Revisiting the Transferability of Chosen- to Known-plaintext Attacks and Applications to Round-reduced AES

Xiaomeng Sun[1], Eik List[2], and Wenying Zhang[1]

[1] School of Information Science and Engineering,
Shandong Normal University, Jinan, China
sunxiaomeng98@126.com, zhangwenying@sdnu.edu.cn
[2] School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore, Singapore
eiklist@ntu.edu.sg

Abstract. Differential-based attacks represent the best known results for many block ciphers. Such attacks usually demand that the adversary can choose plaintexts (CP) or ciphertexts (CC) in subspaces to satisfy differential trails. However, many widespread modes of operation or applications prohibit the adversary from directly choosing inputs for the majority of primitive calls. While Biham and Shamir already suggested a straightforward trade-off for standard differential attacks in their work on the DES, studies on advanced differential-based types, such as impossible-differential, rectangle, or mixture attacks, have surprisingly received little attention so far.
In this work, we study applications of differential-based attacks in the random known-plaintext model (RKP) of the above. For the AES as the probably most widespread block cipher, we derive the best existing distinguishers and attacks in the RKP model on all versions, improving earlier results by at least one round. Interestingly, we show that Demirci-Selçuk meet-in-the-middle attacks with differential enumeration, which require much related data, can also be effective in that setting without approaching the full codebook too closely. For several of our attacks, we showcase differences between the models as trails that lead to the best known attack complexities under chosen data are suboptimal in the RKP model, and can be replaced by better trails. While our results do not threaten the security of the full AES, and their complexities are too high to represent any threats, we hope to inspire cryptographers to also consider attacks in the RKP for future attacks.

Keywords: Differential cryptanalysis · block cipher · AES

1 Introduction

1.1 Block-cipher Attack Settings

Block ciphers are the workhorses of symmetric-key cryptography, with various applications for manifold cryptographic goals from encryption, authentication,

integrity, or one-wayness in complex protocols. Consequently, a thorough understanding of their security against known attacks is a critical task. In particular, the security of the Advanced Encryption Standard (AES) [23,59] is of enormous practical relevance as it is probably the most widespread cipher. Unsurprisingly, since its proposal, and even more so since its standardization, it has received tremendous analysis.

In general, primitive cryptanalysis categorizes attacks by adversarial capabilities into known-plaintext, chosen-plaintext, chosen-ciphertext, and adaptively chosen-data attacks. As attacks have been focusing on maximizing the number of rounds that can be broken more efficiently than exhaustive key search and with less data than the full codebook, chosen-data attacks have become the de facto standard models. Over time, the models were further strengthened to consider also adversaries that can choose key relations. Besides, multi-user, weak-key, and even chosen-key attacks emerged. Such stronger models are relevant, as block ciphers are versatile primitives for various applications. Some settings demand strong security, e.g., the usage as a compression or one-way function. However, with the increasing adversarial capabilities, the gap between very strong models and more practically motivated ones has widened.

Known-plaintext attacks have remained relevant on many occasions, as choosing plaintexts and collecting the results is often not possible, or could lead to the adversary's detection. Additionally, many modes of operation may further restrict the inputs to the primitive compared to the theoretical models. For example, randomized variants of CTR mode [35], as employed in various AE schemes or wide-block ciphers such as AES-GCM [36] or AES-GCM-SIV [43,44], allow only forward queries and restrict the set of legitimate inputs. Moreover, various modes also encrypt previously randomized values. For example, the PRF in SIV [45,60] generates a known pseudorandom authentication tag that is then used as the initial value for CTR-mode encryption.

For the AES as the most widespread block cipher, the previous decade brought various new insights and novel techniques: the multiple-of-n property [17,42], mixture differentials [39], yoyos [62], exchange attacks [5], small-bias truncated differentials [3,41], retracing boomerangs [31,32], the combination of FFT and partial sums [30], and new representations of its key schedule [49,50]. Those have led to new records for distinguishers on up to six rounds, new attacks on seven rounds, and new improved practical attacks on five and six rounds. But how those techniques translate into attacks in the more restrictive known-plaintext model is not clear. The best attacks have remained undefeated for over a decade.

1.2 Related Work

Studying differential attacks in the known-plaintext model is as old as the public record of differential cryptanalysis itself. Biham and Shamir suggested already in their seminal analysis of the Data Encryption Standard (DES) [9,10] that a chosen-plaintext differential attack requiring 2^r pairs could be theoretically converted into a known-plaintext attack with $2^{(r+n+1)/2}$ data. However, the approach remained unexplored since then. One generation after Biham and

Table 1: Best single-key key-recovery attacks on round-reduced AES in the random known-plaintext model. $^{(*)}$ = we argue in the Appendix H that the data does not suffice. #Rds. = #rounds, memory in #AES states.

Version	Attack type	#Rds.	Time	Data	Memory	Reference
128	MitM	5/10	2^{120}	1	2^{96}	[16,25]
	MitM	5/10	2^{120}	1	2^{40}	[21]
	MitM	6/10	2^{120}	$2^{108.5}$	$2^{109.5}$	[15]
	Rectangle	**6/10**	$\mathbf{2^{105.8}}$	$\mathbf{2^{99.1}}$	$\mathbf{2^{103.2}}$	**Sect. 4**
	DS-MitM	**7/10**	$\mathbf{2^{125.1}}$	$\mathbf{2^{124}}$	$\mathbf{2^{125}}$	**Sect. 8**
	Imp. differential	**7/10**	$\mathbf{2^{118.1}}$	$\mathbf{2^{116.5}}$	$\mathbf{2^{117.5}}$	**Sect. 7**
192	Zero correlation	6/12	$2^{188.4}$	2^{128}	2^{128}	[13]
	MitM	7/12	2^{184}	1	2^{72}	[21]
	Imp. differential	**7/12**	$\mathbf{2^{126}}$	$\mathbf{2^{117.5}}$	$\mathbf{2^{118.5}}$	**Appendix G**
	DS-MitM	**8/12**	$\mathbf{2^{179}}$	$\mathbf{2^{124}}$	$\mathbf{2^{128.6}}$	**Appendix I**
256	Zero correlation	6/14	$2^{188.4}$	2^{128}	2^{128}	[13]
	MitM	8/14	2^{248}	1	2^{72}	[21]
	Imp. differential$^{(*)}$	8/14	2^{220}	2^{118}	2^{145}	[46]
	Imp. differential	**8/14**	$\mathbf{2^{228.5}}$	$\mathbf{2^{116.8}}$	$\mathbf{2^{217}}$	**Appendix H**
	DS-MitM	**9/14**	$\mathbf{2^{247.7}}$	$\mathbf{2^{124}}$	$\mathbf{2^{194.1}}$	**Appendix J**

Shamir [9,10], Bouillaguet et al. [15, Sect. 8] proposed "an attack on 6-round AES, using a low-probability differential" that was "the best known attack on 6-round AES in the known-plaintext model", and has stood as the best attack so far. Thus, explicit studies have remained necessary to clarify the potential of transferring attacks from chosen- to known-data models. More generally, attacks with multiple (but few) related texts, such as boomerangs, or mixtures, etc., have not received attention so far. The understanding is even lower for techniques that consider sets of more text and pair relations, such as integral or Demirci-Selçuk Meet-in-the-Middle (DS-MitM) attacks. Intuitively, they seem to prohibit any straightforward transformation unless the amount of required data approaches the full codebook very closely.

Considerable work on the relations between attacks with different adversarial capabilities has also been conducted in provable security. In that direction, the focus was on deriving security guarantees against adversaries that can choose inputs from components with security guarantees only against random inputs. Maurer and Pietrzak showed compositions of weak components to PRFs [56]; Myers showed black-box compositions that did not yield adaptive security [58]. Minematsu and Tsunoo [57] proposed symmetric encryption from wPRFs. Maurer et al. [55] studied conversions of attacks on Feistel ciphers from the chosen- to the known-plaintext model by considering the applicability of attacks on PRFs to wPRFs. In contrast, we exploit the internals of the studied primitive, whereas provable security usually treats the primitives as black boxes.

1.3 Contributions

In this work, we revisit the relation of general differential-based attacks from the chosen plain-/ciphertext to the random known-plaintext (RKP) model. In the course, we study the most effective existing attack types on round-reduced AES with a single secret key, which are all in chosen-data models (or their adaptive variants). We show that impossible- and truncated-differential attacks can be effectively applied in the RKP model, although with higher data complexity. Beyond those, we generalize the approach to techniques with related pairs, such as rectangle and mixture attacks, and demonstrate new attacks in the RKP model as well. For those, we show in examples that the effectiveness of trails does not straightforwardly translate from one model to the other; preferable trails in the CP model may not produce the minimal attack complexity in the RKP model. For some of our mixture and rectangle attacks, choosing seemingly less efficient trails allows us to reduce the complexities.
We also demonstrate that DS-MitM attacks, which require many related texts, can use trade-offs that allow their use in the RKP model as well. While it would be easy with a data complexity very close to the full codebook, we strive to restrict the attack to a more meaningful factor of using at most 1/16 of it, which demands a more sophisticated technique. For the AES-128, we show that an attack in the RKP model poses additional constraints, necessitating the use of only certain differential trails. Tables 1 and 2 compare our results with the best previous attacks on round-reduced AES in the known-plaintext model from the literature. They outline that we can add at least one round over the previous understanding. For the interested reader, Table 3 in Appendix A provides the best chosen-data key-recovery attacks on round-reduced versions of the AES in single-key models for comparison.

1.4 Outline

After briefly recalling the necessary preliminaries on the AES, we study generic conversions from the CP to the RKP model for differential, impossible-differential, boomerang, and mixture attacks in Section 3. Next, we show conversions of the best known six-round distinguishers and attacks of the round-reduced AES in the follow-up sections. It includes a rectangle attack in Section 4, a mixture distinguisher in Section 5, and a truncated-differential distinguisher in Section 6. Then, it presents conversions of the seven-round impossible-differential attack in Section 7, and the seven-round DS-MitM attack in Section 8. We discuss and conclude this work in Section 9. In the extended version, we show similar applications, such as a rectangle distinguisher as well as impossible-differential and DS-MitM attacks on round-reduced AES-192 and AES-256.

2 Preliminaries

General Notations. For positive integers p, n, we denote by $\mathbb{F}_2^n$ the extension of the finite field of characteristic 2. We represent functions and variables by

Table 2: Best single secret-key distinguishers on round-reduced AES in the random known-plaintext model. #Rds. = #rounds, memory in #AES states).

Attack type	#Rds.	Time	Data	Memory	Reference
Zero correlation	4	2^{128}	2^{128}	negl.	[12]
Conditional linear	4	$2^{125.6}$	$2^{125.6}$	negl.	[1,2]
Mixture differential	**6**	$\mathbf{2^{118.1}}$	$\mathbf{2^{115.5}}$	$\mathbf{2^{117.5}}$	**Sect. 5**
Rectangle	**6**	$\mathbf{2^{117.7}}$	$\mathbf{2^{113.5}}$	$\mathbf{2^{115.5}}$	**Appendix C**
Truncated differential	**6**	$\mathbf{2^{111.9}}$	$\mathbf{2^{106.8}}$	$\mathbf{2^{108.4}}$	**Sect. 6**

upper-case and indices by lower-case letters, sets by calligraphic letters (such as $\mathcal{C}$). We employ typewriter font for hexadecimal values (such as `0xff`). Let $X, Y \in \mathbb{F}_p^n$ for some positive integer n in the following. Then, we denote by $X \| Y$ the concatenation of X and Y, by $X \oplus Y$ their elementwise XOR. For a non-negative integer i, we write $[i..j]$ for $\{i, i+1, \ldots, j\}$. $\langle i \rangle_m$ denotes the m-bit encoding of an integer i, e.g. $\langle 135 \rangle_8 = (10000111)$.

The AES. The Advanced Encryption Standard [23,59] is a substitution-permutation network that transforms 16-byte plaintexts through 10, 12, or 14 rounds for key sizes of 128, 192, or 256 bits, respectively. Each state byte is interpreted as an element of the finite field $\mathbb{F}_{2^8}$. The state itself can be viewed either as a linear array of 16 elements in $\mathbb{F}_{2^8}^{16}$ or, also commonly, as a 4×4 matrix in $\mathbb{F}_{2^8}^{4\times 4}$. Almost each round consists of the operations SubBytes (SB), ShiftRows (SR), MixColumns (MC), and AddRoundKey[K^i] (AK[K^i]), i.e. a round-key addition with a round key K^i. Before the first round, an additional whitening key K^0 is XORed to the state; the final round omits the MixColumns operation. For compactness, we will sometimes write $\overline{\mathsf{SB}}$, $\overline{\mathsf{SR}}$, and $\overline{\mathsf{MC}}$ for SB^{-1}, SB^{-1}, and MC^{-1}, respectively. We write $X[i]$ for the i-th byte of an internal state X, and we interchangeably also use $X[i][j]$ to denote the byte in the i-th row and j-th column of state X with the usual byte ordering of

$$\begin{bmatrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} (0,0) & (0,1) & (0,2) & (0,3) \\ (1,0) & (1,1) & (1,2) & (1,3) \\ (2,0) & (2,1) & (2,2) & (2,3) \\ (3,0) & (3,1) & (3,2) & (3,3) \end{bmatrix}.$$

When using $X[i][j]$, we treat the indices as taken modulo four by default. Note that we will use the notation $X[i_1, i_2]$ as a short form for $X[i_1], X[i_2]$. We denote by X^i, Y^i, Z^i, and W^i the internal states in the i-th round directly after the application of AddRoundKey, SubBytes, ShiftRows, and MixColumns, respectively. We denote by $\mathsf{R}[K^i] \stackrel{\text{def}}{=} \mathsf{AK}[K^i] \circ \mathsf{MC} \circ \mathsf{SR} \circ \mathsf{SB}$ one AES round, and denote by $\widehat{\mathsf{R}}[K^i] \stackrel{\text{def}}{=} \mathsf{AK}[K^i] \circ \mathsf{SR} \circ \mathsf{SB}$ the reduced final round. We denote by $\widehat{K}^i = \mathsf{MC}^{-1}(K^i)$ an equivalent key of K^i, and denote by $\widehat{W}^i$ the state after AddRoundKey if we exchange the order of MixColumns and AddRoundKey in the i-th round. We use $\mathbf{M} = \mathsf{circ}(2, 3, 1, 1)$ to denote the circular MixColumns matrix.

3 Generic Conversions

3.1 General Setting

Leaving advanced concepts of guess-and-determine, internal-state, or internal-difference guessing aside, a general differential chosen-plaintext attack needs the following. Suppose we can split a given cipher E into parts $E : \mathbb{F}_2^k \times \mathbb{F}_2^n \to \mathbb{F}_2^n$ with k-bit key and n-bit blocks into $E = E_{\text{out}} \circ E_{\text{dist}} \circ E_{\text{in}}$. Moreover, let spaces $\mathcal{D}_{\text{in}}$, $\mathcal{D}_{\text{out}}$, $\mathcal{D}_x$, $\mathcal{D}_y \subseteq \mathbb{F}_2^n$. Further assume that there is a differential $\Delta X \xrightarrow{E_{\text{dist}}} \Delta Y$ through E_{dist}. Let the adversary gather $D = 2^d$ plaintexts that can be partitioned into 2^s structures of texts such that for each pair of distinct texts P, P' in a structure, $P \oplus P' \in \mathcal{D}_{\text{in}}$. They are influenced by k_{in} key bits from a key space $\mathcal{K}_{\text{in}}$ through E_{in}. The goal is to find pairs in a difference space $\mathcal{D}_x$ that serves as inputs to a distinguisher through E_{dist}, and is mapped probabilistically to output pairs in an output-difference space $\mathcal{D}_y$ after E_{dist} that allows to distinguish E_{dist} with non-negligible advantage from a random permutation. At the ciphertext side, the adversary filters all pairs except those whose ciphertext differences lie in a certain space $\mathcal{D}_{\text{out}}$. Those pairs are influenced by k_{out} key bits from a key space $\mathcal{K}_{\text{out}}$ through E_{out}. The adversary searches for keys in $\mathcal{K}_{\text{in}} \cup \mathcal{K}_{\text{out}}$ that map plaintext pairs from $\mathcal{D}_{\text{in}}$ and their corresponding ciphertext pairs from $\mathcal{D}_{\text{out}}$ to pairs with difference in $\mathcal{D}_x$ through E_{in} and to pairs with difference in $\mathcal{D}_y$ through E_{out}^{-1}.
Let $\delta_{\text{in}} =^{\text{def}} \log_2(|\mathcal{D}_{\text{in}}|)$, $\delta_{\text{out}} =^{\text{def}} \log_2(|\mathcal{D}_{\text{out}}|)$ and assume that for random nonzero $\Delta P \in \mathcal{D}_{\text{in}}$, and $\Delta C \in \mathcal{D}_{\text{out}}$, and over the random choice of the key K, we define c_{in} and c_{out} for the number of bit conditions (cf. [19,18]) that have to be fulfilled for a plaintext pair to form a difference in $\mathcal{D}_x$ and for a ciphertext pair to form a difference in $\mathcal{D}_y$, respectively:

$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : E_{\text{in}}(K, P) \oplus E_{\text{in}}(K, P \oplus \Delta P) \in \mathcal{D}_x\right] \approx 2^{-c_{\text{in}}} \quad \text{and}$$
$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : E_{\text{out}}^{-1}(K, C) \oplus E_{\text{out}}^{-1}(K, C \oplus \Delta C) \in \mathcal{D}_y\right] \approx 2^{-c_{\text{out}}}.$$

3.2 Standard, Truncated-, and Impossible-differential Attacks

For a differential, truncated-differential, or impossible-differential attack, we assume that there is a nontrivial (i.e., nonzero) differential $\Delta X \xrightarrow{E_{\text{dist}}} \Delta Y$ through E_{dist} with probability 2^{-p}. We assume that $\Delta X \in \mathcal{D}_x$ and $\Delta Y \in \mathcal{D}_y$:

$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : E_{\text{dist}}(K, X) \oplus E_{\text{dist}}(K, X \oplus \Delta X) \in \mathcal{D}_y, \Delta X \in \mathcal{D}_x\right] = 2^{-p}.$$

Chosen-plaintext Model. In a chosen-plaintext attack, the adversary can form up to $\binom{D}{2} \approx 2^{2d-1}$ pairs from $D =^{\text{def}} 2^d$ plaintexts. However, given $|\mathcal{D}_{\text{in}}| = 2^{\delta_{\text{in}}}$, the number of pairs whose input difference is in $\mathcal{D}_{\text{in}}$, N_{CP}, is limited to

$$N_{\text{CP}} \approx 2^{d+\min(d,\delta_{\text{in}})-1} = 2^{d+d'-1} = \begin{cases} 2^{2d-1} & \text{if } |\mathcal{D}_{\text{in}}| \geq 2^d \\ 2^{d+\delta_{\text{in}}-1} & \text{otherwise}, \end{cases}$$

where we define $d' =^{\text{def}} \min(d, \delta_{\text{in}})$. The number of pairs that survive the initial filter, #SP, i.e., the pairs whose ciphertext pairs satisfy $C \oplus C' \in \mathcal{D}_{\text{out}}$, is approximately

$$\mathbb{E}\left[\#\mathsf{SP}_{\text{CP}}^{\text{Diff}}\right] = N_{\text{CP}} \cdot \left(\frac{|\mathcal{D}_{\text{out}}|}{2^n - 1} \cdot (1 - 2^{-p}) + 2^{-p}\right) \approx \frac{2^{d+d'}}{2} \cdot \left(\frac{2^{\delta_{\text{out}}}}{2^n} + 2^{-p}\right). \tag{1}$$

Random Known-plaintext Model. In the distinct RKP model, the adversary can use all data and obtain $N_{\text{RKP}} \approx 2^{2d-1}$ pairs from $D = 2^d$ texts. Thereupon, it has to discard all but those pairs $((P, C), (P', C'))$ that satisfy $P \oplus P' \in \mathcal{D}_{\text{in}}$ and $C \oplus C' \in \mathcal{D}_{\text{out}}$. Thus, one can expect that the number of surviving pairs averages

$$\begin{aligned}\mathbb{E}\left[\#\mathsf{SP}_{\text{RKP}}^{\text{Diff}}\right] &= \frac{|\mathcal{D}_{\text{in}}|}{2^n - 1} \cdot N_{\text{RKP}} \cdot \left(\frac{|\mathcal{D}_{\text{out}}|}{2^n - 1} \cdot (1 - 2^{-p}) + 2^{-p}\right) \\ &\approx \frac{2^{2d+\delta_{\text{in}}}}{2^{n+1}} \cdot \left(\frac{2^{\delta_{\text{out}}}}{2^n} + 2^{-p}\right). \end{aligned} \tag{2}$$

Remark 1. This reflects the conversion proposed in [9,10], where a differential attack requiring 2^r pairs in the chosen-plaintext model could be transformed into an attack with $2^{n/2} \cdot \sqrt{2^{r+1}}$ random known plaintexts (cf. [11]).

We can derive a ratio for the expected number of surviving pairs from 2^d texts between the chosen-plaintext and the random known-plaintext model:

$$\frac{\mathbb{E}\left[\#\mathsf{SP}_{\text{CP}}^{\text{Diff}}\right]}{\mathbb{E}\left[\#\mathsf{SP}_{\text{RKP}}^{\text{Diff}}\right]} = \frac{(1)}{(2)} \approx \frac{2^{d+\min(d,\delta_{\text{in}})-1}}{2^{2d-1+\delta_{\text{in}}-n}} = 2^{n-\max(d,\delta_{\text{in}})}. \tag{3}$$

Thus, given an attack of that type in the CP model, the corresponding reduction to a random known-plaintext attack differs mainly in the data that the adversary has to collect (but also to store, filter, and process) for obtaining the same number of required pairs. Let $2^{d_{\text{CP}}}$ and $2^{d_{\text{RKP}}}$ be the number of plaintext-ciphertext pairs that are required in the corresponding models to acquire N pairs. Then, the data complexity of an RKP attack must satisfy

$$d_{\text{RKP}}^{\text{Diff}} \geq d_{\text{CP}}^{\text{Diff}} + \frac{n - \max\left(d_{\text{CP}}^{\text{Diff}}, \delta_{\text{in}}\right)}{2} \tag{4}$$

to apply an attack that was defined for the chosen-plaintext model.

