{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$. Let the subspace $\mathcal{X}_I$ be defined as

$$\mathcal{X}_I = \langle \{e_{i,j}\}_{(i,j)\in I} \rangle \equiv \left\{ \bigoplus_{(i,j)\in I} \alpha_{i,j} \cdot e_{i,j} \,\middle|\, \forall \alpha_{i,j} \in \mathbb{F}_{2^8} \right\}.$$

In other words, $\mathcal{X}_I$ is the set of elements given by linear combinations of $\{e_{i,j}\}_{(i,j)\in I}$, where $e_{i,j} \in \mathbb{F}_{2^8}^{4\times 4}$ has a single 1 in row i and column j.

Theorem 3. *For each $I \subseteq \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$ and for each $a \in \mathcal{X}_I^{\perp}$, there exists one and only one $b \in \mathcal{Y}_I^{\perp}$ such that*

$$R(\mathcal{X}_I \oplus a) = \mathcal{Y}_I \oplus b$$

where $\mathcal{Y}_I = \mathrm{MC} \circ \mathrm{SR}(\mathcal{X}_I)$.

The complete proof[7] of this Theorem is given in Appendix A. Such subspace trail cannot be extended on two rounds for any generic $\mathcal{X}_I$, due to the non-linear S-Box operation of the next round (that *can* destroy the linear relations that hold among the bytes).

3.2 Weak-key subspace trail of AES: A concrete example

Before going on, we present a (proper) weak-key subspace trail for AES. For simplicity, initially we work with a simpler S-Box, that is we replace the AES S-Box with the inverse one

$$\forall x \in GF(2^8): \qquad \mathbb{S}(x) = \begin{cases} 1/x \equiv x^{254}, & \text{if } x \neq 0, \\ 0 & \text{otherwise} \end{cases}$$

To achieve our goal, the idea is to find subspaces $V, W \subset GF(2^8)$ of dimension two and/or four s. t.

$$\mathbb{S}(V \oplus v) \subseteq W \oplus w$$

for *certain* (not all) $v, w \in GF(2^8)$, where $V \neq W$ in general. E. g. the subspace V of dimension four defined as[8]

$$V = \langle [1, 12, 80, 176], 0 \rangle \subseteq \{x \in GF(2^8) \,|\, x^{256} \oplus x = 0\}$$

is invariant under the S-Box – that is, $\mathbb{S}(V) = V$, since $\mathbb{S}(x)^{256} \oplus \mathbb{S}(x) = [(x^{254})]^{256} \oplus x^{254} = x^{254} \oplus x^{254} = 0$ (remember that $x^{2^n-1} = 1$ for all $x \in GF(2^n)$), while its cosets $V \oplus v$ for $v \neq 0$ are in general not invariant.

In [Bra+05], several subspaces $V, W \subset GF(2^8)$ of dimension two and four are defined such that $V \neq W$ and $\mathbb{S}(V \oplus v) \subseteq W \oplus w$. In particular, in there authors found 85 disjoint input subspaces of dimension 2 together with the corresponding output subspaces, and 17 disjoint input subspaces of dimension 4 together with the corresponding output subspaces of the AES, like

$$\mathbb{S}\left(V \equiv \langle [2, 24, 97, 160], 0 \rangle\right) = \left(W \equiv \langle [6, 40, 88, 139], 0 \rangle\right).$$

[7] Here we limit ourselves to highlight that for each $I \subset \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$, there exists $J \subseteq \{(i,j)\}_{0\leq i,j\leq 3}$ such that $\mathrm{SR}(\mathcal{X}_I) = \mathcal{X}_J$ (or equivalently $\mathrm{SR}^{-1}(\mathcal{X}_I) = \mathcal{X}_J$).

[8] About the notation, the flats are denoted by $\langle [a_1, \ldots, a_d], b \rangle$, where b represents the coset and $a_1, \ldots, a_d$ the d basis vectors of the subspace. Here the vectors are denoted by their radius-2 notation, i. e. $x = x_1 + 2 \cdot x_2 + \cdots + 2^{n-1} \cdot x_n \in \mathbb{Z}$ corresponds with the vector $x = (x_1, \ldots, x_n)$.

This can be used to set up a weak-subspace trail for 1-round AES, e.g.

$$\mathcal{V} \oplus x \equiv \begin{bmatrix} \langle [2,24,97,160] \rangle & x_{0,1} & x_{0,2} & x_{0,3} \\ x_{1,0} & \langle [2,24,97,160] \rangle & x_{1,2} & x_{1,3} \\ x_{2,0} & x_{2,1} & \langle [2,24,97,160] \rangle & x_{2,3} \\ x_{3,0} & x_{3,1} & x_{3,2} & \langle [2,24,97,160] \rangle \end{bmatrix}$$

$$\xrightarrow{\mathrm{MC} \circ \mathrm{SR} \circ \mathbb{S}(K^w \oplus \cdot)} \mathcal{W} \oplus y \equiv \begin{bmatrix} \langle [6,40,88,139] \rangle & y_{0,1} & y_{0,2} & y_{0,3} \\ \langle [6,40,88,139] \rangle & y_{1,1} & y_{1,2} & y_{1,3} \\ \langle [6,40,88,139] \rangle & y_{2,1} & y_{2,2} & y_{2,3} \\ \langle [6,40,88,139] \rangle & y_{3,1} & y_{3,2} & y_{3,3} \end{bmatrix}$$

for random value of $x \in \mathcal{D}_0^{\perp} \equiv \mathcal{D}_{1,2,3}$ (where $y \in \mathcal{C}_{1,2,3}$). In this case, the class of weak keys K^w corresponds to the subspace $\mathcal{V} \oplus \mathcal{D}_{1,2,3}$ (where $\mathcal{V} \subset \mathcal{D}_0$), where each byte of the key in the first diagonal belongs to the subspace $\langle [2,24,97,160] \rangle$, while the bytes in the other diagonals can take any possible values. Similarly, it is possible to set up different and longer weak-key subspace trails.

Finally, analogous results can be obtained for the real AES, since the AES S-Box is affine equivalent to $I(x)$, that is

$$\text{AES-S-Box}(x) = \alpha \cdot \mathbb{S}(x) \oplus \beta$$

where α is a 8×8 binary (invertible) matrix and β is a constant ($\beta = 0\text{x}63$). In other words, the previous weak-key subspace trail holds if the subspace W is replaced by $\alpha \cdot W \oplus \beta$.

4 Weak-key invariant subspace trail and key schedule

Let the subspace $\mathcal{IS}$ be defined as

$$\mathcal{IS} \coloneqq \left\{ \begin{bmatrix} a & b & a & b \\ c & d & c & d \\ e & f & e & f \\ g & h & g & h \end{bmatrix} \;\middle|\; \forall a,b,c,d,\ldots,h \in \mathbb{F}_{2^8} \right\} \tag{2}$$

This subspace is invariant under a key-less round $R(\cdot) = \mathrm{MC} \circ \mathrm{SR} \circ \mathrm{SB}(\cdot)$, since

$$\mathrm{SB}(\mathcal{IS}) = \mathcal{IS} \qquad \mathrm{SR}(\mathcal{IS}) = \mathcal{IS} \qquad \mathrm{MC}(\mathcal{IS}) = \mathcal{IS}.$$

This subspace[9] – already presented and used in e. g. [Cha+17; Guo+13; Le+04] – can easily be found by extending the result of Algorithm 1. It will be our starting point in order to set up a weak invariant subspace trail for all versions of AES.

AES Key-Schedule As we are going to show, *the possibility to set up a weak invariant subspace trail depends on the concrete value of the secret key and of the key schedule details.* The problem to design a strong key schedule has been largely studied and discussed in the literature. Usually, the target that a key schedule must satisfy is resistance against related-key attacks, while the problem of weak-keys is in general less considered – and sometimes strange effects can occur, see [Kra+17]. However, presence of weak-keys can have a devastating effect on the security of a cipher. As a concrete examples, we cite the invariant attack on Midori64 [Guo+16], or on AES instantiated with the key schedule [Kho+17] recently proposed at FSE 2018 (as we are going to show in Sect. 4.2).

In the following we consider several AES key schedules present in the literature, and for each one of them we discuss the possibility to set up a weak invariant subspace trail. In more details, we consider three categories of key schedule:

[9] We mention that such subspace also confirms and provides an example of a recent result from Liu and Rijmen [LR17], who proved that if an invariant subspace of a key-less AES round exists, it has dimension at most 8 (see [LR17, Theorem 3] for details).

- the simplest key schedule is given by *identical subkeys* or by subkeys defined as the XOR of the whitening key and round constants – this category has been largely studied in [Bei+17], recently published at Crypto 2017;

- another category of key schedule is given by (linear) *permutation* of the byte positions: each subkey is the result of a particular permutation applied to the whitening key – e. g. the key schedule recently proposed at FSE 2018 [Kho+17];

- finally, we consider *AES-like key schedules* – besides the original AES key schedule, we consider the variant proposed at SAC 2010 by Nikolić [Nik11].

For each case, we present a set of weak-keys for which the invariant subspace trail can be set up[10]. To do this, our strategy is to look for keys that satisfy the following two properties: *(1st)* belong to the invariant subspace $\mathcal{IS}$ and *(2nd)* for which the "next round sub-key" generated by the key schedule belongs to the invariant subspace $\mathcal{IS}$. In other words, in order to find weak-keys, we initially focus on a set of 2^{64} keys – denoted by K_{weak} – "equal" to the subspace $\mathcal{IS}$ just defined, and among them we identify the keys that satisfy the second requirement just given.

Using these results as starting point, in the next section we present weak-key secret-key distinguishers for round-reduced AES, that is we show how to extend the secret key distinguishers on AES on more rounds in the case in which the whitening key is a weak-key.

4.1 Identical round keys and weak round constants

The simplest possible key schedule[11] (mainly used for lightweight ciphers) is probably obtained as follows: the r-th round subkey $k[r]$ is simply given by the XOR of the whitening key K and a round constant $RC[r]$, that is $k[r] = K \oplus RC[r]$.

Consider the subspace $\mathcal{IS}$ previously defined. If for each round r the subkey $K \oplus RC[r]$ belongs to this subspace, then it is possible to set up a weak invariant subspace trail for a set of weak-keys for an arbitrary number of rounds. In particular, if $k[r] \in \mathcal{IS}$ then

$$\mathcal{IS} \xrightarrow{\text{MC} \circ \text{SR} \circ \text{SB}(\cdot)} \mathcal{IS} \xrightarrow{\cdot \oplus k[r]} \mathcal{IS} \tag{3}$$

This property, and similar symmetries in the AES round transformation, are folklore. Only to provide a concrete example in the literature, we mention the invariant attack on Midori64 [Guo+16], where authors set up an invariant subspace attack on the full cipher due to a bad choice of the round constants.

For completeness, we mention that with a proper choice of round constants, such properties can be easily avoided, as showed in details in Beierle et al. [Bei+17]. Even though we do not know of a method to generically rule out weak subspace trails, we do not know of such properties for such a key schedule with random round constants either.

4.2 Key schedule based on permutation of the byte positions

Another possible category of key schedule exploits permutation of the byte positions: each subkey is the result of a particular permutation applied to the whitening key. A concrete example of key schedule based on permutation has been proposed at FSE 2018 [Kho+17]. This "*new fully linear key schedule that can be used to replace the one in AES-128*" (see [Kho+17, Sect. 6]) is basically a permutation on the key state byte positions, where the

[10] In the following, we consider only invariant subspace trails of length at least two.

[11] Our choice to include this very simple key schedule is mainly due to pedagogical reasons, which can help to make the application to real AES easier to understand.

key state update function is defined as follows

$$\begin{pmatrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{pmatrix} \rightarrow \begin{pmatrix} 11 & 15 & 3 & 7 \\ 12 & 0 & 4 & 8 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \end{pmatrix}$$

Regarding security, even though no S-Box nor round constant is used in this key schedule, authors prove there are more active S-Boxes in the related-key model than for AES-128. However, consider the previous subspace $\mathcal{IS}$ defined in Eq. (2) and assume that the whitening key belongs to such subspace. It follows that any subkey generated by the previous permutation belongs this subspace (due to particular symmetries of the permutation), which implies the possibility to set up an "infinitely-long" weak invariant subspace Eq. (3) for a set of weak-keys. Similar results can be obtained also for one of the key-schedules recently proposed at SAC 2018 [Der+18, Sect. 4.2] – see Appendix B.2 for more details.

We mention that a simple way to avoid such invariant subpsace attack is to add random round-constants. For completeness, authors of [Kho+17] also propose to "tweak this design (without increasing the tracking effort) by adding an S-Box layer every round to the entire first row of the key state". Due to the analysis just proposed, this operation does not improve the security against the presented invariant subspace attack. However, this problem can be easily fixed by applying an S-Box layer every round to the entire (e.g.) first column.

4.3 AES key schedule

Next, we present the AES original key schedule. Since in the following we present chosen-key distinguishers that depend on the AES key schedule, we present this case in detail.

4.3.1 Invariant subspace – weak-keys of AES-128

Under one of the 2^{32} weak-keys in K_{weak}

$$K_{\text{weak}} \coloneqq \left\{ \begin{bmatrix} A & A & A & A \\ B & B & B & B \\ C & C & C & C \\ D & D & D & D \end{bmatrix} \;\middle|\; \forall A, B, C, D \in \mathbb{F}_{2^8} \right\} \tag{4}$$

the subspace $\mathcal{IS}$ is mapped into a coset of $\mathcal{IS}$ after two complete AES rounds. In more details, given $k \in K_{\text{weak}}$, let $\hat{k}$ be the corresponding subkey after 2 rounds of the key schedule (where $\hat{k} \notin K_{\text{weak}}$ in general). It follows that

$$\mathcal{IS} \xrightarrow{R^2 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}$$

where $R(\cdot) \equiv \text{ARK} \circ \text{MC} \circ \text{SR} \circ \text{SB}(\cdot)$, that is *$\mathcal{IS}$ forms a weak invariant subspace of length 2*. In order to prove this result, it is sufficient to note that

1. $K_{\text{weak}} \subseteq \mathcal{IS}$, which implies that $\mathcal{IS} \oplus k = \mathcal{IS}$ for all $k \in K_{\text{weak}}$;
2. the first round key derived from the key-schedule of K_{weak} – denoted by K'_w – is a subset of $\mathcal{IS}$

$$K'_w \equiv \begin{bmatrix} \text{SB}(B) \oplus A \oplus R[1] & \text{SB}(B) \oplus R[1] & \text{SB}(B) \oplus A \oplus R[1] & \text{SB}(B) \oplus R[1] \\ \text{SB}(C) \oplus B & \text{SB}(C) & \text{SB}(C) \oplus B & \text{SB}(C) \\ \text{SB}(D) \oplus C & \text{SB}(D) & \text{SB}(D) \oplus C & \text{SB}(D) \\ \text{SB}(A) \oplus D & \text{SB}(A) & \text{SB}(A) \oplus D & \text{SB}(A) \end{bmatrix}$$

4.3.2 Invariant subspace – weak-keys of AES-256

For the case AES-256, a set of 2^{128} weak-keys is given by

$$K_{\text{weak}} \coloneqq \left\{ \begin{bmatrix} A^0 & A^1 & A^0 & A^1 & E^0 & E^1 & E^0 & E^1 \\ B^0 & B^1 & B^0 & B^1 & F^0 & F^1 & F^0 & F^1 \\ C^0 & C^1 & C^0 & C^1 & G^0 & G^1 & G^0 & G^1 \\ D^0 & D^1 & D^0 & D^1 & H^0 & H^1 & H^0 & H^1 \end{bmatrix} \;\middle|\; \begin{array}{c} \forall A^i, \dots, H^i \in \mathbb{F}_{2^8} \\ \forall i = 0, 1 \end{array} \right\}$$

Under any of such keys, the subspace $\mathcal{IS}$ is mapped after two complete rounds into a coset of $\mathcal{IS}$, that is $\mathcal{IS} \xrightarrow{R^2 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}$, where $\hat{k}$ is the corresponding subkey after 2 rounds of the key schedule.

For the follow-up, we also present three subspaces of K_{weak} for which it is possible to construct a longer invariant subspace trail:

3-round: working with any of the 2^{96} keys that satisfy $A^0 = A^1, \dots D^0 = D^1$, the subspace $\mathcal{IS}$ is mapped after three complete rounds into a coset of $\mathcal{IS}$, that is $\mathcal{IS} \xrightarrow{R^3 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}'$ where $\hat{k}'$ is the subkey after 3 rounds.

4-round: working with any of the 2^{64} keys that satisfy $A^0 = A^1, \dots, H^0 = H^1$, the subspace $\mathcal{IS}$ is mapped after four complete rounds into a coset of $\mathcal{IS}$, that is $\mathcal{IS} \xrightarrow{R^4 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}''$ where $\hat{k}''$ is the subkey after 4 rounds.

5-round: working with any of the 2^{32} keys that satisfy $A^0 = A^1 = B^0 = \dots = D^0 = D^1 = 0$ and $E^0 = E^1, \dots \quad H^0 = H^1$, the subspace $\mathcal{IS}$ is mapped after five complete rounds into a coset of $\mathcal{IS}$, that is $\mathcal{IS} \xrightarrow{R^5 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}'''$ where $\hat{k}'''$ is the subkey after 5 rounds.

The complete expressions of the subkeys involved for the previous results are given for completeness in Appendix B.

4.3.3 Invariant subspace – weak-keys of AES-192

For the case AES-192, a set[12] of 2^{64} weak-keys is given by

$$K_{\text{weak}} \equiv \left\{ \begin{bmatrix} A & E \oplus \text{SB}^{-1}(D \oplus H) & A & E \oplus \text{SB}^{-1}(D \oplus H) & E & \text{SB}^{-1}(D \oplus H) \\ B & F \oplus \widehat{\text{SB}}^{-1}(A \oplus E) & B & F \oplus \widehat{\text{SB}}^{-1}(A \oplus E) & F & \widehat{\text{SB}}^{-1}(A \oplus E) \\ C & G \oplus \text{SB}^{-1}(B \oplus F) & C & G \oplus \text{SB}^{-1}(B \oplus F) & G & \text{SB}^{-1}(B \oplus F) \\ D & H \oplus \text{SB}^{-1}(C \oplus G) & D & H \oplus \text{SB}^{-1}(C \oplus G) & H & \text{SB}^{-1}(C \oplus G) \end{bmatrix} \;\middle|\; \forall A, \dots, H \in \mathbb{F}_{2^8} \right\}$$

where $\widehat{\text{SB}}^{-1}(\cdot) = \text{SB}^{-1}(\cdot \oplus R[1])$. Under any of such keys, the subspace $\mathcal{IS}$ is mapped after two complete rounds into a coset of $\mathcal{IS}$, that is $\mathcal{IS} \xrightarrow{R^2 \circ \text{ARK}(\cdot)} \mathcal{IS} \oplus \hat{k}$, where $\hat{k}$ the corresponding subkey after 2 rounds of the key schedule.

4.4 AES-like key schedule

Finally, a possible variant of the AES key schedule has been proposed at SAC 2010 by Nikolić [Nik11]. This variant is obtained by introducing a small change in the current AES key schedule, which allows to improve the security against related-key attacks. In short, for obtaining each column of the new subkey, the new key schedule *always* uses rotation by

[12] We highlight that this subset is *not* a subspace, as for AES-128 and AES-256.

Table 2: *Secret-key properties for round-reduced AES.* In the following, we list the properties for round-reduced AES which are independent of the secret key and the corresponding number of rounds, comparing the case of weak-keys. "Number of keys" denotes the number of keys (with respect to the total space) for which a particular property holds up to r Rounds.

Property	Version of AES	Rounds	Number of keys	Reference
Trunc. Diff.	AES-128/192/256	3	2^{128} / 2^{192} / 2^{256}	folklore
	AES-128/192/256	5	2^{32} / 2^{64} / 2^{128}	§ 5.1
	AES-256	6/7/8	2^{96} / 2^{64} / 2^{32}	§ 5.1
Multiple-of-n	AES-128/192/256	5	2^{128} / 2^{192} / 2^{256}	[Gra+17]
	AES-128/192/256	6	2^{32} / 2^{64} / 2^{128}	§ 5.2
	AES-256	7/8/9	2^{96} / 2^{64} / 2^{32}	§ 5.2

one byte up of the previous subkey column, while AES uses a rotation only when obtaining the subkey column with an index multiple of N_k ($N_k = 4, 6, 8$ for AES-128,-192,-256).

