XPX: Generalized Tweakable Even-Mansour with Improved Security Guarantees

Bart Mennink

Dept. Electrical Engineering, ESAT/COSIC, KU Leuven, and iMinds, Belgium
bart.mennink@esat.kuleuven.be

Abstract. We present XPX, a tweakable blockcipher based on a single permutation P. On input of a tweak $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ and a message m, it outputs ciphertext $c = P(m \oplus \Delta_1) \oplus \Delta_2$, where $\Delta_1 = t_{11}k \oplus t_{12}P(k)$ and $\Delta_2 = t_{21}k \oplus t_{22}P(k)$. Here, the tweak space $\mathcal{T}$ is required to satisfy a certain set of trivial conditions (such as $(0,0,0,0) \notin \mathcal{T}$). We prove that XPX with any such tweak space is a strong tweakable pseudorandom permutation. Next, we consider the security of XPX under related-key attacks, where the adversary can freely select a key-deriving function upon every evaluation. We prove that XPX achieves various levels of related-key security, depending on the set of key-deriving functions and the properties of $\mathcal{T}$. For instance, if $t_{12}, t_{22} \neq 0$ and $(t_{21}, t_{22}) \neq (0, 1)$ for all tweaks, XPX is XOR-related-key secure.

XPX generalizes Even-Mansour (EM), but also Rogaway's XEX based on EM, and tweakable EM used in Minalpher. As such, XPX finds a wide range of applications. We show how our results on XPX directly imply related-key security of the authenticated encryption schemes Prøst-COPA and Minalpher, and how a straightforward adjustment to the MAC function Chaskey and to keyed Sponges makes them provably related-key secure.

Keywords. XPX, XEX, Even-Mansour, tweakable blockcipher, related-key security, Prøst, COPA, Minalpher, Chaskey, Keyed Sponges.

1 Introduction

Even-Mansour Blockcipher. A blockcipher $E : \mathcal{K} \times \{0,1\}^n \to \{0,1\}^n$ is a function that is a permutation on $\{0,1\}^n$ for every key $k \in \mathcal{K}$. The simplest way of designing a blockcipher is the Even-Mansour construction [19, 20]: it is built on top of a single n-bit permutation P:

$$\mathrm{EM}_{k_1,k_2}(m) = P(m \oplus k_1) \oplus k_2\,. \tag{1}$$

See also Figure 1. In the classical indistinguishability security model, this construction achieves security up to approximately $2^{n/2}$ queries, both for the case where the keys are independent [19, 20] as well as for the case where $k_1 = k_2$ [18]. On the downside, this construction clearly does not achieve security against related-key distinguishers that may freely choose an offset δ to transform the key. Indeed, for any $\delta \neq 0$, we have $\mathrm{EM}_{k_1,k_2}(m) = \mathrm{EM}_{k_1\oplus\delta,k_2}(m \oplus \delta)$. Recently, Farshim and Procter [21] and Cogliati and Seurin [14] reconsidered the security of Even-Mansour in the related-key security model. The former considered the case of $k_1 = k_2$, and derived minimal conditions on the set of key-deriving functions such that EM is related-key secure. The latter showed that if $k_1 = \gamma_1(k)$ and $k_2 = \gamma_2(k)$ for two almost perfect nonlinear permutations γ_1, γ_2 [34], the construction is XOR-related-key secure. Karpman showed how to transform related-key distinguishing attacks on EM to key recovery attacks [23].

Even though our focus is on the single-round Even-Mansour (1), we briefly elaborate on its generalization, the iterated $r \geq 1$ round Even-Mansour construction:

$$\mathrm{EM}[r]_{k_1,\ldots,k_{r+1}}(m) = P_r(\cdots P_1(m \oplus k_1) \cdots \oplus k_r) \oplus k_{r+1}\,,$$

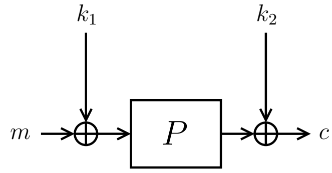

Fig. 1: EM

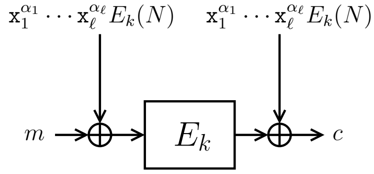

Fig. 2: XEX

where $P_1, \ldots, P_r$ are n-bit permutations. It has been proved that this construction tightly achieves $\mathcal{O}(2^{rn/(r+1)})$ security in the single-key indistinguishability model [8,12,13,25,39]. It has furthermore been analyzed in the chosen-key indifferentiability model [2,26], and the related-key indistinguishability model [14,21]. As our work centers around the 1-round Even-Mansour of (1), we will not discuss these results in detail; we refer to Cogliati and Seurin [14] for a recent and complete discussion of the state of the art.

Tweakable Blockciphers. A tweakable blockcipher $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ generalizes over E by ways of an additional parameter, the tweak $t \in \mathcal{T}$. The tweak is a public parameter which brings additional flexibility to the cipher. In more detail, $\widetilde{E}$ is a family of permutations on $\{0,1\}^n$, indexed by $(k,t) \in \mathcal{K} \times \mathcal{T}$. Liskov et al. [29] formalized the principle of tweakable blockciphers, and introduced two modular constructions based on a classical blockcipher. One of their proposals is the following:

$$\mathrm{LRW}_{k,h}(t,m) = E_k(m \oplus h(t)) \oplus h(t)\,,$$

where h is a universal hash function taken from a family of hash functions H. This construction is proven to achieve security up to $2^{n/2}$ queries. Rogaway [37] introduced XEX: it generalizes over LRW by eliminating the universal hash function (and thus by halving the key size) and by replacing it by an efficient tweaking mechanism based on E_k. In more detail, he suggested the use of masking $\varDelta = \mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell} E_k(N)$ for some pre-defined generators $\mathtt{x}_1, \ldots, \mathtt{x}_\ell \in \mathrm{GF}(2^n)$:

$$\mathrm{XEX}_k((\alpha_1, \ldots, \alpha_\ell, N), m) = E_k(m \oplus \varDelta) \oplus \varDelta\,. \tag{2}$$

If the generators and the tweak space are defined such that the $\mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell}$ are unique and unequal to 1 for all tweaks, XEX achieves birthday bound security [10,31,37]. Along with XEX, Rogaway also considered XE, its cousin which only masks the inputs to E and achieves PRP instead of SPRP security. Here, ℓ is usually a small number, and the generators and the tweak space are defined in such a way that adjusting the tweak is very cheap. For instance, practical applications with $n = 128$ often take $\ell \leq 3$ and $(\mathtt{x}_1, \mathtt{x}_2, \mathtt{x}_3) = (2,3,7)$, and an allowed tweak space would be $[1, 2^{n/2}] \times [0,10] \times [0,10] \times \{0,1\}^n$.

Sasaki et al. [38] recently introduced the "Tweakable Even-Mansour" (TEM) for the purpose of the Minalpher authenticated encryption scheme. TEM is a variant of XEX with E_k replaced by a public permutation P:

$$\mathrm{TEM}_k((\alpha_1, \ldots, \alpha_\ell, N), m) = P(m \oplus \varDelta) \oplus \varDelta\,, \tag{3}$$

where $\varDelta = \mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell} \big(k \| N \oplus P(k \| N)\big)$ for some generators $\mathtt{x}_1, \ldots, \mathtt{x}_\ell \in \mathrm{GF}(2^n)$. (The masking is in fact slightly different, but adjusted for the sake of presentation; cf. Section 6.3 for the details.)

These constructions all achieve approximately birthday bound security, and extensive research has been performed on achieving beyond birthday bound security for tweakable blockciphers [27, 28, 30, 32, 36]. Because this is out of scope for this article, we will not go into detail; we refer to Mennink [30] for a recent and complete discussion of the state of the art.

Application of XEX and TEM. Tweakable blockciphers find a wide spectrum of applications, most importantly in the area of authenticated encryption and message authentication. For instance, XEX has been originally introduced for the authenticated encryption scheme OCB1 and the message authentication code PMAC [37], and its idea has furthermore been adopted in 17 out of 57 initial submissions to the CAESAR [9] competition for the design of a new authenticated encryption scheme: AEZ, CBA, COBRA, COPA, Deoxys, ELmD, iFeed, Joltik, KIASU, Marble, OCB, OTR, POET, and SHELL are directly inspired by XE or XEX; OMD transforms XE to a random function setting; and Minalpher uses TEM. Finally, the Prøst submission is simply a permutation P, which is (among others) plugged into COPA and OTR in an Even-Mansour mode. We note that OTR internally uses XE, while COPA uses XEX with $N = 0$ (see also Section 6.2).

Related-Key Security of XEX and TEM. XEX resists related-key attacks if the underlying blockcipher is sufficiently related-key secure. However, this premise is not necessarily true if Even-Mansour is plugged into XEX, as is done in Prøst-COPA and Prøst-OTR. In fact, Dobraunig et al. [17] derived a related-key attack on Prøst-OTR. This attack uses that the underlying XE-with-EM construction is not secure under related-key attacks, and it ultimately led to the withdrawal of Prøst-OTR. The attack exploits the nonce N that is used in the masking. Karpman [23] generalized the attack to a key recovery attack. Because COPA uses XEX without nonce (hence with $N = 0$), the attack of Dobraunig et al. does not seem to be directly applicable to Prøst-COPA. Nevertheless, it is unclear whether a variant of it generalizes to Prøst-COPA.

1.1 Our Contribution

We present the tweakable blockcipher XPX. It can be seen as a generalization of TEM as well as of XEX with integrated Even-Mansour, and due to its generality it has direct implications for various schemes in literature. In more detail, XPX is a tweakable blockcipher based on an n-bit permutation P. It has a key space $\{0,1\}^n$, a tweak space $\mathcal{T} \subseteq (\{0,1\}^n)^4$ (see below), and a message space $\{0,1\}^n$. It is defined as

$$\mathrm{XPX}_k((t_{11}, t_{12}, t_{21}, t_{22}), m) = P(m \oplus \Delta_1) \oplus \Delta_2 \,,$$

with $\Delta_1 = t_{11}k \oplus t_{12}P(k)$ and $\Delta_2 = t_{21}k \oplus t_{22}P(k)$. Note that XPX boils down to the original Even-Mansour blockcipher by taking $\mathcal{T}_{\mathrm{EM}} = \{(1, 0, 1, 0)\}$. It also generalizes XEX based on Even-Mansour and with $N = 0$, by defining $\mathcal{T}_{\mathrm{XEX}}$ to be a tweak space depending on $(\alpha_1, \ldots, \alpha_\ell)$ (cf. Section 3 for the details).

Valid Tweak Sets. Obviously, XPX is not secure for any possible tweak space $\mathcal{T}$. For instance, if $(0, 0, 0, 0) \in \mathcal{T}$, the scheme is trivially insecure. Also, if $(1, 0, 0, 1) \in \mathcal{T}$, an attacker can easily distinguish by observing that $\mathrm{XPX}_k((1, 0, 0, 1), 0) = 0$. Therefore, it makes sense to limit the tweak space in some way, and we define the notion of valid tweak spaces. This condition eliminates the trivial cases (such as above two) and allows us to focus on the "interesting" tweaks. We remark that $\mathcal{T}_{\mathrm{EM}}$ and $\mathcal{T}_{\mathrm{XEX}}$ are valid tweak spaces.

Single-Key Security. As a first step, we consider the security of XPX in the traditional single-key indistinguishability setting, and we prove that if $\mathcal{T}$ is a valid set, then XPX achieves strong PRP (SPRP) security up to about $2^{n/2}$ queries. The proof is performed in the ideal permutation model, and uses Patarin's H-coefficient technique [35] which has found recent adoption in, among others, generic blockcipher analysis [12–15, 30] and security of message authentication algorithms [4, 16, 33].

Related-Key Security. Next, we consider the security of XPX in the related-key setting, where for every query, the adversary can additionally choose a function to transform the key. We focus on the following two types of key-deriving function sets:

- $\Phi_{\oplus}$: the set of functions that map k to $k \oplus \delta$, for any offset δ;
- $\Phi_{P\oplus}$: the set of functions that map either k to $k \oplus \delta$ or $P(k)$ to $P(k) \oplus \delta$, for any offset δ.

The first set, $\Phi_{\oplus}$, has been formally introduced alongside the formal specification of related-key security by Bellare and Kohno [5]. It is the most logical choice, given that the maskings in XPX itself are XORed into the state. We remark that Cogliati and Seurin [14] also use $\Phi_{\oplus}$ in their related-key analysis of Even-Mansour. The second set, $\Phi_{P\oplus}$, is a natural generalization of $\Phi_{\oplus}$, noting that the masks in XPX are of the form $t_{i1}k \oplus t_{i2}P(k)$. For the case of $\Phi_{P\oplus}$, we assume that the underlying permutation is available for the key-deriving functions. Albrecht et al. [1] showed how to generalize the setting of Bellare and Kohno [5] to primitive-dependent key-deriving functions. In this work, we consider the related-key security for XPX in a security model that is a straightforward generalization of the models of Bellare and Kohno and Albrecht et al. to tweakable blockciphers.

For the two key-deriving sets $\Phi_{\oplus}$ and $\Phi_{P\oplus}$, we show that XPX achieves the following levels of related-key security:

if $\mathcal{T}$ is valid, and for all tweaks:	security	rk
$t_{12} \neq 0$	PRP	$\Phi_{\oplus}$
$t_{12}, t_{22} \neq 0$ and $(t_{21}, t_{22}) \neq (0, 1)$	SPRP	$\Phi_{\oplus}$
$t_{11}, t_{12} \neq 0$	PRP	$\Phi_{P\oplus}$
$t_{11}, t_{12}, t_{21}, t_{22} \neq 0$	SPRP	$\Phi_{P\oplus}$

In brief, if $P(k)$ does not drop from the masking Δ_1 (resp. maskings Δ_1, Δ_2) the scheme achieves PRP (resp. SPRP) related-key security under $\Phi_{\oplus}$. To achieve related-key security under $\Phi_{P\oplus}$, we require that this condition holds for both k and $P(k)$. The requirement "$(t_{21}, t_{22}) \neq (0, 1)$" is technically equivalent to the requirement for XEX that $\mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell} \neq 1$ for all tweaks: if the conditions were violated, both schemes can be attacked in a similar way.

1.2 Applications

XPX appears in many constructions or modes (either directly or indirectly), and our findings have natural implications. We exemplify this for authenticated encryption and for message authentication codes.

Firstly, Prøst-COPA is related-key secure for both key-deriving function sets $\Phi_{\oplus}$ and $\Phi_{P\oplus}$. The crux behind this observation is that the XEX-with-EM evaluations in Prøst-COPA are in fact XPX evaluations with $t_{11}, t_{12}, t_{21}, t_{22} \neq 0$ for all tweaks. A similar observation can be made for Minalpher, with an additional technicality that the key k in TEM is not of full size.

Secondly, we consider the Chaskey permutation-based MAC function by Mouha et al. [33]. We first note that the proof of [33] is implicitly using XPX with a tweak space of size $|\mathcal{T}| = 3$. Next, we show that if we slightly adjust Chaskey to use permuted key $P(k)$ instead of k, it achieves XOR-related-key security. Similar findings can be made for keyed Sponges.

1.3 Outline

Section 2 introduces preliminary notation as well as the security models targeted in this work. XPX is introduced in Section 3. In Section 4, the notion of valid tweak spaces is defined and justified. XPX is analyzed for the various security models in Section 5. We apply the results on XPX to authenticated encryption in Section 6 and to MACs in Section 7.

2 Preliminaries

By $\{0,1\}^n$ we denote the set of bit strings of length n. Let $\mathrm{GF}(2^n)$ be the field of order 2^n. We identify bit strings from $\{0,1\}^n$ and finite field elements in $\mathrm{GF}(2^n)$. This is done by representing a string $a = a_{n-1}a_{n-2}\cdots a_1 a_0 \in \{0,1\}^n$ as polynomial $a(\mathtt{x}) = a_{n-1}\mathtt{x}^{n-1} + a_{n-2}\mathtt{x}^{n-2} + \cdots + a_1\mathtt{x} + a_0 \in \mathrm{GF}(2^n)$ and vice versa. There is additionally a one-to-one correspondence between $[0, 2^n - 1]$ and $\{0,1\}^n$, by considering $a(2) \in [0, 2^n - 1]$. For $a, b \in \{0,1\}^n$, we define addition $a \oplus b$ as addition of the polynomials $a(\mathtt{x}) + b(\mathtt{x}) \in \mathrm{GF}(2^n)$. Multiplication $a \otimes b$ is defined with respect to the irreducible polynomial $f(\mathtt{x})$ used to represent $\mathrm{GF}(2^n)$: $a(\mathtt{x}) \cdot b(\mathtt{x}) \bmod f(\mathtt{x})$.

For integers $a \geq b \geq 1$, we denote by $(a)_b = a(a-1)\cdots(a-b+1) = \frac{a!}{(a-b)!}$ the falling factorial power. If $\mathcal{M}$ is some set, $a \xleftarrow{\$} \mathcal{M}$ denotes the uniformly random drawing of m from $\mathcal{M}$. The size of $\mathcal{M}$ is denoted by $|\mathcal{M}|$. By $\mathsf{Perm}(\mathcal{M})$ we denote the set of all permutations on $\mathcal{M}$.

A blockcipher $E : \mathcal{K} \times \mathcal{M} \to \mathcal{M}$ is a function such that for every key $k \in \mathcal{K}$, the mapping $E_k(\cdot) = E(k, \cdot)$ is a permutation on $\mathcal{M}$. For fixed k its inverse is denoted by $E_k^{-1}(\cdot)$. A tweakable blockcipher $\widetilde{E}$ is a function $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ such that for every $k \in \mathcal{K}$ and tweak $t \in \mathcal{T}$, the mapping $\widetilde{E}_k(t, \cdot) = \widetilde{E}(k, t, \cdot)$ is a permutation on $\mathcal{M}$. Like before, its inverse is denoted by $\widetilde{E}_k^{-1}(\cdot, \cdot)$. Denote by $\widetilde{\mathsf{Perm}}(\mathcal{T}, \mathcal{M})$ the set of tweakable permutations, i.e., the set of all families of permutations on $\mathcal{M}$ indexed with $t \in \mathcal{T}$.

