Quantum Attacks on 1K-AES and PRINCE

Article *in* The Computer Journal · February 2022

DOI: 10.1093/comjnl/bxab216

CITATIONS
0

READS
35

6 authors, including:

Binbin Cai
Beijing University of Posts and Telecommunications
8 PUBLICATIONS **57** CITATIONS

SEE PROFILE

Wu Yusen
University of Western Australia
14 PUBLICATIONS **5** CITATIONS

SEE PROFILE

Su-Juan Qin
Beijing University of Posts and Telecommunications
135 PUBLICATIONS **3,008** CITATIONS

SEE PROFILE

Fei Gao
Ecole des hautes études en santé publique
244 PUBLICATIONS **6,336** CITATIONS

SEE PROFILE

https://doi.org/10.1093/comjnl/bxab216

Quantum Attacks on 1K-AES and PRINCE

Bin-Bin Cai[1,2], Yusen Wu[3], Jing Dong[1], Su-Juan Qin[1], Fei Gao[1,*] and Qiao-Yan Wen[1]

[1]*State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China*
[2]*State Key Laboratory of Cryptology, P.O. Box 5159, Beijing, 100878, China*
[3]*Department of Physics, The University of Western Australia, Perth, WA 6009, Australia*
**Corresponding author: gaof@bupt.edu.cn*

By introducing the BHT algorithm into the slide attack on 1K-AES and the related-key attack on PRINCE, we present the corresponding quantum attacks in this paper. In the proposed quantum attacks, we generalize the BHT algorithm to the situation where the number of marked items is unknown ahead of time. Moreover, we give an implementation scheme of classifier oracle based on Quantum Phase Estimation algorithm in presented quantum attacks. The complexity analysis shows that the query complexity, time complexity and memory complexity of the presented quantum attacks are all $\mathcal{O}(2^{n/3})$ when the success probability is about 63%, where n is the block size. Compared with the corresponding classical attacks, the proposed quantum attacks can achieve subquadratic speed-up under the same success probability no matter on query complexity, time complexity or memory complexity. Furthermore, the query complexity of the proposed quantum slide attack on 1K-AES is less than Grover search on 1K-AES by a factor of $2^{n/6}$. When compared with the Grover search on PRINCE, the query complexity of the presented quantum attack on PRINCE is reduced from $\mathcal{O}(2^n)$ to $\mathcal{O}(2^{n/2})$. When compared with the combination of Grover and Simon's algorithms on PRINCE, the query complexity of our quantum attack on PRINCE is reduced from $\mathcal{O}(n \cdot 2^{n/2})$ to $\mathcal{O}(2^{n/2})$. Besides, the proposed quantum slide attack on 1K-AES indicates that the quantum slide attack could also be applied on Substitution-Permutation Network construction, apart from the iterated Even-Mansour cipher and Feistel constructions.

Keywords: 1K-AES; PRINCE; BHT algorithm; Grover algorithm

Received 9 October 2021; Revised 29 November 2021; Editorial Decision 22 December 2021
Handling editor: Xinyi Huang

1. INTRODUCTION

Quantum computing, which is initiated in 1980s, employs properties of quantum states such as superposition and entanglement to perform computing. Different from classical computing that the state of a single bit is zero or one, the basic unit operated in quantum computing is a qubit being any superposition of two states zero and one [1]. The most exciting fact of quantum computing is that it has been shown to outperform the classical computing in solving certain problems [2], such as simulating quantum systems [3], factoring large numbers [4], unsorted database search [5, 6], solving linear systems of equations [7, 8] and machine learning [9–14]. These achievements stimulated scientists to search for quantum algorithms that could solve problems in other areas outperforming their classical counterparts. Among the enormous applications, cryptanalysis is one of the suitable scenarios and gives rise to the field of quantum cryptanalysis [15]. Evidently, the research of quantum cryptanalysis is significant in both theory and applications, since it stimulates the development of post-quantum cryptography.

In 1994, Shor [4] proposed a quantum polynomial-time algorithm to solve the factorization and discrete logarithm problems, which posed a fatal threat to the security of public-key cryptography such as RSA and Elliptic Curve Cryptography. In 1996, Grover [5] presented a quantum algorithm to search unsorted database, which can provide a quadratic speed-up for brute force attacks in symmetric primitives.