As we show in detail in Appendix B.3, even if this change improves the security against related-key attack, it is possible to get the same results just presented for the original AES key schedule also in this case.

5 Weak-key secret-key distinguishers for AES

As a first application of the invariant subspaces just found, we are going to show that *under the assumption of weak-keys* it is possible to extend the secret-key distinguishers present in the literature to more rounds. In the following, we present in detail only the results for AES-128 for the encryption/forward direction (analogous results hold also in the decryption/backward direction). Similar results can be obtained also for AES-192 and AES-256, using the corresponding weak-keys and weak invariant subspace trails previously defined. The results – which have been practically tested using a C/C++ implementation – are summarized in Table 2.

Important Remark. We emphasize that *from now on we assume that the secret key is a weak-key* (that is, a key in the set K_{weak} as described previously), and that all the following results are independent of the details of the S-Box and of the MixColumns operation.

5.1 Subspace trail distinguishers

In the case of AES, it is possible to set up subspace trail distinguishers for 3 and 4-round AES. Both are *independent* of the secret-key, of the details of the S-Box and of the MixColumns matrix (assuming branch number equal to five). In particular, the first one exploits the fact that

$$\Pr\left[R^3(x) \oplus R^3(y) \in \mathcal{M}_J \;\middle|\; x \oplus y \in \mathcal{D}_I\right] = (2^8)^{-4|I|+|I|\cdot|J|}$$

as showed in details in [Gra+16], while for a random permutation Π the previous probability is equal to

$$\Pr\left[\Pi(x) \oplus \Pi(y) \in \mathcal{M}_J \mid x \oplus y \in \mathcal{D}_I\right] = (2^8)^{-16+4|J|}. \tag{5}$$

For the impossible differential case, the idea is to exploit the fact that $\mathcal{M}_I \cap \mathcal{D}_J = \{0\}$ for $|I| + |J| \leq 4$. Thus, it follows that (see [BK01; Gra+16] for details)

$$\Pr\left[R^4(x) \oplus R^4(y) \in \mathcal{M}_J \;\middle|\; x \oplus y \in \mathcal{D}_I\right] = 0 \qquad \text{if } |I| + |J| \leq 4$$

while for a random permutation Π the probability is given by Eq. (5).

In the following, we show that it is possible to extend the previous subspace trail distinguisher for up to 5 rounds in the case of weak-keys. For simplicity, we focus on the case of AES-128. As we have just seen, the subspace $\mathcal{IS}$ is mapped into a coset $\mathcal{IS} \oplus a$ after two rounds if the secret key is a weak-key. In other words, given two plaintexts $x, y \in \mathcal{IS}$, then $R^2(x) \oplus R^2(y) \in \mathcal{IS}$ under a weak-key. By definition of $\mathcal{IS}$ and $\mathcal{D}_I$, note that[13]

$$\Pr\left[z \in \mathcal{D}_I \mid z \in \mathcal{IS}\right] = \begin{cases} 2^{-32} & I \equiv \{0,2\},\{1,3\} \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where we assume that $z \notin \mathcal{D}_L$ for all $L \subseteq \{0,1,2,3\}$ s.t. $|L| < |I| < 4$. This is the starting point for our results, together with the fact that $\Pr\left[z \in \mathcal{D}_{0,2}\right] = \Pr\left[z \in \mathcal{D}_{1,3}\right] = 2^{-64}$ for a generic text z.

5.1.1 Weak-key subspace trail over 4-round AES-128

Since $R^2(\mathcal{D}_I \oplus a) = \mathcal{M}_I \oplus b$ (that is $\Pr\left[R^2(x) \oplus R^2(y) \in \mathcal{M}_I \;\middle|\; x \oplus y \in \mathcal{D}_I\right] = 1$), it follows that for an AES permutation and for a weak-key

$$\Pr\left[R^4(x) \oplus R^4(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] = 2^{-32} \qquad \text{if } I \equiv \{0,2\},\{1,3\},$$

while for a random permutation Π the probability is equal to 2^{-64} (see Eq. (5)).

This fact can also be re-written using the subspace trail notation.

Proposition 1. *Consider 2^{64} plaintexts in the subspace $\mathcal{IS}$, and the corresponding ciphertexts after 4-rounds AES-128 encrypted under a weak-key $k \in \mathcal{K}_{weak}$.*

With probability 1, the ciphertexts are distributed as follows:

- *there exist 2^{32} (in 2^{64}) different cosets of $\mathcal{M}_{0,2}$ s.t. each one of them contains exactly 2^{32} ciphertexts;*
- *there exist 2^{32} (in 2^{64}) different cosets of $\mathcal{M}_{1,3}$ s.t. each one of them contains exactly 2^{32} ciphertexts.*

For a random permutation, each one of the two previous properties is satisfied with probability

$$\binom{2^{64}}{2^{32}} \cdot \prod_{i=0}^{2^{32}-1} \left[\left(2^{-64}\right)^{2^{32}-1} \cdot \left(1 - i \cdot 2^{-64}\right)\right] \approx 2^{-2^{70}}.$$

Proof. As already showed, a subspace $\mathcal{IS}$ is mapped into a coset of $\mathcal{IS}$ after 2 rounds AES-128 under a weak-key. By definition of $\mathcal{IS} \oplus a$, the first and the third diagonals (resp. the second and the fourth) are equal. This means that:

- there are 2^{32} texts that are equal in the first and the third diagonals, and that differ in the second and in the fourth ones. By definition, these 2^{32} texts belong to the same coset of $\mathcal{D}_{1,3}$. It follows that after 2-round encryption, the 2^{64} texts are divided into 2^{32} different cosets of $\mathcal{D}_{1,3}$;
- equivalently, there are 2^{32} texts that are equal in the second and in the fourth diagonals, and that differ first and the third ones. By definition, these 2^{32} texts belong to the same coset of $\mathcal{D}_{0,2}$. It follows that after 2-round encryption, the 2^{64} texts are divided into 2^{32} different cosets of $\mathcal{D}_{0,2}$.

[13] Observe that the first and the third diagonals of each text in $\mathcal{IS}$ are equal, as well as the second and the fourth ones.

The result follows immediately from the fact that each coset of $\mathcal{D}_I$ is mapped into a coset of $\mathcal{M}_I$ after 2-round AES encryption – see Theorem 1.

In the case of a random permutation, note that

- there are $\binom{2^{64}}{2^{32}}$ different ways to divide 2^{64} texts in sets of 2^{32} elements;
- for each set, 2^{32} texts are equal on two diagonals with prob. $\left(2^{-64}\right)^{2^{32}-1}$;
- the probability that these two diagonals are different for each set is equal to $\prod_{i=0}^{2^{32}-1} \frac{2^{64}-i}{2^{64}} = \prod_{i=0}^{2^{32}-1}\left(1 - i \cdot 2^{-64}\right)$.

As a result, the probability for the case of a random permutation is given by

$$\begin{aligned}
&\binom{2^{64}}{2^{32}} \cdot \prod_{i=0}^{2^{32}-1}\left[\left(2^{-64}\right)^{2^{32}-1} \cdot \overbrace{\left(1 - i \cdot 2^{-64}\right)}^{\leq 1}\right] \leq \binom{2^{64}}{2^{32}} \cdot \left(2^{-64}\right)^{2^{64}-2^{33}+1} \approx \\
&\approx \frac{1}{\sqrt{2\pi \cdot (2^{32}-1)}} \cdot \frac{\left(2^{64}\right)^{2^{64}}}{\left(2^{32}\right)^{2^{32}} \cdot \left(2^{64}-2^{32}\right)^{2^{64}-2^{32}}} \cdot \left(2^{-64}\right)^{2^{64}-2^{33}+1} \approx \\
&\approx \left(2^{32}\right)^{2^{32}} \cdot (1-2^{-32})^{2^{64}-2^{32}} \cdot \left(2^{-64}\right)^{2^{64}-2^{33}+1} \approx 2^{-2^{70}}
\end{aligned}$$

using Stirling's approximation $x! \approx x^x \cdot e^{-x} \cdot \sqrt{2\pi \cdot x}$. □

5.1.2 Weak-key subspace trails over 5-round AES-128

Exploiting the fact that $\Pr[x \in \mathcal{C}_J \mid x \in \mathcal{M}_I] = (2^8)^{-4|I|+|I|\cdot|J|}$ (see e. g. [Gra+16] for details) together with Eq. (6), it is possible to set up a 5-round weak-key subspace trail distinguisher on AES.

Proposition 2. *Let $I \subseteq \{0,1,2,3\}$ fixed. Then, the following probability holds:*

$$\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{weak}\right] = 2^{-95+16\cdot|I|} + 2^{-128+32\cdot|I|}. \quad (7)$$

Since for a random permutation Π the probability is equal to $(2^{-32})^{4-|I|}$, it is possible to distinguish the two cases.

Proof. To compute the previous probability, we first recall the *law of total probability*. Given a finite (or countably infinite) partition $B_1, \ldots, B_n$ of a sample space events in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ s. t. first $B_i \cap B_j = \emptyset$ for each $i \neq j$ and second $\bigcup_i B_i$ is the entire sample space, then

$$\Pr[A] = \sum_{i=1}^{n} \Pr[A|B_i] \cdot \Pr[B_i]$$

It follows that for a fixed I:

$$\begin{aligned}
&\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] = \\
=&\left\{\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; R^4(x) \oplus R^4(y) \in \mathcal{M}'\right] \times \Pr\left[R^4(x) \oplus R^4(y) \in \mathcal{M}'\right]\right\} + \\
&+\left\{\Pr\left[R^4(x) \oplus R^4(y) \notin \mathcal{M}'\right] \times \Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; R^4(x) \oplus R^4(y) \notin \mathcal{M}'\right]\right\} = \\
=&\, 2^{-64+16\cdot|I|} \cdot 2^{-31} + 2^{-32\cdot(4-|I|)} \cdot (1-2^{-31}) \simeq 2^{-95+16\cdot|I|} + 2^{-128+32\cdot|I|}
\end{aligned}$$

where $\mathcal{M}' = \mathcal{M}_{0,2} \cup \mathcal{M}_{1,3}$. □

To have concrete numbers:

- if $|I| = 3$ (I fixed), then

$$\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] \simeq 2^{-32} + 2^{-47},$$

  while for a random permutation $\Pr\left[\Pi(x) \oplus \Pi(y) \in \mathcal{M}_I \mid x, y \in \mathcal{IS}\right] = 2^{-32}$;

- if $|I| = 2$ (I fixed), then

$$\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] = 3 \cdot 2^{-64},$$

  while for a random permutation $\Pr\left[\Pi(x) \oplus \Pi(y) \in \mathcal{M}_I \mid x, y \in \mathcal{IS}\right] = 2^{-64}$;

- if $|I| = 1$ (I fixed), then

$$\Pr\left[R^5(x) \oplus R^5(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] = 2^{-79}.$$

  while for a random permutation $\Pr\left[\Pi(x) \oplus \Pi(y) \in \mathcal{M}_I \mid x, y \in \mathcal{IS}\right] = 2^{-96}$.

To better highlight this difference, focus on the last case. Note that given a subspace $\mathcal{IS}$ it is possible to construct $\binom{2^{64}}{2} = 2^{63} \cdot (2^{64} - 1) \simeq 2^{127}$ different pairs of texts. In the case e. g. $|I| = 1$, approximately $2^{127} \cdot 2^{-79} = 2^{48}$ different pairs of ciphertexts belong to the same coset of $\mathcal{M}_I$ for $|I| = 1$ fixed versus $2^{127} \cdot 2^{-96} = 2^{31}$ of a random permutation.

5.2 Weak-key "Multiple-of-n" property

At Eurocrypt 2017, Grassi et al. [Gra+17] presented the first property on 5-round AES (and AES-like ciphers) which is independent of the secret key and of the details of the S-Box and of the MixColumns (assuming branch number equal to 5). The result can be summarized as follows. Given $2^{32 \cdot |I|}$ plaintexts in the same coset of a diagonal space $\mathcal{D}_I$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ after 5-round AES is always a multiple of 8. This result can be used to set up a distinguisher, since for a random permutation the same property holds with probability 1/8.

In the case of a weak-key, we are able to extend the previous result up to 6-round AES-128. The obtained results are proposed in the following theorems.

Theorem 4. *Let $\mathcal{IS}$ and $\mathcal{M}_I$ be the subspaces defined as before for a fixed I with $1 \leq |I| \leq 3$. Assume that the whitening key is a weak-key, that is it belongs to the set K_{weak} as defined in Eq. (4). Given 2^{64} plaintexts in $\mathcal{IS}$, the number n of different pairs[14] of ciphertexts (c^i, c^j) for $i \neq j$ that belong to the same coset of $\mathcal{M}_I$ (that is $c^i \oplus c^j \in \mathcal{M}_I$) has the following property independently of the details of the S-Box:*

- *for 5-round AES-128, the number of collisions n is a multiple of 128, that is $\exists n' \in \mathbb{N}$ such that $n = 128 \cdot n'$;*
- *for 6-round AES-128, the number of collisions n is a multiple of 2, that is $\exists n' \in \mathbb{N}$ such that $n = 2 \cdot n'$.*

We emphasize that similar properties hold also in the *decryption* direction. In the following, we prove the theorem just given. In order to do this, we exploit a strategy similar to the one already proposed in [Gra+17] and recently re-visited in [Bou+19], where authors re-formulate the "multiple-of-8" property as an immediate consequence of an equivalence relation on the input pairs, under which the difference at the output of the round function is invariant. For this reason, we limit ourselves to highlight the main points of the proof, and we refer to [Bou+19; Gra+17] for more details.

[14] Two pairs (s, t) and (t, s) are considered to be equivalent.

Proof. First of all, note that the invariant subspace $\mathcal{IS}$ is mapped into a coset of $\mathcal{IS}$ after 2-round encryption, and similarly a coset of $\mathcal{M}_I$ is mapped into a coset of $\mathcal{D}_I$ after 2-round decryption, that is

$$\forall k \in K_{\text{weak}}: \qquad \mathcal{IS} \xrightarrow[\text{prob. } 1]{R^2(\cdot)} \mathcal{IS} \oplus a \xrightarrow{R(\cdot) \text{ or } R^2(\cdot)} \mathcal{D}_I \oplus a' \xrightarrow[\text{prob. } 1]{R^2(\cdot)} \mathcal{M}_I \oplus b'$$

Thus, the idea is to *focus only on the middle round(s)*, and to prove the following equivalent result. Given 2^{64} plaintexts in a coset of $\mathcal{IS}$, the number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ that belong to the same coset of $\mathcal{D}_I$ (that is $c^i \oplus c^j \in \mathcal{D}_I$) after 1 or 2 round(s) has the following property:

- for 1-round AES, the number of collisions n is a multiple of 128;
- for 2-round AES, the number of collisions n is a multiple of 2.

5-round AES Given a pair of texts $t^1, t^2 \in \mathcal{IS} \oplus a$, we are going to prove that there exist other pair(s) of texts $s^1, s^2 \in \mathcal{IS} \oplus a$ such that

$$R(t^1) \oplus R(t^2) \in \mathcal{D}_I \;\Leftrightarrow\; R(s^1) \oplus R(s^2) \in \mathcal{D}_I.$$

where the texts s^1, s^2 are given by any different combination of the generating variables of t^1, t^2.

By definition of $\mathcal{IS}$, let t^1 and t^2 be as

$$t^i = a \oplus \begin{bmatrix} x_0^i & x_4^i & x_0^i & x_4^i \\ x_1^i & x_5^i & x_1^i & x_5^i \\ x_2^i & x_6^i & x_2^i & x_6^i \\ x_3^i & x_7^i & x_3^i & x_7^i \end{bmatrix}, \qquad \text{that is} \quad t^i = a \oplus \bigoplus_{j=0}^{7} x_j^i \cdot (e_j \oplus e_{j+8}). \tag{8}$$

where $x_{l,j}$ or $x_{l+4\times j}$ denotes the byte in the l-th row and in the j-th column. For simplicity, let $t^i \equiv (x_0^i, x_1^i, x_2^i, x_3^i, x_4^i, x_5^i, x_6^i, x_7^i)$.

Consider initially the case in which all the generating variables are different, that is $x_j^1 \neq x_j^2$ for $j = 0, 1, \ldots, 7$. Let S be the set of pairs of texts $s^1, s^2 \in \mathcal{IS} \oplus a$ defined by swapping some generating variables of t^1 and t^2. In particular, given t^1 and t^2, the set S_{t^1,t^2} contains all 128 pairs of texts (s^1, s^2) for all $I \subseteq \{0, 1, 2, 3, 4, 5, 6, 7\}$ where

$$s^1 = a \oplus \bigoplus_{j=0}^{7} \left\{ \left[\left(x_j^1 \cdot \delta_j(I) \right) \oplus \left(x_j^2 \cdot \left[1 - \delta_j(I) \right] \right) \right] \cdot \left(e_j \oplus e_{j+8} \right) \right\}$$

$$s^2 = a \oplus \bigoplus_{j=0}^{7} \left\{ \left[\left(x_j^2 \cdot \delta_j(I) \right) \oplus \left(x_j^1 \cdot \left[1 - \delta_j(I) \right] \right) \right] \cdot \left(e_j \oplus e_{j+8} \right) \right\}$$

where the pairs (s^1, s^2) and (s^2, s^1) are considered to be equivalent, and where $\delta_x(A)$ is the Dirac measure defined as

$$\delta_x(A) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

As we are going to show, since

$$\forall (s^1, s^2) \in S_{t^1,t^2}: \qquad R(t^1) \oplus R(t^2) = R(s^1) \oplus R(s^2),$$

it follows that

$$\forall (s^1, s^2) \in S_{t^1,t^2}: \qquad R(t^1) \oplus R(t^2) \in \mathcal{D}_I \;\Leftrightarrow\; R(s^1) \oplus R(s^2) \in \mathcal{D}_I.$$

This is due to the facts that *the S-Box operation works independently on each byte and that the XOR-sum is commutative*. To show this, we propose the detailed computation for the byte in position (0, 0) – analogous for the other cases – of the previous difference

$$\begin{aligned}(R(t^1) \oplus R(t^2))_{0,0} &= \text{0x02} \cdot [\text{SB}(x_0^1 \oplus a'_{0,0}) \oplus \text{SB}(x_0^2 \oplus a'_{0,0})] \oplus \\ &\oplus \text{0x03} \cdot [\text{SB}(x_5^1 \oplus a'_{1,1}) \oplus \text{SB}(x_5^2 \oplus a'_{1,1})] \oplus [\text{SB}(x_2^1 \oplus a'_{2,2}) \oplus \\ &\oplus \text{SB}(x_2^2 \oplus a'_{2,2})] \oplus [\text{SB}(x_7^1 \oplus a'_{3,3}) \oplus \text{SB}(x_7^2 \oplus a'_{3,3})] = (R(s^1) \oplus R(s^2))_{0,0}\end{aligned}$$

where $a'_{i,i}$ for $i = 0, 1, 2, 3$ depends on the initial constant a that defines the coset of $\mathcal{IS}$ and on the secret key. Since each set S_{t^1,t^2} has cardinality 128, in the case in which one focuses on the pairs of texts with different generating variables, it follows that the multiple-of-128 property previously defined holds.