3.3 Mixture-differential Attacks

Conditional-differential attacks were first introduced for the cryptanalysis of stream ciphers [48], but are well-known in the analysis of block ciphers, hash functions, or compression functions. The adversary constructs (at least) two pairs (X_0, X_0') and (X_1, X_1'), and studies the conditional probability that the

latter will be mapped to (Y_1, Y_1') with $Y_1 \oplus Y_1' \in \mathcal{D}_y'$ through E_{dist} for a certain difference space $\mathcal{D}_y'$ if the former pair (X_0, X_0') was mapped to a pair (Y_0, Y_0') through E_{dist} with $Y_0 \oplus Y_0' \in \mathcal{D}_y$:

$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : E_{\text{dist}}(X_1) \oplus E_{\text{dist}}(X_1') \in \mathcal{D}_y' | E_{\text{dist}}(X_0) \oplus E_{\text{dist}}(X_0') \in \mathcal{D}_y\right] .$$

Mixture differentials [39] are variants of conditional differentials, where related pairs (X_i, X_i') are constructed from an initial pair (X_0, X_0') by an exchange function. In this context, the n-bit strings X_i are interpreted as m w-bit words such that $n = m \cdot w$. The exchange function $\rho : \mathbb{F}_2^m \times \mathbb{F}_{2^w}^m \to \mathbb{F}_{2^w}^m$ takes a bit vector $\alpha \in \mathbb{F}_2^m$ and two elements X_0, $X_1 \in \mathbb{F}_{2^w}^m$ and outputs

$$\rho_\alpha(X_0, X_1) \stackrel{\text{def}}{=} (X_{\alpha[0]}[0], \ldots, X_{\alpha_{m-1}}[m-1]) .$$

For all mixture pairs $(X_i, X_i') = (\rho_{\alpha_i}(X_0, X_0'), \rho_{\alpha_i}(X_0', X_0))$, it holds that $X_i \oplus X_i' = X_0 \oplus X_0'$ for some exchange vector α_i. Define $\delta_y' =^{\text{def}} \log_2(|\mathcal{D}_y'|)$ and $\delta_{\text{out}}' =^{\text{def}} \log_2(|\mathcal{D}_{\text{out}}'|)$, and let $\mathsf{E}_Y =^{\text{def}} E_{\text{dist}}(K, X) \oplus E_{\text{dist}}(K, X \oplus \Delta X) \in \mathcal{D}_y$. We define $2^{-p'}$ as the probability for an average related pair

$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : E_{\text{dist}}(K, X_i) \oplus E_{\text{dist}}(K, X_i \oplus \Delta X) \in \mathcal{D}_y' | \mathsf{E}_Y\right] = 2^{-p'} .$$

Chosen-plaintext Model. In the following, we restrict our attention to attacks where structures are quartets, i.e., they contain two pairs, an initial pair and a conditional pair. In a chosen-plaintext mixture attack with two pairs, the adversary collects $D = 2^d$ texts that can produce $N_{\text{CP}} \approx 2^{d+d'-1}/4$ structures. The number of structures that survive the initial filter, i.e., whose ciphertext differences are in $\mathcal{D}_{\text{out}}$ and $\mathcal{D}_{\text{out}}'$, respectively, is approximately

$$\begin{aligned}\mathbb{E}\left[\#\mathsf{SQ}_{\text{CP}}^{\text{Mix2}}\right] &= N_{\text{CP}} \cdot \left(\frac{|\mathcal{D}_{\text{out}}| \cdot |\mathcal{D}_{\text{out}}'|}{(2^n-1)\cdot(2^n-3)} \cdot (1 - 2^{-(p+p')}) + 2^{-(p+p')}\right) \\ &\approx \frac{2^{d+d'}}{2^2} \cdot \left(\frac{2^{\delta_{\text{out}}+\delta_{\text{out}}'}}{2^{2n}} + 2^{-(p+p')}\right) . \end{aligned} \tag{5}$$

Random Known-plaintext Model. In the RKP model, the adversary can use all pairs and obtain $N_{\text{RKP}} = 2 \cdot 3 \cdot \binom{2^d}{4} \approx 2^{4d-2}$ pairs of pairs from $D = 2^d$ texts. The factors arise as there are three ways of combining the texts into pairs, and two ways of mixing a pair with a fixed exchange vector or its bitwise inverse. Here, the adversary searches for pairs of pairs $((P, C), (P', C'))$ and $((P_1, C_1), (P_1', C_1'))$ such that the first pair $P \oplus P' = \mathcal{D}_{\text{in}}$ is in the desired input space and (P_1, P_1') is a mixture of (P, P'). We obtain

$$\begin{aligned}\mathbb{E}\left[\#\mathsf{SP}_{\text{RKP}}^{\text{Mix2}}\right] &= \frac{|\mathcal{D}_{\text{in}}| \cdot N_{\text{RKP}}}{(2^n-1)_3} \cdot \left(\frac{|\mathcal{D}_{\text{out}}| \cdot |\mathcal{D}_{\text{out}}'|}{(2^n-1)\cdot(2^n-3)} \cdot \left(1 - 2^{-(p+p')}\right) + 2^{-(p+p')}\right) \\ &\approx \frac{2^{4d+\delta_{\text{in}}}}{2^{3n+2}} \cdot \left(\frac{2^{\delta_{\text{out}}+\delta_{\text{out}}'}}{2^{2n}} + 2^{-(p+p')}\right) . \end{aligned} \tag{6}$$

As a result, the ratio for the expected number of surviving pairs from 2^d texts between the CP and the RKP model for mixtures with two pairs becomes

$$\frac{\mathbb{E}\left[\#\mathsf{SP}_{\mathrm{CP}}^{\mathrm{Mix2}}\right]}{\mathbb{E}\left[\#\mathsf{SP}_{\mathrm{RKP}}^{\mathrm{Mix2}}\right]} = \frac{(5)}{(6)} \approx \frac{2^{d+\min(d,\delta_{\mathrm{in}})-2}}{2^{4d+\delta_{\mathrm{in}}-3n-2}} = 2^{3n-2d-\max(d,\delta_{\mathrm{in}})}\,. \tag{7}$$

The data complexity of an RKP mixture-differential attack with two pairs must therefore satisfy

$$d_{\mathrm{RKP}}^{\mathrm{Mix2}} \geq \frac{2d_{\mathrm{CP}}^{\mathrm{Mix2}} + 3n - \max\left(d_{\mathrm{CP}}^{\mathrm{Mix2}}, \delta_{\mathrm{in}}\right)}{4} \tag{8}$$

to apply an attack that was defined for the chosen-plaintext model.

3.4 Rectangle Attacks

Boomerang attacks [63] are adaptive plain-and-ciphertext attacks that trace quartets of plaintexts through a (partial) cipher, shift their ciphertexts by differences, and trace the so-shifted ciphertexts back through the inverse cipher. They were subsequently refined by multiple differentials in the middle [7], and adding a transition in the middle part [22,33]. The adversary splits the cipher $E_{\mathrm{dist}} = E_2 \circ E_m \circ E_1$, and exploits the existence of two trails $\Delta_x \xrightarrow{E_1} \Delta_u$ and $\Delta_v \xrightarrow{E_2} \Delta_y$, where the middle part undergoes a more precise evaluation of the map $\Delta_u \xrightarrow{E_m} \Delta_v$ for both pairs through the middle. The probability through the first part is usually approximated by

$$\Pr\left[E_1(K,X) \oplus E_1(K,X') \in \mathcal{D}_u\right] = 2^{-p}\,.$$

Amplified-boomerang [47] and rectangle attacks [7] turned adaptive two-way boomerangs into chosen-plaintext (or chosen-ciphertext) attacks. Given two input pairs (X,X') and $(\overline{X},\overline{X}')$ with $X \oplus X' \in \mathcal{D}_x$ that are mapped to (U,U') and $(\overline{U},\overline{U}')$ through E_1, the probability through the second part of a rectangle is usually approximated by

$$\Pr\left[E_2(K,U) \oplus E_2(K,\overline{U}) \in \mathcal{D}_y\right] = 2^{-q}\,.$$

When $\mathcal{D}_u$ and $\mathcal{D}_v$ are subspaces, the probability that $U, U', \overline{U}$, and $\overline{U}'$ are mapped to $V, V', \overline{V}$, and $\overline{V}'$, respectively, such that $V \oplus \overline{V} \in \mathcal{D}_v$ and $V' \oplus \overline{V}' \in \mathcal{D}_v$, is usually approximated as $2 \cdot |\mathcal{D}_v| \cdot 2^{-n}$ (there are two quartet options). Thus, the probability is approximated by

$$\Pr\left[K \twoheadleftarrow \mathbb{F}_2^k : \begin{cases} E_{\mathrm{dist}}(K,X) \oplus E_{\mathrm{dist}}(K,\overline{X}) \in \mathcal{D}_y, \\ E_{\mathrm{dist}}(K,X \oplus \Delta X) \oplus E_{\mathrm{dist}}(K,\overline{X} \oplus \Delta\overline{X}) \in \mathcal{D}'_y, \\ \Delta X, \Delta\overline{X} \in \mathcal{D}_x \end{cases}\right] \approx \frac{1}{2^{n+2(p+q)}}\,.$$

Chosen-plaintext Model. The adversary can partition the $D = 2^d$ plaintexts into structures. Define $2^{-\widehat{p}} =^{\text{def}} 2^{-2(p+q)}$. The number of quartets whose ciphertexts satisfy $C \oplus \overline{C} \in \mathcal{D}_{\text{out}}$ and $C' \oplus \overline{C'} \in \mathcal{D}'_{\text{out}}$ #SQ is approximately

$$\begin{aligned}\mathbb{E}\left[\#\mathsf{SQ}_{\text{CP}}^{\text{Rect}}\right] &= N_{\text{CP}} \cdot \left(\frac{|\mathcal{D}_{\text{out}}| \cdot |\mathcal{D}'_{\text{out}}|}{(2^n-1)\cdot(2^n-3)} \cdot (1-2^{-\widehat{p}}) + 2^{-\widehat{p}}\right) \\ &\approx \frac{2^{2(d+d')}}{4} \cdot \left(\frac{2^{\delta_{\text{out}}+\delta'_{\text{out}}}}{2^{2n}} + 2^{-\widehat{p}}\right) . \end{aligned} \tag{9}$$

Random Known-plaintext Model. In the RKP model, the adversary can again combine all data and obtain $N_{\text{RKP}} = 2 \cdot 3 \cdot \binom{2^d}{4} \approx 2^{4d-2}$ rectangle quartets from $D = 2^d$ texts. There are three ways of combining plaintexts into pairs and two ways of combining the ciphertexts into two pairs each. In contrast to mixtures, the pairs may employ different input-difference spaces $\mathcal{D}_{\text{in}}$ and $\mathcal{D}'_{\text{in}}$, where we define $\delta'_{\text{in}} = |\mathcal{D}'_{\text{in}}|$. One can expect that the number of surviving quartets with the initial filters on the plaintext side is

$$\begin{aligned}\mathbb{E}\left[\#\mathsf{SQ}_{\text{RKP}}^{\text{Rect}}\right] &= \frac{|\mathcal{D}_{\text{in}}| \cdot |\mathcal{D}'_{\text{in}}| \cdot N_{\text{RKP}}}{(2^n-1)\cdot(2^n-3)} \cdot \left(\frac{|\mathcal{D}_{\text{out}}| \cdot |\mathcal{D}'_{\text{out}}|}{(2^n-1)\cdot(2^n-3)} \cdot (1-2^{-\widehat{p}}) + 2^{-\widehat{p}}\right) \\ &\approx \frac{2^{\delta_{\text{in}}+\delta'_{\text{in}}+4d}}{2^{2n+2}} \cdot \left(\frac{2^{\delta_{\text{out}}+\delta'_{\text{out}}}}{2^{2n}} + 2^{-\widehat{p}}\right) . \end{aligned} \tag{10}$$

In the case of $\delta'_{\text{in}} = \delta_{\text{in}}$, it must hold in an RKP rectangle attack that

$$d_{\text{RKP}}^{\text{Rect}} \geq \frac{2d_{\text{CP}}^{\text{Rect}} + 2n + 2 - (\delta_{\text{in}} + \max\left(d_{\text{CP}}^{\text{Rect}}, \delta_{\text{in}}\right))}{4} \tag{11}$$

to apply an attack that was defined for the chosen-plaintext model.

3.5 DS-MitM Attacks

DS-MitM attacks [24] are key-recovery variants based on an observation from [38]. A sequence of partial outputs derived from a set of related texts can take considerably fewer values than a random sequence. An adversary maps a set of related texts through the distinguishing part of a cipher by guessing the required internal state and key variables, and stores the possible output sequences or sets in a table. From [34], the number of variables could be further reduced if the set is formed in relation to a pair that follows a differential trail. For round-reduced AES, attacks often start from a δ-set, with a set of 2^8 texts differing only in a single active byte. The adversary is interested in recovering the values or differences of the set in a byte or a linear combination of several bytes after the distinguisher [26,27,29,51,53]. The number of related texts, m, must be chosen such that the number of random matched byte sequences between entries in the offline table and candidates from the online phase is sufficiently small to filter the involved key bytes. More precisely, it should be lower than the product of the

number of surviving pairs and 2^b for b guessed key bits in the steps around the distinguisher. In the attacks above on round-reduced AES, this often required more than 16 related texts, and up to 256 texts in a δ-set that iterated over all values in a single input byte, as in [26,27,34,29].

Chosen-plaintext Model. The resulting chosen-plaintext attacks were efficient. An adversary could form up to $\binom{D}{2} \approx 2^{2d-1}$ pairs from $D =^{\text{def}} 2^d$ plaintexts. Given $|\mathcal{D}_{\text{in}}| = 2^{\delta_{\text{in}}}$, the number of pairs whose input difference is in $\mathcal{D}_{\text{in}}$, is calculated as for standard (truncated) differential attacks. Let $\mathcal{D}_{\text{out}}$ be an output difference space. For attacks that consider a distinguisher based on differential enumeration [34], $\mathcal{D}_{\text{out}} \subset \mathbb{F}_2^n$; otherwise, as in low-data chosen-plaintext attacks, $\mathcal{D}_{\text{out}}$ may equal $\mathbb{F}_2^n \setminus \{0^n\}$. Similarly to differential attacks, the number of pairs that survive the initial filter, #SP, i.e., the pairs whose ciphertext pairs satisfy $C \oplus C' \in \mathcal{D}_{\text{out}}$ for some space $\mathcal{D}_{\text{out}}$, can be calculated as:

$$\mathbb{E}\left[\#\mathsf{SP}_{\text{CP}}^{\text{DS}}\right] = \mathbb{E}\left[\#\mathsf{SP}_{\text{CP}}^{\text{Diff}}\right] . \tag{12}$$

Random Known-plaintext Model. For finding the initial pairs that satisfy the desired differential through the distinguisher, the adversary can use all data and obtain $N_{\text{RKP}} \approx 2^{2d-1}$ pairs from $D = 2^d$ texts. However, collecting the related data becomes significantly more challenging in the RKP model. Using 2^d random known plaintexts, the probability for a text to be in the available data is approximately 2^{d-n}. To ensure a high probability that sufficiently many related texts for a pair of plaintexts and ciphertexts that follow the trail in the middle rounds are available, 2^{d-n} must not be too small. However, we strive to avoid data complexities too close to the codebook (e.g. the one from the seven-round attack in [37], which was valuable as a pioneering result on AES) and set $d \leq n-4$ as an upper bound, which is challenging as the data has to be collected, stored, and more data lowers the meaningfulness of even academic attacks.
Using random data, the increasing data complexity will result in significantly more wrong pairs that must be tested, i.e., whose ciphertext difference lies in $\mathcal{D}_{\text{out}}$ when using differential enumeration. From an analysis of the configurations of the number of in- and output bytes in the offline trails, the complexity for testing the number of surviving pairs can more easily become the bottleneck of the attack. Thus, using only differential trails that reduce the number of surviving pairs may be beneficial for RKP attacks.
We introduce a time-data trade-off from a set of parameters θ_k-out-of-m related texts. In the offline phase, the adversary gathers the possible matches for m related texts, and computes and stores all $\binom{m}{\theta_k}$ sets. In the online phase, for every candidate pair that may satisfy the differential through the distinguishing part of the cipher, the adversary guesses the key material for one part of the cipher, usually at the beginning. Then, it derives the first m related texts, traces them back to their would-be plaintexts, and looks up which of them exists in the available collected data. If k texts for $k \geq \theta_k$ are available, the adversary constructs the set of the lexicographically first θ_k texts from them, and conducts

the attack on that subset of related texts. Thus, the attack requires that at least θ_k among m related texts are available in the given data, whose probability is

$$q_{\text{succ}} \stackrel{\text{def}}{=} 1 - \left(\sum_{k=0}^{\theta_k - 1} \binom{m}{k} \cdot (2^{d-n})^k \cdot (1 - 2^{d-n})^{m-k} \right) ,$$

which must be chosen to be non-negligible. As an alternative, the adversary could conduct the online phase first, as in [14,28]. The potential disadvantage of this approach is higher memory complexity if the number of surviving pairs in the online phase exceeds the number of entries to be stored in the offline phase. In the CP model, given a differential through the distinguisher with probability 2^{-p}, the adversary collects $2^{d_{\text{CP}}}$ texts that can be combined to $2^{d+d'-1} \geq 2^p$ pairs, so that at least one of them fulfills the differential, and approximately $2^{d+d'-1+\delta_{\text{out}}-n}$ survive on average. In the RKP model, it must hold

$$d_{\text{RKP}} \geq \frac{p + 1 + \delta_{\text{in}} - n}{2} . \tag{13}$$

However, the data complexity is determined mainly by the goal of obtaining a non-negligible probability that the related texts are available with respect to a pair that satisfies the differential through the distinguisher.

4 Six-round Rectangle Key-recovery Attack

We can formulate a six-round rectangle attack, as shown in Figure 1.

4.1 Attack in the Chosen-plaintext Model

When using the trails in a chosen-plaintext attack, the adversary can start from plaintext structures of two fixed active diagonals. The adversary searches for pairs $((P, C), (\overline{P}, \overline{C}))$ such that $C \oplus \overline{C}$ is active in only one inverse diagonal, cf. Figure 1. Eventually, it will search for pairs of such pairs, i.e., to combine them with other pairs $((P', C'), (\overline{P}', \overline{C}'))$ to form quartets.

The probability that a pair has only a single active diagonal after the first round in ΔW^1 is approximately $4 \cdot 2^{-48-48} \approx 2^{-94}$ for two pairs. Two such pairs are each mapped to a single active inverse diagonal in ΔZ^3. With a probability of approximately 2^{-32}, it holds that ΔZ^3 is the same for both pairs. From there, if $X^4 \oplus \overline{X}^4$ is active in only a single diagonal, which has probability $4 \cdot 2^{-96}$, the same difference holds automatically for $X'^4 \oplus \overline{X}'^4$. Then, both pairs are mapped to a difference ΔW^4 with a single active byte with probability $4 \cdot 2^{-24-24}$, which is guaranteed to be mapped to $C \oplus \overline{C}$ and $C' \oplus \overline{C}'$ having a single active inverse diagonal. Thus, the probability for a good quartet is approximately $2^{-46-48} \cdot 2^{-32} \cdot 2^{-94} \cdot 2^{-22-24} = 2^{-266}$, whereas it is $4 \cdot 2^{-96 \cdot 2} \approx 2^{-190}$ for a random pair. Given 2^7 plaintext structures of 2^{64} plaintexts yields $2^7 \cdot \binom{2^{64}}{2} \approx 2^{134}$ pairs (P, P') in total. Those can be combined to approximately

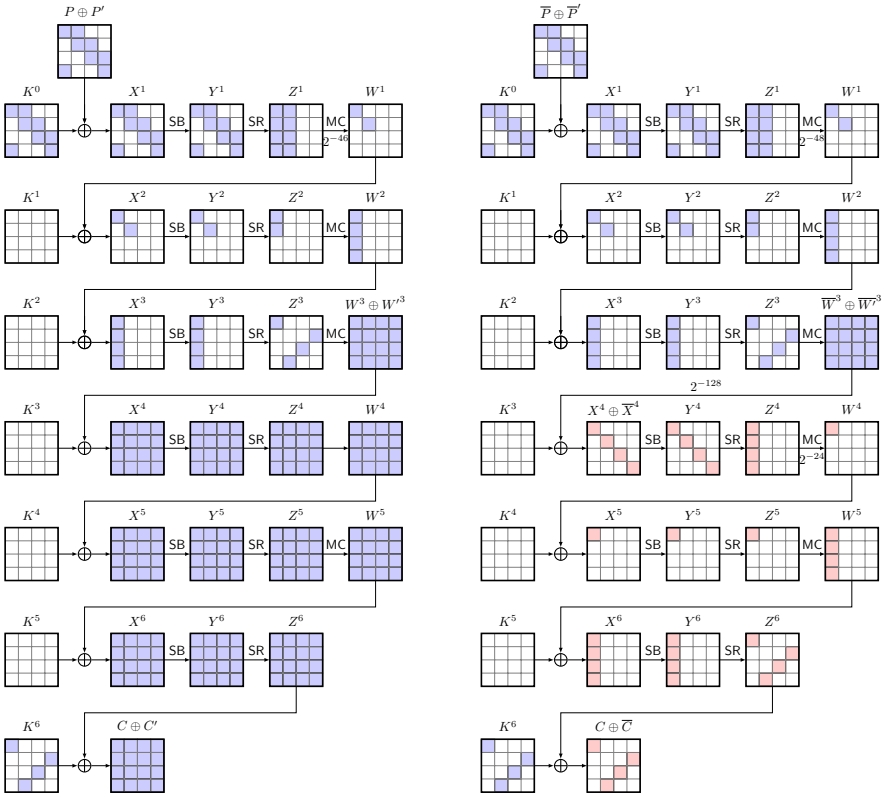


Fig. 1: Six-round rectangle key-recovery attack. Left: The trail between pairs (P, P') to (C, C') (blue). Right: The trail between $(\overline{P}, \overline{P}')$ to $(W^3, \overline{W}^3)$ at the top (blue) and the trail from $(X^3, \overline{X}^3)$ to $(C, \overline{C})$ (red). Colored bytes of plaintext states are active in the differential trails. Colored key bytes are guessed.

$2 \cdot \binom{2^{134}}{2} \approx 2^{268}$ quartets $(P, \overline{P}, P', \overline{P'})$. We expect 2^2 good quartets among $2^{268-190} = 2^{78}$ surviving quartets on average.

The attack starts with collecting the 2^{71} plaintext-ciphertext pairs and storing them four times into a table $\mathcal{T}$ indexed by the 96-bit values from all except one inverse ciphertext diagonal, which costs 2^{71} encryptions and $4 \cdot 2^{71} = 2^{73}$ lookups. Note that the memory can be reused and $2 \cdot 2^{71}$ states in one table suffice. Then, the adversay finds $4 \cdot \binom{2^{71}}{2} \cdot 2^{-96} \approx 2^{47}$ pairs $((P, C), (\overline{P}, \overline{C})$.