Moreover, Simon algorithm [16] could also affect the security of symmetric cryptography. In 2010, Kuwakado and Morri [17] presented a quantum distinguisher for 3-round Feistel cipher by utilizing Simon algorithm. It can distinguish between the cipher and a random permutation with $\mathcal{O}(n)$ queries, where n is the block size. Furthermore, Kuwakado and Morri [18] also proposed a quantum key-recovery attack on Even-Mansour cipher based on Simon algorithm. Compared with the classical key-recovery attack, this quantum attack achieves exponential speed-up. At CRYPTO 2016, Kaplan *et al.* [19] showed that LRW construction, a few modes of operation (such as CBC-MAC, PMAC and GMAC) and several CAESAR candidates can be broken by Simon algorithm. In addition, quantum slide attacks based on Simon algorithm have also been found to threaten the security of iterated Even-Mansour cipher [19] and some Feistel ciphers [20].

At Asiacrypt 2017, Leander *et al.* [21] combined Grover and Simon's algorithms to break the FX-based block ciphers. Based on this combination scheme, several quantum key-recovery attacks on (generalized) Feistel ciphers [20, 22–25] were proposed.

Apart from the above algorithms, researchers also studied quantum cryptanalysis with the help of other quantum algorithms. In 2018, Bonnetain and Naya-Plasencia [26] generalized Kuperberg's algorithm [27] for the hidden shift problem and built quantum attacks on FX variants and Poly1305. In 2020, Hao *et al.* [28] proposed a new quantum period finding algorithm based on the Bernstein-Vazirani algorithm [29].

Our contributions. In this paper, we study the quantum slide attack on 1K-AES and quantum related-key attack on PRINCE. The main contributions of this paper include the following two aspects.

1. We propose the quantum slide attack on 1K-Advanced Encryption Standard (1K-AES) by introducing the BHT algorithm [30]. It implies that the quantum slide attack could also be applied on substitution-permutation network (SPN) construction, apart from the iterated Even-Mansour cipher and Feistel constructions [19, 20]. In the proposed quantum attack, we generalize BHT algorithm to the situation where the number of marked items is not known in advance. Moreover, we give an implementation scheme of classifier oracle based on Quantum Phase Estimation algorithm [6]. The complexity analysis shows that the query complexity, time complexity and memory complexity of the presented quantum attack are all $\mathcal{O}(2^{n/3})$ when the success probability is around 63%. Compared with the classical slide attack on 1K-AES [31], the presented quantum attack can achieve subquadratic speed-up under the same success probability no matter on query complexity, time complexity or memory complexity. Moreover, the proposed quantum slide attack on 1K-AES reduces the query complexity by a factor of $2^{n/6}$ as compared with Grover search on 1K-AES.

2. We found that the generalized BHT algorithm could also be introduced into the related-key attack on PRINCE. Hence, we propose the quantum related-key attack on PRINCE. For the proposed quantum related-key attack on PRINCE that can recover the first subkey, the query complexity, time complexity and memory complexity are $\mathcal{O}(2^{n/3})$ when the success probability is about 63%. After retrieving the first subkey, the other subkey can be recovered by Grover search. Therefore, the query complexity increases to $\mathcal{O}(2^{n/2})$ when we consider the whole quantum attack on PRINCE. When compared with the Grover search on PRINCE, the query complexity of the whole quantum attack is reduced from $\mathcal{O}(2^n)$ to $\mathcal{O}(2^{n/2})$. When compared with the combination of Grover and Simon's algorithms [21] on PRINCE, the query complexity of our attack is reduced from $\mathcal{O}(n \cdot 2^{n/2})$ to $\mathcal{O}(2^{n/2})$.

Organization. The rest of paper is organized as follows. In the next section, some essential algorithms and attacks are introduced. In Section 3, the quantum slide attack on 1K-AES and quantum related-key attack on PRINCE are proposed. The complexity analysis of presented quantum attacks is provided in Section 4. Finally, a short conclusion is given in Section 5.

2. PRELIMINARIES

In this section, some preliminary algorithms and attacks are given.

2.1. Grover algorithm

Assume there is only one marked item in the N-scale datum, where $N = 2^n$. The Grover algorithm can find the marked item in $\mathcal{O}(\sqrt{N})$ query complexity by executing the following steps.