What happens if some variables are equal, e. g. $x_j^1 = x_j^2$ for $j \in J \subseteq \{0, \ldots, 7\}$ with $|J| \geq 1$? In this case, the difference $R(t^1) \oplus R(t^2)$ is independent of the value of $x_j^1 = x_j^2$ for each $j \in J$ (e. g. consider the difference $(R(t^1) \oplus R(t^2))_{0,0}$ in the byte (0,0) just given). As a result, the idea is to consider all the different pairs of texts given by swapping one or more variables x_l^1 and x_l^2 for $l = 0, 1, \ldots, 7$, *where x_j for $j \in J$ can take any possible value in $\mathbb{F}_{2^8}$*. Note that in the case in which $0 \leq |J| < 8$ variables are equal, it is possible to identify

$$\underbrace{2^{7-|J|}}_{\text{by swapping different gen. variables}} \times \underbrace{2^{8\cdot|J|}}_{\text{due to equal gen. variables}} = 2^{7\cdot(1+|J|)} = 128^{1+|J|}$$

different texts s^1 and s^2 in $\mathcal{IS} \oplus a$ that satisfy the condition $R(t^1) \oplus R(t^2) = R(s^1) \oplus R(s^2)$. More formally, given t^1 and t^2, the set S_{t^1,t^2} contains all $2^{7\cdot(1+|J|)}$ pairs of texts (s^1, s^2) for all $I \subseteq \{0, 1, 2, 3, 4, 5, 6, 7\} \setminus J$ and for all $\alpha_0, \ldots, \alpha_{|J|} \in \mathbb{F}_{2^8}$ where

$$s^1 = a \oplus \bigoplus_{j \in \{0,\ldots,7\} \setminus J} \left\{ \left[\left(x_j^1 \cdot \delta_j(I) \right) \oplus \left(x_j^2 \cdot \left[1 - \delta_j(I) \right] \right) \right] \cdot \left(e_j \oplus e_{j+8} \right) \right\} \oplus \bigoplus_{j \in J} \alpha_j \cdot \left(e_j \oplus e_{j+8} \right)$$

$$s^1 = a \oplus \bigoplus_{j \in \{0,\ldots,7\} \setminus J} \left\{ \left[\left(x_j^2 \cdot \delta_j(I) \right) \oplus \left(x_j^1 \cdot \left[1 - \delta_j(I) \right] \right) \right] \cdot \left(e_j \oplus e_{j+8} \right) \right\} \oplus \bigoplus_{j \in J} \alpha_j \cdot \left(e_j \oplus e_{j+8} \right)$$

In conclusion, given plaintexts in the same coset of $\mathcal{IS}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{D}_I$ after one round is a multiple of 128.

6-round AES: Super-S-Box In order to prove the previous claim, we use the "Super-S-Box" notation [DR06] – introduced by the designers of AES, where

$$\text{Super-S-Box}(\cdot) = \text{SB} \circ ARK \circ \text{MC} \circ \text{SB}(\cdot).$$

Given a pair of texts $t^1, t^2 \in \mathcal{IS} \oplus a$, we prove that there exist other pair(s) of texts $s^1, s^2 \in \mathcal{IS} \oplus a$ such that

$$R^2(t^1) \oplus R^2(t^2) \in \mathcal{D}_I \;\Leftrightarrow\; R^2(s^1) \oplus R^2(s^2) \in \mathcal{D}_I$$

where the texts s^1, s^2 are obtained by swapping the diagonals of t^1, t^2. In more details, if the diagonals are different (i. e., $[x_0^1, x_5^1, x_2^1, x_7^1] \neq [x_0^2, x_5^2, x_2^2, x_7^2]$ and $[x_1^1, x_4^1, x_3^1, x_6^1] \neq [x_1^2, x_4^2, x_3^2, x_6^2]$), given t^1 and t^2 defined as in (8)

$$\text{SR}(t^i) \equiv \Big(\underbrace{[x_0^i, x_5^i, x_2^i, x_7^i]}_{1st \text{ and } 3rd \text{ columns}} , \underbrace{[x_1^i, x_4^i, x_3^i, x_6^i]}_{2nd \text{ and } 4th \text{ columns}} \Big)$$

where $\mathrm{SR}(\cdot)$ denotes the ShiftRows operation, then s^1 and s^2 are defined as

$$\mathrm{SR}(s^i) \equiv \big(\underbrace{[x_0^{3-i}, x_5^{3-i}, x_2^{3-i}, x_7^{3-i}]}_{1st \text{ and } 3rd \text{ columns}}, \underbrace{[x_1^i, x_4^i, x_3^i, x_6^i]}_{2nd \text{ and } 4th \text{ columns}}\big).$$

To prove the previous fact, we first recall that 2-round encryption can be rewritten using the Super-S-Box notation

$$R^2(\cdot) = \mathrm{ARK} \circ \mathrm{MC} \circ \mathrm{SR} \circ \text{Super-S-Box} \circ \mathrm{SR}(\cdot).$$

Thus, we are going to prove that

$$\text{Super-S-Box}(\hat{t}^1) \oplus \text{Super-S-Box}(\hat{t}^2) \in \mathcal{W}_I \;\Leftrightarrow\; \text{Super-S-Box}(\hat{s}^1) \oplus \text{Super-S-Box}(\hat{s}^2) \in \mathcal{W}_I$$

where

$$\hat{t}^i = \mathrm{SR}(t^i) \in \mathcal{IS} \oplus \mathrm{SR}(a) \qquad \text{and} \qquad \hat{s}^i = \mathrm{SR}(s^i) \in \mathcal{IS} \oplus \mathrm{SR}(a)$$

for $i = 1, 2$ (note that $t^i, s^i \in \mathcal{IS} \oplus a$) and where the subspace $\mathcal{W}_I$ is defined as

$$\mathcal{W}_I = \mathrm{SR}^{-1}\,\mathrm{MC}^{-1}(\mathcal{D}_I).$$

Note that the first and the third columns of $\hat{t}^i$ and $\hat{s}^i$ are equal, as well as the second and the fourth columns. Similar to the 5-round case, *since the first and the second columns (and so the third and the fourth ones) of $\hat{t}^1$ and $\hat{t}^2$ depend on different and independent variables, since the Super-S-Box works independently on each column and since the XOR-sum is commutative,* it follows that

$$\text{Super-S-Box}(\hat{t}^1) \oplus \text{Super-S-Box}(\hat{t}^2) = \text{Super-S-Box}(\hat{s}^1) \oplus \text{Super-S-Box}(\hat{s}^2)$$

which implies the thesis.

What happens if one diagonal is in common for the two texts, e.g. $[x_0^1, x_5^1, x_2^1, x_7^1] = [x_0^2, x_5^2, x_2^2, x_7^2]$ (analogous for $[x_1^1, x_4^1, x_3^1, x_6^1] = [x_1^2, x_4^2, x_3^2, x_6^2]$)? As before, in this case the difference $R^2(t^1) \oplus R^2(t^2)$ is independent of the values of such diagonal. It follows that the pair of texts s^1 and s^2 can be constructed as

$$\mathrm{SR}(s^i) \equiv \big(\underbrace{[x_0^{3-i}, x_5^{3-i}, x_2^{3-i}, x_7^{3-i}]}_{1st \text{ and } 3rd \text{ columns}}, \underbrace{[\alpha_0, \alpha_5, \alpha_2, \alpha_7]}_{2nd \text{ and } 4th \text{ columns}}\big)$$

or

$$\mathrm{SR}(s^i) \equiv \big(\underbrace{[x_1^i, x_4^i, x_3^i, x_6^i]}_{1st \text{ and } 3rd \text{ columns}}, \underbrace{[\alpha_0, \alpha_5, \alpha_2, \alpha_7]}_{2nd \text{ and } 4th \text{ columns}}\big)$$

where $\alpha_0, \alpha_5, \alpha_2, \alpha_7$ can take any possible values in $\mathbb{F}_{2^8}$. Note that in the case, it is possible to identify $2 \cdot 2^{32} = 2^{33} \geq 2$ different texts s^1 and s^2 in $\mathcal{IS} \oplus a$ that satisfy the condition $R^2(t^1) \oplus R^2(t^2) = R^2(s^1) \oplus R^2(s^2)$. □

Generic Results on 5-round AES

In a similar way, it is possible to prove the following Theorem.

Theorem 5. *Let $\mathcal{IS}$ and $\mathcal{X}_I$ be the subspaces defined as before, for an arbitrary $I \subset \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0 \leq i,j \leq 3}$. Assume that the whitening key is a weak-key, i. e. it belongs to the set K_{weak} defined in Eq. (4). Given 2^{64} plaintexts in $\mathcal{IS}$, the number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ that belong to the same coset of $\mathcal{X}_I$ (i.e. $c^i \oplus c^j \in \mathcal{X}_I$) has the following property independently of the details of the S-Box:*

- *for 5-round AES-128, the number of collisions n is a multiple of 2, that is $\exists n' \in \mathbb{N}$ such that $n = 2 \cdot n'$.*

To prove this result, first of all note that with probability 1

$$\forall k \in K_{\text{weak}}: \qquad \mathcal{IS} \xrightarrow[\text{prob. } 1]{R^2(\cdot)} \mathcal{IS} \oplus a \xrightarrow{R^2(\cdot)} \mathcal{Y}_I \oplus a' \xrightarrow[\text{prob. } 1]{R(\cdot)} \mathcal{X}_I \oplus b'$$

where $\mathcal{Y}_I = SR^{-1} \circ MC^{-1}(\mathcal{X}_I)$ as showed in Theorem 3 (remember that $\text{SB}(\mathcal{X}_I) = \mathcal{X}_I$).

Using the same technique as before (i. e. working with the Super-S-Box notation and by swapping the generating diagonals of a pair of texts), the idea is to focus on the middle rounds only, and to show that given 2^{64} plaintexts in a coset of $\mathcal{IS}$, the number n of different pairs of ciphertexts (c^i, c^j) for $i \neq j$ that belong to the same coset of $\mathcal{Y}_I$ (that is $c^i \oplus c^j \in \mathcal{Y}_I$) after 2 rounds is always a multiple of 2.

Multiple-of-n property for AES-192/256

As we have seen in Section 4.3, it is possible to set up a weak invariant subspace of length two for 2^{64} weak-keys of AES-192, and a weak invariant subspace of length two/three/four/five for $2^{128}/2^{96}/2^{64}/2^{32}$ weak-keys of AES-256. Due to the argumentations just given, it follows that the multiple-of-128 property holds for up to 7-round AES-256, while the multiple-of-2 property holds for up to 9-round AES-256.

5.3 Practical experiments

Most of the previous properties have been practically verified[15]. Here we briefly present the practical results and we compare them with the theoretical ones.

All our distinguishers are based on $\mathcal{IS}$ and their practical verification requires at least 2^{64} reduced-round AES encryptions. For this reason, we performed our experiments on small-scale AES [Cid+05], where each word is composed of 4-bit instead of 8 (note that all previous results are independent of the details of the S-Box operation). This implies that the dimension of $\mathcal{IS}$ reduces to 32 bits from 64.

Practical Results. For Theorem 4, we performed 5-round and 6-round encryptions of $\mathcal{IS}$ for more than 100 randomly chosen weak-keys in K_{weak}. We counted the collisions in each of the four inverse diagonals space $\mathcal{ID}$ and observed the multiple-of-128 and multiple-of-2 properties hold for 5-round and 6-round encryptions, respectively. Moreover, we also verified the multiple-of-n for up to 8-round AES-256 property for the corresponding case of weak-keys. Similar tests have been performed in order to check the multiple-of-2 property on the subspaces $\mathcal{X}_I$ as defined in Definition 8 for each $|I| \leq 4$. Due to increased time and memory complexity, these properties were not verified for $|I| > 4$.

For the follow-up, we also performed the same experiments in decryption direction. In particular, we performed small-scale AES decryptions of $\mathcal{IS}$ for more than 100 randomly chosen keys in K_{weak}. We counted the collisions in each of the four diagonals and observed that multiple-of-128 and multiple-of-2 properties for 4-round and 5-round decryptions, respectively. We got similar results for the multiple-of-2 property on the subspaces $\mathcal{X}_I$. The experiment results coincide with the results provided in Tables 2 and 3.

6 New chosen-key distinguishers for AES in the single-key setting

In this section we present new chosen-key distinguishers for AES in the single-key setting. In particular, as major results, we are able to present *the first candidate 10-round chosen-key distinguisher for AES-128* and *a 14-round candidate chosen-key distinguisher for*

[15] The source codes of the distinguishers/attacks will be made public with the publication of this paper. It is also part of the submission as supplement material.

Table 3: *AES Chosen-Key Distinguishers.* The computation cost is the cost to generate N-tuples of plaintexts/ciphertexts. "SK" denotes a chosen-key distinguisher in the Single-Key setting, while "RK" denotes a chosen-key distinguisher in the Related-Key setting. Distinguishers proposed in this paper are in bold. For completeness, we mention that the known-key distinguishers presented in Gilbert [Gil14] are excluded from this Table due to the arguments reported in Section 6.1.2.

AES	Rounds	Computations	Property	SK	RK	Reference
AES-128	8	2^{24}	Multiple Diff. Trail	✓		[Der+12]
	9	2^{55}	Multi-Collision Diff.		✓	[Fou+13]
	10 (full)	$\mathbf{2^{64}}$	**Multiple-of-n** (2^{32} keys)	✓		§ 6.5.1
	10 (full)	$\mathbf{2^{64}}$	**Multiple-of-n** (1 key)	✓		§ 6.5.2
AES-192	**10**	$\mathbf{2^{64}}$	**Multiple-of-n** (2^{32} keys)	✓		App. G.2.2
	10	$\mathbf{2^{64}}$	**Multiple-of-n** (1 key)	✓		App. G.2.3
	11	$\mathbf{2^{64}}$	**Multiple-of-n** (1 key)	✓		App. G.2.4
AES-256	9	2^{24}	Multiple Diff. Trail	✓		[Der+12]
	13	$\mathbf{2^{64}}$	**Multiple-of-n** (2^{32} keys)	✓		App. G.3.2
	13	$\mathbf{2^{64}}$	**Multiple-of-n** (1 key)	✓		App. G.3.3
	14 (full)	$\mathbf{2^{64}}$	**Multiple-of-n** (1 key)	✓		App. G.3.4
	14 (full)	2^{120}	Multi-Collision Diff.		✓	[Bir+09]

AES-256, both in the single-key setting. All the distinguishers that we present are based on the (practically verified) multiple-of-n property proposed in Section 5.2.

The goal of an open-key distinguisher is to differentiate between a block cipher E which allows to generate plaintext/ciphertext pairs which exhibit a rare relation, even for a small set of keys or a single key, and an ideal cipher Π that does not have such a property. However, this poses a definitional problem as it was shown already in [Can+04] that any concrete implementable cipher (like the AES) can be trivially distinguished from an ideal cipher. To the best of our knowledge, finding a proper formal definition that captures the intuition behind chosen-key distinguishers has been a challenging task for the last fifteen years and is still an open problem.

We do not attempt to address this formalization challenge here, but proceed in the way that is custom in the literature to describe chosen-key distinguisher: *(1st)* describe the rare property (see Section 6.2), *(2nd)* show that it can be efficiently constructed for the block cipher usually using an inside-out approach (see Section 6.3 for 9-round AES-128 and Section 6.5 for 10-round AES-128), and *(3rd)* argue or prove in some model that any generic method is less efficient or has low success probability (see Section 6.4).

Our results are summarized in Table 3. As before, we give all the details for the AES-128 case, and we refer to Appendix G for the corresponding distinguishers on AES-192 and AES-256.

6.1 Open-key distinguishers – state of the art for AES

6.1.1 Chosen-key distinguishers – State of the art for AES

To the best of our knowledge, the only chosen-key distinguisher for AES in the single-key setting is proposed in [Der+12]. In their paper, the chosen-key model asks the adversary to find two plaintexts/ciphertexts pairs and a key such that the two plaintexts are equal in 3 diagonals and the two ciphertexts are equal in 3 anti-diagonals (if the final

MixColumns is omitted). Equivalently, using the subspace trail notation, the goal is to find $(p^1, c^1 \equiv R^8(p^1))$ and $(p^2, c^2 \equiv R^8(p^2))$ for $p^1 \neq p^2$ s.t. $p^1 \oplus p^2 \in \mathcal{D}_I$ and $c^1 \oplus c^2 \in \mathcal{M}_J$ for a certain $I, J \subseteq \{0, 1, 2, 3\}$ s.t. $|I| = |J| = 1$.

This problem is equivalent to the one proposed in [GP10; Jea+14] in the known-key scenario. In particular, the main (and only) difference between the known-key and chosen-key distinguishers is related to the freedom to choose the key, and consequently to the computational cost. In more details, due to this freedom, for the 8-round AES-128 case it is possible to find the required pairs of plaintexts/ciphertexts with 2^{24} computations instead of 2^{44}, while the computational cost in the case of an ideal cipher is of 2^{64} in both cases. For completeness, a similar result is proposed for 9-round AES-256.

The chosen-key model has been popularized some years before by Biryukov et al. [Bir+09], since a distinguisher in this model has been extended to a related-key attack on full AES-256. A related distinguisher for 9-round AES-128 has been proposed by Fouque et al. [Fou+13]. Both the chosen-key distinguisher proposed in these papers are in the related-key setting. Here we briefly recall them, but we emphasize that we do not consider related-keys in this article. In [Bir+09], authors show that it is possible to construct a *q-multicollision* on Davies-Meyer compression function using AES-256 in time $q \cdot 2^{67}$, whereas for an ideal cipher it would require on average $q \cdot 2^{\frac{q-1}{q+1}128}$ time complexity. A similar approach has been exploited in [Fou+13] to set up the first chosen-key distinguisher for 9-round AES-128. Here, the chosen-key model asks the adversary to find a pair of keys (k, k') satisfying $k \oplus k' = \delta$ with a *known (fixed)* difference δ, and a pair of messages $(p^1, c^1 \equiv R^9(p^1))$ and $(p^2, c^2 \equiv R^9(p^2))$ conforming to a partially instantiated differential characteristic in the data part.

We conclude observing that an attack/distinguisher with *no* key difference is (logically) harder, since the attacker has less freedom.

6.1.2 Gilbert's Known-Key Distinguisher for AES

For completeness, we mention that a 10-round known-key distinguisher for AES has been proposed by Gilbert [Gil14] at Asiacrypt 2014. In order to highlight the main differences with our work, we briefly recall it here. Here, the known-key model asks the adversary to find a set of 2^{64} (plaintext, ciphertext) pairs, that is (p^i, c^i) for $i = 0, \ldots, 2^{64} - 1$, and a key k with the following properties[16]:

1. there exists a key k^0 s. t. the bytes of $\{R_{k^0}(p^i)\}_i$ are uniformly distributed, or equivalently that the texts $\{R_{k^0}(p^i)\}_i$ are uniformly distributed among the cosets of $\mathcal{D}_I$ for each I with $|I| = 3$;

2. there exists a key k^{10} s. t. the bytes of $\{\mathrm{MC}^{-1} \circ R^{-1}_{k^{10}}(c^i)\}_i$ are uniformly distributed (MC^{-1} denotes the inverse MixColumns operation), or equivalently that the texts $\{R^{-1}_{k^{10}}(c^i)\}_i$ are uniformly distributed among the cosets of $\mathcal{M}_J$ for each J with $|J| = 3$.

We emphasize that *such properties are not verified* directly *by the plaintexts and by the ciphertexts but after one round encryption/decryption*, and they involve *keys* k^0 *and* k^{10} *that can be different from the "real" encryption subkeys derived from* k.

We briefly recall that the probability that 2^{64} (plaintext, ciphertext) generated by a random permutation satisfy the previous property is $2^{-7\,200}$. Thus, *given* $2^{64} + 2^8$ *plaintexts/ciphertexts, the probability to find among them a subset of* 2^{64} *pairs of texts that satisfy the previous property is close to 1.*

[16] For this distinguisher, we abuse the notation k^r to denote a key of a certain round r. We emphasize that k^r is not necessarily equal to the secret key, that is k^r can be different from the r-th subkey. In other words, it is only required that such a key exists, and not that it is equal to the real secret key.

As a result, a distinguisher based on the Gilbert's technique is different from all the previous distinguishers up to 8 rounds present in the literature. *For all distinguishers up to 8-round (and for the distinguishers proposed in this paper), the property/relation $\mathcal{R}$ – that the N-tuple of (plaintexts, ciphertexts) must satisfy – does not involve any operation of the block cipher E.* When $\mathcal{R}$ does not re-use operations of E, this provides some heuristic evidence that this distinguisher can be considered *meaningful*. On the other hand, *the previous Gilbert's like distinguishers do not satisfy this requirement, since in these cases the property/relation $\mathcal{R}$ involves and re-uses some operations of E.* The *crucial point* is that instead to consider properties "directly" on the plaintexts/ciphertexts, the idea is to show that there exist certain keys for which some properties hold after one round encryption/decryption.

In [Gil14], argumentation are given in order to support such known-key distinguishers and why they should not be systematically ruled out as if they were *artificial* (see [Gil14, Section 3] for details). On the other hand, even if Gilbert's known-key distinguisher leads to statements on more rounds of AES than ever before (without related keys), in the same paper it is also observed that its "impact on the security of [...] AES when used as a known key primitive, e. g. in a hash function construction, is questionable" ([Gil14, Abstract]).