Among pairs that share the same single active inverse ciphertext diagonal, the adversary can search for quartets. It can discard quartets whose pairs do not suggest the same used bytes in K^6 and do not share the same plaintext structures. A good quartet consists of two pairs, so that the plaintexts in each pair originate from the same one of 2^7 plaintext structures. For all pairs, the adversary looks up $4 \cdot 2^8$ candidates K^6 that can lead to a single active byte in $Z^5 \oplus \overline{Z}^5$, and

stores the results in a table indexed by the keys and structure indices. This costs about $4 \cdot 2^{45} \cdot 4 \cdot 2^8 = 2^{58}$ memory accesses, and can reuse previous memory.
The probability that two random pairs suggest the same key K^6 is about $4 \cdot 2^8 \cdot 2^8 \cdot 2^{-32} = 2^{-14}$, and $2^{-7-7} = 2^{-14}$ that they share the same structure indices. Given four options for inverse ciphertext diagonals and four options for the active byte in ΔZ^5, we expect approximately $4 \cdot \binom{2^{45}}{2} \cdot 2 \cdot 4 \cdot 2^8 \cdot 2^8 \cdot 2^{-32} \cdot 2^{-7-7} = 2^{64}$ quartets. The factor two represents the fact that a quartet can be built by $((P, P'), (\overline{P}, \overline{P}'))$ and $((P, \overline{P}'), (\overline{P}, P'))$, which differ only in the structure indices. Identifying both needs one memory access when storing the seven-bit structure indices in order.
For each quartet, the adversary looks up the $4 \cdot 2^8$ keys $K^0[0, 5, 10, 15]$ that produce a single active byte in ΔW^1 with $2 \cdot 4 \cdot 2^{64} = 2^{67}$ lookups to a precomputed hash table. The number of surviving quartets is approximately $4 \cdot 2^8 \cdot 2^8 \cdot 2^{64} \cdot 2^{-32} = 2^{50}$. The adversary can repeat the procedure for the surviving quartets with $2 \cdot 2^{50} = 2^{51}$ lookups, and obtain $2^8 \cdot 2^8 \cdot 2^{50} \cdot 2^{-32} = 2^{34}$ mostly false-positive quartets on average, while preserving all good quartets.
Each surviving quartet produces a 96-bit key candidate. However, the positions of only the plaintext diagonals are fixed. Thus, quartets must suggest the same 64-bit key candidates $K^0[0, 3, 4, 5, 9, 10, 14, 15]$ from the active plaintext-difference diagonals. The correct key can usually be identified among a few key candidates (if any) suggested at least twice, with $\binom{2^{34}}{2} \cdot 2^{-64} \approx 2^3$ random collisions on average. The costs are 2^{34} encryption equivalents. For those, the adversary can guess 2^{64} options for the missing key bits of K^0, compute the key schedule to check if they transfer to the known values of K^6, and test the surviving candidates. This costs at most 2^{64} encryptions.
The attack complexity is dominated $2^{71} + 2^{64}$ encryption equivalents, and $2^{73} + 2^{58} + 2^{67} + 2^{51} + 2^{34} \approx 2^{73}$ memory accesses. Approximating the latter by a full encryption equivalent each yields $2^{73.4}$ encryption equivalents in total. It needs 2^{71} chosen plaintexts, and memory for 2^{72} AES states.

4.2 Conversion to the Random Known-plaintext Model

We use the same trails in the corresponding RKP attack. Here, the plaintext pair (P, P') can have two active diagonals. Thus, the probability that a pair (P, P') lies in $\mathcal{D}_{\text{in}}$ is approximately $6 \cdot 2^{64}/2^{128} \approx 2^{-61.4}$. The probability that two random ciphertexts $(C, \overline{C})$ have a difference in $\mathcal{D}_{\text{out}}$ is approximately $4 \cdot 2^{32}/2^{128} = 2^{-94}$. Given 2^d plaintexts, we can build $3 \cdot 2 \cdot \binom{2^d}{4} = 2^{4d-2}$ quartets to find a right quartet, given three options of combining plaintexts to pairs, and two options for forming a quartet from two ciphertext pairs. The probability of desired plaintexts to form quartets $(P, P', \overline{P}, \overline{P'})$ with both $P \oplus P'$ and $\overline{P} \oplus \overline{P}'$ to have the same two diagonals active is $6 \cdot 2^{64} \cdot 2^{64}/2^{2 \cdot 128} \approx 2^{-125.4}$. Since the probability of a right quartet is 2^{-266}, to obtain on average 2^3 right quartets, the adversary needs

$$d \geq (125.4 + 266 + 3 + 2)/4 \approx 99.1\,. \tag{14}$$

With $2^{99.1}$ random known plaintexts P, the adversary can form $2^{197.2}$ pairs $((P, C), (\overline{P}, \overline{C}))$. At the ciphertext side, it uses the four options for the single

active inverse diagonal to have a 94-bit filter, and therefore reduces the pairs $((P, C), (\overline{P}, \overline{C}))$ to $2^{103.2}$. Such pairs can be combined to $2 \cdot \binom{2^{103.2}}{2} = 2^{206.4}$ quartets; using the plaintext filter reduces this number to $2^{206.4} \cdot 2^{-61.4-64} \approx 2^{81}$ surviving quartets on average. For each such surviving quartet, the adversary can further filter whether the ciphertexts share the same (out of four options) single active inverse diagonal for reducing them to 2^{79}. The adversary identifies candidate keys K^0 and K^6, and discards quartets whose pairs do not suggest at least one shared key. This allows wrong quartets to survive with probability only approximately $4 \cdot 2^{8+8-32} \cdot 4 \cdot 2^{16+16-64} = 2^{-44}$, as before, to reduce them to 2^{35}. On average, the adversary can expect 2^3 good quartets.
In contrast to the CP attack, the active diagonals between quartets may also differ on the plaintext side. Thus, good quartets do not necessarily overlap in any active plaintext diagonals. However, approximating the number of good quartets as a random variable following a binomial distribution $\mathcal{B}(N, 2^{-p})$ with $N \approx 2^{269}$ and $2^{-p} \approx 2^{266}$, the cumulative probability for at least $k \geq 6$ good quartets is greater than 0.808. Since there exist only six options for the two active plaintext diagonals, there will be at least one collision of good quartets that suggest the same values for K^0. Over all quartets, we can expect $\binom{2^{35}}{2} \cdot 2^{-64} \cdot 1/6 \approx 2^{2.4}$ key collisions. For those, the adversary can guess 2^{64} options for the missing key bits of K^0, compute the key schedule to check if they transfer to the known values of K^6, and test the surviving candidates. Then, the adversary can guess 2^{32} options for the 32 missing key bits and test each key with one encryption each. The detailed steps are given in Appendix B. The attack needs

- $4 \cdot 2^{32+8} = 2^{42}$ memory accesses to get precomputed $4 \cdot 2^8$ key candidates $K^0[0, 5, 10, 15]$ for a given input diagonal difference, corresponding to the four possible positions of a single active byte in ΔW^1.
- $4 \cdot 2^{32+8} = 2^{42}$ memory accesses to get precomputed $4 \cdot 2^8$ key candidates $K^6[0, 7, 10, 13]$ for a given input diagonal difference, corresponding to the four possible positions of a single active byte in ΔZ^5.
- $2^{99.1}$ encryptions by the oracle.
- $4 \cdot 2^{99.1} = 2^{101.1}$ memory accesses to store the texts into four tables for the four options of a single active inverse diagonal in ΔZ^6, and to find $4 \cdot 2^{101.2}$ pairs $(P, \overline{P}, C, \overline{C})$. Note that the four options can define an outer loop.
- Furthermore, it uses in total $6 \cdot 4 \cdot 2^{101.2} \approx 2^{105.8}$ memory accesses to store the pairs into six tables, where a table uses the same two diagonals of both plaintexts from a pair as index, and the tables differ in the diagonal indices they use. For each position of the active inverse ciphertext diagonal, we expect $6 \cdot 2 \cdot \binom{2^{101.2}}{2} \cdot 2^{-64-64} \approx 2^{77}$ quartets, and 2^{79} quartets in total.
- $4 \cdot (2^{79+8} + 2^{79+8}) = 2^{90}$ memory accesses to get precomputed key candidates K^6 for the four options of the active byte, and two pairs of 2^{79} quartets. We expect $4 \cdot 2^{79+16-32} = 2^{65}$ surviving quartets.
- $4 \cdot (2^{65+8} + 2^{65+8}) = 2^{76}$ memory accesses to get key candidates for the first active plaintext diagonal, filtering the quartets down to $4 \cdot 2^{65+16-32} = 2^{51}$.
- $2 \cdot 2^{51+8} = 2^{60}$ memory accesses to get key candidates for the second active plaintext diagonal, filtering the quartets down to $2^{51+16-32} = 2^{35}$.

- 2^{35} memory accesses to find colliding key candidates at the plaintext side. For each surviving pair of quartets, the adversary guesses the missing 64-bit key with $2^{2.4} \cdot 2^{64} = 2^{66.4}$ encryptions on average.

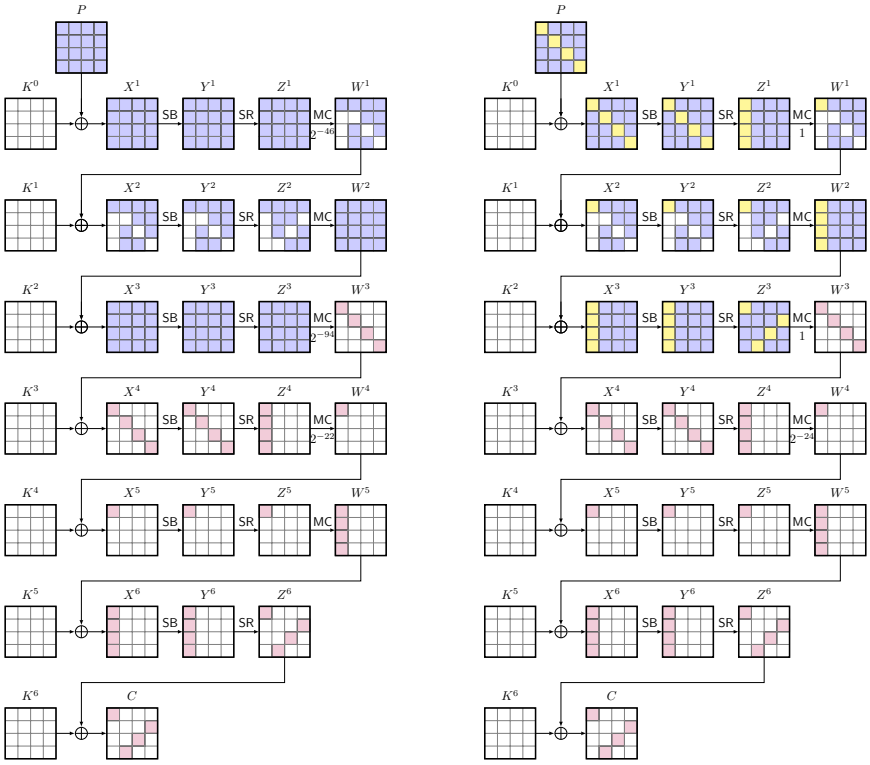
Randomized accesses to huge tables are considerably slower than to small tables that can be cached. We follow earlier works and approximate each memory access to huge tables with a full (here, six-round) encryption equivalent. The time complexity is $2^{42} + 2^{99.1} + 2^{105.8} + 2^{81} + 2^{90} + 2^{76} + 2^{60} + 2^{35} + 2^{66.4} \approx 2^{105.8}$ encryption equivalents. The data complexity is $2^{99.1}$ random known plaintext-ciphertext pairs, and the memory at most $2^{101.2} \cdot 4 = 2^{103.2}$ AES states.

5 Six-round Mixture Distinguisher

Next, we illustrate the transformation of a mixture-differential distinguisher to the random known-plaintext model. If we started from the six-round distinguisher by Bardeh and Rønjom from [5], which starts from pairs with three active plaintext diagonals, and is recalled in the Appendix D the resulting time, data, and memory complexities would be 2^{126} encryption equivalents, $2^{117.5}$ random known plaintexts, and $2^{118.5}$ state equivalents of memory. By choosing a different distinguisher, which is illustrated in Figure 2 and would require more time and data than the original result from [5] in the CP model, we can reduce time and data complexities in the RKP model.

5.1 Parameters in the Chosen-plaintext Model

In the chosen-plaintext model, the attack can start from plaintext structures that iterate over a fixed number of pairwise distinct random values (drawn with replacement) in each plaintext diagonal. In a structure, pairs require all plaintext-difference diagonals to be active. First, the adversary searched for pairs $((P, C), (P', C'))$ that follow the trail of Figure 2, i.e., whose ciphertext difference $C \oplus C'$ has only a single active inverse diagonal. Among those, it restricts its attention to those pairs (P, P') whose mixture pairs $(\overline{P}, \overline{P}')$, generated from exchanging the first plaintext diagonal of P and P', such that the difference $\overline{C} \oplus \overline{C}'$ had only a single active inverse diagonal. The probability that the first pair has only a single active diagonal after the first round is approximately $4 \cdot 2^{-48}$, and approximately $4 \cdot 2^{-96}$ to lead to a single active diagonal with $W^3 \oplus W'^3$ after three rounds. If that is the case, the mixed pair would follow the truncated differential trail having the same difference $\overline{W}^3 \oplus \overline{W'}^3$ with probability one. If the difference of both pairs reduces to a single active byte after the fourth round with probability about 2^{-22-22}, both are mapped to a ciphertext difference with a single active inverse diagonal after six rounds without a final MixColumns operation. Thus, the probability is approximately $2^{-46-94-44} = 2^{-184}$ for a pair and its mixture pair for the real six-round AES, and $2^{-94 \cdot 2} = 2^{-188}$ for a random permutation. The cardinalities of the plaintext- and ciphertext-difference spaces are $|\mathcal{D}_{\text{in}}| \approx 2^{128}$ and $|\mathcal{D}_{\text{out}}| \approx 4 \cdot 2^{32} = 2^{34}$, respectively. The attack needs 2^{93} chosen plaintexts to form approximately 2^{185} pairs in 2^{184} mixture structures.



(a) The first trail of the six-round mixture distinguisher.

(b) The related second trail of the six-round mixture distinguisher.

Fig. 2: The mixture trails on six-round AES. Colored bytes are active in the differential trails. Both figures show two differentials between two related pairs: one differential with all active bytes for one pair, and a differential where one diagonal (colored in red) is exchanged to form a second pair.

5.2 Parameters in the Random Known-plaintext Model

In the RKP model, 2^d plaintexts can form $\binom{2^d}{4} \approx 2^{4d}/24$ quartets, that can be partitioned into two pairs in three distinct ways, with the single exchanged plaintext diagonal could be in either of the two plaintexts in the second pair, yielding approximately 2^{4d-2} mixture quartets $(P, P', \overline{P}, \overline{P}')$. We have expanded the plaintext-difference space to $|\mathcal{D}_{\text{in}}| = 2^{128}$. The probability that $\overline{P}, \overline{P}'$ is a diagonal mixture from (P, P') exchanging the first diagonal is approximately $2^{-128} \cdot 2^{-128} = 2^{-256}$. Moreover, we consider only pairs with $b = 20$ plaintext bits at fixed (choosable by the adversary) positions in the three non-exchanged diagonals to be equal in P and P', which then automatically holds also in $\overline{P}, \overline{P}'$ when they are a mixture pair of (P, P'). To have 2^{184} mixture quartets, we need $d \geq (2 + 256 + 20 + 184)/4$, which yields $d \geq 115.5$.

Algorithm 1 Six-round RKP mixture distinguisher.

function $\mathcal{A}$
  Collect $2^{115.5}$ known plaintext-ciphertext pairs (P, C)
  Store them into a table $\mathcal{Q}$
  for all active inverse ciphertext diagonal indices $j \in \{0, 1, 2, 3\}$ **do**
    Store the pairs into a new table $\mathcal{C}$ indexed by the inverse diagonals of C except for the i-th diagonal and the 20 fixed bits in P
    Identify the approximately $4 \cdot \binom{2^{115.5}}{2} \cdot 2^{-116} \approx 2^{116}$ pairs $((P, C), (P', C'))$
    for all pairs **do**
      for all plaintext diagonal indices $j \in \{0, 1, 2, 3\}$ **do**
        if mixture pair $(\overline{P}, \overline{P}')$ with the j-th plaintext diagonal is in $\mathcal{Q}$ **then**
          if $\overline{C} \oplus \overline{C}'$ has only one active inverse diagonal **then**
            return "real"
  return "random"

The attack steps are detailed in Algorithm 1. The attack can use a table $\mathcal{Q}$ to store the $2^{115.5}$ plaintext-ciphertext pairs (P, C), and then re-index them by the 20 selected bits in P and the values in C in all but the j-th inverse diagonal for all j. Hence, the adversary can identify $4 \cdot \binom{2^{115.5}}{2} \cdot 2^{-20-96} \approx 2^{116}$ pairs $((P, C), (P', C'))$, and look up in $\mathcal{Q}$ whether the mixture pair is also given. Since this covers two ways of matching P, P' to $\overline{P}, \overline{P}'$ or the latter pair in reverse order, and since 20 fixed bits are equal between $\overline{P}$ and $\overline{P}'$ and can be used only once for matching, the adversary can find approximately $\binom{2^{116}}{2} \cdot 2 \cdot 2^{-128-128+20} \approx 2^{-4}$ wrong quartets, but one correct quartet following the trail on average. The attack requires $2^{115.5}$ encryptions and $4 \cdot 2^{115.5} + 2^{116} = 2^{117.94}$ memory accesses to huge tables. Approximating the latter by one six-round encryption equivalent, this yields $2^{118.1}$ encryption equivalents. The attack requires $2^{115.5}$ random known plaintexts and $2^{115.5} \cdot 2 \cdot 2 = 2^{117.5}$ state equivalents of memory.

6 Six-round Truncated-differential Distinguisher

Bao et al. [3] as well as Grassi and Rechberger [40,41] developed truncated-differential distinguishers with very small bias. The works differed in their perspective: the latter two works laid out statistical foundations and focused on many angles of the probabilities of truncated differentials through five-round AES. In contrast, Bao et al. came from the perspective of extending the integral attack to a probabilistic four-round differential, which they extended to five and six rounds thereafter. Rønjom [61] developed a generic framework that uses iterated transition matrices to compute the sum of probabilities over all possible trails, which matched the theoretical results. All those works approximated the AES to have an ideal S-box. Chang et al. [20] refined Bao et al.'s results by studying the distribution of trails with the AES S-box, showing that the bias for the real AES (under random-key assumptions) was larger than assumed in [3].

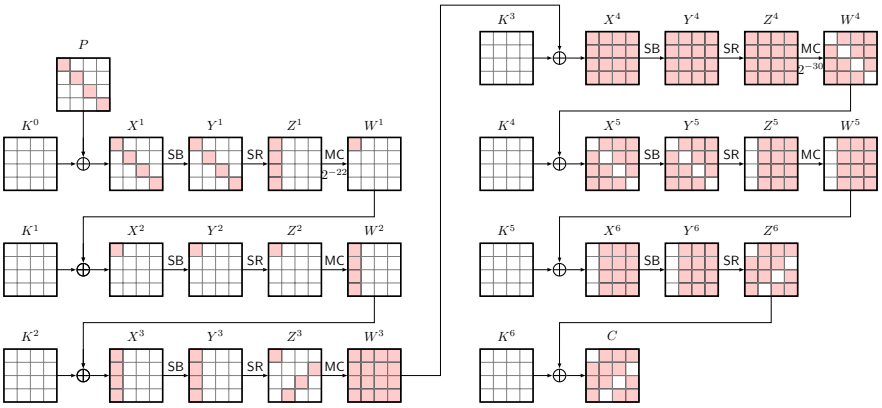


Fig. 3: The six-round truncated-differential distinguisher [3,20] that we transfer to the RKP model. Colored state cells represent active bytes in the trails.

6.1 Evaluating the Probability

Chang et al. derived separate average probabilities p_{i_s,i_e} for all combinations of a starting-byte index i_s to an ending inactive byte index i_e that take the internals of the AES S-box into account. Under the simplifying assumption that inactive bytes in different columns were independent, we can derive the probabilities that a pair with a single active byte in the first column is mapped to a pair through four rounds of AES with the first inverse diagonal being inactive is approximately

$$\begin{aligned}&(2^{-8}+2^{-31.982})\cdot(2^{-8}+2^{-31.774})\cdot(2^{-8}+2^{-31.988})\cdot(2^{-8}+2^{-31.816})\\ \approx\ &2^{-32}+2^{-53.887}\,.\end{aligned}$$

The probability of having the output diagonals 1, 2, or 3 inactive, is approximately $2^{-32}+2^{-53.797}$, $2^{-32}+2^{-53.868}$, and $2^{-32}+2^{-53.792}$, respectively. Moreover, the probabilities of starting from other input diagonals are permutations of those four probabilities. In contrast, for a random permutation, the probability of an inactive inverse diagonal in the ciphertexts is approximately

$$p_{\text{rand}} = (2^{96}-1)/(2^{128}-1) \approx 2^{-32}-2^{-128}\,.$$

6.2 Statistical Framework

We use the statistical framework used in [3,40,41]. Assuming that the normal distribution approximates the binomial distribution sufficiently well for large values of N, the number of independent Bernoulli trials (corresponding to the number of pairs in a truncated-differential attack) N has to satisfy

$$N \geq \frac{2\left(p_{\text{real}}(1-p_{\text{real}}) + \frac{\sigma_{\text{real}}}{\sigma_{\text{rand}}}\cdot p_{\text{rand}}(1-p_{\text{rand}})\right)}{(p_{\text{real}}-p_{\text{rand}})^2}\cdot \mathsf{erfinv}(2p_{\text{succ}}-1)^2\,, \qquad (15)$$

where erfinv denotes the inverse error function and p_{succ} the desired success probability; for $p_{\text{succ}} = 0.95$, it holds that $\mathsf{erfinv}(2 \cdot p_{\text{succ}} - 1)^2 \approx 1.353$. When considering all pairs from all values of a single input diagonal as in [3,40,41], the variance ratio of real five-round AES to a random permutation is beyond one, and up to $\frac{\sigma_{\text{real}}}{\sigma_{\text{rand}}} \approx 16$ when starting from mixture structures as in [40,41]. However, this ratio is approximately one when random pairs are considered.