Note that a blockcipher is a special case of a tweakable blockcipher with $|\mathcal{T}| = 1$, and hence it suffices to restrict our analysis to tweakable blockciphers. In this work, we target the design of a tweakable blockcipher $\widetilde{E}$ from an underlying permutation P, which is modeled as a perfectly random permutation $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M})$. In Section 2.1 we describe the single-key security model and in Section 2.2 the related-key security model. We give a description of Patarin's technique for bounding distinguishing advantages in Section 2.3.

2.1 Single-Key Security Model

Consider a tweakable blockcipher $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ based on a random permutation $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M})$. Let $\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T}, \mathcal{M})$ be an ideal tweakable permutation. The single-key security of $\widetilde{E}$ is informally captured by a distinguisher $\mathcal{D}$ that has adaptive oracle access to either $(\widetilde{E}_k, P)$, for some secret key $k \xleftarrow{\$} \mathcal{K}$, or $(\widetilde{\pi}, P)$. The distinguisher always has two-directional access to P. It may or may not have two-directional access to the construction oracle ($\widetilde{E}_k$ or $\widetilde{\pi}$) depending on whether we consider PRP or strong PRP security. The distinguisher is computationally unbounded, deterministic, and it never makes duplicate queries.

Security Definitions. More formally, we define the PRP security of $\widetilde{E}$ based on P as

$$\mathbf{Adv}_{\widetilde{E}}^{\mathrm{prp}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\widetilde{E}_k,P^{\pm}} = 1\right] - \mathbf{Pr}\left[\mathcal{D}^{\widetilde{\pi},P^{\pm}} = 1\right]\right| ,$$

and the strong PRP (SPRP) security of $\widetilde{E}$ based on P as

$$\mathbf{Adv}_{\widetilde{E}}^{\mathrm{sprp}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\widetilde{E}_k^{\pm},P^{\pm}} = 1\right] - \mathbf{Pr}\left[\mathcal{D}^{\widetilde{\pi}^{\pm},P^{\pm}} = 1\right]\right| ,$$

where the probabilities are taken over the random selections of $k \xleftarrow{\$} \mathcal{K}$, $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M})$, and $\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T},\mathcal{M})$. For $q, r \geq 0$, we define by

$$\mathbf{Adv}_{\widetilde{E}}^{\mathrm{(s)prp}}(q,r) = \max_{\mathcal{D}} \mathbf{Adv}_{\widetilde{E}}^{\mathrm{(s)prp}}(\mathcal{D})$$

the security of $\widetilde{E}$ against any single-key distinguisher $\mathcal{D}$ that makes q queries to the construction oracle ($\widetilde{E}_k$ or $\widetilde{\pi}_k$) and r queries to the primitive oracle.

2.2 Related-Key Security Model

We generalize the security definitions of Section 2.1 to related-key security using the theoretical framework of Bellare and Kohno [5] and Albrecht et al. [1]. The generalization is similar to the one of Cogliati and Seurin [14] with the difference that tweakable blockciphers are considered (and that we consider more general key-deriving functions).

Related-Key Oracle. In related-key attacks, the distinguisher may query its construction oracle not just on $\widetilde{E}_k$, but on $\widetilde{E}_{\varphi(k)}$ for some function φ chosen by the distinguisher. This function may vary for the different construction queries, but should come from a prescribed set. Let Φ be a set of key-deriving functions (a *KDF-set*). For a tweakable blockcipher $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$, we define a *related-key oracle* $\mathsf{RK}[\widetilde{E}] : \mathcal{K} \times \Phi \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ as

$$\mathsf{RK}[\widetilde{E}](k,\varphi,t,m) = \mathsf{RK}[\widetilde{E}]_k(\varphi,t,m) = \widetilde{E}_{\varphi(k)}(t,m) \,.$$

For fixed φ its inverse is denoted $\mathsf{RK}[\widetilde{E}]_k^{-1}(\varphi,t,c) = \widetilde{E}_{\varphi(k)}^{-1}(t,c)$. Denote by $\widetilde{\mathsf{RK\text{-}Perm}}(\Phi,\mathcal{T},\mathcal{M})$ the set of tweakable related-key permutations, i.e., the set of all families of permutations on $\mathcal{M}$ indexed with $(\varphi,t) \in \Phi \times \mathcal{T}$.

Security Definitions. For a KDF-set Φ, we define the related-key (strong) PRP (RK-(S)PRP) security of $\widetilde{E}$ based on P as

$$\mathbf{Adv}_{\Phi,\widetilde{E}}^{\text{rk-prp}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\mathsf{RK}[\widetilde{E}]_k,P^{\pm}} = 1\right] - \mathbf{Pr}\left[\mathcal{D}^{\widetilde{\mathsf{RK}\pi},P^{\pm}} = 1\right]\right| ,$$

$$\mathbf{Adv}_{\Phi,\widetilde{E}}^{\text{rk-sprp}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\mathsf{RK}[\widetilde{E}]_k^{\pm},P^{\pm}} = 1\right] - \mathbf{Pr}\left[\mathcal{D}^{\widetilde{\mathsf{RK}\pi}^{\pm},P^{\pm}} = 1\right]\right| ,$$

where the probabilities are taken over the random selections of $k \xleftarrow{\$} \mathcal{K}$, $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M})$, and $\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK\text{-}Perm}}(\Phi,\mathcal{T},\mathcal{M})$. For $q, r \geq 0$, we define by

$$\mathbf{Adv}_{\Phi,\widetilde{E}}^{\text{rk-(s)prp}}(q,r) = \max_{\mathcal{D}} \mathbf{Adv}_{\Phi,\widetilde{E}}^{\text{rk-(s)prp}}(\mathcal{D})$$

the security of $\widetilde{E}$ against any related-key distinguisher $\mathcal{D}$ that makes q queries to the construction oracle ($\mathsf{RK}[\widetilde{E}]_k$ or $\widetilde{\mathsf{RK}\pi}$) and r queries to the primitive oracle. If for some $k \in \mathcal{K}$ there exist two distinct $\varphi, \varphi' \in \Phi$ such that $\varphi(k) = \varphi'(k)$ with non-negligible probability, $\widetilde{\mathsf{RK}\pi}_k$ behaves as two independent tweakable permutations for these two key-deriving functions but $\mathsf{RK}[\widetilde{E}]_k$ does not. In this case, $\mathcal{D}$ can easily distinguish (it corresponds to the collision-resistance property in [5]).

Key-Deriving Functions. Note that for $\Phi_{\mathrm{id}} = \{\varphi : k \mapsto k\}$, we simply have $\mathbf{Adv}^{\text{rk-(s)prp}}_{\Phi_{\mathrm{id}},\widetilde{E}}(\mathcal{D}) = \mathbf{Adv}^{\text{(s)prp}}_{\widetilde{E}}(\mathcal{D})$, and we will sometimes view single-key security as related-key security under KDF-set Φ_{id}. Two other KDF-sets we consider in this work are the following:

$$\begin{aligned}\Phi_{\oplus} &= \{\varphi_\delta : k \mapsto k \oplus \delta \mid \delta \in \mathcal{K}\}\,, \\ \Phi_{P\oplus} &= \{\varphi_{\delta,\epsilon} : k \mapsto P^{-1}(P(k) \oplus \epsilon) \oplus \delta \mid \delta, \epsilon \in \mathcal{K}, \delta = 0 \vee \epsilon = 0\}\,.\end{aligned} \tag{4}$$

We regularly simply write $\delta \in \Phi_{\oplus}$ to say that $\varphi_\delta \in \Phi_{\oplus}$, and similarly write $(\delta, \epsilon) \in \Phi_{P\oplus}$ to say that $\varphi_{\delta,\epsilon} \in \Phi_{P\oplus}$.

Note that every $\varphi_\delta \in \Phi_{\oplus}$ satisfies $\varphi_\delta = \varphi_{\delta,0} \in \Phi_{P\oplus}$, and hence $\Phi_{\oplus} \subseteq \Phi_{P\oplus}$ by construction. The side condition "$\delta = 0 \vee \epsilon = 0$" for $\Phi_{P\oplus}$ deserves an additional explanation. In our scheme XPX, the in- and outputs will be masked using the values $(k, P(k))$. A function $\varphi_\delta \in \Phi_{\oplus}$ (or, equivalently, $\varphi_{\delta,0} \in \Phi_{P\oplus}$) transforms these values to $(k \oplus \delta, P(k \oplus \delta))$. The set $\Phi_{P\oplus}$ generalizes the strength of the attacker by also transforming $P(k)$ under XOR. In more detail, for any ϵ, $\varphi_{0,\epsilon} \in \Phi_{P\oplus}$ transforms $(k, P(k))$ to $(P^{-1}(P(k) \oplus \epsilon), P(k) \oplus \epsilon)$. From a theoretical point, it may be of interest to drop the side condition from $\Phi_{P\oplus}$. This would, however, make the security analysis of XPX much more complicated and technically demanding.

2.3 Patarin's Technique

We use the H-coefficient technique by Patarin [35] and Chen and Steinberger [13], and we introduce it for our definitions of related-key security. Recall that these definitions simplify to single-key security by using KDF-set Φ_{id}.

Let $P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M})$, and $\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\Phi, \mathcal{T}, \mathcal{M})$. Let $k \xleftarrow{\$} \mathcal{K}$ and $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ be a tweakable blockcipher based on P. Consider any fixed deterministic distinguisher $\mathcal{D}$ for the RK-(S)PRP security of $\widetilde{E}$. It has access to either the *real world* $\mathcal{O}_{\mathrm{re}} = (\mathsf{RK}[\widetilde{E}]_k^{(\pm)}, P^{\pm})$ or the *ideal world* $\mathcal{O}_{\mathrm{id}} = (\widetilde{\mathsf{RK}\pi}^{(\pm)}, P^{\pm})$ and its goal is to distinguish both. Here, the distinguisher has inverse query access to the construction oracle if and only if we are considering *strong* PRP security (hence the brackets around $\pm$). The information that $\mathcal{D}$ learns from the interaction with $\mathcal{O}_{\mathrm{re}}/\mathcal{O}_{\mathrm{id}}$ is collected in a view v. Denote by X_{re} (resp. X_{id}) the probability distribution of views when interacting with $\mathcal{O}_{\mathrm{re}}$ (resp. $\mathcal{O}_{\mathrm{id}}$). Let $\mathcal{V}$ be the set of all attainable views, i.e., views that occur in the ideal world with non-zero probability.

Lemma 1 (Patarin's Technique). *Let $\mathcal{D}$ be a deterministic distinguisher. Consider a partition $\mathcal{V} = \mathcal{V}_{\mathrm{good}} \cup \mathcal{V}_{\mathrm{bad}}$ of the set of attainable views. Let $0 \le \varepsilon \le 1$ be such that for all $v \in \mathcal{V}_{\mathrm{good}}$,*

$$\mathbf{Pr}[X_{\mathrm{re}} = v] \ge (1 - \varepsilon)\mathbf{Pr}[X_{\mathrm{id}} = v]\,. \tag{5}$$

Then, the distinguishing advantage satisfies $\mathbf{Adv}(\mathcal{D}) \le \varepsilon + \mathbf{Pr}[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}]$.

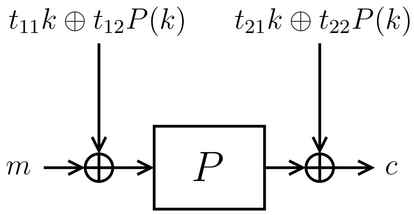

Fig. 3: XPX

A proof of this lemma is given in [12,13]. The idea of the technique is that only few views are significantly more likely to appear in $\mathcal{O}_{\mathrm{id}}$ than in $\mathcal{O}_{\mathrm{re}}$. In other words, the ratio (5) is close to 1 for all but the "bad" views. Note that taking a large $\mathcal{V}_{\mathrm{bad}}$ implies a higher $\mathbf{Pr}\left[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}\right]$, while a small $\mathcal{V}_{\mathrm{bad}}$ implies a higher ε. The definition of what views are "bad" is thus a tradeoff between the two terms.

Let $v_C = \{(\varphi_1, t_1, m_1, c_1), \ldots, (\varphi_q, t_q, m_q, c_q)\}$ be a view on a construction oracle. We say that a tweakable related-key permutation $\widetilde{\mathsf{RK}\pi} \in \widetilde{\mathsf{Perm}}(\Phi, \mathcal{T}, \mathcal{M})$ *extends* v_C, denoted $\widetilde{\mathsf{RK}\pi} \vdash v_C$, if $\widetilde{\mathsf{RK}\pi}(\varphi, t, m) = c$ for each $(\varphi, t, m, c) \in v_C$. Note that if $\widetilde{E} : \mathcal{K} \times \mathcal{T} \times \mathcal{M} \to \mathcal{M}$ is a tweakable blockcipher and $k \in \mathcal{K}$, then $\mathsf{RK}[\widetilde{E}]_k \in \widetilde{\mathsf{Perm}}(\Phi, \mathcal{T}, \mathcal{M})$ and the definition reads $\mathsf{RK}[\widetilde{E}]_k \vdash v_C$. Similarly, if $v_P = \{(x_1, y_1), \ldots, (x_r, y_r)\}$ is a primitive view, we say that a permutation $P \in \mathsf{Perm}(\mathcal{M})$ *extends* v_P, denoted $P \vdash v_P$, if $P(x) = y$ for each $(x, y) \in v_P$.

3 XPX

Let P be any n-bit permutation. We present the tweakable blockcipher XPX that has a key space $\{0,1\}^n$, a tweak space $\mathcal{T} \subseteq (\{0,1\}^n)^4$, and a message and ciphertext space $\{0,1\}^n$. Formally, XPX : $\{0,1\}^n \times \mathcal{T} \times \{0,1\}^n \to \{0,1\}^n$ is defined as

$$\begin{aligned} \mathrm{XPX}_k((t_{11}, t_{12}, t_{21}, t_{22}), m) = P(m \oplus \Delta_1) \oplus \Delta_2 \text{ , where } \Delta_1 &= t_{11}k \oplus t_{12}P(k) \text{ ,} \\ \text{and } \Delta_2 &= t_{21}k \oplus t_{22}P(k) \text{ .} \end{aligned} \tag{6}$$

XPX is depicted in Figure 3. The design is general in that $\mathcal{T}$ can (still) be any set, and we highlight two examples.

- **Even-Mansour.** XPX meets the single-key Even-Mansour construction (1) by fixing $\mathcal{T} = \{(1,0,1,0)\}$. More generally, if $|\mathcal{T}| = 1$, we are simply considering an ordinary (not a tweakable) blockcipher;
- **XEX with Even-Mansour.** XPX covers XEX based on Even-Mansour with $N = 0$ by taking

$$\mathcal{T} = \left\{ \begin{matrix} (\mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell} \oplus 1, \mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell}, \\ \mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell} \oplus 1, \mathtt{x}_1^{\alpha_1} \cdots \mathtt{x}_\ell^{\alpha_\ell}) \end{matrix} \;\middle|\; (\alpha_1, \ldots, \alpha_\ell) \in \mathbb{I}_1 \times \cdots \times \mathbb{I}_\ell \right\} ,$$

  where $\mathtt{x}_1, \ldots, \mathtt{x}_\ell$ and tweak space $\mathbb{I}_1 \times \cdots \times \mathbb{I}_\ell$ are as described in Section 1. In this case, $(\alpha_1, \ldots, \alpha_\ell)$ is in fact the "real" tweak, and $(t_{11}, t_{12}, t_{21}, t_{22})$ is a function of $(\alpha_1, \ldots, \alpha_\ell)$.

Further applications follow in Sections 6 and 7. Obviously, XPX does not achieve security for all choices of $\mathcal{T}$; e.g., if $(1,0,1,1) \in \mathcal{T}$, then we have

$$\mathrm{XPX}_k((1,0,1,1),0) = k\,. \tag{7}$$

In Section 4, we derive a minimal set of conditions on $\mathcal{T}$ to make the XPX construction meaningful. Then, in Section 5 we prove that XPX is secure in various settings, from single-key (S)PRP security to RK-SPRP security for the key-deriving function sets of Section 2.2.

4 Valid Tweak Sets

To eliminate trivial cases such as (7), we define a set of minimal conditions $\mathcal{T}$ needs to satisfy in order for XPX to achieve a reasonable level of security. In more detail, we define the notion of a *valid* tweak space $\mathcal{T}$. After the definition, we present the rationale and a proposition showing that XPX is insecure if $\mathcal{T}$ is invalid.