Moreover, in order to support such a new kind of distinguisher, it is claimed in Gilbert [Gil14] that *(1st)* it seems technically difficult to use a stronger property than the uniform distribution one to extend an 8-round known-key distinguisher to a 10-round one and *(2nd)* it is impossible to use the same technique in order to extend a distinguisher for more than 2 rounds. Recently, both claims have been disproved in Grassi and Rechberger [GR17], in which authors exploit the same technique to propose *(1st)* a distinguisher on 10-round AES based on truncated differential trails and *(2nd)* the *first distinguisher on 12-round AES* obtained by extending an 8-round distinguisher.

As a result, the problem to set up a 9 (or more) rounds open-key distinguisher in the single-key setting for AES-128 without exploiting the Gilbert's technique (i. e. that exploits a property which can be directly verified on the plaintexts/ciphertexts without any key-guessing) is still open. In the following, we are able to provide the *first* solution to this problem.

6.2 The "Simultaneous Multiple-of-n" property

In our distinguisher, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^9(p^i))$ for $i = 0, \ldots, 2^{64} - 1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that the following "*simultaneous multiple-of-n*" property is satisfied:

- for each $J, I \subseteq \{0, 1, 2, 3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ are a multiple of $128 = 2^7$;
- for each $J, I \subset \{(0,0), (0,1), \ldots, (3,2), (3,3)\} \equiv \{(i,j)\}_{0 \le i,j \le 3}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathrm{MC}(\mathcal{X}_I)$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_J$ are a multiple of 2, where $\mathcal{X}$ is defined as in Definition 8.

For the follow-up, we remark and highlight that the subspaces $\mathcal{X}$ are independent, in the sense that e. g. the fact that the multiple-of-2 property is satisfied by $\mathcal{X}_I$ and/or $\mathcal{X}_J$ does not imply anything on $\mathcal{X}_{I \cup J}$ and vice-versa. This is due to the fact that given $\mathcal{X}_I$ and $\mathcal{X}_J$, then $\mathcal{X}_I \cup \mathcal{X}_J \subsetneqq \mathcal{X}_{I \cup J}$ if $\mathcal{X}_{I \cup J} \neq \mathbb{F}_{2^8}^{4 \times 4}$. As a result, any information about the multiple-of-n property on $\mathcal{X}_I, \mathcal{X}_J$ (and so $\mathcal{X}_I \cup \mathcal{X}_J$) is useless to derive information about the multiple-of-n property on $\mathcal{X}_{I \cup J} \setminus (\mathcal{X}_I \cup \mathcal{X}_J)$ – assuming $\mathcal{X}_{I \cup J} \neq \mathbb{F}_{2^8}^{4 \times 4}$.

6.3 9-round chosen-key distinguisher for AES-128

To find a set of 2^{64} plaintexts/ciphertexts with the required "simultaneous multiple-of-n" property, the distinguisher exploits the fact that *the required property can be fulfilled by starting in the middle with a suitable set of texts.* In particular, the idea is simply to *choose the key such that the subkey of the 4-th round k^4 belongs the subset K_{weak} defined as in Eq. (4).* Thus, consider the invariant subspace $\mathcal{IS}$ defined as in Eq. (2), and define the 2^{64} plaintexts as the 4-round decryption of $\mathcal{IS}$ and the corresponding ciphertexts as the 5-round encryption of $\mathcal{IS}$. Due to the secret-key distinguishers just presented, this set satisfies the required "simultaneous multiple-of-n" property.

In more details, due to the assumption on the key (that is, $k^4 \in K_{\text{weak}} \subseteq \mathcal{IS}$), note that the subspace $\mathcal{IS}$ is mapped into a coset of $\mathcal{IS}$ after two rounds of encryption and one round of decryption, that is

$$\forall k^4 \in K_{\text{weak}}: \qquad \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus \tilde{k}.$$

Due to the results of Section 5.2 and since $k^4 \in K_{\text{weak}}$, the multiple-of-n properties hold with probability 1 on the plaintexts and on the ciphertexts

$$\text{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus \tilde{k} \xrightarrow{R^3(\cdot)} \text{Multiple-of-}n$$

It follows that the required set can be constructed using 2^{64} computations. Moreover, we emphasize that our experiments on the secret-key distinguishers of Section 5.2 implies the *practical verification of this distinguisher.* What remains is to give arguments as to why producing that property simultaneously on the plaintext and ciphertext side of an ideal cipher is unlikely to be as efficient.

6.4 Achieving the "Simultaneous Multiple-of-n" property generically

In this case, the adversary faces a family of random and independent *ideal ciphers*[17] $\{\Pi(K,\cdot), K \in \{0,1\}^k\}$, where $k = 128, 192, 256$ respectively for the cases AES-128/192/256. His goal is to find a key k and a set of 2^{64} plaintexts/ciphertexts $(p^i, c^i = \Pi(k, c^i))$ s.t. the "simultaneous multiple-of-n" property is satisfied. As we are going to show, *the probability to find a set of 2^{64} plaintexts/ciphertexts pairs (X_i, Y_i) that satisfies the "simultaneous multiple-of-n" property for a random permutation is upper bounded by $2^{-65\,618}$.*

As first thing, we discuss the freedom to choose the key. Since the adversary does not know the details of the ideal cipher Π, he does not have any advantage to choose a particular key instead of another one. For this reason, in the following we limit to consider the case in which the permutation Π is instantiated by a key chosen at random in the set $\{0,1\}^k$.

Our goal is to prove that the success probability of any oracle algorithm of overall time complexity upper bounded by 2^{64} is negligible[18].

Proposition 3. *Given a perfect random permutation Π or Π^{-1} of $\{0,1\}^{128}$ (e. g. instantiated by an ideal cipher with a fixed key uniformly chosen at random in $\{0,1\}^k$), consider $N = 2^{64}$ oracle queries made by any algorithm $\mathcal{A}$ to the perfect random permutation Π or Π^{-1}. Denote this set of 2^{64} plaintexts/ciphertexts pairs by (X_i, Y_i) for $i = 0, \ldots, 2^{64}-1$, where $Y_i = \Pi(X_i)$. The probability that $\mathcal{A}$ outputs a set of 2^{64} plaintexts/ciphertexts pairs (X_i, Y_i) for $i = 0, \ldots, 2^{64}-1$ that satisfies the "simultaneous multiple-of-n" property is upper bounded by $2^{-65\,618}$.*

[17] An ideal cipher Π is defined as $\Pi : (k,p) \in \{0,1\}^k \times \{0,1\}^n \to c = \Pi(k,p) \in \{0,1\}^n$ such that for each fixed $k \in \{0,1\}^k$, $\Pi(k,\cdot)$ is a permutation, that is $\exists \Pi^{-1}(k,\cdot) : \{0,1\}^n \to \{0,1\}^n$ s. t. $\Pi^{-1}(k, \Pi(k,\cdot)) = \mathbb{I}(\cdot)$ where $\mathbb{I}(\cdot)$ is the identity function.

[18] We highlight that the proof of Proposition 3 is based on a strategy (very) similar to the one proposed in Gilbert [Gil14] to prove that the uniform distribution (on 8- and 10-round AES) is generically hard.

Proof. For completeness, consider first the case of a dishonest algorithm $\mathcal{A}$. Given $N-1$ pairs (X_i, Y_i) generated by the perfect random permutation Π or Π^{-1}, assume the player chooses X_N and Y_N in order to satisfy the "simultaneous multiple-of-n" property. If at least one of the N pairs (X_i, Y_i) output by $\mathcal{A}$ does not result from a query X_i to Π or a query Y_i to Π^{-1}, then the probability that for this pair $Y_i = \Pi(X_i)$ and consequently the success probability of $\mathcal{A}$ is upper bounded[19] by $\frac{1}{2^{128}-(N-1)}$.

From now one, we consider only the case of honest algorithm $\mathcal{A}$, that is we assume all the pairs (X_i, Y_i) result from queries to Π or Π^{-1}. Consider a (random) set of $2^{64}-1$ plaintexts/ciphertexts pairs $\{(X_i, Y_i)\}_{i=0,\ldots,2^{64}-2}$ such that there exists (at least) one plaintext/ciphertext pair $(\hat{X}, \hat{Y})$ for which the required multiple-of-n property is satisfied. By assumption, the player can always find $\hat{X}'$ (resp. $\hat{Y}'$) such that the "simultaneous multiple-of-n" property is satisfied for the plaintexts (resp. for the ciphertexts). However, the oracle's answer $\hat{Y}'$ (resp. $\hat{X}'$) is *uniformly drawn* from $\{0,1\}^{128} \setminus \{Y_1, Y_2, \ldots, Y_{2^{64}-1}\}$ (resp. from $\{0,1\}^{128} \setminus \{X_1, X_2, \ldots, X_{2^{64}-1}\}$). Therefore, *the probability that the answer to the N-th query allows the output of $\mathcal{A}$ to satisfy property $\mathcal{R}$ (i. e. multiple-of-n) is upper bounded by* $(2^{-1})^{2^{16}-16} \cdot (2^{-7})^{14} = 2^{-65\,618} \simeq 2^{-2^{16}}$ since

- there are $\sum_{i=1}^{15} \binom{16}{i} = 2^{16} - 2$ different subspaces $\mathcal{X}_I$ for which the multiple-of-2 property holds, and among them there are 14 subspaces $\mathcal{M}_I$ for which the multiple-of-128 property holds;
- the probability that the number of collisions is a multiple of N is (approximately) $1/N$.

In order to prove this second point, we first show that the probabilistic distribution of the number of collisions is a binomial distribution[20].

Given a set of n pairs texts, consider the event that m pairs belong to the same coset of a subspace $\mathcal{X}$. As first thing, we show that the probabilistic distribution of number of collisions is simply described by a *binomial distribution*. By definition, a binomial distribution with parameters n and p is the discrete probability distribution of the number of successes in a sequence of n independent yes/no experiments, each of which yields success with probability p. In our case, given n pairs of texts, each of them satisfies or not the above property/requirement with a certain probability. Thus, this model can be described using a binomial distribution, for which the mean μ and the variance σ^2 are respectively given by $\mu = n \cdot p$ and $\sigma^2 = n \cdot p \cdot (1-p)$.

In our case, the number of pairs is given by $\binom{2^{64}}{2} \simeq 2^{127}$, the probability that a pair of texts belong to the same coset of $\mathcal{X}_I$ is equal to $2^{-8\cdot(16-|I|)}$, while it is equal to $2^{-32\cdot(4-|J|)}$ for the subspaces $\mathcal{D}_J$ and $\mathcal{M}_J$. A detailed analysis of these probabilities[21] is provided in Appendix D.

Probability that "the number of collision is even" is (approximately) 1/2 – Case:

[19] Note that there are 2^{128} different pairs (X, Y). If $N-1$ are already given, the probability that $Y_i = \Pi(X_i)$ holds is $(2^{128} - (N-1))^{-1}$.

[20] We highlight that the fact that "the probability that the number of collisions is a multiple of N is $1/N$" is obvious *if* the probabilistic distribution of the number of collisions is a uniform one, which is not the case.

[21] Given n texts generated by a random permutation $\Pi(\cdot)$, one can construct $\binom{n}{2}$ different pairs which are *not* independent. For example, consider a pair of texts (t^1, t^2). Given another text t^3, if $t^1 \oplus t^3 \in \mathcal{X}$ and $t^2 \oplus t^3 \in \mathcal{X}$, then (t^1, t^2) belong to the same coset of $\mathcal{X}$ with prob. 1 (by definition of subspace). Thus, one may think that the probability that (t^1, t^2) are in the same coset of $\mathcal{X}$ is different than $2^{-8\cdot(16-|I|)}$. In Appendix D, we prove that even if the pairs are not independent, the probability that each pair (t^1, t^2) satisfies the property to belong to the same coset of $\mathcal{X}$ is exactly $2^{-8\cdot(16-|I|)}$. Similar arguments hold for the subspaces $\mathcal{D}$ and $\mathcal{M}$.

subspaces $\mathcal{X}_I$. The probability that the number of collisions is even is given by

$$\frac{1}{2} + \frac{1}{2} \cdot (1-2p)^n$$

where note that n is an even number. In our case, since $n \simeq 2^{127}$ and $2^{-120} \leq p \leq 2^{-8}$ (where the prob. 2^{-120} and 2^{-8} correspond resp. to the cases $|I| = 15$ and $|I| = 1$), the previous probability is well approximated by $1/2 + 1/2 \cdot (1-2^{-7})^{2^{127}} \approx 1/2$.

In order to prove the previous result, let X be a binomial distribution $X \sim \mathcal{B}(n,p)$. Combining the facts that

$$\Pr[X \text{ even}] + \Pr[X \text{ odd}] = \sum_{k=0}^{n} \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k} = [(1-p)+p]^n = 1$$
$$\Pr[X \text{ even}] - \Pr[X \text{ odd}] = \sum_{k=0}^{n} \binom{n}{k} \cdot (-p)^k \cdot (1-p)^{n-k} = [(1-p)-p]^n$$

where

$$\Pr[X \text{ even}] = \sum_{k=0}^{n/2} \binom{n}{2k} \cdot p^{2k} \cdot (1-p)^{n-2k}$$
$$\Pr[X \text{ odd}] = \sum_{k=0}^{n/2-1} \binom{n}{2k+1} \cdot p^{2k+1} \cdot (1-p)^{n-2k-1},$$

it follows that $\Pr[X \text{ even}] = \frac{1}{2} + \frac{1}{2} \cdot (1-2p)^n$.

Probability that "the number of collision is a multiple of N" is (approximately) $1/N$ – Case: subspaces $\mathcal{M}_J$ and $\mathcal{D}_J$. In order to prove this result, we first approximate the binomial distribution with a normal one. De Moivre-Laplace Theorem claims that the normal distribution is a good approximation of the binomial one *if* the skewness of the binomial distribution – given by $(1-2p)/\sqrt{n \cdot p \cdot (1-p)}$ – is close to zero. In our case, since $n \simeq 2^{127}$ and $2^{-96} \leq p \leq 2^{-32}$ (where the prob. 2^{-96} and 2^{-32} correspond resp. to the cases $|J| = 3$ and $|J| = 1$), it follows that $2^{-47.5} \leq skew \leq 2^{-15.5}$, which means that the normal approximation is sufficiently good. Thus, we approximate the binomial distribution with a normal one $\mathcal{N}(\mu = n \cdot p, \sigma^2 = n \cdot p \cdot (1-p))$, where the probability density function is given by $\varphi(x) = \frac{1}{\sqrt{2\pi \cdot \sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$.

In this case, what is the probability that the multiple-of-N collisions is satisfied? To answer this question, it is sufficient to sum all the probabilities where the number of collisions is a multiple-of-N (for $N \in \mathbb{N}$ and $N \neq 0$), that is

$$\sum_{x \in \mathbb{Z}} \frac{1}{\sqrt{2\pi \cdot \sigma^2}} e^{-\frac{(N \cdot x - \mu)^2}{2\sigma^2}} = \frac{1}{N} \cdot \underbrace{\sum_{x \in \mathbb{Z}} \frac{1}{\sqrt{2\pi \cdot \tilde{\sigma}^2}} e^{-\frac{(x-\tilde{\mu})^2}{2\tilde{\sigma}^2}}}_{=1 \text{ by definition}} = \frac{1}{N}$$

where $\tilde{\mu} = \mu/N$ and $\tilde{\sigma}^2 = \sigma^2/N^2$. Obviously, if $N = 1$, then this probability is equal to 1. □

What happens if the adversary performs more than 2^{64} computations? To answer this question, we first compute the probability that a *random* set of 2^{64} plaintexts/ciphertexts generated by the same key satisfies the "simultaneous multiple-of-n" property. As we have just seen, the "simultaneous multiple-of-n" property is satisfied with probability $(2^{-65\,618})^2 = 2^{-131\,236} \simeq 2^{-2^{17.002}}$ (see Appendix D for details). As a result, given $2^{64} + 2^{12}$

random texts, the player can find a set of 2^{64} texts that satisfy the required property both on the plaintexts and on the ciphertexts, since it is possible to construct

$$\binom{2^{64}+2^{12}}{2^{64}} \approx \frac{(2^{64})^{2^{12}}}{2^{12}!} \simeq 2^{2^{17.7}}$$

different sets of 2^{64} texts (where $n! \simeq (n/e)^n \cdot \sqrt{2\pi n}$ by Stirling's approximation).

On the other hand, *the cost to identify the right* 2^{64} *texts among all the others is in general much higher than* 2^{64} *computations.* Indeed, to have a chance of success higher than 95%, one must consider approximately $3 \cdot 2^{131\,236}$ different sets, since $1-(1-2^{-131\,236})^{3\cdot 2^{131\,236}} \simeq 1-e^{-3} \equiv 0.95$, which implies an overall cost much higher than the cost of the shortcut player.

Moreover, consider the following. Given a set of random texts, suppose to change one plaintext in order to modify the number of collisions in the subspace $\mathcal{X}_I$ (or/and $\mathcal{D}_I$) for a particular I. The problem is that all the other numbers of collisions in the subspace $\mathcal{X}_J$ (or/and $\mathcal{D}_J$) for all $J \neq I$ change. Even if it is possible to have control of these numbers, also the numbers of collisions among the ciphertexts in each subspace $\mathrm{MC}(\mathcal{X}_K)$ and $\mathcal{M}_K$ change, and in general it is not possible to predict such change in advance. In particular, we recall that the number of collisions in a subspace $\mathcal{D}_I$ (resp. $\mathcal{M}_I$) is on average $2^{127} \cdot 2^{-128+32\cdot|I|} = 2^{32\cdot|I|-1} \gg 1$, which implies that the change in one text modifies all the numbers of collisions in each subspaces $\mathcal{D}_I$ or/and $\mathcal{M}_I$ for each $I \subseteq \{0,1,2,3\}$. Similarly, the number of pairs of texts with $1 \leq |J| \leq 15$ equal bytes (that is, that belong to the same coset of a particular subspace $\mathcal{X}_J$) is on average equal to $2^{127} \cdot 2^{-8\cdot|J|} \geq 2^{127} \cdot 2^{-8\cdot 15} = 2^7$, which implies that the change in one text modifies all the numbers of collisions in each subspaces $\mathcal{X}_J$ or $\mathrm{MC}(\mathcal{X}_J)$ for each $J \subseteq \{e_{i,j}\}_{0\leq i,j\leq 3}$. We conjecture that that *there is no (efficient) strategy* – that does not involve brute force research – *to fulfill the required "simultaneous multiple-of-n" property* for which the cost is approximately of 2^{64} computations (or lower). The problem to *formally* prove this fact is left for future work.

Remarks Before going on, we highlight that this claim/result is not true in general if one considers *only* the multiple-of-n property (for $n < 8$) in the subspaces $\mathcal{D}_I$ and $\mathcal{M}_J$, that is, not for the generic subspaces $\mathcal{X}$. In particular, in Appendix G.4 we consider a distinguisher on full AES-192 which is based on the simultaneous multiple-of-2 property both on the plaintexts (in $\mathcal{D}_I$) and the ciphertexts (in $\mathcal{M}_J$). In that section, we present a strategy that an adversary who does not know the key can use to generate a set of texts with the required property $\mathcal{R}$ *at (almost) the same cost* of one who knows the key.

Finally, for a broader understanding of the role of the invariant subspace in the previous distinguishers, in Appendix F we discuss the (im)possibility to set up an open-key distinguisher using the multiple-of-8 property proposed in [Gra+17] for more than 8-round AES.

6.5 Chosen-key distinguisher for 10-round AES-128

To set up the chosen-key distinguisher for 10-round AES-128, two possible approaches can be considered:

- use the previous distinguisher on 9-round as a starting point and *add one round at the beginning (or at the end)* by exploiting a weaker property on the plaintexts (or on the ciphertexts);
- use the previous distinguisher on 9-round as a starting point and *add one round in the middle* by using the remaining degrees of freedom in the choice of the key.