6.3 Attack Parameters in the Chosen-plaintext Model

Each structure of 2^{32} texts guaranteed a ratio of approximately $4 \cdot (2^8 - 1)/(2^{32} - 1)$ pairs with a single active byte after the first round that are assumed to follow the differential, whereas the remaining pairs were assumed to follow the transition with the probability of a random permutation. Thus, the probability through six rounds of AES could be approximated by

$$p_{\mathsf{AES},6} \approx \left(4 \cdot \frac{2^8 - 1}{2^{32} - 1}\right) \cdot p_{\mathsf{AES},5} + \left(1 - 4 \cdot \frac{2^8 - 1}{2^{32} - 1}\right) \cdot p_{\text{rand}} \,.$$

When we consider random pairs close to uniformly distributed over all input diagonals and output inverse diagonals, we obtain an average probability

$$p_{\mathsf{AES},5} \approx 2^{-32} + 2^{-53.835} \,.$$

When considering any active inverse diagonal in the output, we obtain approximately $p_{\mathsf{AES},5} \approx 2^{-30} + 2^{-51.833}$ in contrast to $p_{\text{rand}} \approx 2^{-30} - 2^{-61.415}$. Then,

$$p_{\mathsf{AES},6} \approx 2^{-30} + 2^{-74.058} \,.$$

For $\frac{\sigma_{\text{real}}}{\sigma_{\text{rand}}} \approx 16$, one obtains $N \geq 2^{121.64}$ pairs, yielding $2^{58.64}$ structures of 2^{32} chosen plaintexts each, or $2^{90.64}$ chosen plaintexts. For $\frac{\sigma_{\text{real}}}{\sigma_{\text{rand}}} \approx 1$, $N \geq 2^{118.55}$ pairs, yielding $2^{55.55}$ structures of 2^{32} plaintexts each, or $2^{87.55}$ chosen plaintexts.

6.4 Parameters in the Random Known-plaintext Model

For the truncated-differential distinguisher, we use the evaluation of the biases of differential trails through round-reduced AES from their work in Table 4. The cardinalities of the plaintext- and ciphertext-difference spaces of the adapted attack are $|\mathcal{D}_{\text{in}}| \approx 2^{34}$ and $|\mathcal{D}_{\text{out}}| \approx 2^{98}$. In the RKP model, we obtain $2^{2d-1-n+\delta_{\text{in}}}$ pairs in the desired plaintext space from 2^d plaintexts. That is, we can use a slightly larger plaintext space than in the original chosen-plaintext attack, as all plaintext pairs with a single active diagonal can lead to a single active byte. To have $N \geq 2^{118.55}$ pairs that start from a single active diagonal, we need

$$2^{2d-1-n+\delta_{\text{in}}} = 2^{118.55} \,,$$

which yields $d = 106.8$, i.e., $2^{106.8}$ random known plaintexts that can form approximately $2^{212.6} \geq 2^{118.55+94}$ pairs, as in the original attack. The steps of the attack are given in Algorithm 2. The computational complexity consists of

Algorithm 2 Six-round RKP truncated-differential distinguisher.

function $\mathcal{A}$
  Collect $2^{106.8}$ known plaintext-ciphertext pairs (P, C)
  Store them into a table $\mathcal{Q}$
  count $\leftarrow 0$
  for all active plaintext diagonal indices $i \in \{0, 1, 2, 3\}$ **do**
    for all active inverse ciphertext diagonal indices $j \in \{0, 1, 2, 3\}$ **do**
      Store the plaintext-ciphertext pairs (P, C) in a second table $\mathcal{T}$ indexed by all but the i-th diagonals of P and the j-th inverse diagonals of C
      for all colliding pairs **do**
        count $\leftarrow$ count $+ 1$
  if count $\geq 2^{88.6} - 2^{56.2}$ **then**
    return "real" ▷ We expect count $\approx 2^{88.6} + 2^{44.6}$ on average
  return "random" ▷ We expect count $\approx 2^{88.6} - 2^{57.2}$ for a random permutation

- $2^{106.8}$ encryptions,
- $2 \cdot 4 \cdot 4 \cdot 2^{106.8} \approx 2^{111.8}$ memory accesses to tables, and
- $2^{88.6} + 2^{44.6}$ increment operations on average for six-round AES.

Approximating each lookup into a huge table as a full encryption equivalent, we obtain $2^{106.8} + 2^{111.8} + 2^{88.6} + 2^{44.6} \approx 2^{111.9}$ encryption equivalents. The memory complexity is given by $2^{106.8} \cdot 3 \approx 2^{108.4}$ state equivalents.

7 Impossible-differential Attack on Seven-round AES-128

Here, we consider the attack by Leurent and Pernot [49,50], who slightly improved the attack by Boura et al. [18,19].

7.1 Original Attack Parameters

The cardinalities of plaintext- and ciphertext-difference spaces are $|\mathcal{D}_{\text{in}}| \approx 2^{64}$ and $|\mathcal{D}_{\text{out}}| \approx 2^{32}$ as the key-recovery procedure fixes the spaces of interests. The attack uses structures of 2^{64} chosen plaintexts that produced approximately $2^{2\delta_{\text{in}}-1} \approx 2^{127}$ pairs. Given an initial filter of $2^{\delta_{\text{out}}-n} \approx 2^{-96}$, there are on average

$$2^{s_p} = 2^{2\delta_{\text{in}}-1+\delta_{\text{out}}-n} \approx 2^{31}$$

surviving pairs per structure. There are $c_{\text{in}} = 46$ bit conditions on the plaintext-side guessing trail and $c_{\text{out}} = 22$ bit conditions on the ciphertext-side guessing trail. Moreover, there are $k_{\text{in}} = 80$ key bits and $k_{\text{out}} = 32$ key bits involved, with a union of $k_{\text{atk}} = k_{\text{in}} \cup k_{\text{out}} = 112$ key bits, which has a gap of $a = k - k_{\text{atk}} = 128 - 112 = 16$ bits. Therefore, the number of surviving pairs 2^d must satisfy

$$\left(1 - 2^{-c_{\text{in}}+c_{\text{out}}}\right)^{2^d} \leq 2^{-a}$$

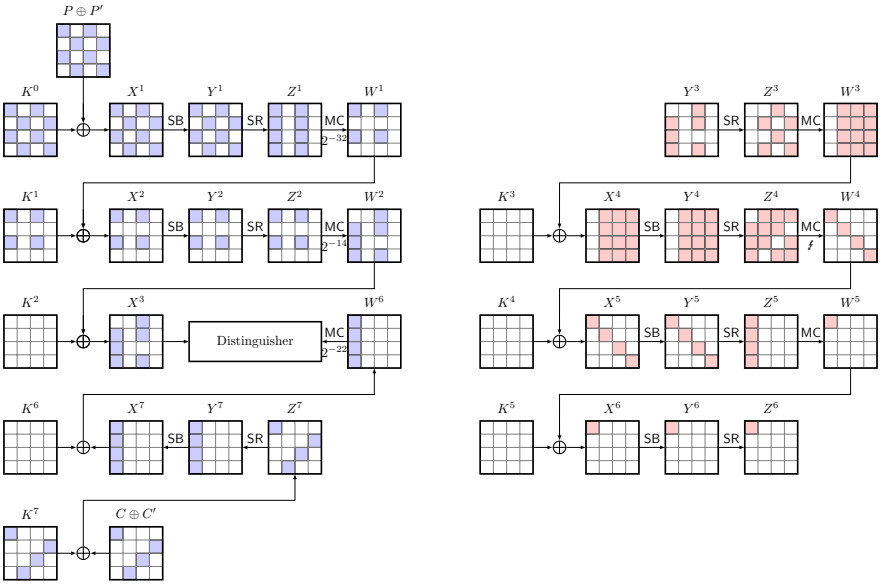


Fig. 4: Seven-round impossible-differential attack on AES-128. Left: The online key-recovery phase. Right: The impossible-differential distinguisher. Colored state bytes are active in the differential trails. Colored key bytes are filtered.

which yields $2^d \geq 2^{71.5}$ and thus $2^s = 2^d/2^{s_p} = 2^{71.5-31} = 2^{40.5}$ structures. This corresponds to $2^{s+\delta_{\text{in}}} = 2^{40.5+64} = 2^{104.5}$ chosen plaintexts to have a bits of advantage. Finally, the attack can guess the remaining a bits not covered in the attack to recover the full key with one encryption per remaining full candidate in $2^{k-a} = 2^{112}$ encryptions on average.

7.2 Parameters in the Random Known-plaintext Model

In the RKP model, pairs occur randomly in the desired plaintext space. Given 2^p plaintexts, we can form 2^{2p-1} pairs. From those, approximately

$$2^{2p-1+\delta_{\text{in}}+\delta_{\text{out}}-2n} = 2^{2(p-n)-1+\delta_{\text{in}}+\delta_{\text{out}}}$$

pairs are surviving pairs, i.e., lie in the desired plain- and ciphertext spaces. Thus, to have the same number of surviving pairs 2^d as before, we need

$$\begin{aligned} 2(p-n) - 1 + \delta_{\text{in}} + \delta_{\text{out}} &= d \\ ((2n+1) - \delta_{\text{in}} - \delta_{\text{out}} + d)/2 &= p\,, \end{aligned}$$

which yields $p = (257 - 96 + 72)/2 = 116.5$. Therefore, the attack needs $2^{116.5}$ random known plaintext-ciphertext pairs on average.
The attack collects and stores them, that is $2^{117.5}$ plaintext-ciphertext states, and identifies approximately $\binom{2^{116.5}}{2} \cdot 2^{-64-96} = 2^{72}$ surviving pairs. For each, it

can use a few table lookups to identify $2^{112} \cdot 2^{-68} = 2^{44}$ keys that it can discard on average. For that purpose, it has a table $\mathcal{K}$ of 2^{112} bits that represent the key candidates, or 2^{105} AES state equivalents. Thus, the time complexity consists of

- $2^{116.5}$ encryptions,
- $2^{116.5}$ lookup and store accesses each into a huge table,
- $2^{72} \cdot 8$ lookups into smaller tables whose size is negligible,
- $2^{72} \cdot 2^{44}$ bit sets into the huge table $\mathcal{K}$,
- 2^{112} sequential lookups to find surviving keys,
- and at most 2^{100} final encryptions,

which can be upper bounded by $2^{116.5}+2^{100} \approx 2^{116.5}$ encryptions and $2^{117.5}+2^{72} \cdot 8+2^{72+44}+2^{112} \approx 2^{117.5}$ memory accesses into huge tables. Approximating one lookup by one encryption equivalent produces a total of $2^{116.5}+2^{117.5} \approx 2^{118.1}$ encryption equivalents. The memory complexity is at most $2^{117.5}+2^{105} \approx 2^{117.5}$ state equivalents.

7.3 Further Applications to Round-reduced AES-192 and AES-256

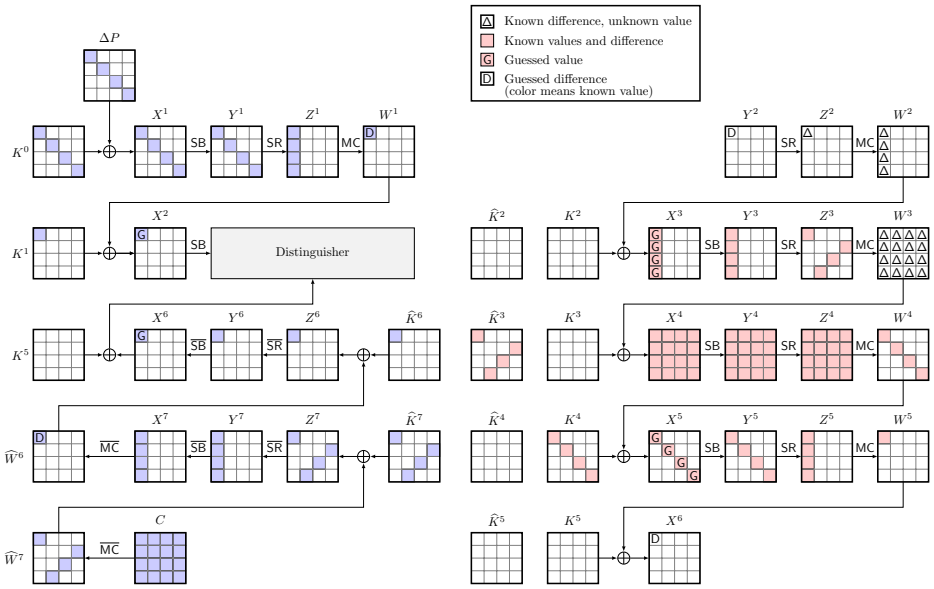
In the Appendix G, we provide two further impossible-differential attacks in the RKP model on seven rounds of AES-192 and eight rounds of AES-256, respectively. Both improve over the decade-old six-round zero-correlation attack from [13] and the six-round meet-in-the-middle attack from [15] that, to the best of our knowledge, have remained the best attacks on those AES variants in the RKP model so far. The attack on seven-round AES-192 uses a similar trail as the attack on seven rounds of AES-128, which led to lower overall complexities compared to the earlier results from [52].
For AES-256, Kara had proposed an updated eight-round attack in the RKP model [46]. We remark on the complexity estimation in the Appendix H arguing that the data complexity seems not to suffice. There, we also note that his attack may be salvaged straightforwardly but leads to higher data and time complexities than our result, as we can replace its trail with the better one from [54].

8 DS-MitM Attack on Seven-round AES-128

In the following, we illustrate with an adaptation of the attack on seven-round AES-128 that also attacks with various related texts, such as DS-MitM attacks, can be transformed to the RKP model.

8.1 Attack in the Chosen-plaintext Model

The original attack uses several differential trails: from two active bytes after the first round to a single active byte after five rounds, and from one active byte after the first round to two active bytes in the same column after five rounds. We expect the reader to be familiar with the attack on seven-round AES-128 from [27] or refer to it for details.



(a) The outer rounds of the online filtering and key-recovery phase.

(b) Distinguisher trail.

Fig. 5: Key-recovery rounds and distinguisher for one of 2^6 differential trails in the distinguisher from the DS-MitM attack on seven-round AES-128. Colored and marked bytes are active (different between texts).

8.2 Attack in the Random Known-plaintext Model

In contrast to the original attack, in the RKP model, the related texts are not guaranteed to be part of the available data. We conduct the online phase first and store the results before conducting the offline phase. From an analysis of the configurations of the number of in- and output bytes in the offline trails, the complexity for testing the number of surviving pairs easily becomes the bottleneck. Dong et al.'s value constraints can be used at the end to reduce the number of available pairs. Since the adversary must keep track of plaintext-ciphertext pairs, storing and testing for pairs of 2-to-1 trails can easily become the bottleneck of the attack. Thus, we consider only 1-to-1 trails, i.e., truncated differentials from a single active byte in ΔY^2 to a single active byte in ΔX^6.

Online Phase. The online phase differs from that in [27] in several aspects:

1. There are no plaintext structures.
2. It uses only trails from a single active byte to a single active byte.
3. It further restricts the attack to trails from a single active byte to a single active byte from the first active plaintext diagonal and the first column of $\Delta\widehat{W}^7 = \mathsf{MC}^{-1}(\Delta C)$.

The adversary observes 2^d plaintext-ciphertext known pairs and stores them in a table corresponding to the values of the plaintexts in all diagonals except the fixed (e.g., the first) diagonal and the values of ciphertexts without the fixed inverse diagonal of $\widehat{W}^7$. It identifies $\binom{2^d}{2} \cdot 2^{-96-96} \approx 2^{2d-193}$ pairs of plaintext-ciphertext pairs. For each pair, the adversary guesses all possible indices of the single active byte position after the first round and after six rounds, and its approximately 2^8 possible differences. It uses precomputed hash tables to obtain the approximately 2^{10} corresponding possible keys.
We constrain the active byte in $\Delta Y^2 = Y^2 \oplus Y'^2$ to α differences, in the manner introduced by Dong et al. [29]. If the constraint is not satisfied, we discard the current pair-key combination. Similarly, we restrict the 255 options to β values for the difference in the active byte in $\Delta X^6 = X^6 \oplus X'^6$.
From ΔY^2, the adversary iterates over the first m nonzero differences of the active byte, computes backwards to their corresponding plaintexts, looks up the available plaintexts in the data, and identifies the lexicographically first k available ciphertexts among those. If $k < \theta_k$, i.e., the number of available plain- and ciphertexts among the first m nonzero differences is below a threshold, it discards the current pair. Then, for the first k found plaintexts among the first m plaintexts, it looks up the corresponding ciphertexts. It guesses the difference in the possible single active byte in $\Delta\widehat{W}^6$, and looks up the 2^{10} would-be values of the corresponding active inverse diagonal of Z^7 from precomputed hash tables to derive the bytes of $\widehat{K}^7$. It guesses the 2^8 key candidates for the active byte in ΔX^6. For all pairs with $k \geq \theta_k$, the adversary computes from the ciphertexts to the active byte in X^6 for the current key guess and derives the differences to the fixed text for all k further texts. It uses the differences of the lead pair at ΔX^6 as well as the differences of the texts at $X_i^6[j] \oplus X^6[j]$ for the first $i = 1..k$ texts at the active byte into a table $\mathcal{T}_{\mathsf{online}}$, indexed by that vector and the indices of the first available k texts, $i_1, \ldots, i_k$, and stores a reference to the plaintext pair, and the induced keys, and all other given byte differences ΔX_i^6 for $i > k$ and their indices into the table entry.

Offline Phase. The offline phase works similarly as in the attack from [27] with several differences: Using row-wise rotational symmetry that was not observed by [27], the procedure needs only 16 tables and trail computations instead of 2^6 to capture all $4 \cdot 4$ combinations of active byte positions in the first column of X^2 and positions in the first diagonal of X^6.
The adversary restricts the difference in the active byte of ΔY^2 to $\alpha \geq 1$ fixed chosen non-zero eight-bit constants. Similarly, it restricts the difference in the active byte in ΔX^6 to $\beta \geq 1$ fixed chosen non-zero eight-bit constants. The adversary traces the approximately 2^8 related texts through the offline phase. Afterwards, it first extracts all $\binom{m}{k}$ combinations of k out of the first m differences to the first text. Let $i_1, \ldots, i_k$ be the k out of $\{1, \ldots, m\}$ indices. For all those combinations, it looks up the string of $\Delta X^6, \Delta X_1^6, \ldots, \Delta X_k^6, i_1, \ldots, i_k$ if it is stored in $\mathcal{T}_{\mathsf{offline}}$. For all matches, it uses the remaining available texts in the table entry to further filter matches. All matches that survive this stage produce a key candidate: 32 bits of $K^0[0, 5, 10, 15]$, 32 bits of $\widehat{K}^7[0, 7, 10, 13]$, eight bits

of K^1, eight bits of $\widehat{K}^6$, as well as 32 bits of $\widehat{K}^3$, and 32 bits of K^4. Even with the key-schedule representations by [49,50], it is not easy to filter them further. However, the adversary can collect them in a table and count the occurrences. Finally, it outputs the key candidates with the highest counts. The detailed steps are provided in the Appendix F.

Online-phase Complexity. The probability of pairs to fulfill the trails from one active plaintext diagonal to a single active ciphertext diagonal is approximately $2^{-96} \cdot 2^{-96} = 2^{-192}$. Given 2^d data, we expect 2^{2d-193} pairs that fulfill this filter. The probability of having a single active byte in the first diagonal after the first round and in the first column after five rounds is $4 \cdot 2^{-120} \cdot 4 \cdot 2^{-120} = 2^{-236}$. We expect 2^{2d-237} correct pairs, i.e., pairs that follow the offline-phase trail, and to have a fraction of approximately 2^{-44} of correct pairs that satisfy the first filter. The complexity of the online phase consists of 2^d encryptions and 2^d lookup-and-store memory accesses into a huge table. Next, the adversary identifies 2^{2d-193} pairs. For each of them, it guesses $4{\cdot}255 \approx 2^{10}$ keys on average, or analogously the α differences in ΔW^1, and encrypts the pairs through the first two rounds. For the non-discarded pairs, it derives 2^8 texts and traces them backwards through the inverse first two rounds. Using 2^{-4} seven-round encryption equivalents as [27] did, this yields $\alpha \cdot 2^{2d-193+10-4} = \alpha \cdot 2^{2d-187}$ encryption equivalents. For each, it computes up to $\alpha \cdot 2^{2d-187} \cdot 2^8 \cdot 2^{-4} = \alpha \cdot 2^{2d-183}$ re-decryptions and needs up to $\alpha \cdot 2^{2d-179}$ memory accesses to a large table to obtain the plaintexts and their corresponding ciphertexts. Next, the adversary takes the $\alpha \cdot 2^{2d-187}$ entries, obtains 2^{10} keys from hash tables for $\widehat{K}^6$, and traces the up to 2^8 texts two rounds backwards to the β differences of ΔX^6 and derives the differences. This costs at most $\alpha \cdot \beta \cdot 2^{2d-187} \cdot 2^{10} \cdot 2^8 \cdot 2^{-4} = \alpha \cdot \beta \cdot 2^{2d-173}$ encryptions, and $\alpha \cdot \beta \cdot 2^{2d-169}$ accesses for the $\alpha \cdot \beta \cdot 2^{2d-169}$ entries on average.
For $d = 124$, $k = 9$, and $m = 144$, we obtain 2^{11} correct pairs and 2^{55} wrong pairs after the first filter, and can expect $2^{11-3} = 2^3$ correct pairs when $\alpha = 1$ and $\beta = 255$, i.e., we have an eight-bit constraint at ΔY^2 and none at ΔX^6. The online phase needs approximately

$$2^d + 2^{2d-187} + 2^{2d-183} + 2^{2d-165} = 2^{124} + 2^{61} + 2^{65} + 2^{83} \approx 2^{124}$$

encryption equivalents and

$$2^d + 2^{2d-179} + 2^{2d-187} + 2^{2d-161} = 2^{124} + 2^{69} + 2^{61} + 2^{87} \approx 2^{124}$$

memory accesses.