1. Prepare a n-qubit register $|0^{\otimes n}\rangle$.
2. Perform n Hadamard gates on the register and the system becomes to
$$|\psi\rangle = H^{\otimes n}|0^{\otimes n}\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle. \tag{1}$$
3. Construct the quantum oracle $O : |x\rangle \xrightarrow{O} (-1)^{f(x)}|x\rangle$, where $f(x) = 1$ if x is the marked item, otherwise $f(x) = 0$.
4. Apply Grover iteration for R ($R \approx \frac{\pi}{4}\sqrt{2^n}$) times then we can get
$$[(2|\psi\rangle\langle\psi| - I)O]^R|\psi\rangle \approx |x'\rangle, \tag{2}$$
where x' is the marked item.
5. Measure the register and obtain x'.

In this way, Grover algorithm can get the marked item with probability close to 1 by applying Grover iteration about $\frac{\pi}{4}\sqrt{2^n}$ times. In other words, the query complexity of Grover algorithm is $\mathcal{O}(\sqrt{2^n})$. Furthermore, the generalization of Grover algorithm (i.e. quantum amplitude amplification, QAA) is given in the following theorem.

THEOREM 2.1 (Brassard, Høyer, Mosca and Tapp [6]). *Let $\mathcal{A}$ be any quantum algorithm applied on q qubits without measurement. Let $\mathcal{B}: \mathbb{F}_2^q \to \{0,1\}$ be a function that classifies the outcomes of $\mathcal{A}$ as good or bad. Let $p > 0$ be the initial success probability that a measurement of $\mathcal{A}|0\rangle$ is good. Set $k = \lfloor \frac{\pi}{4\theta} \rfloor$, where θ is defined as $\sin^2(\theta) = p$. Besides, define the unitary operator $Q = -\mathcal{A}S_0\mathcal{A}^{-1}S_{\mathcal{B}}$, where $S_{\mathcal{B}}$ changes the sign of the good state*

$$|x\rangle = \begin{cases} -|x\rangle, & \text{if } \mathcal{B}(x) = 1 \\ |x\rangle, & \text{if } \mathcal{B}(x) = 0 \end{cases}, \tag{3}$$

whereas S_0 changes the sign of zero state $|0\rangle$ only. Finally, a measurement will yield the good state with probability at least $\max\{1-p, p\}$ after the operation of $Q^k\mathcal{A}|0\rangle$.

Besides, Boyer *et al.* [32] generalized Grover algorithm to the situation where the number of marked items s is unknown in 1998. In this case, the generalized Grover algorithm can find the marked item in time of $\mathcal{O}(\sqrt{N/s})$ as follows. For simplicity, we assume that $1 \le s \le 3N/4$.

1. Initialize $l = 1$ and $\lambda = 8/7$.
2. Select an integer $0 \le i < l$ uniformly at random.
3. Apply Grover iteration for i times on the state $|\psi\rangle$.
4. Measure the register and obtain the outcome j.
5. If j is the marked item, exit. Otherwise, set l to $\min(\lambda l, \sqrt{N})$ and go back to step 2.

2.2. BHT algorithm

In 1997, Brassard *et al.* [30] proposed a quantum algorithm to solve the collision problem. Moreover, it is natural to apply the algorithm to find a claw in pairs of bijection functions. Assume that two functions $F: X \to Z$ and $G: Y \to Z$ with the same codomain, the claw is defined as a pair $x \in X$, $y \in Y$ such that $F(x) = G(y)$. The claw finding problem is to find a claw in F and G under the promise that the claw is existed. Suppose $N = |X| = |Y| = |Z|$, the detailed process of the algorithm **Claw**$(\mathbf{F}, \mathbf{G}, \mathbf{k})$ is as follows.

Claw (F, G, k)

1. Pick an arbitrary subset $K \subseteq X$, where $|K| = k$. Create a table L with size k where each item in L holds a distinct pair $(x, F(x))$ with $x \in K$.
2. Sort L on the basis of the second entry of each item in L.
3. Compute $y_0 = Grover(H, 1)$ where $H: Y \to \{0,1\}$ denotes the function defined by $H(y) = 1$ if and only if a pair $(x_0, G(y))$ appears in L for some arbitrary $x_0 \in K$.
4. Find $(x_0, G(y_0)) \in L$ and output the claw $\{x_0, y_0\}$.