Definition 1. *We say that $\mathcal{T}$ is* valid *if:*

(i) For any $(t_{11},t_{12},t_{21},t_{22}) \in \mathcal{T}$ we have $(t_{11},t_{12}) \neq (0,0)$ and $(t_{21},t_{22}) \neq (0,0)$;

(ii) For any distinct $(t_{11},t_{12},t_{21},t_{22}),(t'_{11},t'_{12},t'_{21},t'_{22}) \in \mathcal{T}$ we have $(t_{11},t_{12}) \neq (t'_{11},t'_{12})$ and $(t_{21},t_{22}) \neq (t'_{21},t'_{22})$;

(iii) If $(1,0,t_{21},t_{22}) \in \mathcal{T}$ for some t_{21},t_{22}:

(a) $t_{21} \neq 0$ and $t_{22} \neq 1$;

(b) For any other $(t'_{11},t'_{12},t'_{21},t'_{22}) \in \mathcal{T}$ and $b \in \{0,1\}$ we have

$$t'_{11} \neq t'_{12}t_{21}(t_{22} \oplus 1)^{-1} \oplus b \text{ and } t'_{22} \neq t'_{21}t_{21}^{-1}(t_{22} \oplus 1) \oplus b\,;$$

(c) For any distinct $(t'_{11},t'_{12},t'_{21},t'_{22}),(t''_{11},t''_{12},t''_{21},t''_{22}) \in \mathcal{T}$ we have

$$t'_{12} \oplus t''_{12} \neq (t'_{11} \oplus t''_{11})t_{21}^{-1}(t_{22} \oplus 1) \text{ and } t'_{22} \oplus t''_{22} \neq (t'_{21} \oplus t''_{21})t_{21}^{-1}(t_{22} \oplus 1)\,;$$

(iv) If $(t_{11},t_{12},0,1) \in \mathcal{T}$ for some t_{11},t_{12}:

(a) $t_{12} \neq 0$ and $t_{11} \neq 1$;

(b) For any other $(t'_{11},t'_{12},t'_{21},t'_{22}) \in \mathcal{T}$ and $b \in \{0,1\}$ we have

$$t'_{11} \neq t'_{12}t_{12}^{-1}(t_{11} \oplus 1) \oplus b \text{ and } t'_{22} \neq t'_{21}t_{12}(t_{11} \oplus 1)^{-1} \oplus b\,;$$

(c) For any distinct $(t'_{11},t'_{12},t'_{21},t'_{22}),(t''_{11},t''_{12},t''_{21},t''_{22}) \in \mathcal{T}$ we have

$$t'_{11} \oplus t''_{11} \neq (t'_{12} \oplus t''_{12})t_{12}^{-1}(t_{11} \oplus 1) \text{ and } t'_{21} \oplus t''_{21} \neq (t'_{22} \oplus t''_{22})t_{12}^{-1}(t_{11} \oplus 1)\,.$$

Conditions (i) and (ii) are basic requirements, in essence guaranteeing that the input to and output of the underlying permutation P is always masked. Conditions (iii) and (iv) are more obscure but are in fact necessary to prevent the key from being leaked. The presence of conditions (iii-a) and (iv-a) is justified by equation (7), but even beyond that, an evaluation $\mathrm{XPX}_k((1,0,t_{21},t_{22}),0)$ for some $t_{21} \neq 0$ and $t_{22} \neq 1$ leaks the value $t_{21}k \oplus (t_{22} \oplus 1)P(k)$ and additional conditions are required. In below proposition, we show that XPX is insecure whenever $\mathcal{T}$ is invalid.

We remark that the second part of condition (ii) and the entire condition (iv) are not strictly needed for PRP security and only apply to SPRP security. We nevertheless included them for completeness.

Proposition 1. *Let $n \geq 1$ and let $\mathcal{T} \subseteq (\{0,1\}^n)^4$ an* invalid *set. We have*

$$\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(5,2) \geq 1 - 1/(2^n - 1)\,.$$

Proof. We consider conditions (i), (ii), and (iii) separately. Condition (iv) is symmetrically equivalent to (iii), and omitted.

Condition (i). Assume, w.l.o.g., that $(0,0,t_{21},t_{22}) \in \mathcal{T}$ for some t_{21}, t_{22}. For any $m \in \{0,1\}^n$ we have $\mathrm{XPX}_k((0,0,t_{21},t_{22}),m) \oplus P(m) = t_{21}k \oplus t_{22}P(k)$. Making these two queries for two different messages $m \neq m'$ gives a collision with probability 1. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n-1)$. Thus, if condition (i) is violated, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(2,2) \geq 1 - 1/(2^n-1)$.

Condition (ii). Assume, w.l.o.g., that $(t_{11},t_{12},t_{21},t_{22}), (t_{11},t_{12},t'_{21},t'_{22}) \in \mathcal{T}$ for some $(t_{21},t_{22}) \neq (t'_{21},t'_{22})$. For any m,

$$\begin{aligned}&\mathrm{XPX}_k((t_{11},t_{12},t_{21},t_{22}),m) \oplus \mathrm{XPX}_k((t_{11},t_{12},t'_{21},t'_{22}),m)\\ &= (t_{21} \oplus t'_{21})k \oplus (t_{22} \oplus t'_{22})P(k)\,.\end{aligned}$$

Making these queries for two different messages $m \neq m'$ gives a collision with probability 1. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n-1)$. Thus, if condition (ii) is violated, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(4,0) \geq 1 - 1/(2^n-1)$.

Condition (iii-a). Suppose $(1,0,t_{21},t_{22}) \in \mathcal{T}$ for some t_{21}, t_{22}. By construction, $\mathrm{XPX}_k((1,0,t_{21},t_{22}),0) = t_{21}k \oplus (t_{22} \oplus 1)P(k)$. If $t_{21} = 0$ or $t_{22} = 1$, this value leaks k or $P(k)$. By making one additional invocation of $P^{\pm}$ the other value is learned as well, giving the distinguisher both $(k, P(k))$. For arbitrary $m \neq 0$, the distinguisher now queries $\mathrm{XPX}_k((1,0,t_{21},t_{22}),m) = c$ and $P(m \oplus k) = y$ and verifies whether $c = y \oplus t_{21}k \oplus t_{22}P(k)$. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n-1)$. Thus, if condition (iii-a) is violated, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(2,2) \geq 1 - 1/(2^n-1)$.

Condition (iii-b). Suppose $(1,0,t_{21},t_{22}) \in \mathcal{T}$ for some t_{21}, t_{22}, and assume $t_{21} \neq 0$ and $t_{22} \neq 1$ (otherwise, the attack of (iii-a) applies). Suppose there is a $(t'_{11},t'_{12},t'_{21},t'_{22}) \in \mathcal{T}$ such that $t'_{22} = t'_{21}t_{21}^{-1}(t_{22} \oplus 1) \oplus b$ for some $b \in \{0,1\}$. This is without loss of generality, as the other case is symmetric and the attack applies by reversing all queries for tweak $(t'_{11},t'_{12},t'_{21},t'_{22})$. We first consider case $b = 0$, case $b = 1$ is treated later.

For $b = 0$: firstly, the attacker queries $\mathrm{XPX}_k((1,0,t_{21},t_{22}),0)$ to receive $c = t_{21}k \oplus (t_{22} \oplus 1)P(k)$. Fix any $c' \in \{0,1\}^n$, and query $\mathrm{XPX}_k^{-1}((t'_{11},t'_{12},t'_{21},t'_{22}),c')$ to receive $m' = t'_{11}k \oplus t'_{12}P(k) \oplus P^{-1}(\mathrm{inp}')$ where $\mathrm{inp}' = c' \oplus t'_{21}k \oplus t'_{22}P(k)$. Eliminating $P(k)$ using c gives

$$\mathrm{inp}' = c' \oplus t'_{22}(t_{22} \oplus 1)^{-1}c \oplus \big(t'_{21} \oplus t'_{22}(t_{22} \oplus 1)^{-1}t_{21}\big)k = c' \oplus t'_{22}(t_{22} \oplus 1)^{-1}c\,,$$

where we use the violation of property (iii-b). Therefore,

$$m' \oplus P^{-1}(c' \oplus t'_{22}(t_{22} \oplus 1)^{-1}c) = t'_{11}k \oplus t'_{12}P(k)\,.$$

This equation is independent of the choice of c'. Making these queries for two different ciphertexts $c' \neq c''$ gives a collision with probability 1. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n-1)$. Thus, if condition (iii-b) is violated with $b = 0$, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(3,2) \geq 1 - 1/(2^n-1)$.

For $b = 1$: in this case we specifically consider $c' = t'_{21}t_{21}^{-1}c$, and have

$$\begin{aligned} \mathrm{inp}' &= t'_{21}t_{21}^{-1}c \oplus t'_{21}k \oplus t'_{22}P(k) \\ &= \big(t'_{21}t_{21}^{-1}(t_{22} \oplus 1) \oplus t'_{22}\big)P(k) = P(k)\,, \end{aligned}$$

using that $c = t_{21}k \oplus (t_{22} \oplus 1)P(k)$ and the violation of property (iii-b). Therefore,

$$\begin{pmatrix} t_{21} & t_{22} \oplus 1 \\ t'_{11} \oplus 1 & t'_{12} \end{pmatrix} \begin{pmatrix} k \\ P(k) \end{pmatrix} = \begin{pmatrix} c \\ m' \end{pmatrix}\,,$$

If this matrix is singular, it implies that $m' = \mathrm{const} \cdot c$ with $\mathrm{const} = t_{21}^{-1}(t'_{11} \oplus 1) = (t_{22} \oplus 1)^{-1}t'_{12}$. For a random tweakable permutation this happens with probability at most $1/2^n$. On the other hand, if it is non-singular, this reveals k and $P(k)$.

For arbitrary $m \neq 0$, the distinguisher now queries $\mathrm{XPX}_k((1, 0, t_{21}, t_{22}), m'') = c''$ and $P(m'' \oplus k) = y$ and verifies whether $c'' = y \oplus t_{21}k \oplus t_{22}P(k)$. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n - 1)$. Thus, if condition (iii-b) is violated with $b = 1$, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(3, 1) \geq 1 - 1/(2^n - 1)$.

Condition (iii-c). Suppose $(1, 0, t_{21}, t_{22}) \in \mathcal{T}$ for some t_{21}, t_{22}, and assume $t_{21} \neq 0$ and $t_{22} \neq 1$ (otherwise, the attack of (iii-a) applies). Suppose there are $(t'_{11}, t'_{12}, t'_{21}, t'_{22})$, $(t''_{11}, t''_{12}, t''_{21}, t''_{22}) \in \mathcal{T}$ such that $t'_{22} \oplus t''_{22} = (t'_{21} \oplus t''_{21})t_{21}^{-1}(t_{22} \oplus 1)$. This is without loss of generality, as the other case is symmetric and the attack applies by reversing all queries for tweaks $(t'_{11}, t'_{12}, t'_{21}, t'_{22}), (t''_{11}, t''_{12}, t''_{21}, t''_{22})$. Firstly, the attacker makes queries $\mathrm{XPX}_k((1, 0, t_{21}, t_{22}), 0)$ to receive $c = t_{21}k \oplus (t_{22} \oplus 1)P(k)$. Now, fix any $c' \in \{0, 1\}^n$, and query

- $\mathrm{XPX}_k^{-1}((t'_{11}, t'_{12}, t'_{21}, t'_{22}), c')$ to receive $m' = t'_{11}k \oplus t'_{12}P(k) \oplus P^{-1}(\mathrm{inp}')$ where $\mathrm{inp}' = c' \oplus t'_{21}k \oplus t'_{22}P(k)$;
- $\mathrm{XPX}_k^{-1}((t''_{11}, t''_{12}, t''_{21}, t''_{22}), c' \oplus (t'_{21} \oplus t''_{21})t_{21}^{-1}c)$ to receive $m'' = t''_{11}k \oplus t''_{12}P(k) \oplus P^{-1}(\mathrm{inp}'')$ where $\mathrm{inp}'' = c' \oplus (t'_{21} \oplus t''_{21})t_{21}^{-1}c \oplus t''_{21}k \oplus t''_{22}P(k)$.

Plugging c into inp' and inp'' gives

$$\begin{aligned} \mathrm{inp}'' &= c' \oplus t'_{21}k \oplus \big(t''_{22} \oplus (t'_{21} \oplus t''_{21})t_{21}^{-1}(t_{22} \oplus 1)\big)P(k) \\ &= c' \oplus t'_{21}k \oplus t'_{22}P(k) = \mathrm{inp}'\,, \end{aligned}$$

where we use the violation of property (iii-c). Therefore,

$$m' \oplus m'' = (t'_{11} \oplus t''_{11})k \oplus (t'_{12} \oplus t''_{12})P(k)\,.$$

This equation is independent of the choice of c'. Making these queries for two different ciphertexts $c' \neq c''$ gives a collision with probability 1. For a random $\widetilde{\pi}$ this happens with probability at most $1/(2^n - 1)$. Thus, if condition (iii-c) is violated, $\mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(5, 0) \geq 1 - 1/(2^n - 1)$.

Conclusion. In any case, a distinguishing attack with success probability at least $1 - 1/(2^n - 1)$ can be performed in at most 5 construction queries and 2 primitive queries. □

5 Security of XPX

In this section, we analyze the security of XPX in various security models. We will focus on valid $\mathcal{T}$ only. Theorem 1 captures all security levels for the three key-deriving function sets of (4).

Theorem 1. *Let $n \geq 1$ and let $\mathcal{T} \subseteq (\{0,1\}^n)^4$ be a* valid *set.*

(a) We have

$$\mathbf{Adv}^{\mathrm{prp}}_{\mathrm{XPX}}(q,r) \leq \mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(q,r) \leq \frac{(q+1)^2 + 2q(r+1) + 2r}{2^n}\,.$$

(b) If for all $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ we have $t_{12} \neq 0$, then

$$\mathbf{Adv}^{\text{rk-prp}}_{\Phi_\oplus,\mathrm{XPX}}(q,r) \leq \frac{\frac{7}{2}q^2 + 4qr}{2^n - q}\,.$$

(c) If for all $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ we have $t_{12}, t_{22} \neq 0$ and $(t_{21}, t_{22}) \neq (0,1)$, then

$$\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_\oplus,\mathrm{XPX}}(q,r) \leq \frac{\frac{7}{2}q^2 + 4qr}{2^n}\,.$$

(d) If for all $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ we have $t_{11}, t_{12} \neq 0$, then

$$\mathbf{Adv}^{\text{rk-prp}}_{\Phi_{P\oplus},\mathrm{XPX}}(q,r) \leq \frac{4q^2 + 4qr}{2^n - q}\,.$$

(e) If for all $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ we have $t_{11}, t_{12}, t_{21}, t_{22} \neq 0$, then

$$\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_{P\oplus},\mathrm{XPX}}(q,r) \leq \frac{4q^2 + 4qr}{2^n}\,.$$

The proof is given in Sections 5.1 to 5.3. In Appendix A, we prove that the conditions $\mathcal{T}$ are minimal, meaning that the security proof cannot go through if the conditions are omitted.

5.1 Proof of Theorem 1(a)

Note that $\mathbf{Adv}^{\mathrm{prp}}_{\mathrm{XPX}}(q,r) \leq \mathbf{Adv}^{\mathrm{sprp}}_{\mathrm{XPX}}(q,r)$ holds by construction, and we will focus on bounding the latter. The proof is a generalization of the proofs of Even-Mansour [4, 18, 19, 33], but difficulties arise due to the tweaks.

Let $k \xleftarrow{\$} \{0,1\}^n$, $P \xleftarrow{\$} \mathsf{Perm}(\{0,1\}^n)$, and $\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T}, \{0,1\}^n)$. Consider any fixed deterministic distinguisher $\mathcal{D}$ for the SPRP security of XPX. In the real world it has access to (XPX_k, P), and in the ideal world to $(\widetilde{\pi}, P)$. It makes q construction queries which are summarized in view $v_1 = \{((t_{11}, t_{12}, t_{21}, t_{22})_1, m_1, c_1), \ldots, ((t_{11}, t_{12}, t_{21}, t_{22})_q, m_q, c_q)\}$. It additionally makes r queries to P, summarized in a view $v_2 = \{(x_1, y_1), \ldots, (x_r, y_r)\}$. As $\mathcal{D}$ is deterministic this properly summarizes the conversation.

To suit the analysis, we generalize our oracles by providing $\mathcal{D}$ with extra data. How these extra data look like, depends on whether or not $\mathcal{T}$ contains tweak tuple $(1, 0, \bar{t}_{21}, \bar{t}_{22})$ or $(\bar{t}_{11}, \bar{t}_{12}, 0, 1)$.[1] Because of their dedicated treatment, we will always refer to these tweak tuples with overlines. Note that, as $\mathcal{T}$ is valid, at most one of the two tweaks is in $\mathcal{T}$, but it may as well be that none of these is allowed.

More formally, *before* $\mathcal{D}$'s interaction with the oracles, we reveal forward construction query $((1, 0, \bar{t}_{21}, \bar{t}_{22}), 0, \bar{c})$ or inverse construction query $((\bar{t}_{11}, \bar{t}_{12}, 0, 1), \bar{m}, 0)$, depending on whether one of the two tweaks is in $\mathcal{T}$, and store the resulting tuple in view v_0. If none of the two tweaks is in $\mathcal{T}$, we simply have $|v_0| = 0$.

[1] Indeed, if (for instance) $(1, 0, \bar{t}_{21}, \bar{t}_{22}) \in \mathcal{T}$, a construction query $((1, 0, \bar{t}_{21}, \bar{t}_{22}), 0)$ will reveal $\bar{c} = \bar{t}_{21}k \oplus (\bar{t}_{22} \oplus 1)P(k)$ and a special analysis is needed.

Then, *after* $\mathcal{D}$'s interaction with its oracles but before $\mathcal{D}$ makes its final decision, we reveal $v_k = \{(k, k^\star)\}$. In the real world, k is the key used for encryption and $k^\star = P(k)$. In the ideal world, $k \xleftarrow{\$} \{0,1\}^n$ will be a randomly drawn dummy key and $k^\star$ will be defined based on k and v_0. If $|v_0| = 0$, then $k^\star \xleftarrow{\$} \{0,1\}^n$. Otherwise, it is the unique[2] value that satisfies

$$\begin{aligned} \bar{t}_{21}k \oplus (\bar{t}_{22} \oplus 1)k^\star = \bar{c} \text{ if } v_0 = \{(1, 0, \bar{t}_{21}, \bar{t}_{22}), 0, \bar{c})\}\,, \text{ or} \\ (\bar{t}_{11} \oplus 1)k \oplus \bar{t}_{12}k^\star = \bar{m} \text{ if } v_0 = \{((\bar{t}_{11}, \bar{t}_{12}, 0, 1), \bar{m}, 0)\}\,. \end{aligned} \tag{8}$$

Clearly, these disclosures are without loss of generality as they may only *help* the distinguisher. The complete view is denoted $v = (v_0, v_1, v_2, v_k)$. Recall that $\mathcal{D}$ is assumed not to make any repeat queries, and hence $v_0 \cup v_1$ and v_2 do not contain any duplicate elements. Note that v_k may collide with v_2, but this will be captured as a bad event.