Offline-phase Complexity. The offline phase computes 16 tables, and traces up to 2^8 texts for $\alpha{\cdot}\beta{\cdot}2^{8\cdot8}$ guesses of internal values with 2^{-4} encryption equivalents per text, i.e., $\alpha \cdot \beta \cdot 2^{72}$ encryption equivalents for $\alpha \cdot \beta \cdot 2^{68}$ table entries in total. For each, it extracts the first $\binom{m}{k}$ options with 2^{-4} matches of the table options with $\alpha \cdot \beta \cdot 2^{68} \cdot \binom{m}{k}$ lookups. For $\alpha = 1$ and $\beta = 255$, we have 2^{76} table entries, out of which a fraction of 2^{-4} applies to a position (i, i') in ΔY^2 and ΔX^6

$$(2^{2d-169-4} \cdot 2^{76-4}) \cdot 2^{-4\cdot(k+1)} \cdot \binom{m}{k} \quad \text{matches} \tag{16}$$

on average. Here, the $k+1$ indicates the difference in a byte in $\Delta X^6[i']$ for the reference pair and k further texts. For each match, it uses the remaining matches to filter the options further. $d = 124$, $k = 9$, and $m = 144$, $\binom{m}{k} \approx 2^{45.7}$, so that the offline phase needs approximately 2^{80} encryptions as well as $2^{76} \cdot 2^{45.7} \approx 2^{121.7}$ memory accesses. From Equation (16), the adversary can expect

$$2^{2 \cdot 124 - 169 - 4} \cdot 2^{76-4} \cdot 2^{-10 \cdot 8} \cdot 2^{45.7} \approx 2^{112.7} \quad \text{matches.}$$

Given 2^{124} distinct random plaintexts, the probability for any text to be available is 2^{-4}. From 256 texts in a δ-sequence, the adversary can expect $256 \cdot 2^{-4} = 16$ texts on average, i.e., seven further bytes in the 2^8 related texts to be available that have not been used before, with another filter of 2^{-56} to obtain $2^{112.7-56} \approx 2^{56.7}$ surviving matches. For each, it stores the union of key information from K^0 and $\widehat{K}^7$ into a table. Given 2^3 correct pairs on average, we can expect the correct key candidate of the 64 bits of K^0 and K^7 to be the only one that is suggested at least four or more times.
To sum up, the phases of the attack combined need 2^{124} random known plaintexts, negligible memory for the hash tables and keys, but memory for 2^{124} plaintext-ciphertext pairs, and 16 would-betables with 2^{64} entries at a time each, or 2^{76} entries. Thus, the memory is dominated by 2^{125} state equivalents for finding ciphertext pairs in the online phase. The attack requires $2^{124} + 2^{80} \approx 2^{124}$ encryption equivalents and $2^{124} + 2^{121.7} + 2^{112.7} + 2^{56.7} \approx 2^{124.3}$ memory accesses. Approximating each memory access to huge tables by one seven-round encryption, the complexity is $2^{124} + 2^{124.2} \approx 2^{125.1}$ encryption equivalents.

9 Conclusion

We revisited and extended the set of relations between differential-based attacks on block ciphers in the chosen- and in the random known-data model. With the AES as probably the most widely deployed and used block cipher and therefore a highly relevant example, we showed explicit adaptations of some of the best distinguishers and attacks to the RKP model, improving over the best attacks on all versions of the AES in that model. The number of required related texts in DS-MitM attacks had made conversions from the CP to the RKP model seem impractical, up to the use of (almost) the full codebook. However, we demonstrated that a probabilistic approach allows for mounting DS-MitM attacks in the RKP model. Our attacks may seem similar to statistical methods, such as statistical integrals [64]. However, while they allowed for reducing the data complexity of integral attacks, their technique still assumed chosen-data adversaries. Naturally, the high complexities render all our attacks purely academic and no threat to the security of the full-round AES, but demonstrate that they may remain applicable in the known-plaintext setting. We leave the study of further implications to other ciphers and techniques as interesting future work.

Acknowledgments. We are highly thankful to the SAC 2026 reviewers for their helpful comments. This work was supported by the Natural Science Foundation of China under grant No. 62272282.

References

1. Tomer Ashur and Erik Takke. A New Linear Distinguisher for Four-Round AES. *IACR Cryptol. ePrint Arch.*, page 398, 2023.
2. Tomer Ashur and Erik Takke. A New Linear Distinguisher for Four-Round AES. *J. Cryptol.*, 38(4):30, 2025.
3. Zhenzhen Bao, Jian Guo, and Eik List. Extended Truncated-differential Distinguishers on Round-reduced AES. *IACR Trans. Symmetric Cryptol.*, 2020(3):197–261, 2020.
4. Navid Ghaedi Bardeh and Vincent Rijmen. New Key-Recovery Attack on Reduced-Round AES. *IACR Trans. Symmetric Cryptol.*, 2022(2):43–62, 2022.
5. Navid Ghaedi Bardeh and Sondre Rønjom. The Exchange Attack: How to Distinguish Six Rounds of AES with $2^{88.2}$ Chosen Plaintexts. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT III*, volume 11923 of *Lecture Notes in Computer Science*, pages 347–370. Springer, 2019.
6. Augustin Bariant and Gaëtan Leurent. Truncated Boomerang Attacks and Application to AES-Based Ciphers. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT IV*, volume 14007 of *Lecture Notes in Computer Science*, pages 3–35. Springer, 2023.
7. Eli Biham, Orr Dunkelman, and Nathan Keller. The Rectangle Attack - Rectangling the Serpent. In Birgit Pfitzmann, editor, *EUROCRYPT*, Lecture Notes in Computer Science, pages 340–357. Springer, 2001.
8. Eli Biham and Nathan Keller. Cryptanalysis of reduced variants of rijndael. In *3rd AES Conference*, volume 230, 2000.
9. Eli Biham and Adi Shamir. Differential Cryptanalysis of the Full 16-Round DES. In Ernest F. Brickell, editor, *CRYPTO*, volume 740 of *Lecture Notes in Computer Science*, pages 487–496. Springer, 1992.
10. Eli Biham and Adi Shamir. *Differential Cryptanalysis of the Data Encryption Standard*. Springer, 1993.
11. Alex Biryukov and Eyal Kushilevitz. From Differential Cryptanalysis to Ciphertext-Only Attacks. In Hugo Krawczyk, editor, *CRYPTO*, volume 1462 of *Lecture Notes in Computer Science*, pages 72–88. Springer, 1998.
12. Andrey Bogdanov and Vincent Rijmen. Zero-Correlation Linear Cryptanalysis of Block Ciphers. *IACR Cryptol. ePrint Arch.*, page 123, 2011.
13. Andrey Bogdanov and Vincent Rijmen. Linear hulls with correlation zero and linear cryptanalysis of block ciphers. *Des. Codes Cryptogr.*, 70(3):369–383, 2014.
14. Xavier Bonnetain, María Naya-Plasencia, and André Schrottenloher. Quantum Security Analysis of AES. *IACR Trans. Symmetric Cryptol.*, 2019(2):55–93, 2019.
15. Charles Bouillaguet, Patrick Derbez, Orr Dunkelman, Pierre-Alain Fouque, Nathan Keller, and Vincent Rijmen. Low-Data Complexity Attacks on AES. *IEEE Trans. Inf. Theory*, 58(11):7002–7017, 2012.
16. Charles Bouillaguet, Patrick Derbez, and Pierre-Alain Fouque. Automatic Search of Attacks on Round-Reduced AES and Applications. In Phillip Rogaway, editor, *CRYPTO*, volume 6841 of *Lecture Notes in Computer Science*, pages 169–187. Springer, 2011.
17. Christina Boura, Anne Canteaut, and Daniel Coggia. A General Proof Framework for Recent AES Distinguishers. *IACR Trans. Symmetric Cryptol.*, 2019(1):170–191, 2019.
18. Christina Boura, Virginie Lallemand, María Naya-Plasencia, and Valentin Suder. Making the Impossible Possible. *J. Cryptology*, 31(1):101–133, 2018.

19. Christina Boura, María Naya-Plasencia, and Valentin Suder. Scrutinizing and Improving Impossible Differential Attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT I*, volume 8873 of *Lecture Notes in Computer Science*, pages 179–199. Springer, 2014.
20. Chengcheng Chang, Meiqin Wang, Ling Sun, and Wei Wang. Improved Truncated Differential Distinguishers of AES with Concrete S-Box. In Takanori Isobe and Santanu Sarkar, editors, *INDOCRYPT*, volume 13774 of *Lecture Notes in Computer Science*, pages 422–445. Springer, 2022.
21. Shiyao Chen, Jian Guo, Eik List, Danping Shi, and Tianyu Zhang. Scrutinizing the Security of AES-Based Hashing and One-Way Functions. In Goichiro Hanaoka and Bo-Yin Yang, editors, *ASIACRYPT I*, volume 16245 of *Lecture Notes in Computer Science*, pages 157–188. Springer, 2025.
22. Carlos Cid, Tao Huang, Thomas Peyrin, Yu Sasaki, and Ling Song. Boomerang Connectivity Table: A New Cryptanalysis Tool. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT II*, Lecture Notes in Computer Science, pages 683–714. Springer, 2018.
23. Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer, 2002.
24. Hüseyin Demirci and Ali Aydin Selçuk. A Meet-in-the-Middle Attack on 8-Round AES. In Kaisa Nyberg, editor, *FSE*, volume 5086 of *Lecture Notes in Computer Science*, pages 116–126. Springer, 2008.
25. Patrick Derbez. *Attaques par Rencontre par le Milieu sur l'AES (Meet-in-the-Middle Attacks on AES)*. PhD thesis, Ecole Normale Supérieure de Paris Paris, France, 2013. https://theses.hal.science/tel-00918146.
26. Patrick Derbez and Pierre-Alain Fouque. Automatic Search of Meet-in-the-Middle and Impossible Differential Attacks. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO II*, volume 9815 of *Lecture Notes in Computer Science*, pages 157–184. Springer, 2016.
27. Patrick Derbez, Pierre-Alain Fouque, and Jérémy Jean. Improved Key Recovery Attacks on Reduced-Round AES in the Single-Key Setting. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT*, volume 7881 of *Lecture Notes in Computer Science*, pages 371–387. Springer, 2013.
28. Patrick Derbez and Léo Perrin. Meet-in-the-Middle Attacks and Structural Analysis of Round-Reduced PRINCE. In Gregor Leander, editor, *FSE*, volume 9054 of *Lecture Notes in Computer Science*, pages 190–216. Springer, 2015.
29. Xiaoli Dong, Jun Liu, Yongzhuang Wei, Wen Gao, and Jie Chen. Meet-in-the-middle attacks on AES with value constraints. *Des. Codes Cryptogr.*, 92(9):2423–2449, 2024.
30. Orr Dunkelman, Shibam Ghosh, Nathan Keller, Gaëtan Leurent, Avichai Marmor, and Victor Mollimard. Partial Sums Meet FFT: Improved Attack on 6-Round AES. In Marc Joye and Gregor Leander, editors, *EUROCRYPT I*, volume 14651 of *Lecture Notes in Computer Science*, pages 128–157. Springer, 2024.
31. Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. The Retracing Boomerang Attack. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT I*, volume 12105 of *Lecture Notes in Computer Science*, pages 280–309. Springer, 2020.
32. Orr Dunkelman, Nathan Keller, Eyal Ronen, and Adi Shamir. The Retracing Boomerang Attack, with Application to Reduced-Round AES. *J. Cryptol.*, 37(3):32, 2024.

33. Orr Dunkelman, Nathan Keller, and Adi Shamir. A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony. In Tal Rabin, editor, *CRYPTO*, Lecture Notes in Computer Science, pages 393–410. Springer, 2010.
34. Orr Dunkelman, Nathan Keller, and Adi Shamir. Improved Single-Key Attacks on 8-Round AES-192 and AES-256. In Masayuki Abe, editor, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 158–176. Springer, 2010.
35. Morris J Dworkin. NIST Special Publication 800-38A 2001 Edition. *NIST Special Publication*, 800:38A, December 2001.
36. Morris J Dworkin. NIST Special Publication 800-38D – Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. Technical report, NIST, November 2007.
37. Niels Ferguson, John Kelsey, Stefan Lucks, Bruce Schneier, Michael Stay, David A. Wagner, and Doug Whiting. Improved Cryptanalysis of Rijndael. In Bruce Schneier, editor, *FSE*, volume 1978 of *Lecture Notes in Computer Science*, pages 213–230. Springer, 2000.
38. Henri Gilbert and Marine Minier. A Collision Attack on 7 Rounds of Rijndael. In *The Third Advanced Encryption Standard Candidate Conference*, pages 230–241. National Institute of Standards and Technology, 2000.
39. Lorenzo Grassi. Mixture Differential Cryptanalysis: a New Approach to Distinguishers and Attacks on round-reduced AES. *IACR Transactions on Symmetric Cryptology*, 2018(2):133–160, 2018.
40. Lorenzo Grassi and Christian Rechberger. New Rigorous Analysis of Truncated Differentials for 5-round AES. *IACR Cryptology ePrint Archive*, 2018:182, 2018.
41. Lorenzo Grassi and Christian Rechberger. Truncated Differential Properties of the Diagonal Set of Inputs for 5-Round AES. In Khoa Nguyen, Guomin Yang, Fuchun Guo, and Willy Susilo, editors, *ACISP*, volume 13494 of *Lecture Notes in Computer Science*, pages 24–45. Springer, 2022.
42. Lorenzo Grassi, Christian Rechberger, and Sondre Rønjom. A New Structural-Differential Property of 5-Round AES. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT II*, volume 10211 of *Lecture Notes in Computer Science*, pages 289–317, 2017.
43. Shay Gueron, Adam Langley, and Yehuda Lindell. AES-GCM-SIV: Nonce Misuse-Resistant Authenticated Encryption. *RFC*, 8452:1–42, 2019.
44. Shay Gueron and Yehuda Lindell. GCM-SIV: Full Nonce Misuse-Resistant Authenticated Encryption at Under One Cycle per Byte. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *ACM CCS*, pages 109–119. ACM, 2015.
45. D. Harkins. Synthetic Initialization Vector (SIV) Authenticated Encryption Using the Advanced Encryption Standard (AES). RFC 5297 (Informational), October 2008.
46. Orhun Kara. Square impossible differential attack and security of AES in known plaintext scenario. *Cryptologia*, 49(2):128–152, 2025.
47. John Kelsey, Tadayoshi Kohno, and Bruce Schneier. Amplified Boomerang Attacks Against Reduced-Round MARS and Serpent. In Bruce Schneier, editor, *FSE*, Lecture Notes in Computer Science, pages 75–93. Springer, 2000.
48. Simon Knellwolf, Willi Meier, and María Naya-Plasencia. Conditional Differential Cryptanalysis of NLFSR-Based Cryptosystems. In Masayuki Abe, editor, *ASIACRYPT*, Lecture Notes in Computer Science, pages 130–145. Springer, 2010.
49. Gaëtan Leurent and Clara Pernot. New Representations of the AES Key Schedule. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT I*, volume 12696 of *Lecture Notes in Computer Science*, pages 54–84. Springer, 2021.

50. Gaëtan Leurent and Clara Pernot. New Representations of the AES Key Schedule. *J. Cryptol.*, 38(1):2, 2025.
51. Leibo Li, Keting Jia, and Xiaoyun Wang. Improved Single-Key Attacks on 9-Round AES-192/256. In Carlos Cid and Christian Rechberger, editors, *FSE*, volume 8540 of *Lecture Notes in Computer Science*, pages 127–146. Springer, 2014.
52. Jiqiang Lu, Orr Dunkelman, Nathan Keller, and Jongsung Kim. New Impossible Differential Attacks on AES. In Dipanwita Roy Chowdhury, Vincent Rijmen, and Abhijit Das, editors, *INDOCRYPT*, volume 5365 of *Lecture Notes in Computer Science*, pages 279–293. Springer, 2008.
53. Jiqiang Lu and Wenchang Zhou. Improved Meet-in-the-Middle Attacks on Nine Rounds of the AES-192 Block Cipher. In Elisabeth Oswald, editor, *CT-RSA*, volume 14643 of *Lecture Notes in Computer Science*, pages 136–159. Springer, 2024.
54. Hamid Mala, Mohammad Dakhilalian, Vincent Rijmen, and Mahmoud Modarres-Hashemi. Improved Impossible Differential Cryptanalysis of 7-Round AES-128. In Guang Gong and Kishan Chand Gupta, editors, *INDOCRYPT*, volume 6498 of *Lecture Notes in Computer Science*, pages 282–291. Springer, 2010.
55. Ueli M. Maurer, Yvonne Anne Oswald, Krzysztof Pietrzak, and Johan Sjödin. Luby-Rackoff Ciphers from Weak Round Functions? In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 391–408. Springer, 2006.
56. Ueli M. Maurer and Krzysztof Pietrzak. Composition of Random Systems: When Two Weak Make One Strong. In Moni Naor, editor, *TCC*, volume 2951 of *Lecture Notes in Computer Science*, pages 410–427. Springer, 2004.
57. Kazuhiko Minematsu and Yukiyasu Tsunoo. Hybrid Symmetric Encryption Using Known-Plaintext Attack-Secure Components. In Dongho Won and Seungjoo Kim, editors, *ICISC*, volume 3935 of *Lecture Notes in Computer Science*, pages 242–260. Springer, 2005.
58. Steven A. Myers. Black-Box Composition Does Not Imply Adaptive Security. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 189–206. Springer, 2004.
59. National Institute of Standards and Technology. FIPS 197. *National Institute of Standards and Technology, November*, pages 1–51, 2001.
60. Phillip Rogaway and Thomas Shrimpton. A Provable-Security Treatment of the Key-Wrap Problem. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *LNCS*, pages 373–390. Springer, 2006.
61. Sondre Rønjom. A Short Note on a Weight Probability Distribution Related to SPNs. *IACR Cryptol. ePrint Arch.*, page 750, 2019.
62. Sondre Rønjom, Navid Ghaedi Bardeh, and Tor Helleseth. Yoyo Tricks with AES. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT I*, volume 10624 of *Lecture Notes in Computer Science*, pages 217–243. Springer, 2017.
63. David A. Wagner. The Boomerang Attack. In Lars R. Knudsen, editor, *FSE*, Lecture Notes in Computer Science, pages 156–170. Springer, 1999.
64. Meiqin Wang, Tingting Cui, Huaifeng Chen, Ling Sun, Long Wen, and Andrey Bogdanov. Integrals Go Statistical: Cryptanalysis of Full Skipjack Variants. In Thomas Peyrin, editor, *FSE*, volume 9783 of *Lecture Notes in Computer Science*, pages 399–415. Springer, 2016.

A Best Attacks on Round-reduced AES in The Chosen-plaintext Model

To aid the reader, Table 3 provides the best key-recovery attacks on all round-reduced versions of AES under a single secret key to allow easy comparison of our results to those.

Table 3: Best single-key key-recovery attacks on round-reduced AES in the chosen-plaintext model. #Rds. = #rounds, memory in #AES states, n/a = not available.

Version	Type	#Rds.	Time	Data	Memory	Reference
128	Integral	7/10	$2^{128} - 2^{119}$	$2^{128} - 2^{119}$	n/a	[37]
	Imp. differential	7/10	$2^{110.9}$	$2^{104.9}$	$2^{71.9}$	[49,50]
	Zero difference	7/10	$2^{110.2}$	$2^{110.2}$	$2^{110.2}$	[4]
	DS-MitM	7/10	$2^{99} + 2^{100}$	2^{97}	2^{98}	[27]
192	Imp. differential	7/12	$2^{139.2}$	$2^{91.2}$	n/a	[52]
	Imp. differential	7/12	$2^{118.8}$ MA	$2^{113.8}$	n/a	[52]
	Integral	8/12	2^{188}	$2^{128} - 2^{119}$	n/a	[37]
	DS-MitM	9/12	$2^{186.4} + 2^{186.8}$	2^{121}	$2^{172.3}$	[53]
	DS-MitM	9/12	$2^{182.8} + 2^{182.8}$	$2^{125.3}$	$2^{173.2}$	[29]
256	Imp. differential	8/14	$2^{229.7}$ MA	$2^{89.1}$	n/a	[52]
	Imp. differential	8/14	$2^{227.8}$ MA	$2^{111.1}$	n/a	[52]
	Integral	8/14	2^{204}	$2^{128} - 2^{119}$	n/a	[37]
	DS-MitM	9/14	$2^{203} + 2^{205}$	2^{120}	2^{203}	[27]
	DS-MitM	9/14	$2^{194.5} + 2^{197.1}$	2^{127}	$2^{162.9}$	[29]

B Six-round Rectangle Key-recovery Attack Algorithm

C Six-round Rectangle Distinguisher

We derive a rectangle variant from a retracing boomerang distinguisher, which transforms to the random known-plaintext model by applying the model in Section 3.4. For this purpose, we consider a truncated variant of the six-round boomerang distinguisher by Bariant and Leurent [6] that is shown Figure 6. In contrast to the distinguisher by Bariant and Leurent, the active diagonal of the plaintext pairs (P, P') is exchanged to derive another pair $(\overline{P}, \overline{P'})$ in our distinguisher.