THEOREM 2.2. (Brassard, Høyer and Tapp [30]). *Given two one-to-one functions $F: X \to Z$ and $G: Y \to Z$, where $|X| = |Y| = |Z| = N$. After k evaluations on F and $\mathcal{O}(\sqrt{N/k})$ evaluations on G, the algorithm* **Claw**$(\mathbf{F}, \mathbf{G}, \mathbf{k})$ *returns a claw, where $1 \le k \le N$. The memory complexity is $\Theta(k)$. And most particularly,* **Claw**$(\mathbf{F}, \mathbf{G}, \mathbf{k})$ *evaluates F and G with $\mathcal{O}(\sqrt[3]{N})$ times and uses space $\Theta(\sqrt[3]{N})$ when $k = \sqrt[3]{N}$.*

2.3. Slide attack on 1K-AES

In block cipher, SPN construction is a series of consecutive mathematical operations. Take a block of plaintext and key as inputs, the ciphertext is generated by performing several alternate layers of substitution boxes and permutation boxes.

AES [33] is one of the representative SPN construction ciphers, and it is an iterated cipher that contains 10/12/14 almost same rounds and the corresponding key size are 128/192/256 bits. Each round composed of four operations: SubBytes (SB), ShiftRows (SR), MixColumns (MC) and AddRoundKey (ARK). In the realistic application of AES cryptography, an additional ARK operation is performed before the first round and the MC operation is omitted in last round. Here, we define 1K-AES as a similar structure with a whitening key before the first round while same round key in all rounds. Besides, the MC operation in the last round of 1K-AES is not omitted. Hence, the encryption of 1K-AES can be described as

$$E_k(P) = F_k \circ F_k \circ \cdots \circ F_k(P \oplus k), \tag{4}$$

where $F_k(x) = ARK_k \circ MC \circ SR \circ SB$ denotes a round function of 1K-AES.

Here, the classical slide attack on 1K-AES [31] is provided. Assume that $P' \oplus k = F_k(P \oplus k)$, i.e. (P, P') is a slide pair of E_k. Therefore, we can obtain $P = k \oplus (SB^{-1} \circ SR^{-1} \circ MC^{-1}(P'))$. Denoting $\bar{P}' = SB^{-1} \circ SR^{-1} \circ MC^{-1}(P')$, then we have $P \oplus \bar{P}' = k$. On the other hand, the corresponding ciphertext must satisfy $C' = k \oplus (MC \circ SR \circ SB(C))$ based on the structure of 1K-AES. Hence, denoting $\bar{C} = MC \circ SR \circ SB(C)$ then we get $C' \oplus \bar{C} = k$. According to these relations, we can obtain $P \oplus \bar{C} = \bar{P}' \oplus C'$. The description of the slide attack on 1K-AES is as follows.

2.4. Related key attack on full PRINCE with the α-reflection property

PRINCE [34] is a low-latency block cipher with a 64-bit block size and a 128-bit secret key k. As presented in Fig. 1, the PRINCE cipher follows the FX construction. Obviously, we can obtain $E_{(k_0||k_1||k_0')}(P) = E_{k_1}(P \oplus k_0) \oplus k_0'$, where $(k_0||k_1) = k$ and $k_0' = (k_0 \ggg 1) \oplus (k_0 \gg 63)$.

In 2013, Jean *et al.* [35] revealed a related-key attack on full PRINCE with the α-reflection property. They introduced a related key $k' = (k_0, k_1 \oplus \alpha)$, where $\alpha =$ 0xc0ac29b7c97c50dd is a constant about the construction of PRINCE$_{core}$. Based on

Algorithm 1 The slide attack on 1K-AES [31]

Ask for the encryption of $2^{n/2}$ known plaintexts (P_i, C_i), here n is the block size.