Throughout, we consider attainable views only. Recall that a view is attainable if it can be obtained in the ideal world. For $v_0 \cup v_1$, this is the case if and only if for any distinct i, i' such that $(t_{11}, t_{12}, t_{21}, t_{22})_i = (t_{11}, t_{12}, t_{21}, t_{22})_{i'}$, we have $m_i \neq m_{i'}$ and $c_i \neq c_{i'}$. For v_2 the condition is equivalent: there should be no two distinct $(x, y), (x', y') \in v_2$ such that $x = x'$ or $y = y'$. Attainability implies for v_k that $k^\star$ satisfies (8) if $|v_0| = 1$.

We say that a view v is *bad* if one of the following conditions holds:

$$\begin{aligned}
&\mathsf{BV}_1: \text{ for some } (x, y) \in v_2 \text{ and } (k, k^\star) \in v_k\text{:} \\
&\qquad \mathsf{BV}_{1a}: \ k = x\,, \text{ or} \\
&\qquad \mathsf{BV}_{1b}: \ k^\star = y\,, \text{ or} \\
&\mathsf{BV}_2: \text{ for some } ((t_{11}, t_{12}, t_{21}, t_{22}), m, c) \in v_1,\ (x, y) \in v_2 \cup v_k, \text{ and } (k, k^\star) \in v_k\text{:} \\
&\qquad \mathsf{BV}_{2a}: \ m \oplus t_{11}k \oplus t_{12}k^\star = x\,, \text{ or} \\
&\qquad \mathsf{BV}_{2b}: \ c \oplus t_{21}k \oplus t_{22}k^\star = y\,, \text{ or} \\
&\mathsf{BV}_3: \text{ for some distinct } ((t_{11}, t_{12}, t_{21}, t_{22}), m, c), ((t'_{11}, t'_{12}, t'_{21}, t'_{22}), m', c') \in v_0 \cup v_1 \\
&\qquad\quad \text{and } (k, k^\star) \in v_k\text{:} \\
&\qquad \mathsf{BV}_{3a}: \ m \oplus t_{11}k \oplus t_{12}k^\star = m' \oplus t'_{11}k \oplus t'_{12}k^\star\,, \text{ or} \\
&\qquad \mathsf{BV}_{3b}: \ c \oplus t_{21}k \oplus t_{22}k^\star = c' \oplus t'_{21}k \oplus t'_{22}k^\star\,.
\end{aligned}$$

Note that every tuple in $v_0 \cup v_1$ uniquely corresponds to an evaluation of the underlying P, namely via (6) where v_k is used as key material. The above conditions cover all cases where two different tuples in v collide at their P evaluation. In more detail, BV_1 covers the case where $v_k = \{(k, k^\star)\}$ collides with a tuple in v_2, BV_2 the case where a tuple in v_1 collides with a tuple in $v_2 \cup v_k$, and BV_3 the case where two tuples in $v_0 \cup v_1$ collide with each other. Note that two different tuples in v_2 never collide (by construction), and that the case of a tuple of v_0 colliding with v_2 is implicitly covered in BV_1. The only remaining case, v_0 colliding with v_k, is not required to be a bad event, as this is the exact way v_k is defined.

In accordance with Patarin's technique (Lemma 1), we derive an upper bound on $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right]$ in Lemma 2, and in Lemma 3 we will prove that $\varepsilon = 1$ works for good views.

Lemma 2. *For Theorem 1(a), we have* $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right] \leq \frac{(q+1)^2 + 2q(r+1) + 2r}{2^n}$.

[2] Because $\mathcal{T}$ is valid, $\bar{t}_{21}, \bar{t}_{22} \oplus 1 \neq 0$ in the former case and $\bar{t}_{11} \oplus 1, \bar{t}_{12} \neq 0$ in the latter.

Proof. Consider a view v in the ideal world $(\widetilde{\pi}, P)$. We will essentially compute

$$\mathbf{Pr}\left[\mathsf{BV}_1 \vee \mathsf{BV}_2 \vee \mathsf{BV}_3\right] \leq \mathbf{Pr}\left[\mathsf{BV}_1\right] + \mathbf{Pr}\left[\mathsf{BV}_2 \mid \neg\mathsf{BV}_1\right] + \mathbf{Pr}\left[\mathsf{BV}_3\right] . \tag{9}$$

We have $k \xleftarrow{\$} \{0,1\}^n$. If $|v_0| = 0$, we would also have $k^\star \xleftarrow{\$} \{0,1\}^n$. If $|v_0| = 1$, the value $k^\star$ is defined based on v_0. In fact, the probability that a transcript is bad is largest in case $|v_0| = 1$ and we consider this case only (the derivation for $|v_0| = 0$ is in fact a simplification of the below one). Without loss of generality, $v_0 = \{((\bar{t}_{11}, \bar{t}_{12}, 0, 1), \bar{m}, 0)\}$, where $\bar{t}_{11} \neq 1$ and $\bar{t}_{12} \neq 0$ by validity of $\mathcal{T}$. By (8), we have

$$k^\star = \bar{t}_{12}^{-1}\big((\bar{t}_{11} \oplus 1)k \oplus \bar{m}\big) .$$

At a high level, we will prove that all bad events become a condition on k once $k^\star$ gets replaced using this equation. We will use validity of $\mathcal{T}$ (and more specifically point (iv)) to show that these are non-trivial conditions (i.e., k never cancels out).

Condition BV_1. Condition BV_{1a} is clearly satisfied with probability $r/2^n$. Regarding BV_{1b}, we have r choices for $(x, y) \in v_2$, and k is a bad key if

$$k = (\bar{t}_{11} \oplus 1)^{-1}(\bar{t}_{12} y \oplus \bar{m}) ,$$

where we use that $\bar{t}_{11} \neq 1$. This happens with probability at most $r/2^n$. Therefore, $\mathbf{Pr}\left[\mathsf{BV}_1\right] \leq 2r/2^n$.

Condition BV_2. Consider any choice of $((t_{11}, t_{12}, t_{21}, t_{22}), m, c) \in v_1$ and $(x, y) \in v_2 \cup v_k$. Regarding BV_{2a}, it is set if

$$t_{11} k \oplus t_{12} \bar{t}_{12}^{-1}\big((\bar{t}_{11} \oplus 1)k \oplus \bar{m}\big) = x \oplus m .$$

This translates to

$$\begin{aligned}
&\big(t_{11} \oplus t_{12}\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1) \oplus 1\big)k = m \oplus t_{12}\bar{t}_{12}^{-1}\bar{m} && \text{if } (x, y) = (k, k^\star) \in v_k ,\\
&\big(t_{11} \oplus t_{12}\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1)\big)k = x \oplus m \oplus t_{12}\bar{t}_{12}^{-1}\bar{m} && \text{if } (x, y) \in v_2 .
\end{aligned}$$

Here, we use that $\neg\mathsf{BV}_1$ holds. Now, if $(t_{11}, t_{12}, t_{21}, t_{22}) = (\bar{t}_{11}, \bar{t}_{12}, 0, 1)$, we necessarily have $m \neq \bar{m}$ as v does not contain any duplicate elements. Then, the key is bad with probability 0 if $(x, y) = (k, k^\star) \in v_k$ and with probability $1/2^n$ otherwise. If $(t_{11}, t_{12}, t_{21}, t_{22}) \neq (\bar{t}_{11}, \bar{t}_{12}, 0, 1)$, the factor in front of k is nonzero as $\mathcal{T}$ is valid (condition (iv-b)), and k satisfies this equation with probability $1/2^n$. Concluding, BV_{2a} is set with probability at most $q(r+1)/2^n$. Regarding BV_{2b}, it is set if

$$t_{21} k \oplus t_{22} \bar{t}_{12}^{-1}\big((\bar{t}_{11} \oplus 1)k \oplus \bar{m}\big) = y \oplus c .$$

As before, this translates to

$$\begin{aligned}
&\big(t_{21} \oplus (t_{22} \oplus 1)\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1)\big)k = c \oplus (t_{22} \oplus 1)\bar{t}_{12}^{-1}\bar{m} && \text{if } (x, y) = (k, k^\star) \in v_k ,\\
&\big(t_{21} \oplus t_{22}\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1)\big)k = y \oplus c \oplus t_{22}\bar{t}_{12}^{-1}\bar{m} && \text{if } (x, y) \in v_2 .
\end{aligned}$$

The remainder of the analysis is the same, showing that BV_{2b} is set with probability at most $q(r+1)/2^n$. Therefore, $\mathbf{Pr}\left[\mathsf{BV}_2\right] \leq 2q(r+1)/2^n$.

Condition BV_3. Consider any two distinct $((t_{11}, t_{12}, t_{21}, t_{22}), m, c)$, $((t'_{11}, t'_{12}, t'_{21}, t'_{22}), m', c') \in v_0 \cup v_1$. If $(t_{11}, t_{12}, t_{21}, t_{22}) = (t'_{11}, t'_{12}, t'_{21}, t'_{22})$, then necessarily

$m \neq m'$ and $c \neq c'$ and BV_3 cannot be satisfied. Otherwise, we have $(t_{11}, t_{12}) \neq (t'_{11}, t'_{12})$ and $(t_{21}, t_{22}) \neq (t'_{21}, t'_{22})$ because of valid $\mathcal{T}$. Plugging $k^\star$ into the equation of BV_{3a} gives

$$\big(t_{11} \oplus t'_{11} \oplus (t_{12} \oplus t'_{12})\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1)\big)k = m \oplus m' \oplus (t_{12} \oplus t'_{12})\bar{t}_{12}^{-1}\bar{m}\,.$$

As before, $t_{11} \oplus t'_{11} \oplus (t_{12} \oplus t'_{12})\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1) \neq 0$: if (t_{11}, t_{12}) or (t'_{11}, t'_{12}) equals $(\bar{t}_{11}, \bar{t}_{12})$ this is due to validity of $\mathcal{T}$ point (iv-b), and otherwise due to point (iv-c). Therefore, k satisfies this equation with probability $1/2^n$. Thus, BV_{3a} is set with probability at most $\binom{q+1}{2}/2^n$. Regarding BV_{3b}, we similarly find

$$\big(t_{21} \oplus t'_{21} \oplus (t_{22} \oplus t'_{22})\bar{t}_{12}^{-1}(\bar{t}_{11} \oplus 1)\big)k = c \oplus c' \oplus (t_{22} \oplus t'_{22})\bar{t}_{12}^{-1}\bar{m}\,,$$

and BV_{3b} is set with probability at most $\binom{q+1}{2}/2^n$. Therefore, $\mathbf{Pr}\left[\mathsf{BV}_3\right] \leq 2\binom{q+1}{2}/2^n \leq (q+1)^2/2^n$.

Conclusion. Using (9), we have $\mathbf{Pr}\left[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}\right] \leq \frac{(q+1)^2 + 2q(r+1) + 2r}{2^n}$. This completes the proof. $\square$

Lemma 3. *For Theorem 1(a), we have* $\mathbf{Pr}\left[X_{\mathrm{re}} = v\right] \geq \mathbf{Pr}\left[X_{\mathrm{id}} = v\right]$ *for any good transcript* $v \in \mathcal{V}_{\mathrm{good}}$.

Proof. For the computation of $\mathbf{Pr}\left[X_{\mathrm{re}} = v\right]$ and $\mathbf{Pr}\left[X_{\mathrm{id}} = v\right]$, it suffices to compute the *fraction of oracles* that could result in view v. Recall that we assume that $\mathcal{D}$ never makes redundant queries, and particularly that $v_0 \cup v_1$ consists of $|v_0| + q$ distinct oracle queries.

In the real world, k will always be a randomly drawn key. The tuples $v_0 \cup v_1$ are construction evaluations and the tuples $v_1 \cup v_k$ are direct permutation evaluations. If $|v_0| = 0$, all of these tuples define a unique P-evaluation, $q + r + 1$ in total. This is because of the fact that we consider good transcripts. If $|v_0| = 1$, the P-evaluations by v_0 and v_k are the same, but apart from that all tuples define unique P-evaluations. So also in this case, we have $q + r + 1$ P-evaluations. Therefore,

$$\begin{aligned}\mathbf{Pr}\left[X_{\mathrm{re}} = v\right] &= \mathbf{Pr}\left[k' \xleftarrow{\$} \{0,1\}^n \;:\; k' = k\right] \cdot \\ &\quad\; \mathbf{Pr}\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \;:\; \mathrm{XPX}_k^P \vdash v_0 \cup v_1 \wedge P \vdash v_2 \cup v_k\right] \\ &= \frac{1}{2^n} \cdot \frac{1}{(2^n)_{q+r+1}}\,.\end{aligned}$$

For the analysis in the ideal world, we group the tuples in $v_0 \cup v_1$ according to the tweak value. Formally, for $t = (t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$, we define

$$\#_t = \left|\{(t, m, c) \in v_0 \cup v_1 \mid m, c \in \{0,1\}^n\}\right|.$$

The computation of $\mathbf{Pr}\left[X_{\mathrm{id}} = v\right]$ now differs depending on whether $|v_0| = 0$ or $|v_0| = 1$. If $|v_0| = 0$:

$$\begin{aligned}\mathbf{Pr}\left[X_{\mathrm{id}} = v \wedge |v_0| = 0\right] &= \mathbf{Pr}\left[k', k^{\star\prime} \xleftarrow{\$} \{0,1\}^n \;:\; k' = k \wedge k^{\star\prime} = k^\star\right] \cdot \\ &\quad\; \mathbf{Pr}\left[\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T}, \mathcal{M}) \;:\; \widetilde{\pi} \vdash v_1\right] \cdot \\ &\quad\; \mathbf{Pr}\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \;:\; P \vdash v_2\right] \\ &= \frac{1}{2^{2n}} \cdot \frac{1}{\prod_t (2^n)_{\#_t}} \cdot \frac{1}{(2^n)_r}\,, \text{ where } \textstyle\sum_t \#_t = q\,.\end{aligned}$$

If $|v_0| = 1$:

$$\begin{aligned}\mathbf{Pr}\left[X_{\text{id}} = v \wedge |v_0| = 1\right] = \mathbf{Pr}&\left[k' \xleftarrow{\$} \{0,1\}^n \ : \ k' = k\right] \cdot \\ \mathbf{Pr}&\left[\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T}, \mathcal{M}) \ : \ \widetilde{\pi} \vdash v_0 \cup v_1\right] \cdot \\ \mathbf{Pr}&\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \ : \ P \vdash v_2\right] \\ = &\ \frac{1}{2^n} \cdot \frac{1}{\prod_t (2^n)_{\#_t}} \cdot \frac{1}{(2^n)_r}, \text{ where } \textstyle\sum_t \#_t = q + 1\,.\end{aligned}$$

In either case,

$$\begin{aligned}\mathbf{Pr}\left[X_{\text{id}} = v\right] &\leq \frac{1}{2^n} \cdot \frac{1}{\prod_t (2^n)_{\#_t}} \cdot \frac{1}{(2^n)_r}, \text{ where } \textstyle\sum_t \#_t = q + 1 \\ &\leq \frac{1}{2^n} \cdot \frac{1}{(2^n)_{q+r+1}} \\ &= \mathbf{Pr}\left[X_{\text{re}} = v\right],\end{aligned}$$

where we use that $(a)_{b_1}(a)_{b_2} \geq (a)_{b_1+b_2}$. This completes the proof. □

5.2 Proof of Theorem 1(b) and 1(c)

The analyses for $\mathbf{Adv}^{\text{rk-prp}}_{\Phi_\oplus,\text{XPX}}(q,r)$ and $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_\oplus,\text{XPX}}(q,r)$ are fairly similar. The definition of the views, as well as the analysis for good transcripts, is the same for both. The only fundamental differences arise at the analysis of bad events, where in the former case we can use that $t_{12} \neq 0$ and that the c-values in v_1 are always random (as $\mathcal{D}$ only makes forward queries), and in the latter case we can use that both $t_{12} \neq 0$ and $t_{22} \neq 0$, and furthermore that $(t_{21}, t_{22}) \neq (0,1)$. Therefore, we discuss the proofs of Theorem 1(b) and Theorem 1(c) in one go and only fork at the analysis of bad events. The proofs are a generalization of the proof of Section 5.1, where the adversary can now make related-key queries.

Let $k \xleftarrow{\$} \{0,1\}^n$, $P \xleftarrow{\$} \mathsf{Perm}(\{0,1\}^n)$, and $\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK\text{-}Perm}}(\Phi_\oplus, \mathcal{T}, \mathcal{M})$. Consider any fixed deterministic distinguisher $\mathcal{D}$ for the RK-(S)PRP security of XPX. In the real world it has access to $(\mathsf{RK}[\widetilde{E}]_k, P)$, and in the ideal world to $(\widetilde{\mathsf{RK}\pi}, P)$. It makes q construction queries which are summarized in view $v_1 = \{(\delta_1, (t_{11}, t_{12}, t_{21}, t_{22})_1, m_1, c_1), \ldots, (\delta_q, (t_{11}, t_{12}, t_{21}, t_{22})_q, m_q, c_q)\}$ (the key-deriving functions are represented by their offsets $\delta_1, \ldots, \delta_q$). It additionally makes r queries to P, summarized in a view $v_2 = \{(x_1, y_1), \ldots, (x_r, y_r)\}$. As $\mathcal{D}$ is deterministic this properly summarizes the conversation. Note that if $\mathcal{D}$ is a PRP distinguisher, the construction queries are all in forward direction, while if it is an SPRP distinguisher, all queries may be in both directions. This does not influence the definition of the view.

As in previous proof, we reveal some additional information to the distinguisher. Note that, as $t_{12} \neq 0$, no tweak of the form $(1, 0, \bar{t}_{21}, \bar{t}_{22})$ exists. In the case of Theorem 1(c), we have that $(t_{21}, t_{22}) \neq (0,1)$ for all tweaks, hence also no tweak of the form $(\bar{t}_{11}, \bar{t}_{12}, 0, 1)$. In the case of Theorem 1(b), such a tweak may exist, but there is no need for a special treatment, as we consider PRP security (see also the discussion before the proof of Proposition 1, where it is explained that condition (iv) is not strictly needed for PRP security but only for SPRP security). In either case, there is no need to disclose additional queries *before* $\mathcal{D}$'s interaction with the oracles (and the notion of v_0 does not carry over from the proof of Theorem 1(a)).