Chosen-Key Distinguishers for AES-192 and (full) AES-256. Before going on, we mention that same techniques are used in Appendix G in order to propose "simultaneous multiple-of-n" distinguishers on 11-round AES-192 and on 14-round AES-256, which is *the first open-key distinguisher on full AES-256 in the single-key setting*. Finally, we also present a chosen-key distinguisher on 10-round AES-128 instantiated with the key-schedule proposed in [Nik11]. We refer to Appendix G for all details.

6.5.1 Chosen-key distinguisher on 10-round AES – Exploit a weaker property

In the first approach, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^{10}(p^i))$ for $i = 0, \ldots, 2^{64}-1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that the following "*simultaneous multiple-of-n*" property is satisfied:

Plaintext: on the plaintexts, we re-use the previous properties:

(1st) for each $J \subseteq \{0,1,2,3\}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_J$ is a multiple of $128 = 2^7$;

(2nd) for each $I \subseteq \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_I$ are a multiple of 2;

Ciphertext: for each $J \subseteq \{0,1,2,3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ is a multiple of 2.

Choosing one of the 2^{32} keys proposed for the 9-round distinguisher given in Section 6.3, it is possible to construct such set with a computational cost of 2^{64}. In more details:

- due to the assumption on the key (that is, $k^4 \in K_{\text{weak}} \subseteq \mathcal{IS}$), note that the subspace $\mathcal{IS}$ is mapped into a coset of $\mathcal{IS}$ after two rounds encryption and one round decryption, that is

$$\forall k^4 \in K_{\text{weak}}: \qquad \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus \tilde{k};$$

- due to the results of Section 5.2, given $k^4 \in K_{\text{weak}}$, (*1st*) the multiple-of-128 property (on $\mathcal{D}_J$) and the multiple-of-2 property (on $\mathcal{X}_I$) hold on the plaintexts while (*2nd*) the multiple-of-2 property (on $\mathcal{M}_J$) holds on the ciphertexts

$$\text{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus \tilde{k} \xrightarrow{R^4(\cdot)} \text{Multiple-of-2}$$

What about an adversary facing a family of random and independent *ideal ciphers*? Due to previous analysis, the property on the plaintexts is satisfied with prob. $2^{-32\,809} \simeq 2^{-2^{15}}$ while the property on the ciphertexts is satisfied with prob. 2^{-14}, for an overall probability of $2^{-32\,809} \cdot 2^{-14} = 2^{-32\,823} \simeq 2^{-2^{15}}$.

In other words, the property on the ciphertexts is much weaker than the property on the plaintexts. This fact can be potentially used to generate a set of 2^{64} plaintexts/ciphertexts with the required properties with a data cost of $3 \cdot 2^{78}$. Indeed, the attacker can easily generate a set of 2^{64} plaintexts that satisfy the "Multiple-of-n" property as described before (e.g. he can generate such set using the fact that the 4-round AES decryption of $\mathcal{IS}$ – namely $R^4(\mathcal{IS})$ – has the required "Multiple-of-n" property). Then, he simply asks the oracle for the corresponding ciphertexts, which satisfy the "Multiple-of-2" property with prob. 2^{-14}. By repeating this process $3 \cdot 2^{14}$, the probability of success[22] is higher than 95%. The cost of such strategy (which includes both the generation of the texts and the check that the property is satisfied) is *at least* of 2^{78}.

[22] The probability of success is given by $1 - (1 - 2^{-14})^{3 \cdot 2^{14}} \geq 0.95$.

Even if this attack is faster than 2^{128}, its cost is still (much) bigger than 2^{64}, which is the cost to generate the required set of plaintexts/ciphertexts for the case of 10-round AES. Remember that the goal in an open-key distinguisher is indeed to be able to generate the requires set of plaintexts/ciphertexts with a *similar* (or even the same) cost for AES (or the studied cipher) and for the ideal cipher. In this case, it is very unlikely that any generic attack can get close to that: *even if we would allow unlimited time, the data complexity of a generic attack would still need to be higher than* 2^{64}. Indeed, working as in the 9-round case, a simple brute force attack requires at least[23] $2^{64} + 2^{11}$ plaintexts/ciphertexts in order to find a set of 2^{64} plaintexts with the required properties. For all these reasons and same as for the 9-round case (see our arguments from Section 6.4), we conjecture that the data/computational cost of an adversary to generate such set is (much) higher than 2^{64} computations.

6.5.2 Chosen-key distinguisher on 10-round AES – Exploit degrees of freedom in the weak-key

In the second approach, the idea is to extend it to 10-round by *adding one round in the middle* using the remaining degrees of freedom in the choice of the key.

In more details, referring to the 9-round distinguisher proposed in Section 6.3, if the subkey k^4 of the 4-th round belongs in K_{weak} (defined as in Eq. (4) and Section 4), it follows that

$$\textit{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus a \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^{2}(\cdot)} \mathcal{IS} \oplus b \xrightarrow{R^{3}(\cdot)} \textit{Multiple-of-}n$$

In other words, one exploits the fact that the subspace $\mathcal{IS}$ is mapped into a coset of it after 2-round encryption and 1-round decryption for any subkey in K_{weak}.

By simple computation, there is a key in K_{weak} for which the subspace $\mathcal{IS}$ is mapped into a coset of it after two rounds decryption. In more details, for the key $\hat{k} \in K_{\text{weak}}$ defined by

$$\hat{k} \equiv (A = 0\text{x}63 \oplus R[5], B = 0\text{x}63, C = 0\text{x}63, D = 0\text{x}63) \in K_{\text{weak}}$$

it follows that

$$\mathcal{IS} \oplus a \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^{2}(\cdot)} \mathcal{IS} \oplus b.$$

To see this, it is sufficient to compute one round of the key schedule

$$\begin{bmatrix} A \oplus 0\text{x}63 \oplus R[5] & 0 & 0 & 0 \\ B \oplus 0\text{x}63 & 0 & 0 & 0 \\ C \oplus 0\text{x}63 & 0 & 0 & 0 \\ D \oplus 0\text{x}63 & 0 & 0 & 0 \end{bmatrix} \xrightarrow{\text{1-round } \textit{Key Schedule}} K_{\text{weak}} \equiv \begin{bmatrix} A & A & A & A \\ B & B & B & B \\ C & C & C & C \\ D & D & D & D \end{bmatrix},$$

and to look for a key in K_{weak} that belongs to $\mathcal{IS}$ one round before. As a result, it follows that for the key $\hat{k} \equiv (A = 0\text{x}63 \oplus R[5], B = 0\text{x}63, C = 0\text{x}63, D = 0\text{x}63) \in K_{\text{weak}}$ it is possible to set up a distinguisher on 10 rounds[24] since

$$\textit{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus a \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^{2}(\cdot)} \mathcal{IS} \oplus b \xrightarrow{R^{3}(\cdot)} \textit{Multiple-of-}n$$

[23] Note that $\binom{2^{64}+2^{11}}{2^{64}} \geq 2^{32\,823}$.

[24] For completeness, we discuss the relevance of a distinguisher that can be constructed for a single key (which this does not mean – in general – that it holds for one key only). A single collision/near-collision/ or similar distinguishing property for a block-cipher based compression function or hash function would be also a property of the cipher that holds (depending on the mode) for a single key. Assume this is found with a non-generic approach. This simple example shows that, in principle, properties even for single keys can be interesting.

Using this observation, we can construct the distinguisher. Exactly as before, the chosen-key model asks the adversary to find a set of 2^{64} plaintexts/ciphertexts, i.e. $(p^i, c^i \equiv R^{10}(p^i))$ for $i = 0, \ldots, 2^{64} - 1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that

- for each $J, I \subseteq \{0, 1, 2, 3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ are a multiple of $128 \equiv 2^7$;
- for each $J, I \subseteq \{(0,0), (0,1), \ldots, (3,2), (3,3)\} \equiv \{(i,j)\}_{0 \leq i,j \leq 3}$, the number of different pairs of ciphertexts that belong to the same coset of $MC(\mathcal{X}_I)$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_J$ are a multiple of 2.

Similar to the 9-round case, due to our arguments from Section 6.4 we conjecture that the computational cost of an adversary to generate such set is (much) higher than 2^{64} computations.

6.6 Key schedule and chosen-key distinguisher: an open problem for future research

As showed in detail in Appendix G.1, similar chosen-key distinguishers can be set up for other key schedules present in the literature. As a result, an open problem is left for future research. As we have already re-called, several key schedules have been proposed in the literature in order to improve the security of AES against related-key attacks, while the security against open-key distinguisher always had less attention.

For this reason, for future research, one should look for *a key schedule for which (1st) the security of AES against the related key attack is improved, (2nd) for which it is not possible to set up a chosen-key distinguisher on full AES-128/256 and – if necessary – (3rd) it does not use the S-Box function* (if one requires a lightweight key-schedule composed only of linear operations).

References

[BK01] Eli Biham and Nathan Keller. *Cryptanalysis of Reduced Variants of Rijndael*. unpublished, http://csrc.nist.gov/archive/aes/round2/conf3/papers/35-ebiham.pdf. 2001.

[BK09] Alex Biryukov and Dmitry Khovratovich. "Related-Key Cryptanalysis of the Full AES-192 and AES-256." In: *Advances in Cryptology - ASIACRYPT 2009*. Vol. 5912. LNCS. 2009, pp. 1–18.

[BR14] Andrey Bogdanov and Vincent Rijmen. "Linear hulls with correlation zero and linear cryptanalysis of block ciphers." In: *Designs, Codes and Cryptography* 70.3 (2014), pp. 369–383.

[BS90] Eli Biham and Adi Shamir. "Differential Cryptanalysis of DES-like Cryptosystems." In: *Advances in Cryptology - CRYPTO 1990*. Vol. 537. LNCS. 1990, pp. 2–21.

[Bar+18] Achiya Bar-On, Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. "Improved Key Recovery Attacks on Reduced-Round AES with Practical Data and Memory Complexities." In: *Advances in Cryptology - CRYPTO 2018*. Vol. 10992. LNCS. 2018, pp. 185–212.

[Bei+17] Christof Beierle, Anne Canteaut, Gregor Leander, and Yann Rotella. "Proving Resistance Against Invariant Attacks: How to Choose the Round Constants." In: *Advances in Cryptology - CRYPTO 2017*. Vol. 10402. LNCS. 2017, pp. 647–678.

[Bey18] Tim Beyne. "Block Cipher Invariants as Eigenvectors of Correlation Matrices." In: *ASIACRYPT 2018, Part I*. LNCS. Springer, Heidelberg, Dec. 2018, pp. 3–31. DOI: 10.1007/978-3-030-03326-2_1.

[Bih+99] Eli Biham, Alex Biryukov, and Adi Shamir. "Cryptanalysis of Skipjack Reduced to 31 Rounds Using Impossible Differentials." In: *Advances in Cryptology - EUROCRYPT 1999*. Vol. 1592. LNCS. 1999, pp. 12–23.

[Bir+09] Alex Biryukov, Dmitry Khovratovich, and Ivica Nikolić. "Distinguisher and Related-Key Attack on the Full AES-256." In: *Advances in Cryptology - CRYPTO 2009*. Vol. 5677. LNCS. 2009, pp. 231–249.

[Bou+19] Christina Boura, Anne Canteaut, and Daniel Coggia. "A General Proof Framework for Recent AES Distinguishers." In: *IACR Transactions on Symmetric Cryptology* 2019.1 (2019), pp. 170–191. DOI: 10.13154/tosc.v2019.i1.170-191. URL: https://tosc.iacr.org/index.php/ToSC/article/view/7401.

[Bra+05] An Braeken, Christopher Wolf, and Bart Preneel. "Normality of Vectorial Functions." In: *Cryptography and Coding - 10th IMA International Conference 2005*. Ed. by Nigel P. Smart. Vol. 3796. LNCS. Springer, 2005, pp. 186–200.

[Can+04] Ran Canetti, Oded Goldreich, and Shai Halevi. "The Random Oracle Methodology, Revisited." In: *Journal ACM* 51.4 (2004), pp. 557–594.

[Cha+17] Colin Chaigneau, Thomas Fuhr, Henri Gilbert, Jérémy Jean, and Jean-René Reinhard. "Cryptanalysis of NORX v2.0." In: *IACR Transactions on Symmetric Cryptology* 2017.1 (2017). https://tosc.iacr.org/index.php/ToSC/article/view/589, pp. 156–174.

[Cid+05] C. Cid, S. Murphy, and M. J. B. Robshaw. "Small Scale Variants of the AES." In: *Fast Software Encryption - FSE 2005*. Vol. 3557. LNCS. 2005, pp. 145–162.

[DR02] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[DR06] Joan Daemen and Vincent Rijmen. "Understanding Two-Round Differentials in AES." In: *Security and Cryptography for Networks - SCN 2006*. Vol. 4116. LNCS. Springer, 2006, pp. 78–94.

[Der+12] Patrick Derbez, Pierre-Alain Fouque, and Jérémy Jean. "Faster Chosen-Key Distinguishers on Reduced-Round AES." In: *Progress in Cryptology - INDOCRYPT 2012*. Vol. 7668. LNCS. 2012, pp. 225–243.

[Der+18] Patrick Derbez, Pierre-Alain Fouque, Jérémy Jean, and Baptiste Lambin. "Variants of the AES Key Schedule for Better Truncated Differential Bounds." In: *SAC 2018*. Vol. 11349. LNCS. 2018, pp. 27–49.

[Flu+01] Scott Fluhrer, Itsik Mantin, and Adi Shamir. "Weaknesses in the Key Scheduling Algorithm of RC4." In: *Selected Areas in Cryptography - SAC 2001*. Vol. 2259. LNCS. 2001, pp. 1–24.

[Fou+13] Pierre-Alain Fouque, Jérémy Jean, and Thomas Peyrin. "Structural Evaluation of AES and Chosen-Key Distinguisher of 9-Round AES-128." In: *Advances in Cryptology - CRYPTO 2013*. Vol. 8042. LNCS. 2013, pp. 183–203.

[GP10] Henri Gilbert and Thomas Peyrin. "Super-Sbox Cryptanalysis: Improved Attacks for AES-Like Permutations." In: *Fast Software Encryption - FSE 2010*. Vol. 6147. LNCS. 2010, pp. 365–383.

[GR16] Lorenzo Grassi and Christian Rechberger. "Practical Low Data-Complexity Subspace-Trail Cryptanalysis of Round-Reduced PRINCE." In: *Progress in Cryptology - INDOCRYPT 2016*. Vol. 10095. LNCS. 2016, pp. 322–342.

[GR17] Lorenzo Grassi and Christian Rechberger. *New and Old Limits for AES Known-Key Distinguishers*. Cryptology ePrint Archive, Report 2017/255. 2017.

[Gil14] Henri Gilbert. "A Simplified Representation of AES." In: *ASIACRYPT 2014, Part I*. Ed. by Palash Sarkar and Tetsu Iwata. Vol. 8873. LNCS. Springer, Heidelberg, Dec. 2014, pp. 200–222. DOI: 10.1007/978-3-662-45611-8_11.

[Gra+16] Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. "Subspace Trail Cryptanalysis and its Applications to AES." In: *IACR Trans. Symm. Cryptol.* 2016.2 (2016). http://tosc.iacr.org/index.php/ToSC/article/view/571, pp. 192–225. ISSN: 2519-173X. DOI: 10.13154/tosc.v2016.i2.192-225.

[Gra+17] Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. "A New Structural-Differential Property of 5-Round AES." In: *EUROCRYPT 2017, Part II*. Ed. by Jean-Sébastien Coron and Jesper Buus Nielsen. Vol. 10211. LNCS. Springer, Heidelberg, May 2017, pp. 289–317. DOI: 10.1007/978-3-319-56614-6_10.

[Gra18] Lorenzo Grassi. "Mixture Differential Cryptanalysis: a New Approach to Distinguishers and Attacks on round-reduced AES." In: *IACR Trans. Symmetric Cryptol.* 2018.2 (2018), pp. 133–160. DOI: 10.13154/tosc.v2018.i2.133-160. URL: https://doi.org/10.13154/tosc.v2018.i2.133-160.

[Guo+13] Jian Guo, Ivica Nikolic, Thomas Peyrin, and Lei Wang. *Cryptanalysis of Zorro*. Cryptology ePrint Archive, Report 2013/713. http://eprint.iacr.org/2013/713. 2013.

[Guo+16] Jian Guo, Jérémy Jean, Ivica Nikolic, Kexin Qiao, Yu Sasaki, and Siang Sim. "Invariant Subspace Attack Against Midori64 and The Resistance Criteria for S-box Designs." In: *IACR Transactions on Symmetric Cryptology* 2016.1 (2016), pp. 33–56.

[Her+09] Miia Hermelin, Joo Yeon Cho, and Kaisa Nyberg. "Multidimensional Extension of Matsui's Algorithm 2." In: *Fast Software Encryption - FSE 2009s*. Vol. 5665. LNCS. 2009, pp. 209–227.

[Jea+14] Jérémy Jean, María Naya-Plasencia, and Thomas Peyrin. "Multiple Limited-Birthday Distinguishers and Applications." In: *Selected Areas in Cryptography - SAC 2013*. Vol. 8282. LNCS. 2014, pp. 533–550.

[Jea16] Jérémy Jean. "Cryptanalysis of Haraka." In: *IACR Trans. Symm. Cryptol.* 2016.1 (2016). http://tosc.iacr.org/index.php/ToSC/article/view/531, pp. 1–12. ISSN: 2519-173X. DOI: 10.13154/tosc.v2016.i1.1-12.

[KM07] Orhun Kara and Cevat Manap. "A New Class of Weak Keys for Blowfish." In: *Fast Software Encryption - FSE 2007*. Vol. 4593. LNCS. 2007, pp. 167–180.

[KR07] Lars R. Knudsen and Vincent Rijmen. "Known-Key Distinguishers for Some Block Ciphers." In: *Advances in Cryptology - ASIACRYPT 2007*. Vol. 4833. LNCS. 2007, pp. 315–324.

[Kho+17] Khoongming Khoo, Eugene Lee, Thomas Peyrin, and Siang Sim. "Human-readable Proof of the Related-Key Security of AES-128." In: *IACR Transactions on Symmetric Cryptology* 2017.2 (2017). https://tosc.iacr.org/index.php/ToSC/article/view/638, pp. 59–83.

[Knu95] Lars R. Knudsen. "Truncated and higher order differentials." In: *Fast Software Encryption - FSE 1994*. Vol. 1008. LNCS. 1995, pp. 196–211.

[Knu98] Lars R. Knudsen. *DEAL - A 128-bit Block Cipher*. Technical Report 151, Department of Informatics, University of Bergen, Norway. 1998.

[Kra+17] Thorsten Kranz, Gregor Leander, and Friedrich Wiemer. "Linear Cryptanalysis: Key Schedules and Tweakable Block Ciphers." In: *IACR Trans. Symm. Cryptol.* 2017.1 (2017), pp. 474–505. ISSN: 2519-173X. DOI: 10.13154/tosc.v2017.i1.474-505.

[LH94] Susan K. Langford and Martin E. Hellman. "Differential-Linear Cryptanalysis." In: *Advances in Cryptology - CRYPTO 1994*. Vol. 839. LNCS. 1994, pp. 17–25.

[LR17] Yunwen Liu and Vincent Rijmen. *New Observations on Invariant Subspace Attack*. Cryptology ePrint Archive, Report 2017/278. http://eprint.iacr.org/2017/278. 2017.

[Lam+15] Mario Lamberger, Florian Mendel, Martin Schläffer, Christian Rechberger, and Vincent Rijmen. "The Rebound Attack and Subspace Distinguishers: Application to Whirlpool." In: *Journal of Cryptology* 28.2 (Apr. 2015), pp. 257–296. DOI: 10.1007/s00145-013-9166-5.

[Le+04] Tri Van Le, Rüdiger Sparr, Ralph Wernsdorf, and Yvo Desmedt. "Complementation-Like and Cyclic Properties of AES Round Functions." In: *Advanced Encryption Standard - AES, 4th International Conference*. Vol. 3373. LNCS. 2004, pp. 128–141.

[Lea+11] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. "A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack." In: *Advances in Cryptology - CRYPTO 2011*. Vol. 6841. LNCS. 2011, pp. 206–221.

[Lea+15] Gregor Leander, Brice Minaud, and Sondre Rønjom. "A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro." In: *Advances in Cryptology - EUROCRYPT 2015*. Vol. 9056. LNCS. 2015, pp. 254–283.