1: **for** each plaintext/ciphertext pair (P_i, C_i) **do**
2: Compute the value $\bar{C}_i = MC \circ SR \circ SB(C_i)$;
3: Compute the value $P_i \oplus \bar{C}_i$;
4: Store $(P_i \oplus \bar{C}_i, P_i)$ in a hash table according to the first entry;
5: **end for**
6: **for** each plaintext/ciphertext pair (P_j, C_j) **do**
7: Compute the value $\bar{P}_j = SB^{-1} \circ SR^{-1} \circ MC^{-1}(P_j)$;
8: Compute the value $\bar{P}_j \oplus C_j$;
9: **if** $(P_i \oplus \bar{C}_i = \bar{P}_j \oplus C_j)$ **then**
10: test the key candidate $k = P_i \oplus \bar{P}_j$ by trial encryption;
11: **end if**
12: **end for**

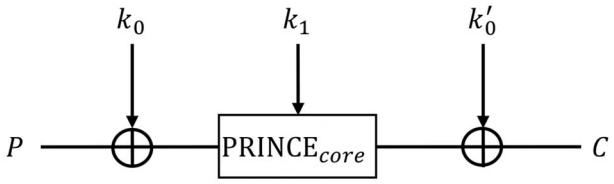

FIGURE 1. The PRINCE cipher.

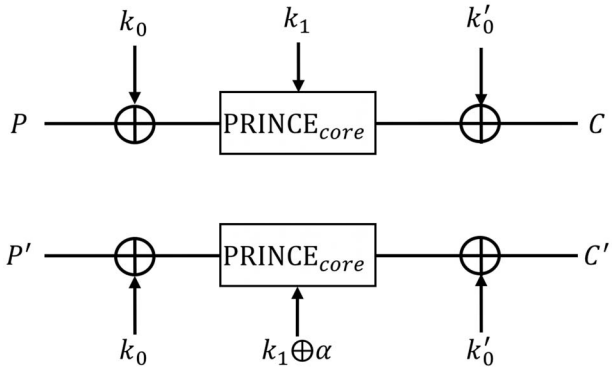

FIGURE 2. Related-key distinguisher on full PRINCE.

the α-reflection property, $D_{(k_0||k_1||k_0')}(\cdot) = E_{(k_0'||k_1\oplus\alpha||k_0)}(\cdot)$ is satisfied. Hence, the following property [35] can be obtained.

Property. As presented in Fig. 2, (P, C) is a plaintext/ciphertext from PRINCE with secret key k, and (P', C') is a plaintext/ciphertext encrypted under related key k' by PRINCE. If $C \oplus P' = k_0 \oplus k_0'$, then $P \oplus C' = k_0 \oplus k_0'$ with probability 1.

Now, the description of the related-key attack on PRINCE is as follows. Note that the test condition $Z_1 = Z_2$ after the collision was found in this algorithm is used to check the related-key distinguisher. After retrieving k_0, k_1 can be recovered by exhaustive search.

Algorithm 2 The related-key attack on PRINCE [35]

Ask for the encryption of $2^{n/2}$ known plaintexts (P_i, C_i) with the real key $k = (k_0, k_1)$, where n is the block size.
Ask for the encryption of $2^{n/2}$ known plaintexts (P_i', C_i') with the related key $k' = (k_0, k_1 \oplus \alpha)$.

1: **for** each plaintext/ciphertext pair (P_i, C_i) **do**
2: Store $(P_i, C_i, P_i \oplus C_i)$ in a hash table according to the last entry;
3: **end for**
4: **for** each plaintext/ciphertext pair (P_i', C_i') **do**
5: Store $(P_i', C_i', P_i' \oplus C_i')$ in a hash table according to the last entry;
6: **if** $(P_i \oplus C_i = P_j' \oplus C_j')$ **then**
7: Compute the value $Z_1 = C_i \oplus P_j'$;
8: Compute the value of $C' = P \oplus Z_1$ with a random plaintext P;
9: Compute the value of $Z_2 = P' \oplus C$, where C is the corresponding ciphertext of P with real key, P' decrypted from C' with related key;
10: **if** $(Z_1 = Z_2)$ **then**
11: Obtain $Z_1 = Z_2 = k_0 \oplus k_0'$ and retrieve k_0 by inverting the bijection;
12: **end if**
13: **end if**
14: **end for**

3. QUANTUM ATTACKS

3.1. Quantum slide attack on 1K-AES

Here, we put forward the quantum slide attack on 1K-AES. The detailed attack can be achieved by executing the following steps. The values of complexity coefficients t_1 and t_2 given here will be discussed in the next section.