After $\mathcal{D}$'s interaction with its oracles but before $\mathcal{D}$ makes its final decision, we reveal the secret key along with additional key material corresponding to the evaluations of $P(k \oplus \delta_i)$. Formally, let $\{\epsilon_1, \ldots, \epsilon_s\}$ be a minimal set such that it includes 0 (w.l.o.g. $\epsilon_1 = 0$) as well as $\delta_1, \ldots, \delta_q$. After $\mathcal{D}$'s interaction we reveal

$$v_k = \{(k_{\epsilon_1}, k^\star_{\epsilon_1}), \ldots, (k_{\epsilon_s}, k^\star_{\epsilon_s})\} .$$

In the real world, we define $k_{\epsilon_i} = k \oplus \epsilon_i$ and $k^\star_{\epsilon_i} = P(k \oplus \epsilon_i)$ for $i = 1, \ldots, s$, where k is the key used for encryption. In the ideal world, $k \xleftarrow{\$} \{0,1\}^n$ will be a randomly drawn dummy key and we define $k_{\epsilon_i} = k \oplus \epsilon_i$ as before. Also $k^\star_{\epsilon_i} \xleftarrow{\$} \{0,1\}^n$ for $i = 1, \ldots, s$ will be dummy keys.

The complete view is denoted $v = (v_1, v_2, v_k)$. Recall that $\mathcal{D}$ is assumed not to make any repeat queries, and hence v_1 and v_2 do not contain any duplicate elements. Note that v_k may contain collisions or may collide with v_2, but this will be captured as a bad event. As before, we only consider attainable views, which can be obtained in the ideal world.

We say that a view v is *bad* if one of the following conditions holds:

BV_1 : for some $(x, y) \in v_2$ and $(k_\delta, k^\star_\delta) \in v_k$:

BV_{1a} : $k_\delta = x$, or

BV_{1b} : $k^\star_\delta = y$, or

BV_2 : for some $(\delta, (t_{11}, t_{12}, t_{21}, t_{22}), m, c) \in v_1$, $(x, y) \in v_2 \cup v_k$, and $(k_\delta, k^\star_\delta) \in v_k$:

BV_{2a} : $m \oplus t_{11} k_\delta \oplus t_{12} k^\star_\delta = x$, or

BV_{2b} : $c \oplus t_{21} k_\delta \oplus t_{22} k^\star_\delta = y$, or

BV_3 : for some distinct $(\delta, (t_{11}, t_{12}, t_{21}, t_{22}), m, c), (\delta', (t'_{11}, t'_{12}, t'_{21}, t'_{22}), m', c') \in v_1$ and $(k_\delta, k^\star_\delta), (k_{\delta'}, k^\star_{\delta'}) \in v_k$:

BV_{3a} : $m \oplus t_{11} k_\delta \oplus t_{12} k^\star_\delta = m' \oplus t'_{11} k_{\delta'} \oplus t'_{12} k^\star_{\delta'}$, or

BV_{3b} : $c \oplus t_{21} k_\delta \oplus t_{22} k^\star_\delta = c' \oplus t'_{21} k_{\delta'} \oplus t'_{22} k^\star_{\delta'}$, or

BV_4 : for some distinct $(k_\delta, k^\star_\delta), (k_{\delta'}, k^\star_{\delta'}) \in v_k$: $k^\star_\delta = k^\star_{\delta'}$.

The cases $\mathsf{BV}_1, \mathsf{BV}_2, \mathsf{BV}_3$ are direct generalizations of the bad events of Section 5.1. New is BV_4, which considers the case where two different tuples in v_k collide. Note that different tuples in v_k *never* collide at the input, because the input values are $k \oplus \epsilon_1, \ldots, k \oplus \epsilon_s$, where $\epsilon_i \neq \epsilon_j$ for any $i, j \in \{1, \ldots, s\}$.

In accordance with Patarin's technique (Lemma 1), we derive an upper bound on $\mathbf{Pr}[X_{\text{id}} \in \mathcal{V}_{\text{bad}}]$ for the case of Theorem 1(b) in Lemma 4 and for the case of Theorem 1(c) in Lemma 5. In Lemma 6 we will prove that $\varepsilon = 1$ works for good views (the same analysis applies to both cases (b) and (c)). The proofs are then completed by noting that $s \leq q$.

Lemma 4. *For Theorem 1(b), we have* $\mathbf{Pr}[X_{\text{id}} \in \mathcal{V}_{\text{bad}}] \leq \frac{q^2 + 2q(r+s) + 2rs + s^2/2}{2^n - q}$.

Proof. The proof generalizes the one of Lemma 2. The most important difference is that now v_k may contain more than one tuple, where the distinguisher may effectively determine the differences between the input values. For the bound on bad views for Theorem 1(b), we know that $\mathcal{D}$ may only make *forward* construction queries, hence the c's in v_1 will always be randomly drawn.

Consider a view v in the ideal world $(\widetilde{\mathsf{RK}\pi}, P)$. We will essentially compute

$$\begin{aligned} \mathbf{Pr}[\mathsf{BV}_1 \vee \mathsf{BV}_2 \vee \mathsf{BV}_3 \vee \mathsf{BV}_4] \leq\ & \mathbf{Pr}[\mathsf{BV}_1] + \mathbf{Pr}[\mathsf{BV}_4] + \\ & \mathbf{Pr}[\mathsf{BV}_2 \mid \neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)] + \\ & \mathbf{Pr}[\mathsf{BV}_3 \mid \neg\mathsf{BV}_4] . \end{aligned} \tag{10}$$

We have $k \xleftarrow{\$} \{0,1\}^n$ with $k_{\epsilon_i} = k \oplus \epsilon_i$ for $i = 1, \ldots, s$, and additionally $k^\star_{\epsilon_i} \xleftarrow{\$} \{0,1\}^n$ for $i = 1, \ldots, s$. We will again show that all bad events have a non-trivial k or $k^\star_{\epsilon_i}$, or a non-trivial c.

Condition $\mathsf{BV_1}$. Consider any choice of $(x, y) \in v_2$ and $(k_\delta, k^\star_\delta) \in v_k$. It satisfies BV_{1a} with probability $1/2^n$, noting that $k_\delta = k \oplus \delta$. It also clearly satisfies BV_{1b} with probability $1/2^n$, as $k^\star_\delta \xleftarrow{\$} \{0,1\}^n$. Therefore, $\mathbf{Pr}[\mathsf{BV}_1] \leq 2rs/2^n$.

Condition $\mathsf{BV_4}$. Consider any two distinct $(k_\delta, k^\star_\delta), (k_{\delta'}, k^\star_{\delta'}) \in v_k$. These satisfy $k^\star_\delta = k^\star_{\delta'}$ with probability $1/2^n$. Therefore, $\mathbf{Pr}[\mathsf{BV}_4] \leq \binom{s}{2}/2^n \leq s^2/2^{n+1}$.

Condition $\mathsf{BV_2}$. Consider any choice of $(\delta, (t_{11}, t_{12}, t_{21}, t_{22}), m, c) \in v_1$ and $(x, y) \in v_2 \cup v_k$. This fixes the corresponding tuple $(k_\delta, k^\star_\delta) \in v_k$. Regarding BV_{2a}, it is set if

$$m \oplus t_{11} k_\delta \oplus t_{12} k^\star_\delta = x \,.$$

Regardless of whether $(x, y) \in v_k$, as $t_{12} \neq 0$ we always have a non-trivial term $k^\star_\delta$. This equation is thus satisfied with probability $1/2^n$. Event BV_{2b} is a bit more technical. The condition is satisfied if

$$c \oplus t_{21} k_\delta \oplus t_{22} k^\star_\delta = y \,.$$

If $(x, y) = (k_\delta, k^\star_\delta) \in v_k$ and additionally $(t_{21}, t_{22}) = (0, 1)$, the equation translates to $c = 0$. As we consider PRP security where the c-values are always randomly generated, this happens with probability at most $1/(2^n - q)$. In any other case, because $\neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)$ holds and that $(t_{21}, t_{22}) \neq (0, 0)$ by validity of $\mathcal{T}$ point (i), there is always a non-trivial k- or $k^\star_\delta$-term involved, and the equation is satisfied with probability $1/2^n$. Therefore, $\mathbf{Pr}[\mathsf{BV}_2 \mid \neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)] \leq 2q(r + s)/(2^n - q)$.

Condition $\mathsf{BV_3}$. Consider any two distinct $(\delta, (t_{11}, t_{12}, t_{21}, t_{22}), m, c)$, $(\delta', (t'_{11}, t'_{12}, t'_{21}, t'_{22}), m', c') \in v_1$. These fix the corresponding tuples $(k_\delta, k^\star_\delta), (k_{\delta'}, k^\star_{\delta'}) \in v_k$.

- If $(\delta, (t_{11}, t_{12}, t_{21}, t_{22})) = (\delta', (t'_{11}, t'_{12}, t'_{21}, t'_{22}))$, then $m \neq m'$ and $c \neq c'$ by attainability of v, and BV_3 is not satisfied by construction;
- If $(t_{11}, t_{12}, t_{21}, t_{22}) \neq (t'_{11}, t'_{12}, t'_{21}, t'_{22})$, condition BV_{3a} translates to

  $$(t_{11} \oplus t'_{11}) k \oplus t_{12} k^\star_\delta \oplus t'_{12} k^\star_{\delta'} = m \oplus m' \oplus t_{11}\delta \oplus t'_{11}\delta' \,.$$

  As $(t_{11}, t_{12}) \neq (t'_{11}, t'_{12})$ by validity of $\mathcal{T}$ point (ii), this equation always contains a k, $k^\star_\delta$, or $k^\star_{\delta'}$, and is satisfied with probability $1/2^n$. The analysis of BV_{3b} is equivalent;
- If $(t_{11}, t_{12}, t_{21}, t_{22}) = (t'_{11}, t'_{12}, t'_{21}, t'_{22})$ but $\delta \neq \delta'$, then the conditions translate to

  $$\begin{aligned} \mathsf{BV}_{3a}:&\ t_{12}(k^\star_\delta \oplus k^\star_{\delta'}) = m \oplus m' \oplus t_{11}(\delta \oplus \delta') \,, \\ \mathsf{BV}_{3b}:&\ t_{22}(k^\star_\delta \oplus k^\star_{\delta'}) = c \oplus c' \oplus t_{21}(\delta \oplus \delta') \,. \end{aligned}$$

  For BV_{3a}, using that $t_{12} \neq 0$ and $\neg\mathsf{BV}_4$, the equation is satisfied with probability $1/2^n$. For BV_{3b}, note that t_{22} may be 0, but c and c' are randomly generated and satisfy the equation with probability at most $1/(2^n - q)$.

Therefore, $\mathbf{Pr}[\mathsf{BV}_3 \mid \neg\mathsf{BV}_4] \leq 2\binom{q}{2}/(2^n - q) \leq q^2/(2^n - q)$.

Conclusion. Using (10), we have $\mathbf{Pr}[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}] \leq \frac{q^2 + 2q(r+s) + 2rs + s^2/2}{2^n - q}$. This completes the proof. □

Lemma 5. *For Theorem 1(c), we have* $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right] \leq \frac{q^2+2q(r+s)+2rs+s^2/2}{2^n}$.

Proof. The proof follows the one of Lemma 4 with the only differences at BV_2 and BV_3. In more detail, we cannot rely on the randomness of the c-values, but we can use that $t_{22} \neq 0$ and $(t_{21}, t_{22}) \neq (0,1)$. For BV_1 and BV_4 the analysis is identical and we find that $\mathbf{Pr}\left[\mathsf{BV}_1\right] \leq 2rs/2^n$ and $\mathbf{Pr}\left[\mathsf{BV}_4\right] \leq s^2/2^{n+1}$.

Condition BV_2. For BV_{2a} the analysis is identical. For BV_{2b}, the condition is satisfied if

$$c \oplus t_{21}k_\delta \oplus t_{22}k_\delta^\star = y\,.$$

If $(x,y) = (k_\delta, k_\delta^\star) \in v_k$, then we use the fact that $(t_{21}, t_{22}) \neq (0,1)$ to observe that the equation always contains a k or a $k_\delta^\star$ and is satisfied with probability $1/2^n$. In any other case, the analysis of Lemma 4 carries over. We find that $\mathbf{Pr}\left[\mathsf{BV}_2 \mid \neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)\right] \leq 2q(r+s)/2^n$.

Condition BV_3. For BV_{3a} the analysis is identical. For BV_{3b}, the analysis is also identical except for the case where $(t_{11}, t_{12}, t_{21}, t_{22}) = (t'_{11}, t'_{12}, t'_{21}, t'_{22})$ but $\delta \neq \delta'$. In this case, BV_{3b} translates to

$$t_{22}(k_\delta^\star \oplus k_{\delta'}^\star) = c \oplus c' \oplus t_{21}(\delta \oplus \delta')\,.$$

Now, using that $t_{22} \neq 0$, this equation is satisfied with probability $1/2^n$. Therefore, $\mathbf{Pr}\left[\mathsf{BV}_3 \mid \neg\mathsf{BV}_4\right] \leq 2\binom{q}{2}/2^n \leq q^2/2^n$.

Conclusion. We have $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right] \leq \frac{q^2+2q(r+s)+2rs+s^2/2}{2^n}$. This completes the proof. □

Lemma 6. *For Theorem 1(b) and Theorem 1(c), we have* $\mathbf{Pr}\left[X_{\text{re}} = v\right] \geq \mathbf{Pr}\left[X_{\text{id}} = v\right]$ *for any good transcript* $v \in \mathcal{V}_{\text{good}}$.

Proof. The proof is a straightforward generalization of the one of Lemma 3, with the simplification that v_0 does not exist. In the real world v now corresponds to $q+r+s$ unique P-evaluations (where s is the number of elements in v_k), and in the ideal world the tuples in v_1 are now grouped according to (offset,tweak). Indeed, $\widetilde{\mathsf{RK}\pi}$ behaves like a random permutation for every different (offset,tweak)-combination.

More formally, in the real world k will always be a randomly drawn key. All tuples in $v_1 \cup v_2 \cup v_k$ correspond to unique P-evaluations, $q+r+s$ in total. Therefore,

$$\begin{aligned}\mathbf{Pr}\left[X_{\text{re}} = v\right] &= \mathbf{Pr}\left[k' \xleftarrow{\$} \{0,1\}^n \;:\; k' = k\right] \cdot \\ &\qquad \mathbf{Pr}\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \;:\; \mathrm{XPX}_k^P \vdash v_1 \wedge P \vdash v_2 \cup v_k\right] \\ &= \frac{1}{2^n} \cdot \frac{1}{(2^n)_{q+r+s}}\,.\end{aligned}$$

For the analysis in the ideal world, we group the tuples in v_1 according to the (offset,tweak) value. Formally, for $(\delta, t) \in \{0,1\}^n \times \mathcal{T}$, we define

$$\#_{\delta,t} = \left|\{(\delta, t, m, c) \in v_1 \mid m, c \in \{0,1\}^n\}\right|\,.$$

We compute $\mathbf{Pr}[X_{\mathrm{id}} = v]$ as follows:

$$\begin{aligned}
\mathbf{Pr}[X_{\mathrm{id}} = v] = \mathbf{Pr}&\left[k', {k^\star_{\epsilon_1}}', \dots, {k^\star_{\epsilon_s}}' \xleftarrow{\$} \{0,1\}^n \;:\; k' = k \wedge {k^\star_{\epsilon_i}}' = k^\star_{\epsilon_i}\right] \cdot \\
\mathbf{Pr}&\left[\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK\text{-}Perm}}(\Phi_\oplus, \mathcal{T}, \mathcal{M}) \;:\; \widetilde{\mathsf{RK}\pi} \vdash v_1\right] \cdot \\
\mathbf{Pr}&\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \;:\; P \vdash v_2\right] \\
&= \frac{1}{2^{(s+1)n}} \cdot \frac{1}{\prod_{\delta,t} (2^n)_{\#_{\delta,t}}} \cdot \frac{1}{(2^n)_r}\,, \text{ where } \textstyle\sum_{\delta,t} \#_{\delta,t} = q\,.
\end{aligned}$$

As before,

$$\begin{aligned}
\mathbf{Pr}[X_{\mathrm{id}} = v] &\le \frac{1}{2^{(s+1)n}} \cdot \frac{1}{(2^n)_q} \cdot \frac{1}{(2^n)_r} \\
&\le \frac{1}{2^n} \cdot \frac{1}{(2^n)_{q+r+s}} \\
&= \mathbf{Pr}[X_{\mathrm{re}} = v]\,,
\end{aligned}$$

where we use that $(a)_{b_1}(a)_{b_2} \ge (a)_{b_1+b_2}$. This completes the proof. □

5.3 Proof of Theorem 1(d) and 1(e)

The analyses for $\mathbf{Adv}^{\text{rk-prp}}_{\Phi_{P\oplus},\mathrm{XPX}}(q, r)$ and $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_{P\oplus},\mathrm{XPX}}(q, r)$ are again fairly similar to each other, the only fundamental differences arising at the analysis of bad events. In more detail, in the former case we can use that $t_{11}, t_{12} \neq 0$ and that the c-values in v_1 are always random (as $\mathcal{D}$ only makes forward queries), and in the latter case we can use that $t_{11}, t_{12}, t_{21}, t_{22} \neq 0$ for all tweak tuples in $\mathcal{T}$. Therefore, we discuss the proofs of Theorem 1(d) and Theorem 1(e) in one go and only fork at the analysis of bad events. The proofs are a generalization of the proofs of Section 5.2, where the adversary can now choose among more key-deriving functions.