C.1 Attack in The Chosen-plaintext Model

The chosen-plaintext attack starts from plaintexts in structures with the first diagonal active and searches for pairs that follow the trail in Figure 6. With

Algorithm 3 Six-round RKP rectangle attack.

function $\mathcal{A}$
  Precompute hash tables $\mathcal{H}_i$ and $\mathcal{H}'_i$ for $i \in \{0, \ldots, 3\}$
  Collect $2^{99.1}$ known plaintext-ciphertext pairs (P, C)
  for all active inverse ciphertext diagonal indices $i \in \{0, 1, 2, 3\}$ **do**
    Store them into $\mathcal{C}$ according to the values in the ciphertexts in three inverse diagonals except diagonal i
    Identify the approximately $4 \cdot \binom{2^{99.1}}{2} \cdot 2^{-96} \approx 2^{103.2}$ pairs $((P, C), (\overline{P}, \overline{C}))$
    Store them into six tables $\mathcal{P}_j$, for $j \in \{0, \ldots, 5\}$, according to their values in two plaintext diagonals of P and P' each, where the values of P and P' in the entry are also lexicographically sorted.
    Identify approximately $4 \cdot 6 \cdot 2 \cdot \binom{2^{101.2}}{2} \cdot 2^{-128} \approx 2^{79}$ quartets
    for all quartets $(P, C), (P', C'), (\overline{P}, \overline{C}), (\overline{P}', \overline{C}')$ **do**
      for all active diagonal indices after the first round, $j' \in \{0, 1, 2, 3\}$ **do**
        Identify 255 key candidates on average for each plaintext pair and their first active diagonal with $\mathcal{H}_{j-j'}$
        if there is no key contained in both sets **then** discard the pair
        Identify 255 key candidates on average for each plaintext pair and their second active diagonal with $\mathcal{H}_{j-j'}$
        if there is no key contained in both sets **then** discard the pair
        Identify $4 \cdot 255$ key candidates on average for each ciphertext pair and their active inverse diagonal with $\mathcal{H}'_i$
        if there is no key contained in both sets **then** discard the pair
      ▷ We expect ca. $2^{79} \cdot 4 \cdot 2^{8+8-32} \cdot 4 \cdot 2^{16+16-64} = 2^{35}$ remaining quartets
      for six options of active plaintext diagonals $i \in \{0, \ldots, 5\}$ **do**
        Store the suggested keys for K^0 into a table $\mathcal{K}_i$ and find key collisions between quartets ▷ We expect ca. $6 \cdot \binom{2^{35}/6}{2} \cdot 2^{-64} \approx 2^{2.4}$ collisions
        for all pairs of quartet with colliding keys **do**
          Guess the 64 unknown key bits of K^0
          Compute the key schedule to K^6
          if does not match the known bits of K^6 **then** discard
          Encrypt any other known plaintext pair with the candidate key
          if the encryption is correct **then return** the key candidate
  return $\perp$

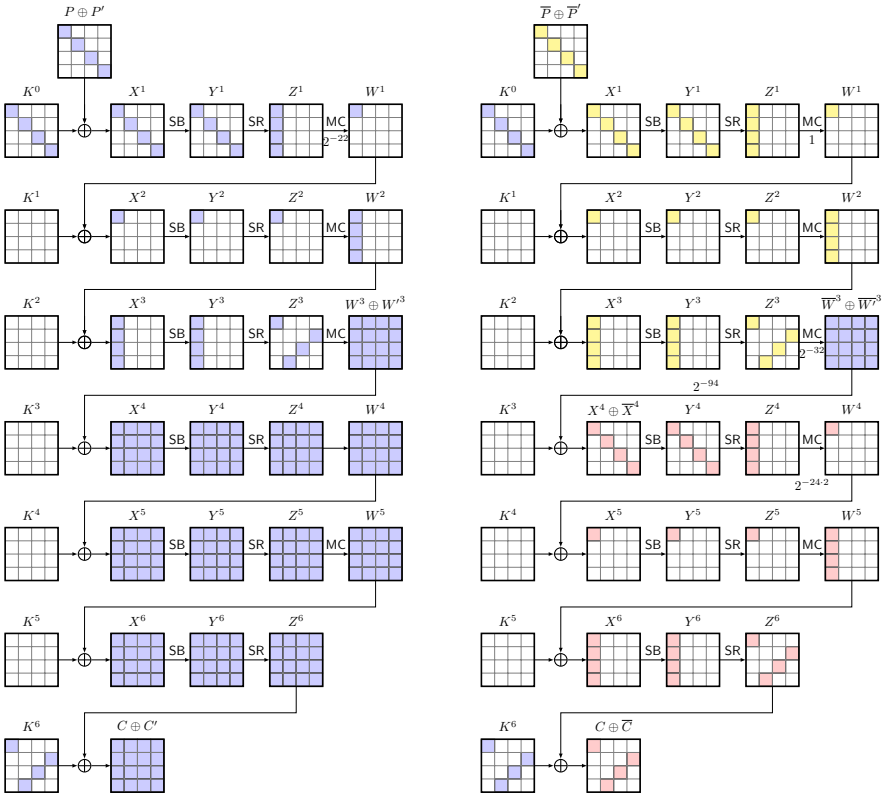


Fig. 6: A six-round rectangle distinguisher derived from a trail in [6]. Left: the trail between pairs (P, P') to (C, C') (blue). Right: the trail between $(\overline{P}, \overline{P}')$ at the top (yellow) and from $(W^2, \overline{W}^2)$ to $(C, \overline{C})$ and from $(P', \overline{P}')$ to $(C', \overline{C}')$ (red). Colored state cells represent active bytes in the differential trails.

probability 2^{-22}, only a single byte is active in ΔW^1. If the second pair $(\overline{P}, \overline{P}')$ shares the same values for the active diagonal with P, P', then its difference is also mapped to a single active byte with probability one for the pair. Thus, we consider a **retracing rectangle** here. Both pairs are then mapped to a single active inverse diagonal in ΔZ^3. With probability 2^{-94}, $X^4 \oplus \overline{X}^4$ is active in only exactly one diagonal. If the second pair has exactly the same difference ΔZ^3 as the first pair, which holds with probability approximately 2^{-32}, it follows that $X'^4 \oplus \overline{X'}^4 = X^4 \oplus \overline{X}^4$. In that case, both pairs are mapped to a difference ΔW^4 with a single active byte at a fixed position with probability $2^{-24 \cdot 2}$, which is mapped to $C \oplus \overline{C}$ and $C' \oplus \overline{C}'$ having a single active inverse diagonal at a fixed position. As a result, the probability for a correct quartet, i.e., a quartet whose pairs follow the described trails, is approximately $p_1 \approx 2^{-22} \cdot 1 \cdot 2^{-94} \cdot 2^{-32} \cdot 2^{-24 \cdot 2} = 2^{-196}$, whereas the probability is approximately $p_0 \approx 2^{-96 \cdot 2} = 2^{-192}$ for

a random pair. Thus, the signal-to-noise ration $s = p_1/p_0 = 2^{-4}$. Approximating the number of quartets to be normally distributed and using

$$q = c \cdot p_1^{-1} \cdot s^{-1}$$

for a small constant c and setting a threshold on the number of quartets as

$$q_t = q \cdot (1 + s) \cdot p_0$$

such that the distinguisher will output "real" whenever the empirically observed number of quartets $\widehat{q} \geq q_t$ and "random" otherwise, the distinguisher has an approximated success probability $\Phi(\sqrt{c}/2)$ where Φ is the cumulative density function of the standard normal distribution. For $c = 4$, $\Phi(\sqrt{4}/2) \approx 0.84$. Given the parameters above, the minimum number of required quartets is then $q = 4 \cdot 2^{196} \cdot 2^4 = 2^{202}$.

The adversary can build plaintext structures that iterate over all values of one fixed plaintext diagonal and leave all other bits constant, which yields $\binom{2^{32}}{2} \approx 2^{63}$ pairs per structure. Using 2^{70} structures produces approximately 2^{133} pairs that can be combined to approximately $2 \cdot \binom{2^{133}}{2} = 2^{266}$ quartets. However, in only a fraction of 2^{-64} of them, $\overline{P}$ shares the same fixed-diagonal value with P and $\overline{P}'$ the same with P'. Thus, approximately $2^{266-64-196} = 2^6$ good quartets can be formed on average. On the other hand, we can expect $2^{266-64-192} = 2^{10}$ random wrong quartets.

In contrast to retracing boomerangs, the adversary has to store all data over all structures, i.e., 2^{102} plaintext-ciphertext pairs, indexed by the value of the three inverse diagonals of the ciphertexts that shall be inactive. The adversary can identify $\binom{2^{102}}{2} \cdot 2^{-96} \approx 2^{107}$ pairs $(P, C, \overline{P}, \overline{C})$ across structures. For each of them, there exists exactly one other pair $(P', C', \overline{P}', \overline{C}')$ that was already queried before and could lead to a quartet, as it must be a mixture plaintext pair. The probability that it also has the desired ciphertext difference is 2^{-96}, i.e., one can expect $2^{107} \cdot 2^{-96}/2 = 2^{10}$ random wrong quartets on average for a random permutation, and $2^{10} + 2^6$ quartets for the real six-round AES. Note that the division by two refers to the fact that both pairs $(P, C, \overline{P}, \overline{C})$ and $(P', C', \overline{P}', \overline{C}')$ refer to the same quartet, i.e., it is irrelevant, which pair is found first. The attack requires

- $2^{70+32} = 2^{102}$ six-round encryptions,
- 2^{102} memory accesses into a table of 2^{102} entries,
- 2^{107} memory accesses to find quartets into a table of 2^{107} entries, and
- $2^{10} + 2^6$ counter increments.

Thus, the complexity stems from 2^{102} encryptions and $2^{102}+2^{107}+2^{10}+2^6 \approx 2^{107}$ memory accesses to huge tables. Approximating each memory access to huge tables by a full six-round encryption, this yields $2^{102} + 2^{107} + 2^{107} \approx 2^{108.7}$ encryption equivalents. It needs $2^{102} \cdot 2 + 2^{107} \cdot 4 \approx 2^{109}$ state equivalents of memory and 2^{102} chosen plaintexts.

Algorithm 4 Six-round rectangle distinguisher.

function $\mathcal{A}$
  Collect $2^{113.5}$ known plaintext-ciphertext pairs (P, C).
  Store them into a table $\mathcal{Q}$
  $\mathsf{count} \leftarrow 0$
  for all active plaintext diagonal indices $i \in \{0, 1, 2, 3\}$ **do**
    for all active inverse ciphertext diagonal indices $j \in \{0, 1, 2, 3\}$ **do**
      Store the plaintext-ciphertext pairs (P, C) into a second table $\mathcal{T}$ indexed by the i-th diagonal of P and all but the j-th inverse diagonals of C
      Identify the approximately $4 \cdot \binom{2^{113.5}}{2} \cdot 2^{-96-32} \approx 2^{100}$ pairs $((P, C), (\overline{P}, \overline{C}))$ in total
      Store the pairs for the current value i into a table $\mathcal{P}$ indexed by the plaintext diagonals except i, lexicographically sorted
    Identify the approximately $4 \cdot 2 \cdot \binom{2^{100}}{2} \cdot 2^{-96-96} \approx 2^{10}$ quartets in total
    for all quartets **do** $\mathsf{count} \leftarrow \mathsf{count} + 1$
  if $x \geq 2^{10} + 2^5$ **then return** "real"
  return "random"

C.2 Conversion to The Random Known-plaintext Model

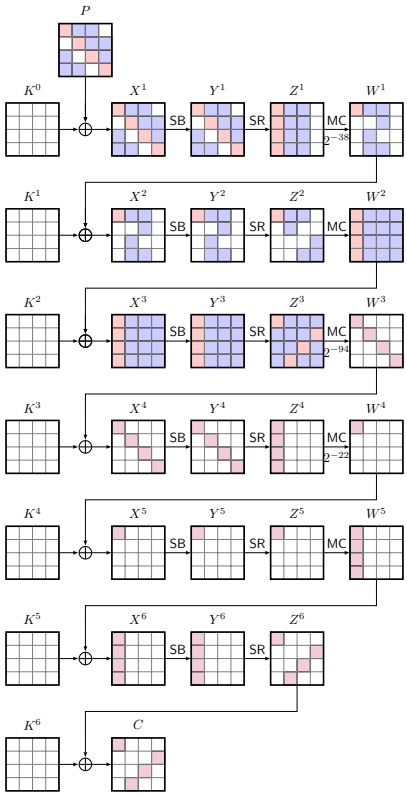
In the RKP model, we can use the same trails but significantly more resources. In the random known-plaintext model, we have plaintext and ciphertext spaces of $|\mathcal{D}_{\text{in}}| = |\mathcal{D}_{\text{out}}| \approx 4 \cdot 2^{32} = 2^{34}$ for P' when we allow any diagonal as the active one. However, the active-diagonal index between P and P' must match that between $\overline{P}$ and $\overline{P}'$. Moreover, there are (almost) all 2^{128} possible options for P, and approximately 2^{96} options for $\overline{P}$, respectively; though, P, P', and $\overline{P}$ determine the value of $\overline{P}'$. Thus, the probability that a random plaintext quartet is in the input space is approximately $1 \cdot 2^{-94} \cdot 2^{-32} \cdot 2^{-128} = 2^{-254}$.
Moreover, we can allow all options for the single active inverse diagonal in the ciphertext differences $C \oplus \overline{C}$ and $C' \oplus \overline{C}'$, so that a random quartet has probability 2^{-188}. Thus, the trail probability becomes $p_1 \approx 2^{-22} \cdot 2^{-32} \cdot 2^{-94} \cdot 2^{-22} \cdot 2^{-22} \approx 2^{-192}$ compared to $p_0 \approx 2^{-188}$ for a wrong quartet. As a result, we want $q \geq 4 \cdot 2^{192} \cdot 2^{-188+192} = 2^{198}$ quartets. Given 2^d plaintexts, we can build $6 \cdot \binom{2^d}{4} \approx 2^{4d-2}$ different pairs of pairs, which can be combined in two ways at the plaintexts side. Thus, the attack needs $4d \geq 2 + 254 + 198$, which yields $d \geq 113.5$. From that, we expect $6 \cdot \binom{2^{113.5}}{4} \approx 2^{452}$ quartets, out of which $2^{452} \cdot 2^{-254} \cdot 2^{-188} \approx 2^{10}$ fulfill the plaintext and ciphertext conditions, plus $2^{452} \cdot 2^{-254} \cdot 2^{-192} \approx 2^6$ correct quartets. The attack consists of the steps outlined in Algorithm 4.
The distinguisher has a time complexity of

- $2^{113.5}$ encryptions,
- $(1 + 4 \cdot 4) \cdot 2^{113.5} = 2^{117.6}$ memory accesses for the four tables $\mathcal{P}_i$,
- $4 \cdot 4 \cdot 2^{98} = 2^{102}$ memory accesses for storing the pairs,
- 2^{102} memory accesses for the checks of the remaining pairs, and
- $2^8 + 2^4$ memory accesses to a counter.

Thus, the time complexity adds up to $2^{113.5}$ encryptions and $2^{117.6}+2^{102}+2^{102}+2^{10}+2^{6} \approx 2^{117.6}$ memory accesses. Approximating each with a full encryption yields $2^{117.7}$ encryption equivalents. The attack needs $2 \cdot 2 \cdot 2^{113.5} = 2^{115.5}$ state equivalents of memory.

D Six-round Mixture Distinguisher

We also consider the six-round mixture distinguisher by Bardeh and Rønjom from [5], that is shown in Figure 7.



(a) Six-round mixture distinguisher [5].

Fig. 7: The mixture distinguisher on six-round AES that we transfer to the RKP model. Colored state cells represent active bytes in the differential trails. Figure 2 contains two differentials between two related pairs: one differential with all active bytes for one pair, and a differential where one diagonal (colored in red) is exchanged to form a second pair.

D.1 Original Attack Parameters in The Chosen-plaintext Model

The original attack started from plaintexts in structures with the first three diagonals active and searched for pairs that followed the trail in Figure 2, and whose mixture pair from exchanging the first plaintext diagonal also followed it. The probability for the first pair to follow the trail through the first round was $4 \cdot 2^{-40}$ and $4 \cdot 2^{-96}$ to lead to a single active diagonal after three rounds, i.e., 2^{-132}. If that is the case, the mixed pair would follow the truncated differential trail until there as well and possess the same difference at ΔW^3 as the first pair. If the difference of both pairs would reduce to have any single active byte after Round 4 with probability $(4 \cdot 2^{-24})^2 = 2^{-44}$, they would both have a single active inverse diagonal after six rounds without a final MixColumns operation. Thus, the probability would be 2^{-176} for a mixture pair of pairs for the real six-round AES, whereas it would be approximately $2^{-94 \cdot 2} = 2^{-188}$ for a random permutation.
The cardinalities of the plaintext- and ciphertext-difference spaces are $|\mathcal{D}_{\text{in}}| = 2^{96}$ and $|\mathcal{D}_{\text{out}}| \approx 4 \cdot 2^{32} = 2^{34}$. Therefore, approximately $2^{2\delta_{\text{in}}-1} = 2^{191}$ pairs could be formed from a structure of 2^{96} plaintext; though, the attack only needed $2^{89.8}$ chosen plaintexts to form 2^{176} mixtures structures.

D.2 Parameters in The Random Known-plaintext Model

In the random known-plaintext model, we can expand the plaintext space to $|\mathcal{D}_{\text{in}}| \approx 4 \cdot 2^{96} = 2^{98}$ texts. Given 2^p random pairwise distinct plaintexts, we can form approximately $\binom{2^p}{4} \approx 2^{4p}/24$ four-tuples of texts. The probability that the second text shares a diagonal with the first text is $4 \cdot 2^{-32} = 2^{n-\delta_{\text{in}}}$, the probability that the third text is a mixture of the first two is 2^{-n}, and the probability that the fourth text is the opposite mixture of the first two is 2^{-n} again. Therefore, we obtain $2^{4p-3n+\delta_{\text{in}}}/24$ mixture quartets from those plaintexts. To have 2^{176} mixture quartets, we need

$$2^{4p-3n+\delta_{\text{in}}}/24 = 2^{176}\,,$$

which yields $4p - 384 + 98 - 4.6 = 176$ and therefore $p \approx 116.7$, i.e., $2^{116.7}$ known random plaintexts are required to form approximately $2^{462-384+98} = 2^{176}$ pairs of pairs and expect one. The attack can use $4 \cdot 4$ tables of $2^{116.7}$ plaintext-ciphertext pairs each to store all plaintexts and ciphertexts; four for each combination of inactive plaintext diagonal and four for each active inverse ciphertext diagonal. Then, it can filter pairs on 32 plaintext bits and 96 ciphertext bits in each table to find $16 \cdot 2^{2 \cdot 116.7-1-128} \approx 2^{108.4}$ pairs that it can store in another table (using the same memory as before). For each, it can look up whether the mixture pair is also contained in the table. Thus, the attack requires $2^{116.7} + 16 \cdot 2 \cdot 2^{116.7} + 4 \cdot 2^{108.4} \approx 2^{121.7}$ computations or memory accesses to huge tables, $2^{121.7}$ states of memory and $2^{116.7}$ random known plaintexts.

E Low-bias Differential Probabilities

For the truncated-differential distinguisher in Section 6, we use the evaluation of the biases of differential trails through round-reduced AES from [20] in Table 4.

Table 4: Probability log decimals through four-round AES from [20] for mapping a single active byte at index i_s to having a single inactive byte at index i_e. Values denote x in $2^{-8} + 2^{-31-x}$.

	i_3					i_3			
i_e	0	1	2	3	i_e	0	1	2	3
0	.982	.816	.988	.774	2	.765	.922	.797	1.002
5	.774	.982	.816	.988	7	1.002	.765	.922	.797
10	.988	.774	.982	.816	8	.797	1.002	.765	.922
15	.816	.988	.774	.982	13	.922	.797	1.002	.765
1	.654	.687	.947	.924	3	.924	.908	.674	.682
6	.924	.654	.687	.947	4	.682	.924	.908	.674
11	.947	.924	.654	.687	9	.674	.682	.924	.908
12	.687	.947	.924	.654	14	.908	.674	.682	.924

F DS-MitM Attack on Seven-round AES-128 Algorithm

Algorithm 5 RKP DS-MitM attack on seven-round AES-128.

function $\mathcal{A}_{\mathsf{online}}(\mathcal{Q}, \mathcal{P}, i, i', \Delta Y^2[i], \Delta X^6[i'])$
  Initialize an empty table $\mathcal{T}_{\mathsf{online}}$
  Let $\Delta Z^6[i'']$ be the byte corresponding to $\Delta X^6[i']$
  for all 255^2 guesses of $\Delta W^1[i], \Delta Z^6[i'']$ **do**
    Compute the pair from P, P' to $Y^2[i]$
    Derive the δ-sequence for all texts $Y_j^2[i]$
    Compute backwards to the corresponding would-be plaintexts P_j
    Lookup the available P_j's in $\mathcal{Q}$;
    Let k be the #available P_j's for the current δ-set
    if $k < \theta_k$ **then** discard the pair
    Construct the lexicographically first θ_k-reduced δ-sequence as plaintexts P_j
    Lookup the corresponding ciphertexts C_j
    Compute from C_j to $X_j^6[i']$ for all available $j \in \{0, \ldots, 255\}$ and store the resulting sequence indexed by its configuration of available text indices and the byte vector $\Delta X_j^6[i']$, and the induced key material as an entry in $\mathcal{T}_{\mathsf{online}}$
  return $\mathcal{T}_{\mathsf{online}}$

function $\mathcal{A}_{\mathsf{offline}}(\mathcal{T}_{\mathsf{online}}, i, i', \Delta Y^2[i], \Delta X^6[i'])$
  Initialize an empty table $\mathcal{T}_{\mathsf{offline}}$
  for all guesses of eight internal byte variables in X^3 and X^5 **do**
    Trace the solutions from $\Delta Y^2[i]$ to $\Delta X^6[i]$ ▷ One solution on average
    Trace the texts in a δ-sequence through the distinguisher to $X_j^6[i']$
    for all k-reductions of the δ-sequence **do**
      Construct the contained text in the δ-sequence vector $\mathbf{x}$
      if $\mathbf{x} \in \mathcal{T}_{\mathsf{online}}$ **then**
        Add $\mathbf{x}$, the matches, and the induced keys as an entry $\mathbf{y}$ in $\mathcal{T}_{\mathsf{offline}}$
  return $\mathcal{T}_{\mathsf{offline}}$

function $\mathcal{A}$
  Precompute four hash tables $\mathcal{H}_i$ and four hash tables $\mathcal{H}'_i$ as in Algorithm 3
  Collect 2^{124} known plaintext-ciphertext pairs (P, C) and insert them into a table $\mathcal{Q}$, according to the values in the plaintext in the diagonals except for the first one and the values in three inverse diagonals of $\widehat{W}^7$ except for the first one
  Identify approximately $\binom{2^{124}}{2} \cdot 2^{-96-96} \approx 2^{47}$ pairs (P, C)
  Store them into a table $\mathcal{P}$
  Initialize a key-candidate table $\mathcal{K}$
  for all $\alpha \times \beta$ active-byte differences $(\Delta Y^2[i], \Delta X^6[i'])$ **do**
    for all guesses of $4 \cdot 4$ active byte indices (i, i') in ΔW^1 and ΔX^6 **do**
      $\mathcal{T}_{\mathsf{online}} \leftarrow \mathcal{A}_{\mathsf{online}}(\mathcal{Q}, \mathcal{P}, i, i', \Delta Y^2[i], \Delta X^6[i']$
      $\mathcal{T}_{\mathsf{offline}} \leftarrow \mathcal{A}_{\mathsf{offline}}(\mathcal{T}_{\mathsf{online}}, i, i', \Delta Y^2[i], \Delta X^6[i'])$
      for all matches in $\mathcal{T}_{\mathsf{offline}}$ **do**
        Store their induced key candidates into $\mathcal{K}$
  Sort $\mathcal{K}$ descendingly
  return $\mathcal{K}$

G Impossible-differential Attack on Seven-round AES-192

In this section, we describe a similar impossible-differential attack on seven rounds as we have given in Section 7 for the AES-128. The key-schedule differences between the individual AES versions prevent us from having two key bytes in K^1 derived from known bytes in K^0. However, by a slight alteration, we can derive at least one key byte in $K^1[11]$. The trails are shown in Figure 8.