1. Pick 2^{t_1} plaintexts P_i randomly, where $i = 0, 1, \ldots, 2^{t_1} - 1$. Ask for encryption of P_i and obtain corresponding ciphertexts C_i.
2. For each plaintext/ciphertext pair (P_i, C_i), compute the value $\widetilde{P}_i = SB^{-1} \circ SR^{-1} \circ MC^{-1}(P_i)$. Then, compute the value $\widetilde{P}_i \oplus C_i$.
3. Construct a table L of size 2^{t_1}. Store 2^{t_1} pairs $(\widetilde{P}_i, \widetilde{P}_i \oplus C_i)$ in L and sort L according to the second entry.
4. Compute $Q_j = \mathbf{QAA}(H, 1)$ where $H : T \to \{0, 1\}$ denotes the function defined by $H(Q) = 1$ if and only if $Q \oplus (MC \circ SR \circ SB(D)) = \widetilde{P}_i \oplus C_i$, here $T = \{t | t \in \{0, 1\}^n\}$ and $|T| = 2^{t_2}$, $D = E_k(Q)$. In this case, the **QAA** algorithm outputs right Q_j with probability P_Q.

5. Find $(\widetilde{P}_i, Q_j \oplus (MC \circ SR \circ SB(D_j))) \in L$ and test the key candidate $k = Q_j \oplus \widetilde{P}_i$ by trial encryption.

It is worth noting that the **QAA** [6] algorithm is adopted in step 4 rather than Grover algorithm [5] since the quantum state searched at this time is not a uniform superposition state with 2^n elements. Besides, the **QAA** [6] algorithm here should be generalized to the situation where the number of marked items is unknown ahead of time. Concretely, it is similar to execute the algorithm like Ref. [32] except that step 3 should be applying QAA unitary operator for i times on the state $|\psi'\rangle = \frac{1}{\sqrt{2^{t_2}}} \sum_{j=0}^{2^{t_2}-1} |Q_j\rangle$.

3.2. Quantum related-key attack on PRINCE

The detailed quantum related-key attack on PRINCE is composed of following steps.

1. Pick 2^{t_1} plaintexts P_i randomly, where $i = 0, 1, \ldots, 2^{t_1} - 1$. Ask for encryption of P_i with secret key and obtain corresponding ciphertexts C_i.
2. For each plaintext/ciphertext pair (P_i, C_i), compute the value $P_i \oplus C_i$.
3. Store 2^{t_1} pairs $(P_i, C_i, P_i \oplus C_i)$ in a table L and sort L based on the last entry.
4. Compute $Q_j = \mathbf{QAA}(H, 1)$ where $H : T \to \{0, 1\}$ denotes the function defined by $H(Q) = 1$ if and only if $Q \oplus D = P_i \oplus C_i$, here D is the corresponding ciphertext of Q with related key k'. The **QAA** outputs right Q_j with probability $P_{Q'}$.
5. Find $(P_i, C_i, Q_j \oplus D_j) \in L$ and test the key candidate k_0 that is retrieved by inverting the bijection of $Q_j \oplus C_i$.

After retrieving k_0, the value of k_0' can be obtained immediately. Finally, k_1 can be recovered by Grover search.

3.3. The implementation scheme of oracle H

Next, the implementation scheme of oracle H is provided. In proposed quantum attacks, the oracle $H : T \to \{0, 1\}$ checks whether the condition $Q_j \oplus (MC \circ SR \circ SB(D_j)) = \widetilde{P}_i \oplus C_i$ (*resp.* $Q_j \oplus D_j = P_i \oplus C_i$) holds, where $j = 0, 1, \ldots, 2^{t_2} - 1$. If the condition is satisfied, the oracle $H(Q_j) = 1$, else $H(Q_j) = 0$. Here, we regard $Q_j \oplus (MC \circ SR \circ SB(D_j))$ (*resp.* $Q_j \oplus D_j$) and $\widetilde{P}_i \oplus C_i$ (*resp.* $P_i \oplus C_i$) as function F_1 and F_2 respectively. The implementation scheme of the oracle H can be achieved with the following theorem.