Let $k \xleftarrow{\$} \{0,1\}^n$, $P \xleftarrow{\$} \mathsf{Perm}(\{0,1\}^n)$, and $\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK\text{-}Perm}}(\Phi_{P\oplus}, \mathcal{T}, \mathcal{M})$. Consider any fixed deterministic distinguisher $\mathcal{D}$ for the RK-(S)PRP security of XPX. In the real world it has access to $(\mathsf{RK}[\widetilde{E}]_k, P)$, and in the ideal world to $(\widetilde{\mathsf{RK}\pi}, P)$. It makes q construction queries which are summarized in view

$$v_1 = \{((\delta_1, \delta_2)_1, (t_{11}, t_{12}, t_{21}, t_{22})_1, m_1, c_1), \dots, ((\delta_1, \delta_2)_q, (t_{11}, t_{12}, t_{21}, t_{22})_q, m_q, c_q)\}\,.$$

Indeed, all queries are made under key-deriving functions of the form $\varphi_{\delta_1,\delta_2}$ where $\delta_1 = 0 \vee \delta_2 = 0$. It additionally makes r queries to P, summarized in a view $v_2 = \{(x_1, y_1), \dots, (x_r, y_r)\}$. As $\mathcal{D}$ is deterministic this properly summarizes the conversation. Note that if $\mathcal{D}$ is a PRP distinguisher, the construction queries are all in forward direction, while if it is an SPRP distinguisher, all queries may be in both directions. This does not influence the definition of the view.

To suit further analysis, we assume that the tuples in v_1 are sorted depending on the key-deriving function that is used. Given that v_1 is an unordered set, this is without loss of generality. In more detail, assume $\mathcal{D}$ made q_0 queries for key-deriving function $\varphi_{0,0}$, q_1 queries for key-deriving functions of the form $\varphi_{\delta,0}$ with $\delta > 0$, and q_2 queries for key-deriving functions of the form $\varphi_{0,\delta}$ with $\delta > 0$. Then, we assume v_1 to be sorted in such a way that

$$(\delta_1,\delta_2)_i = \begin{cases} (0,0) \text{ for } i = 1,\ldots,q_0\,, \\ (\delta_{1,i},0) > (0,0) \text{ for } i = q_0+1,\ldots,q_0+q_1\,, \\ (0,\delta_{2,i}) > (0,0) \text{ for } i = q_0+q_1+1,\ldots,q_0+q_1+q_2\,, \end{cases}$$

where $q_0+q_1+q_2 = q$.

We again reveal some additional information to the distinguisher. As in Section 5.2, there is no need to disclose information prior to the interaction. *After* $\mathcal{D}$'s interaction with its oracles but before $\mathcal{D}$ makes its final decision, we reveal the secret key k and the corresponding $P(k)$, along with additional key material corresponding the tuples $(\varphi_{(\delta_1,\delta_2)_i}(k), P(\varphi_{(\delta_1,\delta_2)_i}(k)))$ for $i = 1,\ldots,q$. Formally, let $\{\epsilon_1,\ldots,\epsilon_{s_1}\}$ be a minimal set that it includes $\delta_{1,q_0+1},\ldots,\delta_{1,q_0+q_1}$, and let $\{\eta,\ldots,\eta_{s_2}\}$ be a minimal set that it includes $\delta_{2,q_0+q_1+1},\ldots,\delta_{2,q_0+q_1+q_2}$. After $\mathcal{D}$'s interaction we reveal $v_k = (v_{k,0}, v_{k,1}, v_{k,2})$, where

$$\begin{aligned} v_{k,0} &= \{(k,k^\star)\}\,, \\ v_{k,1} &= \{(k_{\epsilon_1},k^\star_{\epsilon_1}),\ldots,(k_{\epsilon_{s_1}},k^\star_{\epsilon_{s_1}})\}\,, \\ v_{k,2} &= \{(k^{\star\star}_{\eta_1},k^\star_{\eta_1}),\ldots,(k^{\star\star}_{\eta_{s_2}},k^\star_{\eta_{s_2}})\}\,. \end{aligned}$$

We define $s = 1+s_1+s_2$. The sets $v_{k,0}$ and $v_{k,1}$ are as usual, but $v_{k,2}$ will be defined slightly differently. In the real world, k is the key used for encryption, and $k^\star = P(k)$. The tuples in $v_{k,1}$ are defined as $(k_{\epsilon_i},k^\star_{\epsilon_i}) = (k\oplus\epsilon_i, P(k\oplus\epsilon_i))$. The tuples in $v_{k,2}$ are defined as $(k^{\star\star}_{\eta_i},k^\star_{\eta_i}) = (P^{-1}(k^\star\oplus\eta_i), k^\star\oplus\eta_i)$.[3] In the ideal world, $k,k^\star \xleftarrow{\$} \{0,1\}^n$ will be randomly drawn dummy keys, and we define $k_{\epsilon_i} = k\oplus\epsilon_i$ and $k^\star_{\eta_i} = k^\star\oplus\eta_i$ as before. The values $k^\star_{\epsilon_i} \xleftarrow{\$} \{0,1\}^n$ for $i = 1,\ldots,s_1$ and $k^{\star\star}_{\eta_i}$ for $i = 1,\ldots,s_2$ will be dummy keys.

The complete view is denoted $v = (v_1,v_2,v_k)$. Recall that $\mathcal{D}$ is assumed not to make any repeat queries, and hence v_1 and v_2 do not contain any duplicate elements. Note that v_k may contain collisions or may collide with v_2, but this will be captured as a bad event. As before, we only consider attainable views, which can be obtained in the ideal world.

For the sake of the discussion of the bad events, we will introduce an alternative, unified, notation for tuples in v_k. In more detail, for any offset (δ_1,δ_2) that appears in v_1, we define

$$(k_{\delta_1,\delta_2}, l_{\delta_1,\delta_2}) = \begin{cases} (k,k^\star) \in v_{k,0} \text{ if } \delta_1=\delta_2=0\,, \\ (k_{\delta_1},k^\star_{\delta_1}) \in v_{k,1} \text{ if } \delta_1 \neq 0\,, \\ (k^{\star\star}_{\delta_2},k^\star_{\delta_2}) \in v_{k,2} \text{ if } \delta_2 \neq 0\,. \end{cases} \tag{11}$$

Now, we say that a view v is *bad* if one of the following conditions holds:

BV_1 : for some $(x,y)\in v_2$ and $(k_{\delta_1,\delta_2}, l_{\delta_1,\delta_2}) \in v_k$:

- BV_{1a} : $k_{\delta_1,\delta_2} = x$, or
- BV_{1b} : $l_{\delta_1,\delta_2} = y$, or

BV_2 : for some $((\delta_1,\delta_2),(t_{11},t_{12},t_{21},t_{22}),m,c)\in v_1$, $(x,y)\in v_2\cup v_k$, and $(k_{\delta_1,\delta_2}, l_{\delta_1,\delta_2}) \in v_k$:

- BV_{2a} : $m\oplus t_{11}k_{\delta_1,\delta_2}\oplus t_{12}l_{\delta_1,\delta_2} = x$, or
- BV_{2b} : $c\oplus t_{21}k_{\delta_1,\delta_2}\oplus t_{22}l_{\delta_1,\delta_2} = y$, or

[3] Note that the number of $\star$'s refers to the number of P-evaluations that are needed to compute the value from k. For instance, $k^\star_{\epsilon_1} = P(k\oplus\epsilon_1)$, $k^\star_{\eta_1} = P(k)\oplus\eta_1$, and $k^{\star\star}_{\eta_1} = P^{-1}(P(k)\oplus\eta_1)$.

BV_3 : for some distinct $((\delta_1,\delta_2),(t_{11},t_{12},t_{21},t_{22}),m,c)$,
$((\delta_1',\delta_2'),(t_{11}',t_{12}',t_{21}',t_{22}'),m',c') \in v_1$
and $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}),(k_{\delta_1',\delta_2'},l_{\delta_1',\delta_2'}) \in v_k$:

BV_{3a} : $m \oplus t_{11}k_{\delta_1,\delta_2} \oplus t_{12}l_{\delta_1,\delta_2} = m' \oplus t_{11}'k_{\delta_1',\delta_2'} \oplus t_{12}'l_{\delta_1',\delta_2'}$, or

BV_{3b} : $c \oplus t_{21}k_{\delta_1,\delta_2} \oplus t_{22}l_{\delta_1,\delta_2} = c' \oplus t_{21}'k_{\delta_1',\delta_2'} \oplus t_{22}'l_{\delta_1',\delta_2'}$, or

BV_4 : for some distinct $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}),(k_{\delta_1',\delta_2'},l_{\delta_1',\delta_2'}) \in v_k$:

BV_{4a} : $k_{\delta_1,\delta_2} = k_{\delta_1',\delta_2'}$, or

BV_{4b} : $l_{\delta_1,\delta_2} = l_{\delta_1',\delta_2'}$.

Note that these cases directly generalize the bad events of Section 5.2, using the generalized description of tuples in v_k. The only change appears in BV_4, where now different tuples in v_k may collide *both* at the input and output.

In accordance with Patarin's technique (Lemma 1), we derive an upper bound on $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right]$ for the case of Theorem 1(d) in Lemma 7 and for the case of Theorem 1(e) in Lemma 8. In Lemma 9 we will prove that $\varepsilon = 1$ works for good views (the same analysis applies to both cases (d) and (e)). The proofs are then completed by noting that $s \le q$.

Lemma 7. *For Theorem 1(d), we have* $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right] \le \frac{q^2+2q(r+s)+2rs+s^2}{2^n-q}$.

Proof. The proof is similar to the one of Lemma 4, the most significant changes appear in the analysis of BV_4. However, subtle changes arise for the other events too, because the distinguisher may choose to either XOR k with an offset (resulting in $v_{k,1}$) or $P(k)$ with an offset (resulting in $v_{k,2}$). Recall that $t_{11},t_{12} \neq 0$ and that $\mathcal{D}$ may only make *forward* construction queries, hence the c's in v_1 will always be randomly drawn.

We again bound $\mathbf{Pr}\left[X_{\text{id}} \in \mathcal{V}_{\text{bad}}\right]$ using (10). We have $k,k^\star \xleftarrow{\$} \{0,1\}^n$, and additionally $k^\star_{\epsilon_i} \xleftarrow{\$} \{0,1\}^n$ for $i=1,\dots,s_1$ and $k^{\star\star}_{\eta_i} \xleftarrow{\$} \{0,1\}^n$ for $i=1,\dots,s_2$. We will show that all bad events have one of these keys or a value c as non-trivial term.

Condition BV_1. Consider any choice of $(x,y) \in v_2$ and $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}) \in v_k$. By construction, $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2})$ equals either of $\{(k,k^\star),(k\oplus\delta_1,k^\star_{\delta_1}),(k^{\star\star}_{\delta_2},k^\star\oplus\delta_2)\}$ (cf. (11)), where $k,k^{\star\star}_{\delta_2},k^\star,k^\star_{\delta_1} \xleftarrow{\$} \{0,1\}^n$. Thus, both BV_{1a} and BV_{1b} are satisfied with probability $1/2^n$. Therefore, $\mathbf{Pr}\left[\mathsf{BV}_1\right] \le 2rs/2^n$.

Condition BV_4. Consider any two distinct $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}),(k_{\delta_1',\delta_2'},l_{\delta_1',\delta_2'}) \in v_k$. If $\delta_2=\delta_2'=0$ we are clearly back at the case of Lemma 4, and BV_4 is set with probability $1/2^n$. The case where $\delta_1=\delta_1'=0$ is symmetrically equivalent. Remains the case where (w.l.o.g.) $\delta_1>0$ and $\delta_2'>0$. We particularly have $\delta_2,\delta_1'=0$, and

$$(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}) = (k\oplus\delta_1,k^\star_{\delta_1})\,, \qquad (k_{\delta_1',\delta_2'},l_{\delta_1',\delta_2'}) = (k^{\star\star}_{\delta_2'},k^\star\oplus\delta_{2'})\,, \tag{12}$$

where $k,k^\star_{\delta_1},k^{\star\star}_{\delta_2'},k^\star \xleftarrow{\$} \{0,1\}^n$. In this case, BV_4 is satisfied with probability at most $2/2^n$. Therefore, considering all cases, $\mathbf{Pr}\left[\mathsf{BV}_4\right] \le 2\binom{s}{2}/2^n \le s^2/2^n$.

Condition BV_2. Consider any choice of $((\delta_1,\delta_2),(t_{11},t_{12},t_{21},t_{22}),m,c) \in v_1$ and $(x,y) \in v_2 \cup v_k$. This fixes the corresponding tuple $(k_{\delta_1,\delta_2},l_{\delta_1,\delta_2}) \in v_k$. Regarding BV_{2a}, it is set if

$$m \oplus t_{11}k_{\delta_1,\delta_2} \oplus t_{12}l_{\delta_1,\delta_2} = x\,.$$

Regardless of whether $(x, y) \in v_k$, as $t_{12} \neq 0$ we always have a non-trivial term $l_{\delta_1,\delta_2} \in \{k^\star, k^\star_{\delta_1}, k^\star \oplus \delta_2\}$. This equation is thus satisfied with probability $1/2^n$. Event BV_{2b} is a bit more technical. The condition is satisfied if

$$c \oplus t_{21} k_{\delta_1,\delta_2} \oplus t_{22} l_{\delta_1,\delta_2} = y\,.$$

If $(x, y) \in v_k$, we require that c equals $t_{21} k_{\delta_1,\delta_2} \oplus t_{22} l_{\delta_1,\delta_2} \oplus y$, which happens with probability at most $1/(2^n - q)$. In any other case, if $(x, y) \in v_2$, because $\neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)$ holds and that $(t_{21}, t_{22}) \neq (0, 0)$ by validity of $\mathcal{T}$ point (i), there is always a non-trivial key-term involved, and the equation is satisfied with probability $1/2^n$. Therefore, $\mathbf{Pr}[\mathsf{BV}_2 \mid \neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)] \leq 2q(r + s)/(2^n - q)$.

Condition BV_3. Consider any two distinct $((\delta_1, \delta_2), (t_{11}, t_{12}, t_{21}, t_{22}), m, c)$, $((\delta'_1, \delta'_2), (t'_{11}, t'_{12}, t'_{21}, t'_{22}), m', c') \in v_1$. These fix the corresponding tuples $(k_{\delta_1,\delta_2}, l_{\delta_1,\delta_2})$, $(k_{\delta'_1,\delta'_2}, l_{\delta'_1,\delta'_2}) \in v_k$. If $\delta_2 = \delta'_2 = 0$ we are clearly back at the case of Lemma 4, and the case where $\delta_1 = \delta'_1 = 0$ is symmetrically equivalent (the derivation in this case uses that $t_{11} \neq 0$). Remains the case where (w.l.o.g.) $\delta_1 > 0$ and $\delta'_2 > 0$. We particularly have $\delta_2, \delta'_1 = 0$, and the keys satisfy (12) where $k, k^\star_{\delta_1}, k^{\star\star}_{\delta'_2}, k^\star \xleftarrow{\$} \{0,1\}^n$. As $(t_{11}, t_{12}), (t'_{11}, t'_{12}), (t_{21}, t_{22}), (t'_{21}, t'_{22}) \neq (0, 0)$ by validity of $\mathcal{T}$ point (i), BV_3 always contains one of these four key values, and both BV_{3a} and BV_{3b} are satisfied with probability $1/2^n$. Therefore, considering all cases, $\mathbf{Pr}[\mathsf{BV}_3 \mid \neg\mathsf{BV}_4] \leq 2\binom{q}{2}/(2^n - q) \leq q^2/(2^n - q)$.

Conclusion. Using (10), we have $\mathbf{Pr}[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}] \leq \frac{q^2+2q(r+s)+2rs+s^2}{2^n-q}$. This completes the proof. □

Lemma 8. *For Theorem 1(e), we have* $\mathbf{Pr}[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}] \leq \frac{q^2+2q(r+s)+2rs+s^2}{2^n}$.

Proof. The proof follows the one of Lemma 7 with the only differences at BV_2 and BV_3. In more detail, we cannot rely on the randomness of the c-values, but we can use that $t_{21}, t_{22} \neq 0$. For BV_1 and BV_4 the analysis is identical and we find that $\mathbf{Pr}[\mathsf{BV}_1] \leq 2rs/2^n$ and $\mathbf{Pr}[\mathsf{BV}_4] \leq s^2/2^n$.

Condition BV_2. For BV_{2a} the analysis is identical. For BV_{2b}, the condition is satisfied if

$$m \oplus t_{21} k_{\delta_1,\delta_2} \oplus t_{22} l_{\delta_1,\delta_2} = y\,.$$

Regardless of whether $(x, y) \in v_k$, as $t_{21} \neq 0$ we always have a non-trivial term $k_{\delta_1,\delta_2} \in \{k, k \oplus \delta_1, k^{\star\star}_{\delta_2}\}$. This equation is thus satisfied with probability $1/2^n$. We find that $\mathbf{Pr}[\mathsf{BV}_2 \mid \neg(\mathsf{BV}_1 \vee \mathsf{BV}_4)] \leq 2q(r + s)/2^n$.

Condition BV_3. For BV_3 the analysis is identical, with the difference that for BV_{3b} we now rely on both Lemma 4 and Lemma 5. In more detail, for the case where $(t_{11}, t_{12}, t_{21}, t_{22}) = (t'_{11}, t'_{12}, t'_{21}, t'_{22})$ but $\delta \neq \delta'$, we use the reasoning of Lemma 5 for both the case $\delta_2 = \delta'_2 = 0$ (using that $t_{21} \neq 0$) and the case $\delta_1 = \delta'_1 = 0$ (using that $t_{22} \neq 0$). The remainder of the proof remains unchanged. Therefore, $\mathbf{Pr}[\mathsf{BV}_3 \mid \neg\mathsf{BV}_4] \leq 2\binom{q}{2}/2^n \leq q^2/2^n$.