G.1 Attack Parameters in The Chosen-plaintext Model

The cardinalities of plaintext- and ciphertext-difference spaces in a chosen-plaintext model remain $|\mathcal{D}_{\text{in}}| = 2^{\delta_{\text{in}}} = 2^{64}$ and $|\mathcal{D}_{\text{out}}| = 2^{\delta_{\text{out}}} = 2^{32}$. The attack uses structures of 2^{64} chosen plaintexts each that can be combined to approximately $2^{2\delta_{\text{in}}-1} \approx 2^{127}$ pairs per structure. Given an initial filter of $2^{\delta_{\text{out}}-n} \approx 2^{-96}$, there are on average

$$2^{s_p} = 2^{2\delta_{\text{in}}-1+\delta_{\text{out}}-n} \approx 2^{31}$$

surviving pairs per structure. The number of bit conditions, $c_{\text{in}} = 46$ bit conditions on the plaintext-side guessing trail, and $c_{\text{out}} = 22$ bit conditions on the ciphertext-side guessing trail, remain as before. The only difference is the fact that there are now $k_{\text{in}} = 88$ key bits and $k_{\text{out}} = 32$ key bits involved, with a union of $k_{\text{atk}} = k_{\text{in}} \cup k_{\text{out}} = 120$ key bits. Therefore, the required number of surviving pairs 2^d should satisfy

$$\left(1 - 2^{-c_{\text{in}}+c_{\text{out}}}\right)^{2^d} \leq 2^{-a}$$

which yields $2^d \geq 2^{74}$ for $a \approx 92.33$ and thus $2^s = 2^d/2^{s_p} = 2^{74-31} = 2^{43}$ structures. This corresponds to $2^{s+\delta_{\text{in}}} = 2^{43+64} = 2^{107}$ chosen plaintexts. As a result, the attack can reduce the involved key space from 2^{120} to $2^{120-92.33} \leq 2^{28}$. In combination with the 72 non-involved bits of the secret key, the adversary can perform at most 2^{100} guesses to identify the full secret key with one encryption per remaining full candidate in $2^{k-a} \leq 2^{100}$ encryptions on average.

G.2 Attack Parameters in The Random Known-plaintext Model

In the random known-plaintext model, pairs occur randomly in the desired plaintext space. Given 2^p plaintexts, we can form 2^{2p-1} pairs. From those, approximately

$$2^{2p-1+\delta_{\text{in}}+\delta_{\text{out}}-2n} = 2^{2(p-n)-1+\delta_{\text{in}}+\delta_{\text{out}}}$$

pairs are surviving pairs, i.e., lie in the desired plain- and ciphertext spaces. To have the same number of surviving pairs 2^d as before, we need

$$\begin{aligned} 2(p-n) - 1 + \delta_{\text{in}} + \delta_{\text{out}} &= d \\ ((2n+1) - \delta_{\text{in}} - \delta_{\text{out}} + d)/2 &= p\,, \end{aligned}$$

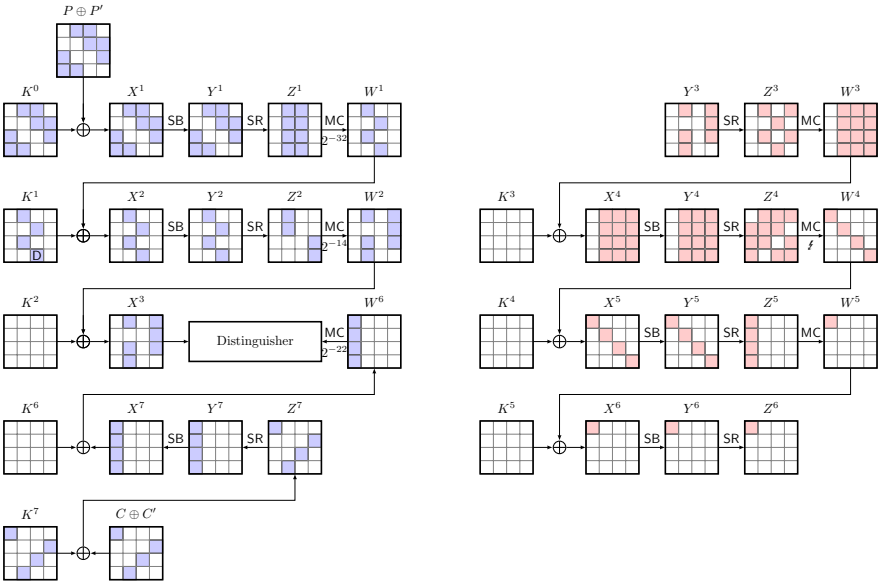


Fig. 8: Seven-round impossible-differential attack on AES-192. Left: the online key-recovery phase. Right: the impossible-differential distinguisher. Colored state cells represent active bytes in the differential trails. Colored key cells are filtered in the attack.

which yields $p = (257 - 64 - 32 + 74)/2 = 117.5$. Therefore, the attack needs $2^{117.5}$ random known plaintext-ciphertext pairs on average.

The attack collects and stores them, that is $2^{118.5}$ plaintext-ciphertext states, and identifies approximately $\binom{2^{117.5}}{2} \cdot 2^{-64-96} = 2^{74}$ surviving pairs. For each, it can use a few table lookups to identify $2^{120} \cdot 2^{-68} = 2^{52}$ keys that it can discard on average. For that purpose, it has a table $\mathcal{K}$ of 2^{120} bits that represent the key candidates, or 2^{113} AES state equivalents. Thus, the time complexity consists of

- $2^{117.5}$ encryptions,
- $2^{117.5}$ lookups into a huge table,
- $2^{74} \cdot 4$ lookups into smaller tables whose size is negligible,
- $2^{74} \cdot 2^{52}$ bit models in the huge table $\mathcal{K}$,
- 2^{120} sequential lookups to find surviving keys,
- and at most 2^{100} final encryptions,

and can be upper bounded by $2^{117.5} + 2^{100}$ encryptions and $2^{117.5} + 2^{76} + 2^{126} + 2^{120} \approx 2^{126}$ memory accesses into huge tables. The memory complexity is upper bounded by $2^{113} + 2^{118.5} \approx 2^{118.5}$ state equivalents.

H Impossible-differential Attack on Eight-round AES-256

In this section, we revisit and improve over the known-plaintext attack by Kara on eight-round AES-256 [46] by replacing the trail from Biham and Keller [8] with that by Mala et al. [54].

H.1 Attack Parameters in The Random Known-plaintext Model

The cardinalities of plaintext- and ciphertext-difference spaces are $|\mathcal{D}_{\text{in}}| = 2^{\delta_{\text{in}}} = 2^{128}$ and $|\mathcal{D}_{\text{out}}| = 2^{\delta_{\text{out}}} = 2^{32}$ as the key-recovery procedure fixes the spaces of interests. The attack original uses all combination of $2^p = 2^{118}$ known plaintexts that can combined to approximately $2^{2p-1} = 2^{235}$ pairs. Given an initial filter of $2^{\delta_{\text{out}}-n} \approx 2^{-96}$, there are on average

$$2^{s_p} = 2^{2p-1+\delta_{\text{out}}-n} \approx 2^{139}$$

surviving pairs per structure.
Our attack differs in the trail used from ΔZ^1 to ΔW^4. The original attack had $c_{\text{in}} = 118$ bit conditions on the plaintext-side guessing trail and $c_{\text{out}} = 22$ bit conditions on the ciphertext-side guessing trail. By replacing the plaintext-side trail, we obtain

- 64 bit conditions from ΔZ^1 to ΔW^1,
- 32 bit conditions from ΔZ^2 to ΔW^2,
- and 14 bit conditions from ΔZ^3 to ΔW^3,

which leads to $c_{\text{in}} = 110$ bit conditions. As in the original attack, there remain $k_{\text{in}} = 192$ key bits from K^0 and K^1 and $k_{\text{out}} = 32$ key bits from K^8 involved, with a union of $k_{\text{atk}} = k_{\text{in}} \cup k_{\text{out}} = 224$ involved key bits. We tested the key-schedule representations by Leurent and Pernot [49,50], but could not identify better attacks, as the ability to deduce a byte in K^8 would require the knowledge of one full row in K^0 and K^1.
For our trail, the required number of suriving pairs 2^d must satisfy

$$\left(1 - 2^{-c_{\text{in}}+c_{\text{out}}}\right)^{2^d} \leq 2^{-a}$$

which yields $2^d \geq 2^{136.5}$ surviving pairs to obtain an advantage of approximately $a = 32.65$ bits. Thus, $2^s = 2^d/2^{s_p} = 2^{136.5+96} = 2^{232.5}$ pairs, which implies the need for approximately $2^{116.8}$ random known plaintexts. As a final step, the attack can guess the remaining a bits not covered in the attack to recover the full key with one encryption per remaining full candidate in $2^{k-a} = 2^{223.5}$ encryptions on average.
Our attack needs memory for $2^{116.8}$ plaintext-ciphertext pairs, and finds $2^{136.5}$ pairs that satisfy its ciphertext filter. Each of them suggests $2^{224-110-22} = 2^{92}$ keys on average, i.e., $2^{136.5+92} = 2^{228.5}$ keys will be looked up and marked, which dominates the overall complexity. In contrast, the attack in [46] ignores the costs for setting flags in a huge table. The attack needs

- $2^{116.8} + 2^{136.5} + 2^{228.5} + 2^{223.5} \approx 2^{228.5}$ operations,
- $2^{116.8}$ plaintext-ciphertext pairs, and
- memory for $2^{117.8} + 2^{224} \cdot 1/128 \approx 2^{217}$ state equivalents.

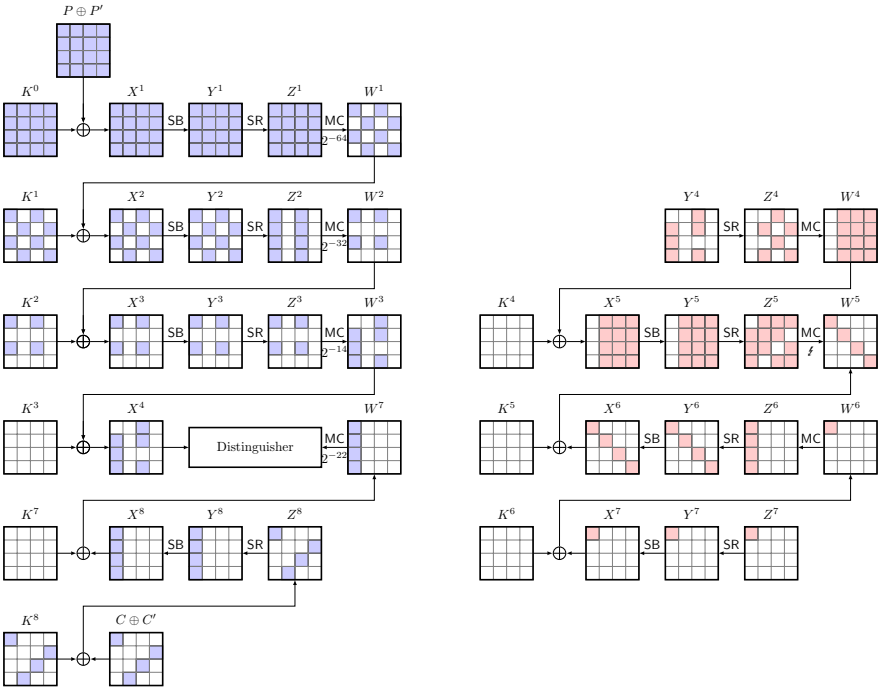


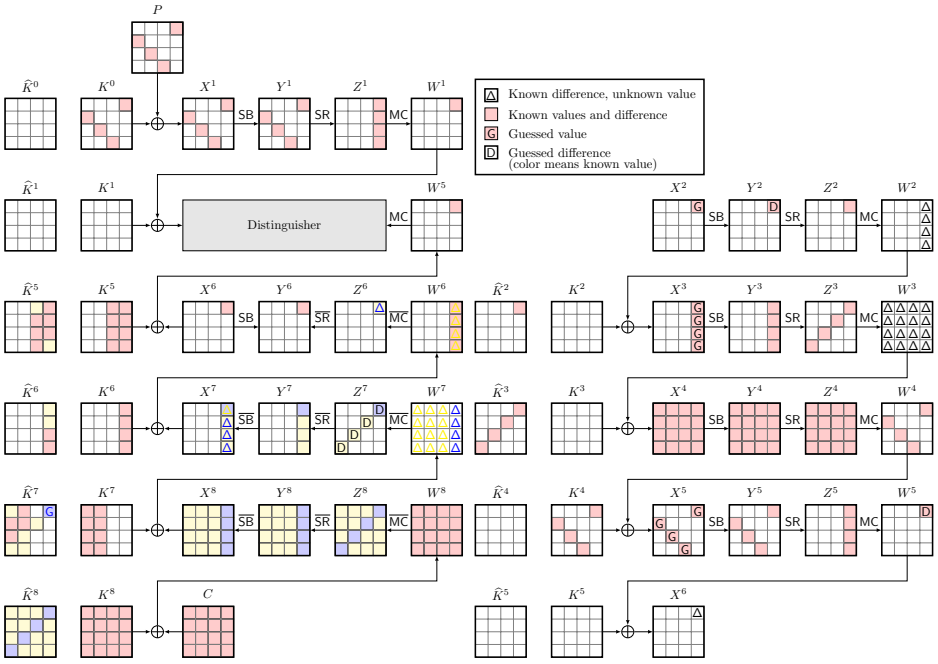
Fig. 9: Eight-round impossible-differential attack on AES-256. Left: the online key-recovery phase. Right: the impossible-differential distinguisher. Colored state cells represent active bytes in the differential trails. Colored key cells are filtered in the attack.

H.2 A Comparison with The RKP Attack in [46]

Kara's work [46] described an attack also on eight rounds of AES-256, using the characteristic from [8] instead, with

- 2^{118} known plaintexts,
- 2^{220} encryption equivalents, and
- 2^{145} state equivalents of memory.

That attack split the key-testing stage into a first one that yielded 32 pairs on average per key from the first two and the final rounds. It claimed that the "probability that a [wrong] key is not eliminated is $2^{-6 \cdot 32} = 2^{-192}$. However, we argue that the approximate calculation should be $(1 - 2^{-6})^{32} \approx 0.604$. That is, we think that the majority of all keys would not be eliminated, and the described attack would have less than a single bit of advantage with the given data complexity, which would fail to reduce the key space sufficiently. On a minor note, the memory complexity is not clear: pairs suggest incorrect keys, which must be memorized in some form. Since the known-plaintext attack in [46] did not make explicit how it kept track of wrong keys, we argue that its



(a) Online key-recovery phase. (b) Offline phase.

Fig. 10: Key-recovery phase (left) and offline phase (right) for the DS-MitM attack on eight-round AES-192.

memory complexity is in the order of storing at least 2^{192} bits for that amount of key candidates for the first filtering step, which yields at least 2^{185} state equivalents of memory and a list of pairs for each pair that would be at least $2^{192} \cdot 2^{139-118} \approx 2^{213}$ references to pairs.

I DS-MitM Attack on Eight-round AES-192

While one could also adapt existing DS-MitM attacks on nine-round AES-192 to the random known-plaintext model, the resulting data complexity would be very close to the full codebook, in the order of $2^{127.5}$ known plaintexts or higher, at the cost of close to 2^{190} encryption equivalents, when aiming at a sufficiently high probability that all related data are available. However, the descriptiveness of such attacks is limited. Striving for attacks with a more meaningful distance to the generic bounds, the following section considers a DS-MitM attack on eight rounds instead.

The attack uses a trail shown in Figure 10b with the online phase shown in Figure 10a.

Offline Phase. The offline phase uses a trail similar to that in the DS-MitM attack on seven-round AES-128, from one active byte in ΔX^2 to one active byte in ΔX^6. We fix the positions to Byte 12 in both states and can fix the number of considered single-byte differences in ΔX^2 to $\alpha < 2^8$. Then, the adversary guesses eight bytes in the active column in X^3 and the active diagonal of X^5 as in Figure 10b. Moreover, it can guess the value of the reference text in the active byte of X^2. From those bytes in X^3 and X^5, the adversary can compute the input and output differences of the SubBytes operation between ΔX^4 and ΔY^4. It obtains one solution on average through the AES S-box for the in- and output differences at ΔX^4 and ΔY^4, and traces the pair and thereupon a reduced δ-sequence of related texts through the distinguisher.

As table index, it uses $\Delta X^2[12]$, $X^6[12]$, $\Delta X^6[12]$ and $\Delta X_j^6[12]$ for subsets of indices j. For that purpose, it enumerates all $\binom{m}{k}$ subsets for $k = 9$ out of the first $m = 144$ indices, which yields $2^{45.7}$ options. Moreover, it stores all remaining values at $\Delta X_j^6[12]$ and the key material into tables $\mathcal{T}_i$, where i represents the current out of $\binom{m}{k}$ subsets.

Offline Complexity. In total, the offline computations result in 10 byte guesses and α options for $\Delta X^2[12]$, i.e. $\alpha \cdot 2^{80}$ entries for the pair and related texts, for which $1 + 4 + 16 + 4 = 25$ out of $8 \cdot 16 = 128$ S-boxes are computed. Fixing $\alpha = 2^2$, we obtain $2^2 \cdot 2^{80} \cdot 2^8 \cdot 25/128 \approx 2^{87.6}$ encryption equivalents and 2^{82} first memory accesses to store the resulting sequences and the obtained key material. For each conforming pair, the adversary stores $2^{45.7}$ options, i.e. $2^{82+45.7} \approx 2^{127.7}$ memory accesses. The memory is on average $3+16$ bytes on average plus 9 bytes for the key information, i.e. $2^{127.7} \cdot 28/16 \approx 2^{128.5}$ state equivalents of memory.

Online Phase. The adversary observes 2^d plaintext-ciphertext known pairs, inverts the final MixColumns operation to $\widehat{W}^8$, and stores the resulting $P, \Delta W^8$ pairs into a table $\mathcal{Q}$ corresponding to the values of the plaintexts in all diagonals except the third (e.g., the rightmost) diagonal. It identifies $\binom{2^d}{2} \cdot 2^{-96} \approx 2^{2d-97}$ pairs of plaintext-ciphertext pairs. Note that, given a random plaintext pair, the trail has a probability of approximately 2^{-120} to have only one active byte at ΔX^2 and approximately 2^{-120} for only one active byte at ΔX^6. Thus, there are 2^{2d-241} correct pairs. For $d = 124$, this implies 2^7 correct pairs among 2^{151} mostly wrong pair candidates. Recall that we fix the number of considered differences of $\Delta X^2[12] = \Delta W^1[12]$ from $2^8 - 1$ to α, which reduces the correct pairs to 2 on average.

For each candidate pair, the adversary guesses the α possible differences at $\Delta W^1[12]$, derives the δ-sequence, and computes backwards to their corresponding would-be plaintexts. Given k available related texts of the δ-sequence, if $k < \theta_k = 9$ among the first $m = 144$ texts, the pair is discarded. For all surviving pairs, the adversary looks up the corresponding ciphertexts C, C', and C_j. It inverts the final MixColumns operation and conducts an internal MitM attack. In the first phase, the adversary conducts the following steps:

1. It **guesses** $\Delta Z^7[3]$, and derives $\Delta Y^7[15]$ in backward, and $\Delta X^8[0..3]$ in forward direction.
2. It deduces $X^8[0..3]$ and $Y^8[0..3]$ given the in- and output differences to the S-box, and derives $Z^8[0,7,10,13]$.
3. It derives $\widehat{K}^8[0,7,10,13]$ from those bytes in Z^8 and $\widehat{W}^8$.
4. It derives $\widehat{W}^7[0..3]$ from $X^8[0..3]$.
5. It **guesses** $\Delta Z^7[6]$, and derives $\Delta Y^7[14]$ and $\Delta X^8[4..7]$.
6. It deduces $X^8[4..7]$ and $Y^8[4..7]$ given the in- and output differences to the S-box, and derives $Z^8[0,7,10,13]$.
7. It derives $\widehat{K}^8[0,7,10,13]$ from those bytes in Z^8 and $\widehat{W}^8$.
8. It derives $\widehat{W}^7[4..7]$ from $X^8[4..7]$.
9. From $\widehat{K}^8[0,1,4,7,10,11,13,14]$, it derives $\widehat{K}^7[3,6]$ and $\widehat{K}^6[12]$.
10. It derives $Z^7[3,6]$ from $\widehat{W}^7[3,6]$ and $\widehat{K}^7[3,6]$.
11. It derives $X^7[14,15]$ and $\Delta X^7[14,15]$ from $Z^7[3,6]$ and $\Delta Y^7[14,15]$.
12. It derives $\Delta W^6[12..15] = \Delta \overline{X}^7[12..15]$ from $\Delta Z^6[13..15] = (0,0,0)$ and $\Delta X^7[15]$.
13. The probability that the so-derived $\Delta \overline{X}^7[14] = \Delta X^7[14]$ is approximately 2^{-8}. If both do not match, the adversary discards the pair; otherwise, it uses $\Delta X^7[12..15] = \Delta W^6[12..15]$.
14. If the pair has not been discarded, the adversary **guesses** $\Delta Z^7[9]$, and derives $\Delta Y^7[9]$ and $\Delta X^8[8..11]$.
15. It deduces $X^8[8..11]$ and $Y^8[8..11]$ given the in- and output differences to the S-box, and derives $Z^8[2,5,8,15]$.
16. It derives $\widehat{K}^8[2,5,8,15]$ from those bytes in Z^8 and $\widehat{W}^8$.
17. It derives $\widehat{W}^7[8..11]$ from $X^8[8..11]$.
18. It deduces $X^7[13]$ and $Y^7[13]$ given the in- and output differences to the S-box.
19. It derives $Z^7[9]$ from $Y^7[13]$, and $\widehat{K}^7[9]$ from $Z^7[9]$ and $\widehat{W}^7[9]$.
20. It stores the 2^{16} non-discarded pairs on average into a table $\mathcal{S}$, indexed by $\Delta Z^6[12]$, containing the known information.

Then, the adversary conducts the second phase:

1. It **guesses** $\Delta Z^7[12]$, and derives $\Delta Y^7[12]$ and $\Delta X^8[12..15]$.
2. It deduces $X^8[12..15]$ and $Y^8[12..15]$ given the in- and output differences to the S-box, and derives $Z^8[3,6,9,12]$.
3. It derives $\widehat{K}^8[3,6,9,12]$ from those bytes in Z^8 and $\widehat{W}^8$.
4. It derives $\widehat{W}^7[12..15]$ from $X^8[12..15]$.
5. It **guesses** $\widehat{K}^7[12]$ and derives $Z^7[12]$, $X^7[12]$, and $\Delta X^7[12]$.
6. It derives $\Delta Z^6[12] = \Delta Y^6[12]$ from $\Delta Z^6[13..15] = (0,0,0)$ and $\Delta W^6[12] = \Delta X^7[12]$.