[Lea+18] Gregor Leander, Cihangir Tezcan, and Friedrich Wiemer. "Searching for Subspace Trails and Truncated Differentials." In: *IACR Trans. Symm. Cryptol.* 2018.1 (2018), pp. 74–100. ISSN: 2519-173X. DOI: 10.13154/tosc.v2018.i1.74-100.

[MS87] Judy H. Moore and Gustavus J. Simmons. "Cycle Structure of the DES with Weak and Semi-Weak Keys." In: *Advances in Cryptology - CRYPTO 1986*. Vol. 263. LNCS. 1987, pp. 9–32.

[Mat94] Mitsuru Matsui. "Linear Cryptanalysis Method for DES Cipher." In: *Advances in Cryptology - EUROCRYPT 1993*. Vol. 765. LNCS. 1994, pp. 386–397.

[Nik11] Ivica Nikolić. "Tweaking AES." In: *Selected Areas in Cryptography - SAC 2010*. Vol. 6544. LNCS. 2011, pp. 198–210.

[Ohk09] Kenji Ohkuma. "Weak Keys of Reduced-Round PRESENT for Linear Cryptanalysis." In: *Selected Areas in Cryptography - SAC 2009*. Vol. 5867. LNCS. 2009, pp. 249–265.

[Røn+17] Sondre Rønjom, Navid Ghaedi Bardeh, and Tor Helleseth. "Yoyo Tricks with AES." In: *ASIACRYPT 2017, Part I*. LNCS. Springer, Heidelberg, Dec. 2017, pp. 217–243. DOI: 10.1007/978-3-319-70694-8_8.

[Røn16] Sondre Rønjom. *Invariant subspaces in Simpira*. Cryptology ePrint Archive, Report 2016/248. http://eprint.iacr.org/2016/248. 2016.

[Tod+16] Yosuke Todo, Gregor Leander, and Yu Sasaki. "Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64." In: *ASIACRYPT 2016, Part II*. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Vol. 10032. LNCS. Springer, Heidelberg, Dec. 2016, pp. 3–33. DOI: 10.1007/978-3-662-53890-6_1.

[Vau96] Serge Vaudenay. "On the weak keys of blowfish." In: *Fast Software Encryption - FSE 1996*. Vol. 1039. LNCS. 1996, pp. 27–32.

[WW16] Yanfeng Wang and Wenling Wu. "New Observations on Piccolo Block Cipher." In: *Topics in Cryptology - CT-RSA 2016*. Vol. 9610. LNCS. 2016, pp. 378–393.

[Wei+12] Lei Wei, Thomas Peyrin, Przemysław Sokołowski, San Ling, Josef Pieprzyk, and Huaxiong Wang. "On the (In)Security of IDEA in Various Hashing Modes." In: *Fast Software Encryption - FSE 2012*. Vol. 7549. LNCS. 2012, pp. 163–179.

A Generic subspace trail of length 1 for AES - Proof

Here we give a complete proof regarding the subspace trail of length 1 set up using the generic subspace $\mathcal{X}$ defined in Section 3.1.

First of all, we recall the definition of $\mathcal{X}$.

Definition 9. Let I a subset of $\{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$. Let the subspace $\mathcal{X}_I$ be defined as

$$\mathcal{X}_I = \langle \{e_{i,j}\}_{(i,j)\in I} \rangle \equiv \left\{ \bigoplus_{(i,j)\in I} \alpha_{i,j} \cdot e_{i,j} \,\middle|\, \forall \alpha_{i,j} \in \mathbb{F}_{2^8} \right\}.$$

In other words, $\mathcal{X}_I$ is the set of elements given by linear combinations of $\{e_{i,j}\}_{(i,j)\in I}$, where $e_{i,j} \in \mathbb{F}_{2^8}^{4\times 4}$ has a single 1 in row i and column j.

Theorem 6. *For each $I \subseteq \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$ and for each $a \in \mathcal{X}_I^{\perp}$, there exists one and only one $b \in \mathcal{Y}_I^{\perp}$ such that*

$$R(\mathcal{X}_I \oplus a) = \mathcal{Y}_I \oplus b$$

where $\mathcal{Y}_I = MC \circ SR(\mathcal{X}_I)$.

Observe that for each $I \subset \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$, there exists $J \subseteq \{(i,j)\}_{0\leq i,j\leq 3}$ such that $SR(\mathcal{X}_I) = \mathcal{X}_J$ (or equivalently $SR^{-1}(\mathcal{X}_I) = \mathcal{X}_J$). As a result, $\{\mathcal{X}_I, MC \circ SR(\mathcal{X}_I)\}$ is a subspace trail of length 1. Let $\mathcal{X}_I$ defined as in Definition 8.

Proof. To prove the Theorem, we simply compute $R(\mathcal{X}_I \oplus a)$. Since SubBytes is bijective and operates on each byte independently, its only effect is to change the coset. In other words, it simply changes the coset $\mathcal{X}_I \oplus a$ to $\mathcal{X}_I \oplus a'$, where $a'_{i,j} = SB(a_{i,j})$ for each $i,j = 0,\ldots,3$. ShiftRows simply moves the bytes of $\mathcal{X}_I \oplus a'$ into $SR(\mathcal{X}_I) \oplus b'$, where $b' = SR(a')$. Since MixColumns is a linear operation, it follows that $MC(SR(\mathcal{X}_I) \oplus b') = MC \circ SR(\mathcal{X}_I) \oplus MC(b') = MC \circ SR(\mathcal{X}_I) \oplus b''$. Key addition then changes the coset to $MC \circ SR(\mathcal{X}_I) \oplus b$. □

B Subkeys and key schedule of AES – Details

B.1 SubKeys of AES-256 – Details about Section 4.3

In order to prove the results proposed for AES-256 in Section 4.3, we list here the subkeys involved. Referring to Section 4.3, consider the set of subkeys defined by K_{weak}.

(1st) Consider the 2^{96} keys that satisfy

$$A^0 = A^1, \quad B^0 = B^1, \quad C^0 = C^1, \quad D^0 = D^1$$

that is

$$\left\{ \begin{bmatrix} A & A & A & A & E^0 & E^1 & E^0 & E^1 \\ B & B & B & B & F^0 & F^1 & F^0 & F^1 \\ C & C & C & C & G^0 & G^1 & G^0 & G^1 \\ D & D & D & D & H^0 & H^0 & H^0 & H^1 \end{bmatrix} \;\middle|\; \forall A,B,C,D,\ldots,H^0,H^1 \in \mathbb{F}_{2^8} \right\}$$

The next subkey is given by

$$\begin{bmatrix} A \oplus SB(F^1) \oplus R[1] & SB(F^1) \oplus R[1] & A \oplus SB(F^1) \oplus R[1] & SB(F^1) \oplus R[1] \\ B \oplus SB(G^1) & SB(G^1) & B \oplus SB(G^1) & SB(G^1) \\ C \oplus SB(H^1) & SB(H^1) & C \oplus SB(H^1) & SB(H^1) \\ D \oplus SB(E^1) & SB(E^1) & D \oplus SB(E^1) & SB(E^1) \end{bmatrix}$$

(2nd) Consider the 2^{64} keys that satisfy

$$A^0 = A^1, \quad B^0 = B^1, \quad C^0 = C^1, \quad D^0 = D^1, \quad \dots, \quad H^0 = H^1$$

The next subkey is given by

$$\begin{bmatrix} SB(\hat{F} \oplus R[1]) \oplus E & SB(\hat{F} \oplus R[1]) & SB(\hat{F} \oplus R[1]) \oplus E & SB(\hat{F} \oplus R[1]) \\ SB(\hat{G}) \oplus F & SB(\hat{G}) & SB(\hat{G}) \oplus F & SB(\hat{G}) \\ SB(\hat{H}) \oplus G & SB(\hat{H}) & SB(\hat{H}) \oplus G & SB(\hat{H}) \\ SB(\hat{E}) \oplus H & SB(\hat{E}) & SB(\hat{E}) \oplus H & SB(\hat{E}) \end{bmatrix}$$

where

$$\hat{E} := SB(E), \quad \hat{F} := SB(F), \quad \hat{G} := SB(G), \quad \hat{H} := SB(H).$$

(3rd) Consider the 2^{32} keys that satisfy

$$A^0 = A^1 = B^0 = \dots = D^0 = D^1 = 0, \quad E^0 = E^1, \quad F^0 = F^1, \quad \dots \quad H^0 = H^1.$$

Then, the next subkeys satisfy

$$\begin{bmatrix} SB(\hat{\hat{G}}) \oplus \hat{F} \oplus R'[2] & SB(\hat{\hat{G}}) \oplus R[2] & SB(\hat{\hat{G}}) \oplus \hat{F} \oplus R'[2] & SB(\hat{\hat{G}}) \oplus R[2] \\ SB(\hat{\hat{H}}) \oplus \hat{G} & SB(\hat{\hat{H}}) & SB(\hat{\hat{H}}) \oplus \hat{G} & SB(\hat{\hat{H}}) \\ SB(\hat{\hat{E}}) \oplus \hat{H} & SB(\hat{\hat{E}}) & SB(\hat{\hat{E}}) \oplus \hat{H} & SB(\hat{\hat{E}}) \\ SB(\hat{\hat{F}}) \oplus \hat{E} & SB(\hat{\hat{F}}) & SB(\hat{\hat{F}}) \oplus \hat{E} & SB(\hat{\hat{F}}) \end{bmatrix}$$

where

$$\hat{\hat{E}} := SB(SB(E)), \qquad \hat{\hat{F}} := SB(\hat{F} \oplus R[1]),$$
$$\hat{\hat{G}} := SB(SB(G)), \qquad \hat{\hat{H}} := SB(SB(H))$$

and $R'[2] := R[1] \oplus R[2]$.

B.2 Key-schedule based on Permutations

In Section 4.2, we discuss the security of the key-schedule based on permutation of the bytes proposed in [Kho+17]. Similar key-schedules have been recently proposed at SAC 2018 [Der+18]. Here we show that also for one of them it is possible to set up an "infinitely-long" weak invariant subspace (again, a simple way to avoid this is to add random round-constants which break the symmetry).

The key-schedule proposed in [Der+18, Sect. 4.2] is defined by the following byte-permutation:

$$\big(8 \quad 1 \quad 7 \quad 15 \quad 10 \quad 4 \quad 2 \quad 3 \quad 6 \quad 9 \quad 11 \quad 0 \quad 5 \quad 12 \quad 14 \quad 13\big).$$

This key-schedule permutation guarantees that at least 20 S-Boxes are active in every possible related-key truncated differential.

It's simple to verify that the subspace

$$\mathcal{IS}^{(1)} := \left\{ \begin{bmatrix} a & a & a & a \\ b & a & b & a \\ a & a & a & a \\ a & a & a & a \end{bmatrix} \middle| \forall a, b \in \mathbb{F}_{2^8} \right\}$$

(where $\mathcal{IS}^{(1)} \subseteq \mathcal{IS}$) is invariant w.r.t. the previous byte permutation. Since $\mathcal{IS}^{(1)} \subseteq \mathcal{IS}$ where $\mathcal{IS}$ is defined as in Eq. (2), this allows to set up an "infinitely-long" weak invariant subspace.

Remark Note that a generic subspace

$$\mathcal{IS}^{(x)} \coloneqq \left\{ \begin{bmatrix} a & a & a & a \\ a & a & a & a \\ a & a & a & a \\ a & a & a & a \end{bmatrix} \middle| \forall a \in \mathbb{F}_{2^8} \right\}$$

is always invariant under a byte-permutation. The previous results are of some importance/relevance since the subspaces $\mathcal{IS}$ and $\mathcal{IS}^{(1)}$ are not trivial (they have dimension 8 and 2, w.r.t. dimension 1 of $\mathcal{IS}^{(x)}$).

B.3 AES-like key schedule

As discussed in Section 4.4, a possible variant of the AES key schedule has been proposed at SAC 2010 by Nikolić [Nik11]. This variant is obtained by introducing a small change in the current AES key schedule, which allows to improve the security against related-key attacks. As we are going to show, even if this change improves the security against related-key attack, it is possible to get the same results just presented for the original AES key schedule also in this case.

For simplicity, we focus on AES-128 for which the new key schedule is defined as $W[i][j] = K[i][j]$ for $j < 4$ and as

$$W[i][j] = \begin{cases} W[i][j-4] \oplus SB(W[i-1][j-1]) \oplus R[i][j/4] & \text{if } j \bmod 4 = 0 \\ W[i-1][j-1] \oplus W[i][j-4] & \text{otherwise} \end{cases}$$

where $i = 0, 1, 2, 3$, $j = 4, \ldots, 43$ and $R[\cdot]$ is an array of predetermined constants. Also in this case it is possible to find a set (*not* a subspace) of weak-keys K'_{weak} for which it is possible to construct a weak invariant subspace trail of length 2 as before, where K'_{weak} is defined as

$$K'_{\text{weak}} \coloneqq \begin{bmatrix} SB(B) \oplus SB(D) \oplus B \oplus R[1] & A & SB(B) \oplus SB(D) \oplus B \oplus R[1] & A \\ SB(A) \oplus SB(C) \oplus C & B & SB(A) \oplus SB(C) \oplus C & B \\ SB(B) \oplus SB(D) \oplus D \oplus R[1] & C & SB(B) \oplus SB(D) \oplus D \oplus R[1] & C \\ SB(A) \oplus SB(C) \oplus A & D & SB(A) \oplus SB(C) \oplus A & D \end{bmatrix}$$

for all $A, B, C, D \in \mathbb{F}_{2^8}$. Similar results hold also for the cases AES-192 and AES-256.

C Secret-Key distinguisher for round-reduced AES in the case of weak-keys – Details

Here we give more details about the secret-key distinguishers for round-reduced AES in the case of weak-keys.

C.1 Weak-key impossible differential over 4-round AES-128

Since $\mathcal{M}_I \cap \mathcal{D}_J = \{0\}$ for $|I| + |J| \leq 4$, it follows that for an AES permutation and for a weak-key

$$\Pr\left[R^4(x) \oplus R^4(y) \in \mathcal{M}_I \;\middle|\; x, y \in \mathcal{IS},\, k \in K_{\text{weak}}\right] = 0 \qquad \text{if } I \neq \{0,2\}, \{1,3\},$$

while for a random permutation Π the probability is given by Eq. (5).

C.2 Zero-sum property

Another distinguisher is the integral one and exploits the zero-sum property. Given a set of 2^8 plaintexts with one active byte[25], the XOR-sum of the corresponding ciphertexts after 3-round is equal to zero, independently of the secret key, of the details of the S-Box and of the MixColumns matrix. Similarly, given a set of 2^{32} plaintexts with one active diagonal, then the XOR-sum of the corresponding ciphertexts after 4-round is equal to zero, that is

$$\bigoplus_{p \in \mathcal{D}_I \oplus a} R^4(p) = 0.$$

Such result holds on 4-round AES-128 also in the case of a weak-key. In particular, under a weak-key and given 2^{64} plaintexts in the subspace $\mathcal{IS}$, it follows that the XOR-sum of corresponding ciphertexts after 4-round is equal to zero, that is

$$\bigoplus_{p \in \mathcal{IS}} R^4(p) = 0. \tag{9}$$

Here we present a theoretical proof.

First of all, we recall the definition of active/constant/balance bytes. Let Γ be a collection of state vectors $X = (x_0, \ldots, x_{2^n-1})$ where $x_i \in \mathbb{F}_{2^m}^{4\times 4}$. Each byte of x_i can be

- Active (A): if all each possible value in $\mathbb{F}_{2^m}$ is assumed the same number of times;
- Balance (B): if the sum of all bytes can predicted (e. g. equal to zero);
- Constant (C): if the values are equal.

Remember that a subspace $\mathcal{IS}$ is mapped into a coset of it after two rounds in the presence of a weak-key. This means that it is sufficiently to prove that

$$\bigoplus_{p \in \mathcal{IS} \oplus \hat{k}} R^2(p) = 0 \qquad \text{for } \hat{k} \in \mathcal{IS}^{\perp}.$$

Observe that each byte of $\mathcal{IS}$ is active. Indeed

$$R(\mathcal{IS} \oplus \hat{k}) = MC \times \begin{bmatrix} \mathrm{SB}(a) & \mathrm{SB}(b) & \mathrm{SB}(a \oplus \hat{k}_{0,2}) & \mathrm{SB}(b \oplus \hat{k}_{0,3}) \\ \mathrm{SB}(c) & \mathrm{SB}(d) & \mathrm{SB}(c \oplus \hat{k}_{1,2}) & \mathrm{SB}(d \oplus \hat{k}_{1,3}) \\ \mathrm{SB}(e) & \mathrm{SB}(f) & \mathrm{SB}(e \oplus \hat{k}_{2,2}) & \mathrm{SB}(f \oplus \hat{k}_{2,3}) \\ \mathrm{SB}(g) & \mathrm{SB}(h) & \mathrm{SB}(g \oplus \hat{k}_{3,2}) & \mathrm{SB}(h \oplus \hat{k}_{3,3}) \end{bmatrix}$$

for each $a, b, \ldots, h \in \mathbb{F}_{2^8}$. Since each column takes a particular value $\mathbb{F}_{2^8}^4$ exactly four times and since the MixColumns matrix is bijective, each byte is active after the MixColumns operation.

What happens after the second round? It is simple to prove that the balance property holds. Indeed, since the S-Box is bijective, each active byte remains active after the S-Box and the ShiftRows operation. Finally, since an active byte satisfies also the balance property (i. e. sum equal to zero), and since the MixColumns operation is linear, the zero-sum property survives after one-round. The result is proved.

[25] A byte is active if every value in $\mathbb{F}_{2^8}$ appears the same number of times, while it is constant if it is fixed to a constant for all texts.

D Proof of Proposition 3 – the pairs are *not* independent!

In order to estimate the number of collisions of Prop. Proposition 3, the probabilistic distribution of number of collisions that the previous computation, we assume that all the pairs are independent. However, this is not the case. However, here we show that the previous approximation is still a good approximation. To do this, we focus on the subspace $\mathcal{M}_J$, but analogous results hold for the subspaces $\mathcal{X}_I$ and $\mathcal{D}_I$. Moreover, we assume for simplicity $|J| = 3$ (analogous for the other cases).

Indeed, consider three texts, that is t^1, t^2 and t^3, and the corresponding three couples, that is $(t^1, t^2), (t^1, t^3)$ and (t^2, t^3). Three possible events can happen:

- if $t^1 \oplus t^2 \in \mathcal{M}_J$ and $t^1 \oplus t^3 \in \mathcal{M}_J$, then $t^2 \oplus t^3 \in \mathcal{M}_J$ with probability 1 (since $\mathcal{M}_J$ is a subspace);
- if $t^1 \oplus t^2 \in \mathcal{M}_J$ and $t^1 \oplus t^3 \notin \mathcal{M}_J$ (or vice-versa), then $t^2 \oplus t^3 \notin \mathcal{M}_J$ with probability 1 (since $\mathcal{M}_J$ is a subspace);
- if $t^1 \oplus t^2 \notin \mathcal{M}_J$ and $t^1 \oplus t^3 \notin \mathcal{M}_J$, then both the events $t^2 \oplus t^3 \in \mathcal{M}_J$ and $t^2 \oplus t^3 \notin \mathcal{M}_J$ are possible; in particular, $t^2 \oplus t^3 \in \mathcal{M}_J$ with *approximately* prob. $2^{-32\cdot(4-|J|)}$.

On the other hand, what is the probability that a pair of texts (p, q) satisfy $p \oplus q \in \mathcal{M}_J$?