Conclusion. Using (10), we have $\mathbf{Pr}[X_{\mathrm{id}} \in \mathcal{V}_{\mathrm{bad}}] \leq \frac{q^2+2q(r+s)+2rs+s^2/2}{2^n}$. This completes the proof. □

Lemma 9. *For Theorem 1(d) and Theorem 1(e), we have* $\mathbf{Pr}[X_{\mathrm{re}} = v] \geq \mathbf{Pr}[X_{\mathrm{id}} = v]$ *for any good transcript* $v \in \mathcal{V}_{\mathrm{good}}$.

Proof. The proof of Lemma 6 carries over directly, with the difference that now v_k contains $1+s_1+s_2=s$ tuples. In more detail, the computation of $\mathbf{Pr}[X_{\mathrm{re}}=v]$ is exactly the same. In the ideal world, we have

$$\begin{aligned}\mathbf{Pr}[X_{\mathrm{id}}=v] = \mathbf{Pr}&\left[\begin{array}{r}k', k^{\star\prime}, {k^{\star}_{\epsilon_1}}', \ldots, {k^{\star}_{\epsilon_{s_1}}}', k^{\star\star\prime}_{\eta_1}, \ldots, {k^{\star\star}_{\eta_{s_2}}}' \xleftarrow{\$} \{0,1\}^n \ : \\ k'=k \wedge k^{\star\prime}=k^{\star} \wedge {k^{\star}_{\epsilon_i}}' = k^{\star}_{\epsilon_i} \wedge k^{\star\star\prime}_{\eta_i} = k^{\star\star}_{\eta_i}\end{array}\right]\cdot \\ &\mathbf{Pr}\left[\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK}\text{-}\mathsf{Perm}}(\Phi_{P\oplus}, \mathcal{T}, \mathcal{M}) \ : \ \widetilde{\mathsf{RK}\pi} \vdash v_1\right]\cdot \\ &\mathbf{Pr}\left[P \xleftarrow{\$} \mathsf{Perm}(\mathcal{M}) \ : \ P \vdash v_2\right] \\ = &\frac{1}{2^{(s+1)n}} \cdot \frac{1}{\prod_{(\delta_1,\delta_2),t} (2^n)_{\#_{(\delta_1,\delta_2),t}}} \cdot \frac{1}{(2^n)_r}\,, \text{ where } \textstyle\sum_{(\delta_1,\delta_2),t} \#_{(\delta_1,\delta_2),t} = q\,,\end{aligned}$$

where $\#_{(\delta_1,\delta_2),t}$ is defined similarly as before. We obtain $\mathbf{Pr}[X_{\mathrm{re}}=v] \geq \mathbf{Pr}[X_{\mathrm{id}}=v]$. $\square$

6 Application to Authenticated Encryption

We will show how XPX applies to the Prøst-COPA [3,24] and Minalpher [38] authenticated encryption schemes. Before doing so, we briefly discuss the security model.

6.1 Security Model

Authenticated encryption covers the case where both privacy and authenticity of data is required. In more detail, an authenticated encryption scheme consists of an encryption function Enc and a decryption function Dec. Enc gets as input a key, nonce, associated data, and message, and outputs a ciphertext and a tag. Dec gets as input a key, nonce, associated data, ciphertext, and tag, and it either outputs a message (if the authentication is correct) or a dedicated $\perp$ symbol.

Let $\mathsf{AE} = (\mathsf{Enc}, \mathsf{Dec})$ be an authenticated encryption scheme, and let $\mathcal{P}$ be an idealized primitive upon which AE is based (optional, for instance a blockcipher or permutation). Let k be a randomly drawn key. Let \$ be a function with the same interface as E_k, but that returns fresh and random answers to every query. Let $\perp$ be a function that outputs $\perp$ on every query. We define the privacy of AE based on $\mathcal{P}$ as

$$\mathbf{Adv}^{\mathrm{priv}}_{\mathsf{AE}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\mathsf{Enc}_k,\mathcal{P}^{\pm}}=1\right] - \mathbf{Pr}\left[\mathcal{D}^{\$,\mathcal{P}^{\pm}}=1\right]\right|\,,$$

and the authenticity of AE based on $\mathcal{P}$ as

$$\mathbf{Adv}^{\mathrm{auth}}_{\mathsf{AE}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\mathsf{Enc}_k,\mathsf{Dec}_k,\mathcal{P}^{\pm}}=1\right] - \mathbf{Pr}\left[\mathcal{D}^{\mathsf{Enc}_k,\perp,\mathcal{P}^{\pm}}=1\right]\right|\,.$$

In both definitions, some conditions on $\mathcal{D}$ may apply (such as the nonce-respecting condition). For $q, \ell, \sigma, r \geq 0$, we define by

$$\mathbf{Adv}^{\mathrm{priv/auth}}_{\mathsf{AE}}(q,\ell,\sigma,r) = \max_{\mathcal{D}} \mathbf{Adv}^{\mathrm{priv/auth}}_{\mathsf{AE}}(\mathcal{D})$$

the security of AE against any distinguisher $\mathcal{D}$ that makes q queries to the construction oracle, each of length at most ℓ and of total size σ, and r queries to the primitive oracle.

So far, the model is in the single-key setting, But it generalizes to related-key security straightforwardly (the way Section 2.2 generalizes Section 2.1). We denote the corresponding related-key security definitions by

$$\mathbf{Adv}^{\mathrm{rk\text{-}priv/auth}}_{\Phi,\mathsf{AE}}(\mathcal{D}) \text{ and } \mathbf{Adv}^{\mathrm{rk\text{-}priv/auth}}_{\Phi,\mathsf{AE}}(q,\ell,\sigma,r)\,,$$

where Φ is some key-deriving function set.

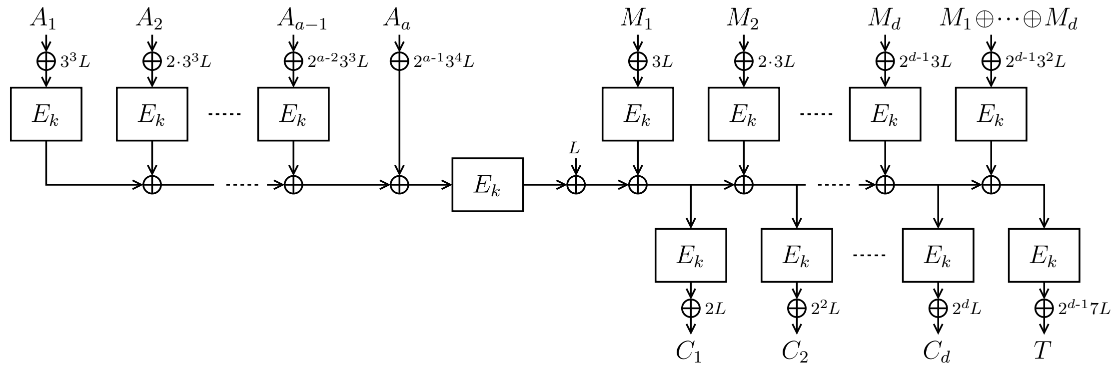

Fig. 4: COPA for integral data. Here, $L = E_k(0)$.

6.2 Prøst-COPA

COPA is an authenticated encryption scheme by Andreeva et al. [3]. COPA for integral message is depicted in Figure 4 (we refer to [3] for the general case). At its core, it is using a blockcipher E in XEX mode (2) with masks $\Delta = 2^\alpha 3^\beta 7^\gamma E_k(0)$, where (α, β, γ) is the tweak coming from tweak space $\{0, \ldots, \ell\} \times \{0, \ldots, 5\} \times \{0, 1\} \backslash \{(0, 0, 0)\} = \mathcal{T}_{\text{COPA}}$.[4]

Before discussing the related-key security of COPA, we quickly revisit the original security proof at a high level. Consider an attacker against COPA that has resources (q, ℓ, σ, r). As a first step, all XEX evaluations in COPA are replaced with a random tweakable permutation $\widetilde{\pi} \xleftarrow{\$} \widetilde{\mathsf{Perm}}(\mathcal{T}_{\text{COPA}}, \{0,1\}^n)$. This step costs us $\mathbf{Adv}^{\text{sprp}}_{\text{XEX}}(2(\sigma+q), r)$. Next, COPA with ideal tweakable permutation is proven to achieve privacy up to bound $A_{\text{priv}}(q, \ell, \sigma) = \frac{\sigma^2}{2^n} + \frac{(\ell+2)(q-1)^2}{2^n}$ and authenticity up to bound $A_{\text{auth}}(q, \ell, \sigma) = \frac{(\sigma+q)^2}{2^n} + \frac{(\ell+2)(q-1)^2}{2^n} + \frac{2q}{2^n}$. Thus:

$$\mathbf{Adv}^{\text{priv/auth}}_{\text{COPA}}(q, \ell, \sigma, r) \leq \mathbf{Adv}^{\text{sprp}}_{\text{XEX}}(2(\sigma+q), r) + A_{\text{priv/auth}}(q, \ell, \sigma)\,.$$

The step towards RK-security of COPA is quite straightforward, noting that an attacker against COPA with ideal tweakable *related-key* permutation has no benefit over an attacker against COPA with ideal tweakable (non-related-key) permutation.

Theorem 2 (RK-security of COPA). *Let Φ be any KDF-set. We have*

$$\mathbf{Adv}^{\text{rk-priv/auth}}_{\Phi,\text{COPA}}(q, \ell, \sigma, r) \leq \mathbf{Adv}^{\text{rk-sprp}}_{\Phi,\text{XEX}}(2(\sigma+q), r) + A_{\text{priv/auth}}(q, \ell, \sigma)\,.$$

Proof. Consider an attacker against COPA that has resources (q, ℓ, σ, r). As a first step, all XEX evaluations in COPA are replaced with a random tweakable related-key permutation $\widetilde{\mathsf{RK}\pi} \xleftarrow{\$} \widetilde{\mathsf{RK\text{-}Perm}}(\Phi, \mathcal{T}_{\text{COPA}}, \{0,1\}^n)$. This step costs $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi,\text{XEX}}(2(\sigma+q), r)$. It remains to consider COPA with $\widetilde{\mathsf{RK}\pi}$. However, as $\widetilde{\mathsf{RK}\pi}$ instantiates an ideal permutation for every different related-key function, every new related-key function instantiates a completely independent instance of COPA. Formally, assume the adversary queries COPA for s different key-deriving functions, $\varphi_1, \ldots, \varphi_s$, where φ_i is used with total resources (q_i, ℓ_i, σ_i). These all instantiate independent versions of COPA, contributing $A_{\text{priv/auth}}(q_i, \ell_i, \sigma_i)$ to

[4] The fact that $(0, 0, 0) \notin \mathcal{T}_{\text{COPA}}$ is important, cf. Rogaway [37] and Minematsu [31] who describe an attack on XEX if $(0, 0, 0)$ were permitted.

the bound, totaling to

$$\sum_{i=1}^{s} A_{\text{priv/auth}}(q_i, \ell_i, \sigma_i) \leq A_{\text{priv/auth}}(q, \ell, \sigma)\,,$$

which completes the proof. □

Prøst-COPA [24], in turn, uses the Prøst permutation in Even-Mansour mode. In other words, Prøst-COPA does not simply use XEX, but XPX with tweak space

$$\mathcal{T}_{\text{Prøst}} = \left\{ \begin{matrix} (2^\alpha 3^\beta 7^\gamma \oplus 1, 2^\alpha 3^\beta 7^\gamma, \\ 2^\alpha 3^\beta 7^\gamma \oplus 1, 2^\alpha 3^\beta 7^\gamma) \end{matrix} \,\middle|\, (\alpha, \beta, \gamma) \in \mathcal{T}_{\text{COPA}} \right\}. \tag{13}$$

Taking any of the KDF-sets $\Phi \in \{\Phi_\oplus, \Phi_{P\oplus}\}$ of (4), we find:

Corollary 1 (RK-security of Prøst-COPA). *For Φ being $\Phi_\oplus$ or $\Phi_{P\oplus}$ of (4), we have*

$$\mathbf{Adv}^{\text{rk-priv/auth}}_{\Phi,\text{Prøst-COPA}}(q, \ell, \sigma, r) \leq \frac{16(\sigma+q)^2 + 8(\sigma+q)r}{2^n} + A_{\text{priv/auth}}(q, \ell, \sigma)\,.$$

Proof. The proof of Theorem 2 generalizes to Prøst-COPA straightforwardly, where $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi,\text{XEX}}(2(\sigma+q), r)$ gets replaced with $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi,\text{XPX}}(2(\sigma+q), r)$. This XPX is instantiated using tweak space $\mathcal{T}_{\text{Prøst}}$ of (13), which is valid and satisfies $t_{11}, t_{12}, t_{21}, t_{22} \neq 0$ for any $(t_{11}, t_{12}, t_{21}, t_{22}) \in \mathcal{T}_{\text{Prøst}}$ (note that $(\alpha, \beta, \gamma) = (0, 0, 0)$ is excluded). Therefore, Theorem 1(c) applies for $\Phi = \Phi_\oplus$ and Theorem 1(e) for $\Phi = \Phi_{P\oplus}$. In the worst case, we find that

$$\mathbf{Adv}^{\text{rk-sprp}}_{\Phi,\text{XPX}}(2(\sigma+q), r) \leq \frac{16(\sigma+q)^2 + 8(\sigma+q)r}{2^n}\,,$$

completing the proof. □

Note that if Prøst-COPA were not to use Prøst permutation in Even-Mansour mode, but if it simply had $E = P$, then the resulting XPX construction would have tweak space

$$\mathcal{T}_{\text{Prøst}'} = \left\{ (0, 2^\alpha 3^\beta 7^\gamma, 0, 2^\alpha 3^\beta 7^\gamma) \mid (\alpha, \beta, \gamma) \in \mathcal{T}_{\text{COPA}} \right\}.$$

This tweak space does not satisfy the conditions of Theorem 1(e) and we can only argue the related-key security of Prøst-COPA under $\Phi_\oplus$.

6.3 Minalpher

Minalpher is an authenticated encryption scheme by Sasaki et al. [38]. At its core, it is using tweakable Even-Mansour TEM of (3): an evaluation of an n-bit permutation with masks[5] $\Delta = 2^\alpha 3^\beta \big(k \| \mathsf{flag} \| N \oplus P(k \| \mathsf{flag} \| N)\big)$, where $(\alpha, \beta, \mathsf{flag}, N)$ is the tweak coming from tweak space

$$\big(\{0, \ldots, \ell\} \times \{0, 1, 2\}\big) \backslash \{(0, 0)\} \times \{\mathsf{flag_m}, \mathsf{flag_{ad}}, \mathsf{flag_{mac}}\} \times \{0, 1\}^{n/2-s} = \mathcal{T}_{\text{Minalpher}}\,.$$

Here, the key k is of size $n/2$ bits, the flag of size s bits, and the nonce N of size $n/2 - s$ bits.

[5] The original specification uses a generator $\mathtt{y}$ instead of 2.

The authors prove, among others, that $\mathbf{Adv}_{\mathrm{TEM}}^{\mathrm{sprp}}(q,r) \leq \mathcal{O}((q+r)^2/2^n + (q+r)/2^{n/2})$. The extra term $\mathcal{O}((q+r)/2^{n/2})$ is new compared to Theorem 1(a), and is caused by the shorter key size. A bit of thought reveals that, because the tweaks $\mathsf{flag}\|N$ are concatenated to k instead of XORed with k, the results of Theorem 1(b-e) generalize to TEM. Here, again, the specific key length needs to be taken into account. In [38], the designers prove that if the underlying TEM is sufficiently strong, Minalpher is a secure authenticated encryption scheme. In a similar fashion as Theorem 2 and Corollary 1, a generalization of Theorem 1(b-e) can be used to argue the related-key security of Minalpher.

7 Application to MAC

Various novel MAC functions, such as the keyed Sponges [4, 6, 11, 22] and Chaskey [33], consist of a sequential application of a permutation, where the key is used to mask the state. We discuss an application of the analysis of XPX to Chaskey in detail, and explain how similar reasoning applies to keyed Sponges. We first briefly discuss the security model.

7.1 Security Model

A MAC function is expected to guarantee authenticity. However, we consider a different security model, namely PRF security. More formally, let MAC be a MAC function that gets as input a key and message, and outputs a tag. Let $\mathcal{P}$ be an idealized primitive upon which MAC is based (optional, for instance a blockcipher or permutation). Let k be a randomly drawn key. Let \$ be a function with the same interface as MAC, but that returns fresh and random answers to every query. We define the PRF security of MAC based on $\mathcal{P}$ as

$$\mathbf{Adv}_{\mathsf{MAC}}^{\mathrm{prf}}(\mathcal{D}) = \left|\mathbf{Pr}\left[\mathcal{D}^{\mathsf{MAC}_k,\mathcal{P}^{\pm}} = 1\right] - \mathbf{Pr}\left[\mathcal{D}^{\$,\mathcal{P}^{\pm}} = 1\right]\right| .$$

For $q, \ell, \sigma, r \geq 0$, we define by

$$\mathbf{Adv}_{\mathsf{MAC}}^{\mathrm{prf}}(q,\ell,\sigma,r) = \max_{\mathcal{D}} \mathbf{Adv}_{\mathsf{MAC}}^{\mathrm{prf}}(\mathcal{D})$$

the security of MAC against any distinguisher $\mathcal{D}$ that makes q queries to the construction oracle, each of length at most ℓ and of total size σ, and r queries to the primitive oracle.

As before, the definition generalizes to related-key security straightforwardly, and we denote the corresponding related-key security definitions by

$$\mathbf{Adv}_{\Phi,\mathsf{MAC}}^{\text{rk-prf}}(\mathcal{D}) \text{ and } \mathbf{Adv}_{\Phi,\mathsf{MAC}}^{\text{rk-prf}}(q,\ell,\sigma,r)\,,$$

where Φ is some key-deriving function set.

7.2 Chaskey

Chaskey is a permutation-based MAC function by Mouha et al. [33]. We consider a small adjustment, called Chaskey′, that processes the initialized state with an evaluation of the permutation. Chaskey and Chaskey′ without final truncation are depicted in Figure 5.