Thereupon, the adversary looks up entries in $\mathcal{S}$ and identifies on average $2^{16+16-8} = 2^{24}$ matches. For each match:

1. It knows $X^7[12..15]$ and $\widehat{K}^6[12]$, and derives $\widehat{W}^6[12..15]$ and $Z^6[12]$ from that and $\widehat{K}^6[12]$.
2. It derives $X^6[12]$ and $\Delta X^6[12]$ from $Z^6[12]$ and $\Delta Z^6[12]$.
3. It further derives $W^5[12]$ and $\Delta W^5[12]$, given $K^5[12]$ from $K^7[0,4]$.
4. For all known related texts $\widehat{W}_j^8$, it computes the corresponding values $W_j^5[12]$.

For each such sequence, it looks up in the offline table $\mathcal{T}_i$ whose i corresponds to the lexicographically first k-out-of-m combination of the available related texts. The adversary can match on $\Delta W^1[12]$ (which has only 2 bits of information), $W^5[12]$, $\Delta W^5[12]$, $\Delta W_j^5[12]$ for $k = 9$ values, and $K^4[6] = K^7[2] \oplus K^7[6]$ as well as $\widehat{K}^2[12] = \widehat{K}^7[12] \oplus \widehat{K}^8[12]$, i.e., 106 bit. There are approximately $2^{2d-97} = 2^{151}$ pairs on average, α values of $\Delta W^1[12]$, and 2^{24} guesses in the final three rounds in the online phase. The adversary looks up matches in the table corresponding to its lexicographically first nine available texts. We can expect $2^{151+2+24} \cdot 2^{82} \cdot 2^{-106} = 2^{153}$ matches. For each match, there are on average seven further texts and therefore bytes available for matching, reducing the resulting number of matches to $2^{153-56} = 2^{97}$ matches on average. Each of them provides a key candidate for the full K^8 and $\widehat{K}^7[9,12]$. Guessing the six remaining bytes and conducting a test encryption will provide the full key at $2^{97} \cdot 2^{48} = 2^{145}$ average encryption equivalents.
The probability that a configuration of at least $\theta_k = 9$ out of the first $m = 144$ related texts exists is approximately

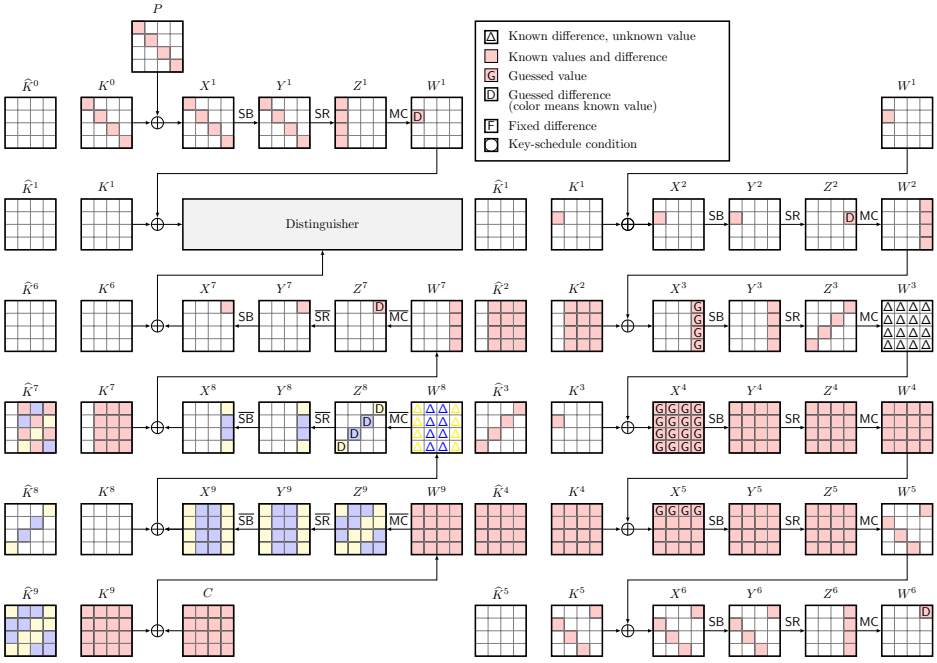
$$1 - \left(\sum_{k=0}^{\theta_k - 1} \binom{m}{k} \cdot 2^{-4k} \cdot (1 - 2^{-4})^{m-k} \right) \approx 0.559\,,$$

and the probability for no correct pair to have a good configuration is $(0.451)^2 \approx 2^{-2.3}$ on average.

Online Complexity. The online phase needs 2^{124} encryptions and memory accesses, and will identify 2^{151} pairs. The pairs will be require $2^{151} \cdot \alpha \cdot 2^8 \cdot 4/128 \approx 2^{156}$ encryption equivalents to obtain the plaintexts, and on average $2^{151} \cdot \alpha \cdot 16 = 2^{157}$ memory accesses to find the corresponding ciphertexts. Given on average 16 available related texts (plus the pair), the final three rounds need $2^{151} \cdot \alpha \cdot (2 \cdot 2^{16} + 18 \cdot 2^{24}) \cdot 21/128 \approx 2^{178.6}$ encryption equivalents and $2^{151} \cdot \alpha \cdot 2^{24} = 2^{177}$ lookups into an offline table. For pairs that have at least one configuration available, the adversary can expect

$$(\alpha \cdot 2^{151+24} \cdot 2^{82}) \cdot 2^{-106} = 2^{153}$$

matches that can be reduced to 2^{97} and tested with 2^{145} encryption equivalents on average to find the correct key. Thus, the complexity is on average $2^{124} + 2^{156} + 2^{178.6} + 2^{145} \approx 2^{178.6}$ encryption equivalents and $2^{124} + 2^{157} + 2^{177} + 2^{153} \approx 2^{177}$ memory accesses; approximating each with a full encryption equivalents yields $2^{177} + 2^{178.6} \approx 2^{179}$ encryption equivalents. The memory is dominated by 2^{124} plaintext-ciphertext pairs or 2^{125} state equivalents, producing $2^{128.6}$ state equivalents together with the offline tables.



(a) Online key-recovery phase. (b) Offline phase.

Fig. 11: Key-recovery phase (left) and offline phase (right) for the DS-MitM attack on nine-round AES-256.

J DS-MitM Attack on Nine-round AES-256

We consider a similar DS-MitM attack as on reduced AES-128 and -192, here on nine rounds of AES-256. The online and offline phases are shown in Figures 11a and 11b, respectively.

Online Phase. The online phase starts as for the DS-MitM attack on AES-192: the adversary observes 2^d known plaintext-ciphertext pairs (P, C) and stores them in a table $\mathcal{Q}$ corresponding to the values of the plaintexts in the rightmost three diagonals. It identifies $\binom{2^d}{2} \cdot 2^{-96} \approx 2^{2d-97}$ pairs of plaintext-ciphertext pairs. Note that, given a random plaintext pair, the trail has a probability of approximately 2^{-120} to have only one active byte at ΔX^2 and approximately 2^{-120} for only one active byte at ΔX^6. Given $d = 124$ and $\alpha = 2^2$, there are $2^{2d-240-(8-\alpha)} = 2$ correct pairs among 2^{151} mostly wrong pair candidates on average.

For each candidate pair, the adversary guesses the α possible differences at $\Delta W^1[1]$, derives the δ-sequence, and computes backwards to their corresponding would-be plaintexts. The adversary iterates through the first m related texts (sorted by the difference to the reference text in $W^1[1]$) and looks up if the

corresponding related plaintexts P_j exist in $\mathcal{Q}$. If less than $\theta_k = 9$ among the first $m = 144$ texts are available, the pair is discarded. For all surviving pairs, the adversary looks up the corresponding ciphertexts C, C', and C_j, inverts the final MixColumns operation, and computes to $X^7[12]$. The detailed steps are given in Algorithm 6. It constructs an entry for storage, using as the table index for an entry the following information:

1. $W^1[1]$, $\Delta W^1[1]$ (which has two bits of information), and $\Delta W^6[12]$,
2. $\Delta W_j^6[12]$ for the lexicographically first available subset of k out of the first m related text indices j,
3. as well as nine key bytes described later in Equations (17) and (18).

Thus, there are $2 + (2 + 9 + 9) \cdot 8 = 162$ bits for the table index that can be used for primary matching later. It stores all remaining available values at $\Delta W_j^6[12]$ plus the remaining of the 24 bytes of the key material (four bytes in K^0, four bytes in $\widehat{K}^8$, and the full K^9) in the online phase in a table $\mathcal{T}_i$, where i represents the current k-out-of-m combination of the available related texts.
and conducts the following steps: For each such sequence, it stores the matching information in an online table $\mathcal{T}_i$, where i represents the lexicographically first available set of θ_k out of the first m related plaintexts.

Online Complexity. The online phase needs 2^{124} encryptions and memory accesses to identify 2^{151} pairs. The adversary requires $2^{151} \cdot \alpha \cdot 2^8 \cdot 4/128 \approx 2^{156}$ encryption equivalents to obtain the would-be plaintexts of related texts, and on average $2^{151} \cdot \alpha \cdot 18 = 2^{157.2}$ memory accesses to find the corresponding ciphertexts, given on average 16 available related texts plus the reference pair. Computing the δ-sequence through the inverted final three rounds needs on average $2^{151} \cdot \alpha \cdot 18 \cdot 2^{40} \cdot 21/144 \approx 2^{194.4}$ encryption equivalents (for 18 texts on average and 21 out of $9 \cdot 16 = 144$ S-boxes in the nine-round cipher), and $2^{151} \cdot \alpha \cdot 2^{40} = 2^{193}$ accesses to store the entries into the tables. Assuming approximately normally distributed available texts, each of the $2^{45.7}$ tables contains about $2^{147.3}$ entries on average.
Thus, the online complexity is on average $2^{124} + 2^{156} + 2^{194.4} \approx 2^{194.4}$ encryption equivalents and $2^{124} + 2^{157} + 2^{193} \approx 2^{193}$ memory accesses. The memory in the online phase is approximately $2 \cdot 2^{124} + 2^{193} \cdot (162 + 15 \cdot 8)/128 = 2^{194.1}$ state equivalents.

Offline Phase. The offline phase uses a trail from a single active byte of $\Delta W^1[1]$ to a single active byte, this time through five rounds, in $\Delta W^6[12]$. The adversary guesses 26 bytes: $\Delta Z^2[13]$, $X^3[12..15]$, the full X^4, $X^5[0, 4, 8, 12]$, and $\Delta W^6[12]$. The detailed computations by the adversary in the offline phase are as follows:

- From $\Delta Z^2[12]$ and $X^3[12..15]$, compute to the full $\Delta W^3 = \Delta X^4$.
- Together with the guessed X^4, compute the full W^4 and ΔX^5.
- From the guess of $X^5[0, 4, 8, 12]$, compute $Z^5[0, 4, 8, 12]$ and $\Delta Z^5[0, 4, 8, 12]$.
- By requiring only the four bytes in the first diagonal of ΔW^5 to be active, derive the full difference ΔZ^5, and thus ΔY^5.

Algorithm 6 Online phase of the RKP DS-MitM attack on nine-round AES-256.

function $\mathsf{computeToX}^7(C, C')$
  Derive ΔY^9 from $C \oplus C'$
  for all guesses of $\Delta Z^8[3, 12]$ **do**
    Derive $\Delta Y^8[12, 15]$ in backward, and $\Delta X^9[0..3, 12..15]$ in forward direction
    Deduce $X^9[0..3, 12..15]$ and $Y^9[0..3, 12..15]$ from $\Delta X^9[0..3, 12..15]$ and ΔY^9
    Derive $\widehat{W}^8[0..3, 12..15]$, $Z^9[0, 3, 6, 7, 9, 10, 12, 13]$, and $\widehat{K}^9[0, 3, 6, 7, 9, 10, 12, 13]$ from Z^9 and $\widehat{W}^9$
    for all guesses of $\Delta Z^8[6, 9]$ **do**
      Derive $\Delta Y^8[6, 9]$ and $\Delta X^9[4..11]$
      Deduce $X^9[4..11]$ and $Y^9[4..11]$ from $\Delta X^9[4..11]$ and $\Delta Y^9[4..11]$
      Derive $Z^8[1, 2, 4, 5, 8, 11, 14, 15]$
      Derive $\widehat{K}^9[1, 2, 4, 5, 8, 11, 14, 15]$ from those bytes in Z^9 and $\widehat{W}^9$
      Derive $\widehat{W}^7[4..11]$ from $X^8[4..11]$
      for all guesses of $\Delta Z^7[12]$ **do**
        Derive $\Delta W^7[12..15] = \Delta X^8[12..15]$
        Deduce $X^8[12..15]$ and $Y^8[12..15]$ from $\Delta X^8[12..15]$ and $\Delta Y^8[12..15]$
        Derive $Z^8[3, 6, 9, 12]$
        From $\widehat{W}^8$ and $Z^8[3, 6, 9, 12]$, derive those bytes in $\widehat{K}^8[3, 12]$
        From $\widehat{K}^9$, derive K^9, then $K^7[4..15]$ from the key schedule
        Derive $W^7[12..15]$, and $Z^7[12]$ to $X^7[12]$ and $X'^7[12]$
  return $X^7[12]$, $X'^7[12]$, $K^7[4..15]$, $\widehat{K}^8[3, 6, 9, 12]$, K^9

function $\mathcal{A}_{\mathsf{online}}(\mathcal{Q}, \mathcal{P})$
  Initialize empty tables $\mathcal{T}_i \leftarrow \emptyset$ for all $i \in 1..\binom{m}{\theta_k}$
  for all candidate pairs $(P, P') \in \mathcal{P}$ **do**
    for all α guesses of $\Delta W^1[12]$ **do**
      Compute from P, P' to $W^1[12]$, $W'^1[12]$
      Derive the δ-sequence at $W_j^1[12]$
      Compute backwards to their corresponding would-be plaintexts P_j
      Lookup the available P_j's in $\mathcal{Q}$
      Let k be the #available P_j's for the current δ-set
      if $k < \theta_k$ **then** discard the pair
      Construct the lexicographically first θ_k-out-of-m reduced δ-sequence from the available plaintexts P_j
      Lookup the corresponding ciphertexts C_j
      $X^7[12], \ldots, K^9 \leftarrow \mathsf{computeToX}^7(C, C')$
      Compute from C_j to $X_j^7[12]$ for all available $j \in \{0, \ldots, 255\}$
      Store the resulting reduced δ-sequence, alongside the induced key material into $\mathcal{T}_i$, indexed by i, its lexicographically first θ_k available text indices, and the byte vector $\Delta X_j^6[12]$
  return The set of sorted tables $\{\mathcal{T}_i\}_i$

- Deduce X^5 from ΔX^5 and ΔY^5, and derive the full ΔX^6.
- Derive $\Delta Y^6[1,6,11,12]$ from $\Delta W^6[12]$, and derive the values for X^6 and Y^6 in those bytes.
- Compute from Y^6 to $W^6[12]$.
- Derive K^4 from X^5 and W^4, $K^5[1,6,11,12]$ from those bytes in W^5 and X^6, $K^2[4..15]$ from K^4 and the key schedule, $\widehat{K}^3[3,6,9,12]$ from those bytes in Z^3 and full X^4, $K^3[0]$ from $S(K^4[13])$ and $K^5[1]$, and $K^1[1]$ from $S(K^2[13])$ and $K^3[1]$.
- Derive further $W^1[1]$ and $\Delta W^1[1]$.
- Discard the current trail if $\Delta W^1[1]$ is not among the α constrained set of differences.
- Compute all related 2^8 texts to $W_j^6[12]$.

The adversary enumerates all $\binom{m}{k}$ subsets for $k = 9$ out of the first $m = 144$ indices, which yields $2^{45.7}$ options, and looks of for matches in the $2^{45.7}$ tables $\mathcal{T}_i$, on 162 bits. Given 2^{202} offline entries, $2^{45.7}$ options, $2^{193-45.7}$ entries in a fixed table, and 162 bits for matching, one can expect $2^{193+202-162} = 2^{233}$ matches on average.

Algorithm 7 Offline phase of the RKP DS-MitM attack on nine-round AES-256.

```
51: function A_offline({T_i}_i)
52:     for all guesses of 26 internal byte variables do
53:         Trace the solutions from ΔW^1[12] to ΔW^6[12]   ▷ One solution on average
54:         Trace the texts in a δ-sequence through the distinguisher to W_j^6[12]
55:         Construct the matching vector for all texts at W_j^6[12]
56:         for all k-reductions of the δ-sequence i do
57:             Isolate the matching information vector for the reduced δ-sequence x
58:             if x ∉ T_i then continue
59:             for all matches x' do
60:                 Extract the information of more available texts and match on those
61:                 if still matches then
62:                     Combine K^8 ‖ K^9 from the match and the current entry
63:                     Derive a key candidate K = K^0 ‖ K^1 with the key schedule
64:                     if a test encryption succeeds under K then
65:                         if a second test encryption succeeds under K then
66:                             return K
67:     return ⊥
```

```
51: function A
52:     Collect 2^124 known plaintext-ciphertext pairs (P, C)
53:     Store them in a table Q, according to the values in the plaintext in the diagonals
        except for the first one
54:     Identify approximately C(2^124, 2) · 2^-96 ≈ 2^153 pairs (P, C)
55:     Store them into a table P
56:     {T_i}_i ← A_online(Q, P)
57:     return A_offline({T_i}_i)
```

and conducts the following steps: For each such sequence, it stores the matching information in an online table $\mathcal{T}_i$, where i represents the lexicographically first available set of θ_k out of the first m related plaintexts.

Matching. Figure 12 shows the combined known information about round keys and equivalent round keys from an entry in the offline tables (in red) and a candidate from the online phase (in blue). From the full (red) K^4 and full (blue) K^9, they can recover the full K^8 and thus the full secret key. Since both online and offline phases contain further independent variables (indicated as G in Figure 12), at least nine further independent byte variables can be easily derived linearly from parts of the offline phase (red, left-hand side) and the online phase (blue, right-hand side):

$$\begin{aligned}
\widehat{K}^3[12] &= \widehat{K}^9[0] \oplus \widehat{K}^9[4] \oplus \widehat{K}^9[8] \oplus \widehat{K}^9[12] \\
K^4[2] \oplus K^4[10] &= K^0[10] \\
K^4[7] \oplus K^4[15] &= K^0[15] \\
K^5[12] &= K^9[4] \oplus K^9[12] \\
K^5[11] &= K^9[3] \oplus K^9[11]\,.
\end{aligned} \tag{17}$$

Moreover, the (blue) values in $\widehat{K}^8[3,6,9,12]$ can be used for matching as follows:

$$\begin{aligned}
\overline{\mathsf{MC}}(K^4[4..7] \oplus K^4[12..15])[0] &= \overline{\mathsf{MC}}(S(K^9[13,14,15,12] \oplus K^9[9,10,11,8]))[0] \oplus \\
&\quad \widehat{K}^8[12] \\
\overline{\mathsf{MC}}(K^4[0..3] \oplus K^4[8..11])[1] &= \overline{\mathsf{MC}}(S(K^9[5,6,7,4] \oplus K^9[13,14,15,12]) \oplus \\
&\quad S(K^9[9,10,11,8] \oplus K^9[13,14,15,12]))[1] \oplus \widehat{K}^8[9] \\
\overline{\mathsf{MC}}(K^4[4..7])[2] &= \overline{\mathsf{MC}}(S(K^9[9,10,11,8] \oplus K^9[13,14,15,12]))[2] \oplus \\
&\quad \widehat{K}^8[6] \\
\overline{\mathsf{MC}}(K^4[0..3])[3] &= \overline{\mathsf{MC}}(S(K^9[5,6,7,4] \oplus K^9[13,14,15,12]) \oplus \\
&\quad S(K^9[9,10,11,8] \oplus K^9[13,14,15,12]))[2] \oplus \widehat{K}^8[3]\,,
\end{aligned} \tag{18}$$

where $\overline{\mathsf{MC}}(x)[i]$ returns y_i where $x = (x_0, x_1, x_2, x_3) \in \mathbb{F}_{2^8}^4$ and $y = (y_0, y_1, y_2, y_3)^\top = \mathbf{M}^{-1} \cdot x^\top$ with $\mathbf{M}^{-1}$ the inverse $\mathsf{MixColumns}$ matrix of the AES.
For each match, there are on average seven further texts and therefore bytes available for matching in the online entry, reducing the resulting number of matches further to $2^{233-56} = 2^{177}$ on average. Each match allows for constructing the full key from $K^8 \| K^9$. Those allow to identify wrong matches by one full encryption in most cases, and to identify the $2^{177-128}$ surviving wrong matches with another full encryption. The probability that a configuration of at least $\theta_k = 9$ out of the first $m = 144$ related texts exists is approximately 0.559 as before on average, and the probability for no correct pair to have a good configuration is $(0.451)^2 \approx 2^{-2.3}$ on average.

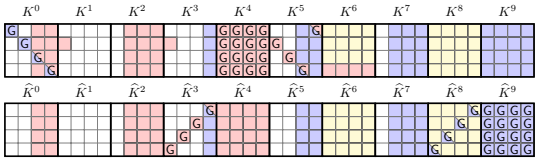


Fig. 12: Key relations. Red = offline phase, blue = online phase, yellow = linearly dependent on known variables from the online and offline phases; G = guessed.

Offline Complexity. In total, the offline computations result in 26 byte guesses, where we limit the options for $\Delta X^2[12]$ to α, i.e. $\alpha \cdot 2^{200}$ entries for the pair and related texts, for which $1 + 4 + 16 + 16 + 4 = 41$ out of $9 \cdot 16 = 144$ S-boxes are computed. Fixing $\alpha = 2^2$, we obtain $(2^{26 \cdot 8} \cdot 2 + 2^2 \cdot 2^{25 \cdot 8} \cdot 2^8) \cdot 41/144 \approx 2^{207.2}$ encryption equivalents. The adversary conducts less than $2^{202} \cdot 2^{45.7} = 2^{247.7}$ accesses into huge tables to identify matches, plus 2^{233} key-schedule computations of a few S-boxes and matches of the further texts, which can be approximated by a full encryption, plus $2^{177} + 2^{49}$ encryptions on average. Thus, the computational complexity can be approximated by $2^{207.2} + 2^{233} + 2^{177} + 2^{49} \approx 2^{233}$ encryption equivalents, plus $2^{247.7}$ memory accesses; approximating the latter by one encryption equivalent each produces the dominant term of the computational complexity of the attack.