To answer this question, first of all, it is important to focus on the previous last event and to theoretically compute a better approximation of this probability. For our goal, we focus on the case $|J| = 3$ with J fixed. We are going to show that the last probability is well approximated by $2^{-32} \cdot (1 - 2^{-32})^{-1}$. Since $t^1 \oplus t^2 \notin \mathcal{M}_J$, it follows that the difference on the J-th anti-diagonal is different from $(0, 0, 0, 0)$, i. e. they can take only one of $2^{32} - 1$ possible values different from $(0, 0, 0, 0)$. Similar consideration holds for $t^1 \oplus t^3 \notin \mathcal{M}_J$. Since $t^2 \oplus t^3 = (t^1 \oplus t^2) \oplus (t^1 \oplus t^3)$, it follows that the difference of the J-th anti-diagonal of $t^2 \oplus t^3$ is equal to zero if the difference of the J-th anti-diagonal of $t^1 \oplus t^2$ is equal to the difference of the J-th anti-diagonal of $t^1 \oplus t^3$. Since this happens with probability $(2^{32} - 1)^{-1}$, it follows that the probability that $t^1 \oplus t^3 \in \mathcal{M}_J$ is

$$(2^{32} - 1)^{-1} = 2^{-32} \cdot (1 - 2^{-32})^{-1} \approx 2^{-32} + 2^{-64} - 2^{-96} + \dots$$

To have more confidence about this fact, note that:

- $t^1 \oplus t^2 \in \mathcal{M}_J$, $t^1 \oplus t^3 \in \mathcal{M}_J$ and $t^2 \oplus t^3 \in \mathcal{M}_J$ occurs with probability $(2^{-32})^2$;
- $t^1 \oplus t^2 \in \mathcal{M}_J$, $t^1 \oplus t^3 \notin \mathcal{M}_J$ and $t^2 \oplus t^3 \notin \mathcal{M}_J$ occurs with probability $2^{-32} \cdot (1 - 2^{-32})$ (similar for the other 3 cases);
- $t^1 \oplus t^2 \notin \mathcal{M}_J$, $t^1 \oplus t^3 \notin \mathcal{M}_J$ and $t^2 \oplus t^3 \notin \mathcal{M}_J$ occurs with probability $(1 - 2^{-32})^2 \cdot (1 - 2^{-32} \cdot (1 - 2^{-32})^{-1})$.

All the other cases have probability 0 (since $\mathcal{M}_J$ is a subspace). By simple computation, the probability of all the possible events is equal to

$$(2^{-32})^2 + 3 \cdot 2^{-32} \cdot (1 - 2^{-32}) + (1 - 2^{-32})^2 \cdot (1 - 2^{-32} \cdot (1 - 2^{-32})^{-1}) = 1,$$

as expected. In other words, if one uses the probability $(1 - 2^{-32})^3$ for the last case, it follows that the probability of all the possible events is equal to $1 - 2^{-96}$, which is obviously wrong.

Thus, *what is the probability that* $t^2 \oplus t^3 \in \mathcal{M}_J$*?* Given the events $A_1, \dots, A_n$ in a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, we recall that

$$Prob\Big(\bigcup_{i=1}^{n} A_i\Big) = \sum_{k=1}^{n}\Big((-1)^{k-1} \sum_{J\subset\{1,\dots,n\},\,|J|=k} Prob\big(\bigcap_{j\in J} A_j\big)\Big),$$

where the last sum runs over all subsets J of the indexes $1, \ldots, n$ which contain exactly k elements. Thus:

$$\Pr\left[t^2 \oplus t^3 \in \mathcal{M}_J\right] = \underbrace{2^{-32} \cdot 2^{-32} \cdot 1}_{\text{1st Case}} + \underbrace{2 \cdot 2^{-32} \cdot (1 - 2^{-32}) \cdot 0}_{\text{2nd Case}} + \\ + \underbrace{(1 - 2^{-32})^2 \cdot 2^{-32} \cdot (1 - 2^{-32})^{-1}}_{\text{3rd Case}} = 2^{-32}.$$

As a result, even if the pairs are not independent, the probability that two texts s and t belong in the same coset of $\mathcal{M}_J$ for $|J| = 3$ is exactly 2^{-32} (analogous for the other cases).

E Chosen-key distinguisher and "high" number of collisions

In Section 6.3, we propose a chosen-key distinguisher for 9-round AES-128.

Referring to the scenario described in Section 6, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^9(p^i))$ for $i = 0, \ldots, 2^{64} - 1$, such that the following property is satisfied

- for each $J, I \subseteq \{0, 1, 2, 3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ are a multiple of $128 = 2^7$;
- for each $J, I \subset \{(0,0), (0,1), \ldots, (3,2), (3,3)\} \equiv \{(i,j)\}_{0 \le i,j \le 3}$, the number of different pairs of ciphertexts that belong to the same coset of $MC(\mathcal{X}_I)$ and the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_J$ are a multiple of 2, where $\mathcal{X}$ is defined as in Definition 8.

Another possible property that can be taken in account is the following:

- for each $J \subseteq \{0, 1, 2, 3\}$ with $|J| = 1$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ is higher than 2^{42}; similarly, for each $I \subseteq \{0, 1, 2, 3\}$ with $|I| = 1$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ is higher than 2^{42}.

Here we show why such property is useless in order to set up the distinguisher.

Chernoff's and Chebyshev's Inequalities For the follow-up, let us recall the Chernoff's Inequality. The generic Chernoff bound for a random variable X is obtained by applying Markov's inequality to e^{tX}. For every $t > 0$:

$$\Pr\left[X \ge a\right] \le \frac{\mathbb{E}\left[e^{t \cdot X}\right]}{e^{t \cdot a}}.$$

In the case of a binomial distribution $X \sim \mathcal{B}(n, p)$ (where $\mu = n \cdot p$), it follows that:

- for any $0 \le t \le 1$:

$$\Pr\left[(X \ge (1 + t) \cdot \mu\right] \leqslant exp(-\mu \cdot t^2/3);$$
$$\Pr\left[(X \le (1 - t) \cdot \mu\right] \leqslant exp(-\mu \cdot t^2/3);$$

- for any $t \ge 1$:

$$\Pr\left[X \ge (1 + t) \cdot \mu\right] \leqslant exp(-\mu \cdot t/3).$$

Secondly, let's recall the Chebyshev's inequality. Let X be a random variable with finite expected value μ and finite non-zero variance σ^2. Then for any real number $k > 0$:

$$\Pr\left[|X - \mu| \geq k\sigma\right] \leq \frac{1}{k^2}.$$

The distinguishing algorithm Assume to use the strategy proposed in Section 6.3 to construct the required set of texts. What is the probability that the property regarding the number of collisions higher than 2^{42} holds?

As we have just seen, given two texts in a coset of $\mathcal{IS}$, they belong to the same coset of $\mathcal{M}_I$ after 3-round for $|I| = 1$ with probability 2^{-79} – see Eq. (7). Note that given 2^{64} plaintexts in a coset of $\mathcal{IS}$ and the corresponding ciphertexts, the distribution probability of the number of pair of ciphertexts that belong to the same coset of $\mathcal{M}_I$ is simply described by a *binomial distribution*. By definition, a binomial distribution with parameters n and p is the discrete probability distribution of the number of successes in a sequence of n independent yes/no experiments, each of which yields success with probability p. In our case, given n pairs of texts, each of them satisfies or not the above property/requirement with a certain probability. Thus, this model can be described using a binomial distribution. We remember that for a random variable X that follows the binomial distribution, that is $X \sim \mathcal{B}(n, p)$, the mean μ and the variance σ^2 are respectively given by $\mu = n \cdot p$ and $\sigma^2 = n \cdot p \cdot (1 - p)$.

In our case, since $n = \binom{2^{64}}{2} \simeq 2^{127}$ and $p \simeq 2^{-79}$, it follows that $\mu \simeq 2^{48}$ and $\sigma^2 \simeq 2^{48}$. Using Chebyshev's inequality, it follows that the second property is verified with probability approximately equal to 1 ($\geq 1 - 2^{-47}$). Indeed, note that

$$\Pr\left[X \geq 2^{42}\right] > \Pr\left[|X - 2^{48}| \leq 2^{48} - 2^{42}\right] \geq 1 - \frac{1}{(2^{24} - 2^{18})^2} \approx 1 - 2^{-47.95}.$$

Thus, this property is achieved without any additional cost.

Adversary. Consider now the strategy that an adversary - who does not know the key - can use to fulfill this second property. In order to estimate the probability that a random set of texts satisfies the second property (regarding the number of collisions higher than 2^{42}), we use again the Chebyshev's inequality. To do this, we first approximate the probabilistic distribution of the number of collisions with a binomial distribution with parameter $n \simeq 2^{127}$ and $p = 2^{-96}$. Since the mean value is equal to $\mu = 2^{31}$ and the variance is equal to $\sigma^2 \simeq 2^{31}$, and using Chernoff's inequality, it follows that the second property is satisfied with probability

$$\Pr\left[X \geq 2^{42} \equiv (1 + (2^{11} - 1)) \cdot 2^{31}\right] \leq exp\left(-\frac{(2^{11} - 1) \cdot 2^{31}}{3}\right) \approx 2^{-2^{41.86}},$$

that is, much smaller than for the distinguishing algorithm.

On the other hand, it is quite simple to find a strategy in order to satisfy this extra-property without affecting the total cost. The goal is to find a set of 2^{64} texts such that the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ for $|I| = 1$ is higher than 2^{42} (analogous for the ciphertexts). For each I with $|I| = 1$, the idea is simply to consider 2^{23} plaintexts in the same coset of $\mathcal{D}_I \oplus a$ for a random a. It is simply to observe that the number of collisions in the same coset of $\mathcal{D}_I$ is at least $\binom{2^{23}}{2} \simeq 2^{45}$, that is the property is fulfilled. Similarly, for each J with $|J| = 1$, the idea is simply to consider 2^{23} ciphertexts in the same coset of $\mathcal{M}_J \oplus b$ for a random b. Using this strategy, $4 \cdot 2^{23} = 2^{25}$ plaintexts and $4 \cdot 2^{23} = 2^{25}$ ciphertexts are chosen, for a total of 2^{26} texts over 2^{64}, that is $2^{64} - 2^{26} \simeq 2^{64}$ are still "free".

It follows that this extra-property can be achieved also in this case with no additional cost. In other words, it can be considered useless in order to set up the known-key distinguisher.

F On the difficulty to set up multiple-of-n open-key distinguishers that do *not* rely on weak-keys

In order to better understand the role of the invariant subspace, and hence the dependence on weak-keys, in the previous construction, we briefly discuss the following problem: is it possible to set up a similar distinguisher using the multiple-of-8 property proposed in [Gra+17] which holds for any key? We conjecture that this is hard.

Given a coset of a diagonal space $\mathcal{D}_I$, the multiple-of-8 property holds (1) after 5-round encryption and (2) after 3-round decryption. It follows that given a coset of $\mathcal{C}_I$ in the middle, then

$$\forall k: \qquad \textit{Multiple-of-8} \xleftarrow{R^{-4}(\cdot)} \mathcal{C}_I \oplus a \xrightarrow{R^{4}(\cdot)} \textit{Multiple-of-8},$$

it is possible to set up a distinguisher on 8 rounds.

To extend this distinguisher to more rounds, a possibility can be to use a coset of $\mathcal{D}_I \oplus \mathcal{M}_J$ in the middle. Here we show why this solution does not work. First of all, observe that

$$\mathcal{D}_I \oplus \mathcal{M}_J \oplus a \equiv \bigcup_{b \in \mathcal{D}_I \oplus a} \mathcal{M}_J \oplus b \equiv \bigcup_{b \in \mathcal{M}_J \oplus a} \mathcal{D}_I \oplus b$$

Thus, consider 5-round encryption (similar for the decryption direction). The number of collisions between the pairs of ciphertexts whose corresponding plaintexts are in the same coset of $\mathcal{D}_I$ is a multiple of 8 with prob. 1. However, it is not possible to claim anything about the the pairs of ciphertexts whose corresponding plaintexts are in the same coset of $\mathcal{M}_J$, or for which one plaintext is in $\mathcal{D}_I \oplus a'$ and the other in $\mathcal{M}_J \oplus b'$. As a result, one looses any multiple-of-n property. A similar argumentation works also in the decryption direction.

As we have just seen, the invariant subspace allows to solve this problem in the case of weak-keys. The problem to set up a known-key distinguisher (for which the key does not satisfy any particular property) that exploits the multiple-of-n property for more than 8-round AES is still open.

G Chosen-key distinguishers for 10-round AES-128, 11-round AES-192 and 14-round AES-256

In Sections 6.3 and 6.5, we have proposed two ways to set up a chosen-key distinguisher for full AES-128. Here we propose the details in the case in which AES is instantiated by the key-schedule defined in [Nik11]. Moreover, using the same strategies, here we present similar results for 11-round AES-192 and 14-round AES-256. Since the strategies used to set up these distinguishers is similar to the ones proposed for AES-128, we refer to Sections 6.3 and 6.5 for all the details and we highlight here the main differences.

G.1 Chosen-key distinguisher on AES-128 instantiated by other key-schedule

In Section 4, we consider key schedules different than the original AES one. One may ask what happens to the chosen-key distinguisher just presented in these cases. Since this is obvious for the cases of identical subkeys (or weak round constants) and for the key schedule proposed in [Kho+17] (due to the possibility to set up "infinitely-long" weak invariant subspace trail), we limit ourselves to discuss the key schedule proposed by I. Nikolić in [Nik11]. Here we briefly show that an analogous 10-round chosen-key distinguisher can be set up for AES-128.

As before, the idea is to find a weak-key such that the invariant subspace $\mathcal{IS}$ is mapped into a coset of it after two rounds decryption and two rounds encryption. By simple computation, there exists a key $\hat{k}'$ in K'_{weak} – defined as in Section 4 – with such property, which is defined by

$$\hat{k}' \equiv (A = C = 0\text{x}63 \oplus R[4],\ B = D = SB(R[5]) \oplus R[4]) \in K'_{\text{weak}}.$$

G.2 Chosen-key distinguisher for 11-round AES-192

G.2.1 9-round AES-192 distinguisher

In order to set up the 9-round distinguisher of AES-192, one exploits the fact that

$$\forall k^4 \in K_{\text{weak}}: \qquad \mathcal{IS} \oplus a \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus b$$

for each key in K_{weak} defined in Section 4.3 where the round constant $R[1]$ that defines K_{weak} must be replaced with $R[4]$.

G.2.2 10-round AES-192 distinguisher – "Weaker" property

As for AES-128, the simplest way to extend the previous distinguisher to 10-round is to exploit a weaker property on (e.g.) the ciphertexts. As a result, *while the property on the plaintexts is unchanged*, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^{10}(p^i))$ for $i = 0, \ldots, 2^{64} - 1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that *for each* $J \subseteq \{0, 1, 2, 3\}$*, the number of different pairs of ciphertexts that belong to the same coset of* $\mathcal{M}_I$ *is a multiple of* 2.

G.2.3 10-round AES-192 distinguisher – Freedom of the key

In order to set up the distinguisher on 10 round, we need a weak invariant subspace trail on 4-round. By simple computation, it is sufficient to choose the subkey[26]

$$\hat{k} \equiv (A = 0, B = 0, C = 0, D = 0, E = 0, F = 0, G = 0, H = 0) \in K_{\text{weak}}$$

(where $R[1]$ that defines K_{weak} must be replaced with $R[5]$)) for which

$$\mathcal{IS} \oplus a \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus b.$$

Due to the results of Section 5.2, the multiple-of-128 property (on $\mathcal{D}_J$) and the multiple-of-2 property (on $\mathcal{X}_I$) hold with probability 1 on the plaintexts while the multiple-of-2 property (on $\mathcal{M}_J$) holds on the ciphertexts

$$\textit{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^2(\cdot)} \mathcal{IS} \oplus \tilde{k} \xrightarrow{R^4(\cdot)} \textit{Multiple-of-}2$$

G.2.4 11-round AES-192 distinguisher

Finally, it is possible to combine the previous two distinguishers on 10-round AES-192 in order to set up a distinguisher on 11-round AES-192. In this case, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^{11}(p^i))$ for $i = 0, \ldots, 2^{64} - 1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that the following "*simultaneous multiple-of-n*" property is satisfied:

[26] For completeness, another possible key can be used. In particular, given the key $\hat{k} \in K_{\text{weak}}$ defined by $\hat{k} \equiv (A = 0\text{x}63 \oplus R[5], B = 0\text{x}63, C = 0\text{x}63, D = 0\text{x}63, E = 0, F = 0, G = 0, H = 0)$ (where $R[1]$ that defines K_{weak} must be replaced with $R[4]$), then for $\mathcal{IS} \oplus a \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^3(\cdot)} \mathcal{IS} \oplus b$. We highlight that there is no key that allows to extend both 1-round forward and 1-round backward.

Plaintext: on the plaintexts, we re-use the previous properties:

(1st) for each $J \subseteq \{0,1,2,3\}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_J$ is a multiple of $128 = 2^7$;

(2nd) for each $I \subset \{(0,0),(0,1),\ldots,(3,2),(3,3)\} \equiv \{(i,j)\}_{0\leq i,j\leq 3}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_I$ are a multiple of 2;

Ciphertext: *for each $J \subseteq \{0,1,2,3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ is a multiple of* 2.

In order to set up the chosen-key distinguisher, the idea is to exploit the fact that for the key $\hat{k} \in K_{\text{weak}}$ defined by $\hat{k} \equiv (A=0, B=0, C=0, D=0, E=0, F=0, G=0, H=0) \in K_{\text{weak}}$ (where $R[1]$ that defines K_{weak} must be replaced with $R[5]$)), it holds that

$$\mathcal{IS} \oplus a \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^{2}(\cdot)} \mathcal{IS} \oplus b.$$

Due to the results of Section 5.2, the multiple-of-128 property (on $\mathcal{D}_J$) and the multiple-of-2 property (on $\mathcal{X}_I$) hold with probability 1 on the plaintexts while the multiple-of-2 property (on $\mathcal{M}_J$) holds on the ciphertexts

$$\textit{Multiple-of-}n \xleftarrow{R^{-3}(\cdot)} \mathcal{IS} \oplus \hat{k} \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^{2}(\cdot)} \mathcal{IS} \oplus \tilde{k} \xrightarrow{R^{4}(\cdot)} \textit{Multiple-of-}2$$

as required.

What about an adversary facing a family of random and independent ideal ciphers? As we showed in detail in Section 6.5, the required properties on the plaintexts and on the ciphertexts hold with prob. $2^{-32\,823} \simeq 2^{-2^{15}}$ for a random set of texts. Due to our argumentations from Section 6.4, we conjecture that the computational cost of an adversary to generate such set is (much) higher than 2^{64} computations.

G.3 Chosen-key distinguisher for (*full*) AES-256

G.3.1 Chosen-Key distinguisher for 12-round AES-256

Similarly, to set up the 12-round distinguisher of AES-256, one exploits the fact that

$$\forall k \in K_{\text{weak}}: \qquad \mathcal{IS} \oplus a \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^{5}(\cdot)} \mathcal{IS} \oplus b$$

for each key in K_{weak} defined in Section 4.3 where

$$A^0 = A^1 = B^0 = \ldots = D^0 = D^1 = 0, \quad E^0 = E^1, F^0 = F^1, \ldots, H^0 = H^1.$$

G.3.2 13-round AES-256 distinguisher – "Weaker" property

As for AES-128, the simplest way to extend the previous distinguisher to 13-round is to exploit a weaker property on (e.g.) the ciphertexts. As a result, *while the property on the plaintexts is unchanged*, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^{13}(p^i))$ for $i = 0, \ldots, 2^{64}-1$ – where all the plaintexts/ciphertexts are generated by *the same key* – such that *for each $J \subseteq \{0,1,2,3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_I$ is a multiple of* 2.

G.3.3 13-round AES-256 distinguisher – Freedom of the key

Another possibility to extend the previous distinguisher to 13-round is to exploit the freedom in the key. In more details, in order to set up the distinguisher on 13 round and using the same argumentation proposed for AES-128, *among the previous weak-keys* the idea is to choose the sub-key defined by

$$\hat{k} \equiv (E^0 = E^1 = 0\text{x}63 \oplus R[5], F^0 = F^1 = 0\text{x}63, \ldots, H^0 = 0, H^1 = 0\text{x}63) \in K_{\text{weak}}$$

for which

$$\mathcal{IS} \oplus a \xleftarrow{R^{-2}(\cdot)} \mathcal{IS} \xrightarrow{R^5(\cdot)} \mathcal{IS} \oplus b.$$

or

$$\hat{k} \equiv (E^0 = E^1 = F^0 = F^1 = \ldots = H^0 = 0, H^1 = 0) \in K_{\text{weak}}$$

for which

$$\mathcal{IS} \oplus a \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^6(\cdot)} \mathcal{IS} \oplus b.$$

G.3.4 Chosen-key distinguisher on *full* AES-256

The previous chosen-key distinguisher covers 13 rounds of AES-256. Here we show that it is possible to consider a weaker property (e.g.) on the plaintexts to cover full AES-256 in the single-key setting. In this case, the chosen-key model asks the adversary to find a set of 2^{64} (plaintexts, ciphertexts), that is $(p^i, c^i \equiv R^{14}(p^i))$ for $i = 0, \ldots, 2^{64} - 1$ – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that the following "*simultaneous multiple-of-n*" property is satisfied:

Plaintext: on the plaintexts, we re-use the previous properties:

> *(1st)* for each $J \subseteq \{0, 1, 2, 3\}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_J$ is a multiple of $128 = 2^7$;
>
> *(2nd)* for each $I \subset \{(0,0), (0,1), \ldots, (3,2), (3,3)\} \equiv \{(i,j)\}_{0 \leq i,j \leq 3}$, the number of different pairs of plaintexts that belong to the same coset of $\mathcal{X}_I$ are a multiple of 2;

Ciphertext: *for each $J \subseteq \{0, 1, 2, 3\}$, the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ is a multiple of* 2.