THEOREM 3.1 (Zhou, Loke, Izaac and Wang [36]). *Let U_a and U_b be two unitary operators such that $U_a|0^{\otimes n}\rangle = |\phi_a\rangle$ and $U_b|0^{\otimes n}\rangle = |\phi_b\rangle$, define the quantum state*

$$|\phi\rangle = \frac{1}{2}(|0\rangle_1(|\phi_a\rangle_2|\phi_b\rangle_3 + |\phi_b\rangle_2|\phi_a\rangle_3) + |1\rangle_1(|\phi_a\rangle_2|\phi_b\rangle_3 - |\phi_b\rangle_2|\phi_a\rangle_3)) \quad (5)$$

and the unitary operator $G = -AS_0A^\dagger S_x$, where $A : |0\rangle_{123} \xrightarrow{A} |\phi\rangle$, $S_0 = I - 2|0\rangle_{123}\langle 0|_{123}$, $S_x = I - 2|0\rangle_1\langle 0|_1$. The value $z(\phi_a, \phi_b) = |\langle\phi_a|\phi_b\rangle|^2$ can be estimated by applying Quantum Phase Estimation algorithm with the unitary operator G on the state $|\phi\rangle$ in $\mathcal{O}(m)$ query complexity of U_a and U_b, where $m = \mathcal{O}(1/\epsilon)$ and $\epsilon = 2^{-p_0}$ is the additive error in Quantum Phase Estimation algorithm with p_0 ancillary qubits.

Proof. First, the preparation of quantum state $|\phi\rangle$ is provided.

1. Prepare the quantum state $|0\rangle_1|0^{\otimes n}\rangle_2|0^{\otimes n}\rangle_3$.
2. Apply the operator U_a and U_b to registers R_2 and R_3 respectively, then the state $|0\rangle_1|\phi_a\rangle_2|\phi_b\rangle_3$ is obtained.
3. Perform the swap test in R_2 and R_3, we get $|\phi\rangle = \frac{1}{2}(|0\rangle_1(|\phi_a\rangle_2|\phi_b\rangle_3 + |\phi_b\rangle_2|\phi_a\rangle_3) + |1\rangle_1(|\phi_a\rangle_2|\phi_b\rangle_3 - |\phi_b\rangle_2|\phi_a\rangle_3))$.

It is easy to find that the quantum state $|\phi\rangle$ can be rewritten as

$$|\phi\rangle = \sin\theta|\phi^0\rangle + \cos\theta|\phi^1\rangle, \quad (6)$$

where $|\phi^0\rangle$ corresponds to the part of $|\phi\rangle$ whose first qubit is $|0\rangle$, $|\phi^1\rangle$ corresponds to the part of $|\phi\rangle$ whose first qubit is $|1\rangle$. It can be easily calculated that $\sin^2\theta = \frac{1}{2} + \frac{1}{2}|\langle\phi_a|\phi_b\rangle|^2$. Hence, we can get $z(\phi_a, \phi_b) = 2\sin^2\theta - 1$. Next, we show how to estimate the value of $z(\phi_a, \phi_b)$ by using the Quantum Phase Estimation algorithm [6]. To do that, we first add ancillary qubits $|0^{\otimes p_0}\rangle$ to the state $|\phi\rangle$. Then we perform $H^{\otimes p_0}$ on the ancillary qubits and the system becomes to $|+\rangle^{\otimes p_0}|\phi\rangle$, where $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. According to the construction of the unitary $G = -AS_0A^\dagger S_x$, it can be expressed as

$$G = e^{\mathbf{i}2\theta}|\phi^+\rangle\langle\phi^+| + e^{-\mathbf{i}2\theta}|\phi^-\rangle\langle\phi^-|, \quad (7)$$

where the phase θ encodes the inner-product information, $e^{\pm\mathbf{i}2\theta}$ and $|\phi^\pm\rangle$ are its eigenvalues and eigenstates (un-normalized), $\mathbf{i} = \sqrt{-1}$. Now, we can obtain

$$|+\rangle^{\otimes p_0}|\phi\rangle \to |\frac{\theta}{\pi}\rangle|\phi^+\rangle + |1 - \frac{\theta}{\pi}\rangle|\phi^-\rangle \quad (8)$$

by running Quantum Phase Estimation algorithm of A on quantum state $|+\rangle^{\otimes p_0}|\phi\rangle$. Since $z(\phi_a, \phi_b) = 2\sin^2\theta - 1$, then we can get $|z(\phi_a, \phi_b))|\phi\rangle$ by performing Sine-gate operation to ancillary register, where the function of Sine-gate operation is to achieve $2\sin^2(\pi \cdot \theta') - 1$ and θ' is the input of the Sine-gate operation. The detailed quantum circuit is illustrated in Fig. 3. ■

Based on **Theorem 3.1**, we can obtain the overlap (i.e. $\langle\phi_a|\phi_b\rangle$) of the functions F_1 and F_2. In our scenario, parameters $p_0 = 1$ and $m = \mathcal{O}(1)$, the oracle H can be constructed as follows.