Mouha et al. [33] proved the security of Chaskey (without the first evaluation of P). It consists of the idea that XORing the key k twice in-between every two consecutive P evaluations gives a blockcipher-based Chaskey using Even-Mansour constructions $m \mapsto P(m \oplus k) \oplus k$, $m \mapsto P(m \oplus 3k) \oplus 2k$, and $m \mapsto P(m \oplus 5k) \oplus 4k$. The security of Chaskey boils down to the advantage of a distinguisher in distinguishing these three constructions

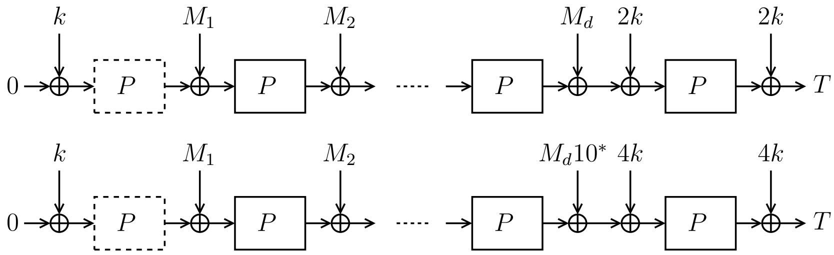

Fig. 5: Chaskey′ for integral messages (top) and fractional messages (bottom). The dashed P's are absent in the original Chaskey.

from three ideal permutations, an advantage the authors dub the "3PRP" security. This 3PRP security is effectively equivalent to the PRP security of XPX with tweak space $\{(1,0,1,0),(3,0,2,0),(5,0,4,0)\} = \mathcal{T}_{\text{Chaskey}}$, and we find:[6]

$$\mathbf{Adv}^{\text{prf}}_{\text{Chaskey}}(q,\ell,\sigma,r) \le \mathbf{Adv}^{\text{prp}}_{\text{XPX}}(\sigma,r) + \frac{2\sigma^2}{2^n}\,.$$

Now, for Chaskey′, The idea is to XOR $P(k)\oplus P(k)$ everywhere in-between two consecutive P evaluations *except for the first two*. In this case, Chaskey′ would simply be using XPX with tweak space

$$\{(0,1,0,1),(2,1,2,0),(4,1,4,0)\} = \mathcal{T}_{\text{Chaskey}'}\,.$$

Note that $\mathcal{T}_{\text{Chaskey}'}$ satisfies the conditions of Theorem 1(b). Similarly to Theorem 2 and Corollary 1, we directly obtain:

Corollary 2 (RK-security of Chaskey′). *For $\Phi_\oplus$ of (4), we have*

$$\mathbf{Adv}^{\text{rk-prf}}_{\Phi_\oplus,\text{Chaskey}'}(q,\ell,\sigma,r) \le \frac{\frac{7}{2}\sigma^2 + 4\sigma r}{2^n-\sigma} + \frac{2\sigma^2}{2^n}\,.$$

7.3 Keyed Sponge

Following [6, 11], Andreeva et al. [4] formalized two Sponges: the inner-keyed Sponge and the outer-keyed Sponge. Gaži et al. [22] generalized these results (among others) to full-state absorption. This construction, to some extent, resembles the Donkey Sponge construction [7]. In a similar fashion as the analysis of Section 7.2, the inner-keyed Sponge [4] and the Donkey Sponge [7] can be adjusted to achieve related-key security.

Acknowledgments. This work was supported in part by the Research Council KU Leuven: GOA TENSE (GOA/11/007). Bart Mennink is a Postdoctoral Fellow of the Research Foundation – Flanders (FWO). The author would like to thank the DTU Compute team for their comments and suggestions.

[6] The authors of [33] effectively consider MAC security instead of PRF security, but the analysis carries over.

References

[1] Albrecht, M., Farshim, P., Paterson, K., Watson, G.: On cipher-dependent related-key attacks in the ideal-cipher model. In: FSE 2011. LNCS, vol. 6733, pp. 128–145. Springer (2011)

[2] Andreeva, E., Bogdanov, A., Dodis, Y., Mennink, B., Steinberger, J.P.: On the indifferentiability of key-alternating ciphers. In: CRYPTO 2013 (I). LNCS, vol. 8042, pp. 531–550. Springer (2013)

[3] Andreeva, E., Bogdanov, A., Luykx, A., Mennink, B., Tischhauser, E., Yasuda, K.: Parallelizable and authenticated online ciphers. In: ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 424–443. Springer (2013)

[4] Andreeva, E., Daemen, J., Mennink, B., Van Assche, G.: Security of keyed sponge constructions using a modular proof approach. In: FSE 2015. LNCS, Springer (2015), to appear

[5] Bellare, M., Kohno, T.: A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: EUROCRYPT 2003. LNCS, vol. 2656, pp. 491–506. Springer (2003)

[6] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: On the security of the keyed sponge construction. SKEW 2011 (2011)

[7] Bertoni, G., Daemen, J., Peeters, M., Van Assche, G.: Permutation-based encryption, authentication and authenticated encryption. DIAC 2012 (2012)

[8] Bogdanov, A., Knudsen, L.R., Leander, G., Standaert, F., Steinberger, J.P., Tischhauser, E.: Key-alternating ciphers in a provable setting: Encryption using a small number of public permutations. In: EUROCRYPT 2012. LNCS, vol. 7237, pp. 45–62. Springer (2012)

[9] CAESAR: Competition for Authenticated Encryption: Security, Applicability, and Robustness (May 2015), `http://competitions.cr.yp.to/caesar.html`

[10] Chakraborty, D., Sarkar, P.: A general construction of tweakable block ciphers and different modes of operations. In: Inscrypt 2006. LNCS, vol. 4318, pp. 88–102. Springer (2006)

[11] Chang, D., Dworkin, M., Hong, S., Kelsey, J., Nandi, M.: A keyed sponge construction with pseudorandomness in the standard model. SHA-3 Candidate Conference 2012 (2012)

[12] Chen, S., Lampe, R., Lee, J., Seurin, Y., Steinberger, J.P.: Minimizing the two-round Even-Mansour cipher. In: CRYPTO 2014 (I). LNCS, vol. 8616, pp. 39–56. Springer (2014)

[13] Chen, S., Steinberger, J.: Tight security bounds for key-alternating ciphers. In: EUROCRYPT 2014. LNCS, vol. 8441, pp. 327–350. Springer (2014)

[14] Cogliati, B., Seurin, Y.: On the provable security of the iterated Even-Mansour cipher against related-key and chosen-key attacks. In: EUROCRYPT 2015. LNCS, vol. 9056, pp. 584–613. Springer (2015)

[15] Dai, Y., Lee, J., Mennink, B., Steinberger, J.P.: Tight security bounds for multiple encryption. In: CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 20–38. Springer (2014)

[16] Datta, N., Nandi, M.: ELmD v1.0 (2014), submission to CAESAR competition

[17] Dobraunig, C., Eichlseder, M., Mendel, F.: Related-key forgeries for Prøst-OTR. In: FSE 2015. LNCS, Springer (2015), to appear

[18] Dunkelman, O., Keller, N., Shamir, A.: Minimalism in cryptography: The Even-Mansour scheme revisited. In: EUROCRYPT 2012. LNCS, vol. 7237, pp. 336–354. Springer (2012)

[19] Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. In: ASIACRYPT '91. LNCS, vol. 739, pp. 201–224. Springer-Verlag, Berlin (1991)

[20] Even, S., Mansour, Y.: A construction of a cipher from a single pseudorandom permutation. Journal of Cryptology 10(3), 151–162 (1997)

[21] Farshim, P., Procter, G.: The related-key security of iterated Even-Mansour ciphers. In: FSE 2015. LNCS, Springer (2015), to appear

[22] Gaži, P., Pietrzak, K., Tessaro, S.: The exact PRF security of truncation: Tight bounds for keyed sponges and truncated CBC. In: CRYPTO 2015. LNCS, Springer (2015), to appear

[23] Karpman, P.: From related-key distinguishers to related-key-recovery on Even-Mansour constructions. Cryptology ePrint Archive, Report 2015/134 (2015)

[24] Kavun, E., Lauridsen, M., Leander, G., Rechberger, C., Schwabe, P., Yalçın, T.: Prøst v1 (2014), submission to CAESAR competition

[25] Lampe, R., Patarin, J., Seurin, Y.: An asymptotically tight security analysis of the iterated Even-Mansour cipher. In: ASIACRYPT 2012. LNCS, vol. 7658, pp. 278–295. Springer (2012)

[26] Lampe, R., Seurin, Y.: How to construct an ideal cipher from a small set of public permutations. In: ASIACRYPT 2013 (I). LNCS, vol. 8269, pp. 444–463. Springer (2013)

[27] Lampe, R., Seurin, Y.: Tweakable blockciphers with asymptotically optimal security. In: FSE 2013. LNCS, vol. 8424, pp. 133–151. Springer (2013)

[28] Landecker, W., Shrimpton, T., Terashima, R.S.: Tweakable blockciphers with beyond birthday-bound security. In: CRYPTO 2012. LNCS, vol. 7417, pp. 14–30. Springer (2012)

[29] Liskov, M., Rivest, R.L., Wagner, D.: Tweakable block ciphers. In: CRYPTO 2002. LNCS, vol. 2442, pp. 31–46. Springer (2002)

[30] Mennink, B.: Optimally secure tweakable blockciphers. In: FSE 2015. LNCS, Springer (2015), to appear
[31] Minematsu, K.: Improved security analysis of XEX and LRW modes. In: SAC 2006. LNCS, vol. 4356, pp. 96–113. Springer (2007)
[32] Minematsu, K.: Beyond-birthday-bound security based on tweakable block cipher. In: FSE 2009. LNCS, vol. 5665, pp. 308–326. Springer (2009)
[33] Mouha, N., Mennink, B., Van Herrewege, A., Watanabe, D., Preneel, B., Verbauwhede, I.: Chaskey: An efficient MAC algorithm for 32-bit microcontrollers. In: SAC 2014. LNCS, vol. 8781, pp. 306–323. Springer (2014)
[34] Nyberg, K., Knudsen, L.R.: Provable security against differential cryptanalysis. In: CRYPTO '92. LNCS, vol. 740, pp. 566–574. Springer (1993)
[35] Patarin, J.: A proof of security in $O(2^n)$ for the Xor of two random permutations. In: ICITS 2008. LNCS, vol. 5155, pp. 232–248. Springer (2008)
[36] Procter, G.: A note on the CLRW2 tweakable block cipher construction. Cryptology ePrint Archive, Report 2014/111 (2014)
[37] Rogaway, P.: Efficient instantiations of tweakable blockciphers and refinements to modes OCB and PMAC. In: ASIACRYPT 2004. LNCS, vol. 3329, pp. 16–31. Springer (2004)
[38] Sasaki, Y., Todo, Y., Aoki, K., Naito, Y., Sugawara, T., Murakami, Y., Matsui, M., Hirose, S.: Minalpher v1 (2014), submission to CAESAR competition
[39] Steinberger, J.: Improved security bounds for key-alternating ciphers via hellinger distance. Cryptology ePrint Archive, Report 2012/481 (2012)

A Minimality of the Conditions of Theorem 1

We show that the conditions we put on $\mathcal{T}$ in Theorem 1 are minimal, in the sense that XPX can be broken if the conditions are omitted. For the validity condition on $\mathcal{T}$, this is already justified by Proposition 1. Below proposition considers the remaining conditions on $\mathcal{T}$ put by parts (b)-(e) of Theorem 1.

Proposition 2. *Let $n \geq 1$ and let $\mathcal{T} \subseteq (\{0,1\}^n)^4$ a* valid *set.*

(b) If $(t_{11}, 0, t_{21}, t_{22}) \in \mathcal{T}$ for some t_{11}, t_{21}, t_{22}, then

$$\mathbf{Adv}^{\text{rk-prp}}_{\Phi_\oplus,\text{XPX}}(4,0) \geq 1 - 1/(2^n-1)\,.$$

(c) If $(t_{11}, t_{12}, t_{21}, 0) \in \mathcal{T}$ or $(t_{11}, t_{12}, 0, 1) \in \mathcal{T}$ for some t_{11}, t_{12}, t_{21}, then

$$\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_\oplus,\text{XPX}}(4,0) \geq 1 - 1/(2^n-1)\,.$$

(d) If $(0, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ for some t_{12}, t_{21}, t_{22}, then

$$\mathbf{Adv}^{\text{rk-prp}}_{\Phi_{P\oplus},\text{XPX}}(4,0) \geq 1 - 1/(2^n-1)\,.$$

(e) If $(t_{11}, t_{12}, 0, t_{22}) \in \mathcal{T}$ for some t_{11}, t_{12}, t_{22}, then

$$\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_{P\oplus},\text{XPX}}(4,0) \geq 1 - 1/(2^n-1)\,.$$

Proof. We consider the four cases separately.

Case (b). Suppose $(t_{11}, 0, t_{21}, t_{22}) \in \mathcal{T}$ for some t_{11}, t_{21}, t_{22}. Fix any $\delta \neq \delta'$ and any $m \in \{0,1\}^n$. The attacker makes the following queries:

- $\text{XPX}_k(\delta, (t_{11}, 0, t_{21}, t_{22}), m)$ to receive $c = t_{21}(k \oplus \delta) \oplus t_{22}P(k \oplus \delta) \oplus P(\text{inp})$ where $\text{inp} = m \oplus t_{11}(k \oplus \delta)$;
- $\text{XPX}_k(\delta', (t_{11}, 0, t_{21}, t_{22}), m \oplus t_{11}(\delta \oplus \delta'))$ to receive $c' = t_{21}(k \oplus \delta') \oplus t_{22}P(k \oplus \delta') \oplus P(\text{inp}')$ where $\text{inp}' = m \oplus t_{11}(\delta \oplus \delta') \oplus t_{11}(k \oplus \delta')$.

By construction, $\text{inp}' = \text{inp}$, and thus

$$c \oplus c' = t_{21}(\delta \oplus \delta') \oplus t_{22}\big(P(k \oplus \delta) \oplus P(k \oplus \delta')\big)\,.$$

This equation is independent of the choice of m. Making these queries for two different messages $m \neq m'$ gives a collision with probability 1. For a random $\widetilde{\mathsf{RK}\pi}$ this happens with probability at most $1/(2^n - 1)$. Thus, $\mathbf{Adv}^{\text{rk-prp}}_{\Phi_\oplus,\text{XPX}}(4, 0) \geq 1 - 1/(2^n - 1)$.

Case (c). If $(t_{11}, t_{12}, t_{21}, 0) \in \mathcal{T}$ for some t_{11}, t_{12}, t_{21} the attack is the inverse of the one for case (b). Now, suppose $(t_{11}, t_{12}, 0, 1) \in \mathcal{T}$ for some t_{11}, t_{12}. The attacker makes the following queries:

- $\text{XPX}_k^{-1}(0, (t_{11}, t_{12}, 0, 1), 0)$ to receive $m = (t_{11} \oplus 1)k \oplus t_{12}P(k)$;
- $\text{XPX}_k(0, (t_{11}, t_{12}, 0, 1), m \oplus \delta)$ for $\delta \neq 0$ to receive

$$\begin{aligned} c_\delta &= P(k) \oplus P(m \oplus \delta \oplus t_{11}k \oplus t_{12}P(k)) \\ &= P(k) \oplus P(k \oplus \delta)\,. \end{aligned}$$

Now, fix any m' and query

- $\text{XPX}_k(\delta, (t_{11}, t_{12}, 0, 1), m')$ to receive $c' = P(m' \oplus t_{11}(k \oplus \delta) \oplus t_{12}P(k \oplus \delta)) \oplus P(k \oplus \delta)$;
- $\text{XPX}_k(0, (t_{11}, t_{12}, 0, 1), m' \oplus t_{11}\delta \oplus t_{12}c_\delta)$ to receive $c'' = P(m' \oplus t_{11}\delta \oplus t_{12}c_\delta \oplus t_{11}k \oplus t_{12}P(k)) \oplus P(k)$.

These queries satisfy $c' \oplus c'' = c_\delta$. For a random $\widetilde{\mathsf{RK}\pi}$ this happens with probability at most $1/(2^n - 1)$. Thus, $\mathbf{Adv}^{\text{rk-sprp}}_{\Phi_\oplus,\text{XPX}}(4, 0) \geq 1 - 1/(2^n - 1)$.

Case (d). Suppose $(0, t_{12}, t_{21}, t_{22}) \in \mathcal{T}$ for some t_{12}, t_{21}, t_{22}. Fix any $\delta \neq \delta'$ and any $m \in \{0, 1\}^n$. The attacker makes the following queries:

- $\text{XPX}_k((0, \delta), (0, t_{12}, t_{21}, t_{22}), m)$ to receive $c = t_{21}P^{-1}(P(k) \oplus \delta) \oplus t_{22}(P(k) \oplus \delta) \oplus P(\text{inp})$ where $\text{inp} = m \oplus t_{12}(P(k) \oplus \delta)$;
- $\text{XPX}_k((0, \delta'), (0, t_{12}, t_{21}, t_{22}), m \oplus t_{12}(\delta \oplus \delta'))$ to receive $c' = t_{21}P^{-1}(P(k) \oplus \delta') \oplus t_{22}(P(k) \oplus \delta') \oplus P(\text{inp}')$ where $\text{inp}' = m \oplus t_{12}(\delta \oplus \delta') \oplus t_{12}(P(k) \oplus \delta')$.

By construction, $\text{inp}' = \text{inp}$, and thus

$$c \oplus c' = t_{21}\big(P^{-1}(P(k) \oplus \delta) \oplus P^{-1}(P(k) \oplus \delta')\big) \oplus t_{22}(\delta \oplus \delta')\,.$$

This equation is independent of the choice of m. Making these queries for two different messages $m \neq m'$ gives a collision with probability 1. For a random $\widetilde{\mathsf{RK}\pi}$ this happens with probability at most $1/(2^n - 1)$. Thus, $\mathbf{Adv}^{\text{rk-prp}}_{\Phi_{P\oplus},\text{XPX}}(4, 0) \geq 1 - 1/(2^n - 1)$.

Case (e). The attack is the inverse of the one for case (d). $\square$