Choosing the key as before and due to the same arguments given for AES-128 and AES-192, the computational cost to construct such set is of 2^{64}.

What about an adversary facing a family of random and independent ideal ciphers? Due to previous analysis, the required properties holds with prob. $2^{-32\,823} \simeq 2^{-2^{15}}$ for a random set of texts. As before, a simple brute force attack requires at least $2^{64} + 2^{11}$ plaintexts/ciphertexts in order to find a set of 2^{64} plaintexts with the required properties. Due to our argumentations from Section 6.4, we conjecture that the computational cost of an adversary to generate such set is (much) higher than 2^{64} computations.

G.4 Open problem – chosen-key distinguishers on full AES-192

We have just seen how to set up chosen-key distinguisher for 11-round AES-192. An open problem is to set up a chosen-key distinguisher in the single-key setting on full AES-192.

As we have seen in Section 5.2, the multiple-of-2 property holds for one more round than the multiple-of-128 property independent of the secret weak-key. Thus, a possible idea/starting point is to exploit such property to construct the distinguishers. Here, the chosen-key model asks the adversary to find a set of 2^{64} plaintexts/ciphertexts (p^i, c^i) – where all the plaintexts/ciphertexts are generated by *the same key* – and a key such that:

- for each $I \subseteq \{0,1,2,3\}$ the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ for each I is a *multiple of 2*; similarly, for each $J \subseteq \{0,1,2,3\}$ the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ for each J is a *multiple of 2*.

Observe that for a random set, the previous properties hold with probability $(2^{-1})^{2\cdot 14} = 2^{-28}$. Note that such property can be potentially used to set up both a distinguisher on full AES-192 and on full AES-256.

Here we discuss this distinguisher, focusing on the adversary strategy. By simple computation, $2^{64}+1$ oracle queries are sufficient to generate the required set with high probability. Indeed, given $2^{64}+1$ texts, it is possible to construct $\binom{2^{64}+1}{2^{64}} = 2^{64}$ different sets of 2^{64} texts. Since each one of them satisfies the required property, the probability to find a good set is close to one[27]. On the other hand, if one considers the total cost of the adversary to find the required sets of 2^{64} texts using a "simple" brute force, the computational cost is in general higher than 2^{64}. In Appendix G.4, we propose a possible strategy that the adversary can use to find the required set. However, even if such strategy has a computational cost of approximately 2^{64} computations, the memory cost is approximately of 2^{96}. It follows that the total cost (computations + memory) can be considered higher than the 2^{64}. For completeness, we highlight that such strategy requires 2^{64} computations only if a random set satisfies the multiple-of-n property with probability at most 2^{-64}, which implies that n must satisfy $n < 8$. It follows that such strategy can not be used for the previous distinguishers where the multiple-of-128 property is exploited.

As a result this can be *a candidate for a possible distinguisher on full AES-192/256*. We leave the open problem to confute or to prove this fact for future work[28]. In conclusion, the problem to set up a chosen-key distinguisher in the single-key setting on full AES-192/256 is still open for future research.

"Simultaneous Multiple-of-2" chosen-key distinguisher – (inefficient) strategy for the adversary

We have just discussed the possibility to set up chosen-key distinguishers on full AES-192/256 by exploiting the multiple-of-2 property. Here we claim that an adversary - who does not know the key - is able to construct a set of 2^{64} plaintexts/ciphertexts (p^i, c^i) such that:

- for each $I \subseteq \{0,1,2,3\}$ the number of different pairs of plaintexts that belong to the same coset of $\mathcal{D}_I$ for each I is a multiple of 2; similarly, for each $J \subseteq \{0,1,2,3\}$ the number of different pairs of ciphertexts that belong to the same coset of $\mathcal{M}_J$ for each J is a multiple of 2.

with a computational cost of 2^{64} computations and a memory cost of 2^{96}. Here we present such strategy.

1) As first thing, the adversary consider a set of 2^{64} random plaintexts and corresponding ciphertexts, and count the number of collisions. To do this, we propose to use the strategy proposed in Algorithm 2 – described in the following – for each possible subspace $\mathcal{D}_I$ and $\mathcal{M}_J$.

The basic idea is to implement the distinguisher using a data structure. Assume $J \subseteq \{0,1,2,3\}$ with $|J| = 3$ is fixed (the other cases are analogous). The goal is to count

[27] In particular, $3 \cdot 2^{28}$ sets are sufficient to find the required sets with probability higher than 95%, since $1-(1-2^{-28})^{3\cdot 2^{28}} \simeq 1-e^{-3} \simeq 95\%$.

[28] In more details, in order to confute this fact one has to find a strategy that the adversary can use for which the total cost is of 2^{64}. To prove this result, one has to show that no strategy exists with the required cost.

Algorithm 2 Count number of collisions in the same coset of $\mathcal{M}_J$ for $|J| = 3$.

Precondition: 2^{64} (plaintext, ciphertext) pairs (p^i, c^i) for $i = 0, \ldots, 2^{64} - 1$.
Postcondition: Number of Collisions n in $\mathcal{M}_J$ for $|J| = 3$.

```
for all j ∈ {0,1,2,3} do
    Let A[0,...,2^32 − 1] an array initialized to zero
    for i from 0 to 2^32 − 1 do
        x ← 0
        for k from 0 to 3 do
            x ← x + MC^{-1}(c^i)_{k,j−k} · 256^k      ▷ MC^{-1}(c^i)_{k,j−k} denotes the byte of MC^{-1}(c^i) in row k and column j − k mod 4
        A[x] ← A[x] + 1    ▷ A[x] denotes the value stored in the x-th address of the array A
    n ← 0                                              ▷ n ≡ Number of Collisions
    for i from 0 to 2^32 − 1 do
        n ← n + A[i] · (A[i] − 1)/2
return n
```

the number of pairs of ciphertexts (c^1, c^2) such that $c^1 \oplus c^2 \in \mathcal{M}_J$, or equivalently

$$MC^{-1}(c^1)_{i,j-i} = MC^{-1}(c^2)_{i,j-i} \tag{10}$$

for all $i = 0, 1, 2, 3$ where $j = \{0, 1, 2, 3\} \setminus J$, and the index is computed modulo 4. To do this, consider an array A of 2^{32} elements completely initialized to zero. The element of A in position x for $0 \leq x \leq 2^{32} - 1$ – denoted by $A[x]$ – represents the number of ciphertexts c that satisfy the following equivalence (in the integer field $\mathbb{N}$):

$$x = c_{0,0-j} + 256 \cdot MC^{-1}(c)_{1,1-j} + MC^{-1}(c)_{2,2-j} \cdot 256^2 + MC^{-1}(c)_{3,3-j} \cdot 256^3.$$

It's simple to observe that if two ciphertexts c^1 and c^2 satisfy (10), then they increment the same element x of the array A. It follows that given $r \geq 0$ texts that the same element x of the array A, then it is possible to construct $\binom{r}{2}$ different pairs of texts that satisfy (10).

For each $\mathcal{M}_J$ and for each $\mathcal{D}_I$, the idea is to store the array $A[\cdot]$ (28 in total). These arrays are then used in the following to construct the required set of texts.

2) If the previous set does not satisfy the required property, the idea is simply to change one text with a random one until the multiple-of-2 property is verified: with high probability, $3 \cdot 2^{28}$ texts are sufficient, for a total of $2^{64} + 3 \cdot 2^{28} \simeq 2^{64}$ oracle queries. Indeed, using $3 \cdot 2^{28}$ texts, the multiple-of-2 property is verified with probability

$$1 - (1 - 2^{-28})^{3 \cdot 2^{28}} \simeq 1 - e^{-3} \simeq 95\%$$

Equivalently, one can use a single random query, for a total of $2^{64} + 1$ queries. Given $2^{64} + 1$ texts, one simply considers $3 \cdot 2^{28}$ different subsets of 2^{64} texts.

3) *How to count the new number of collisions?* Assume we change one text that contributes the entry $A[i]$ of the array with one that contributes $A[j]$. It is simple to observe that the new number of collisions is given by

$$n \leftarrow n - \binom{A[i]}{2} + \binom{A[i]-1}{2} + \binom{A[j]+1}{2} - \binom{A[j]}{2} = n + A[j]^2 + (A[i]^2 - 2 \cdot A[i] + 1).$$

That is, 2 table look-ups are sufficient to compute the new number of collisions for each subspace $\mathcal{M}_I$ and $\mathcal{D}_J$.

On the other hand, a more efficient strategy can be considered. Note indeed that we are only looking for the parity of the number of collisions, and not for the real number. Thus, observe that

$$\binom{A[i]}{2} \equiv \frac{A[i] \cdot (A[i]-1)}{2} = 0 \bmod 2 \qquad \textit{iff} \qquad A[i] = 0 \text{ or } 1 \bmod 4.$$

This implies that each entry of the table can be computed module 4, that is $A[x] \leftarrow (A[x]+1) \bmod 4$. Using this observation, it follows that each entry of the table is composed of only 2 bits, for a total of 4 (possible indexes I with $|I| = 3$) $\times 4 \times 2^{32} = 2^{36}$ bits in the case of subspace of dimension 12 (e. g. $\mathcal{M}_J$ for $|J| = 3$), 6 (possible indexes I with $|I| = 2$) $\times 4 \times 2^{64} = 2^{68.6}$ bits in the case of subspace of dimension 8 and 4 (possible indexes I with $|I| = 1$) $\times 4 \times 2^{96} = 2^{100}$ bits in the case of subspace of dimension 4. It follows that the total memory cost is well approximated by $2^{100}/16 = 2^{96}$ equivalently texts.

H Practical collisions for 7-round AES-256 compressing modes

Many block cipher hashing modes contain XOR of input and output of the cipher. E.g. given an input $x = (x_0, x_1, ..., x_n)$, the corresponding hash $H = (H_0 \equiv IV, H_1, ..., H_n)$ can be produced using

- the Davies–Meyer hash function: $H_i = E_{x_i}(H_{i-1}) \oplus H_{i-1}$;
- the Matyas-Meyer-Oseas hash function $H_i = E_{g(H_{i-1})}(x_i) \oplus x_i$;
- the Miyaguchi–Preneel hash function $H_i = E_{g(H_{i-1})}(x_i) \oplus H_{i-1} \oplus x_i$.

In this section, we show how to produce collisions for some of such constructions exploiting our invariant subspace $\mathcal{IS}$. *Since we assume the attacker is able to choose the initial value IV, we propose our results in the compressing mode.*

Using the result proposed in Appendix G.3.3 and when the first and second round keys (namely, k_1 and k_2) are all zero, we have

$$\mathcal{IS} \oplus k_0 \xleftarrow{R^{-1}(\cdot)} \mathcal{IS} \xrightarrow{R^6(\cdot)} \mathcal{IS} \oplus k_6,$$

where k_0 and k_6 are the initial and final round keys.

Since dimension of $\mathcal{IS}$ is 64, we expect to find a collision with (at least) 2^{32} elements in $\mathcal{IS}$. In fact, since one can construct $\binom{2^{32}}{2} = 2^{32} \cdot (2^{32}-1)/2 \approx 2^{63}$, the probability to find a collision is approximately $1 - (1 - 2^{-64})^{2^{63}} \approx 1 - e^{-1/2} \approx 39.35\%$.

We performed two experiments by encrypting 2^{32} elements in $\mathcal{IS}$ in an inside out fashion by choosing the AES-256 key as

$$[k_0 \| k_1] = [62636363\ 00000000\ 00000000\ 00000000 \| 00000000\ 00000000\ 00000000\ 00000000],$$

which makes first and second round keys zero. In our first experiment we used the smaller invariant subspace[29] $\mathcal{IS}'$ of dimension 32 where every column is identical

$$\mathcal{IS}' \coloneqq \left\{ \begin{bmatrix} a & a & a & a \\ b & b & b & b \\ c & c & c & c \\ d & d & d & d \end{bmatrix} \;\middle|\; \forall a, b, c, d \in \mathbb{F}_{2^8} \right\},$$

[29] This choice is motivated by the fact that $k_1, k_2, k_3, k_4 \in \mathcal{IS}'$. As a result:

$$\mathcal{IS}' \oplus k_0 \xleftarrow{R^{-1}(\cdot)} \mathcal{IS}' \xrightarrow{R^5(\cdot)} \mathcal{IS}' \oplus k_5$$

where $\mathcal{IS}' \oplus k_5 \subseteq \mathcal{IS}$ (since $k_5 \in \mathcal{IS} \setminus \mathcal{IS}'$).

Table 4: Examples of compression collisions for 6 and 7-round AES-256 used in Matyas-Meyer-Oseas construction where $k_0\|k_1$ = 62636363 00000000 00000000 00000000 00000000 00000000 00000000 00000000. Last round contains the matrix multiplication but not the final key addition, which does not affect the collisions. These plaintexts are also collisions for Miyaguchi-Preneel mode and pseudo-collisions for Davies-Meyer mode.

Plaintext	Hash (Plaintext ⊕ Ciphertext)
7-round Collisions	
6407503c0664335f 0664335f0664335f c2e01a46a0837925 a0837925a0837925	4a2e96618b438711 284df5028b438711
fa8cca8ad93ff889 98efa9e9d93ff889 02cc0aa7b96b44b3 60af69c4b96b44b3	79b1f1b3c1415dd7 1bd292d0c1415dd7
6-round Collisions	
b1b602e8d3d5618b d3d5618bd3d5618b d0122734b2714457 b2714457b2714457	f85752eeb3488419 9a34318db3488419
e75dd657853eb534 853eb534853eb534 27f4f3b1459790d2 459790d2459790d2	c99eec4ba84135a3 abfd8f28a84135a3
e00159e982623a8a 82623a8a82623a8a be9c9a2adcfff949 dcfff949dcfff949	09315bee8a3b5978 6b52388d8a3b5978
6eed06230c8e6540 0c8e65400c8e6540 497163fb2b120098 2b1200982b120098	a77bf28d6e087b35 c51891ee6e087b35
345684eb5635e788 5635e7885635e788 5ee850813c8b33e2 3c8b33e23c8b33e2	c82e26780c32ed63 aa4d451b0c32ed63
439ad67621f9b515 21f9b51521f9b515 7e4032701c235113 1c2351131c235113	2ecfb051888f27dd 4cacd332888f27dd

which is the same invariant subspace independently used in the cryptanalysis of constructions using the unkeyed AES round permutation, recently e.g. for cryptanalysis of Haraka hash function in [Jea16], and in the second one we chose 2^{32} random elements in $\mathcal{IS}$.

As a result, we got a 7-round collision in both cases for the Matyas-Meyer-Oseas or Miyaguchi-Preneel compressing functions constructed with 7-round AES-256, where the attacker choose IV $(= H_0)$ as k_0. Note that since AES-256 block size is 128 bits and key size is 256 bits, a $g(\cdot)$ conversion/padding function is used on the output to make it suitable as the key. A very natural function $g(\cdot) : \mathbb{F}_{2^n} \mapsto \mathbb{F}_{2^{2n}}$ that turns out to be good for our purpose is given by

$$g(x) = x \| \underbrace{0 \ldots 0}_{n \text{ bit}} \in \mathbb{F}_{2^{2n}}$$

where $\|$ denotes concatenation. Our collisions for 7-round AES-256 hashing modes are provided in Table 4. Moreover, an perhaps a more natural application, these collisions turn into collisions for Davies-Meyer compressing mode where the message block is fixed to $k_0\|k_1$ and the plaintexts of Table 4 are used as IVs.

To the best of our knowledge, the best known collision attacks on AES compressing modes are the trivial conversion of the Whirlpool attacks of [Lam+15]. They turn into 6-round collision attacks on every key length of AES which require 2^{56} time and 2^{32} memory complexity. Our collisions are on 7 rounds and require 2^{32} time and 2^{32} memory complexity where a time-memory tradeoff is also possible. Our attack is also valid for 6 rounds with the same complexities. It may be conceivable that local collision methods

from [Bir+09] can be adapted to the compression collision setting we consider here. Note however that this approach can not avoid to simultaniously require differences in both the chaining as well as the message input of an AES-256-based compression functions, whereas we only need a difference in one of the two.

I Sage code

Listing 1: Sage code for Algorithm 1

```
def wkst(f, ks, n, max_rnd, finv=None,
         ksinv=None, backwards_at=[]):
    """
    Return the set of all subspace trails containing
    W_{i,alpha}

    INPUT:
        - ''f'' -- function; mapping from
                           F_2^n -> F_2^n
        - ''ks'' -- function; mapping from
                            F_2^n -> F_2^n
        - ''n'' -- integer; the same n as in F_2^n
        - ''max_rnd'' -- integer; upper bound on
                                  the number of
                                  rounds to cover
        - ''finv'' -- function; mapping from
                                F_2^n -> F_2^n
        - ''ksinv'' -- function; mapping from
                                F_2^n x Z -> F_2^n
        - ''backwards_at'' -- list of integers;
                              rounds in which to
                              compute backwards
    """
    if backwards_at != []:
        assert finv is not None
        assert ksinv is not None

    vs = VectorSpace(GF(2), n)
    rnd_cnt = 0
    S = [vs.zero()]
    Us = [(vs.subspace([]),
           vs.subspace([ks(s, rnd_cnt) for s in S])
          )]

    while Us[-1][0].dimension() != n:
        rnd_cnt += 1

        Ui_next = [f(vs.zero())]
        for _ in range(n*1.3):
            u = Us[-1][1].random_element()
            Ui_next.append(f(u))

        Ui_next = vs.subspace(Ui_next)
        Ui_next_prime = vs.subspace(Ui_next.basis()
            + [ks(s, rnd_cnt) for s in S])
        Us += [(Ui_next, Ui_next_prime)]

        if rnd_cnt in backwards_at:
            backwards_at.remove(rnd_cnt)

            Ui_next_prime = Us[-1][1]
            for _ in range(Ui_next_prime.dimension()+10):
                ki = Ui_next_prime.random_element()
```

```
54                  k0 = ksinv(ki, rnd_cnt)
55                  if k0 not in S:
56                      S.append(k0)
57
58              Us = backwards(finv, n, rnd_cnt, Us[-1][0])
59              rnd_cnt = 0
60
61          if max_rnd != 0 and rnd_cnt >= max_rnd:
62              break
63
64      return Us, S
65
66
67  def backwards(finv, n, rnd, Ui):
68      """
69      Given a weak-key-subspace-trail Us, compute
70      backwards  for rnd many rounds, using the
71      inverse round function finv, from its last
72      subspace and return a new starting point
73      (U_0, U_0')
74
75      INPUT:
76          - ''finv'' -- function; mapping from
77                                  F_2^n -> F_2^n
78          - ''n'' -- integer; the same n as in F_2^n
79          - ''rnd'' -- integer; number of rounds
80                                 to invert
81          - ''Ui'' -- subspace;
82      """
83      vs = VectorSpace(GF(2), n)
84      Ui_prev, Ui_prev_prime = Ui, Ui
85
86      # compute rnd many rounds backwards
87      for i in range(rnd, -1, -1):
88          Ui_prev = list(Ui_prev_prime.basis()
89                         + [finv(vs.zero())])
90          for _ in range(n*1.3):
91              u = Ui_prev_prime.random_element()
92              Ui_prev.append(finv(u))
93          Ui_prev = vs.subspace(Ui_prev)
94      return [(Ui_prev, Ui_prev)]
```