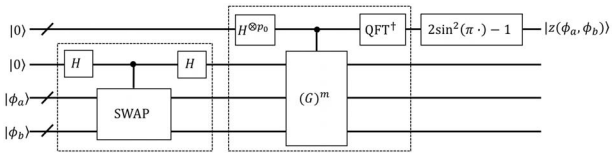

FIGURE 3. The quantum circuit of the overlap calculation $z(\phi_a, \phi_b) = |\langle\phi_a|\phi_b\rangle|^2$ of two quantum states $|\phi_a\rangle$ and $|\phi_b\rangle$. The first dashed box implies the preparation of the quantum state $|\phi\rangle$. Here $\text{SWAP}|\phi_a\rangle|\phi_b\rangle = |\phi_b\rangle|\phi_a\rangle$, which can be realized by basic quantum gates [1]. The second dashed box suggests the Phase Estimation algorithm, where $\text{QFT}^\dagger$ represents the inverse Fourier transformation. The overlap of two quantum states can be estimated in the first register. Here, p_0 ancillary qubits can provide an estimation with $\epsilon = 2^{-p_0}$ additive error, and the iteration time $m = \mathcal{O}(1/\epsilon)$.

1. Prepare the quantum state $|0^{\otimes n}\rangle_1|0^{\otimes n}\rangle_2|0^{\otimes n}\rangle_3|0^{\otimes n}\rangle_4|0\rangle_5 |-\rangle_6$, here $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.
2. Perform t_2 Hadamard transformations on R_1:
$$\frac{1}{\sqrt{2^{t_2}}}\sum_{j=0}^{2^{t_2}-1}|Q_j\rangle_1|0^{\otimes n}\rangle_2|0^{\otimes n}\rangle_3|0^{\otimes n}\rangle_4|0\rangle_5|-\rangle_6. \quad (9)$$
3. Apply the function operator U_{F_1} on R_1 and R_2, we can obtain
$$\frac{1}{\sqrt{2^{t_2}}}\sum_{j=0}^{2^{t_2}-1}|Q_j\rangle_1|F_1(Q_j)\rangle_2|0^{\otimes n}\rangle_3|0^{\otimes n}\rangle_4|0\rangle_5|-\rangle_6. \quad (10)$$
4. Perform t_1 Hadamard transformations on R_3 and access to the QRAM [37], the system state now is
$$\frac{1}{\sqrt{2^{t_2}}}\sum_{j=0}^{2^{t_2}-1}|Q_j\rangle_1|F_1(Q_j)\rangle_2\frac{1}{\sqrt{2^{t_1}}}\sum_{i=0}^{2^{t_1}-1}|P_i\rangle_3|F_2(P_i)\rangle_4|0\rangle_5|-\rangle_6. \quad (11)$$
5. Perform the operations in **Theorem 3.1** and append another swap test in the end, we can obtain
$$\frac{1}{\sqrt{2^{t_2}}}\sum_{j=0}^{2^{t_2}-1}|Q_j\rangle_1|F_1(Q_j)\rangle_2 \frac{1}{\sqrt{2^{t_1}}}\sum_{i=0}^{2^{t_1}-1}|P_i\rangle_3|F_2(P_i)\rangle_4|z(F_1(Q_j),F_2(P_i))\rangle_5|-\rangle_6, \quad (12)$$
where the notation $z(F_1(Q_j), F_2(P_i)) = 1$ if and only if $F_1(Q_j) = F_2(P_i)$.
6. Perform C-NOT operation on R_5 and R_6, where R_5 is the control qubit and R_6 is the target qubit. In this way, one can efficiently mark the target Q_j by flipping its phase.
7. Uncompute the ancillas using the inverse algorithm of steps 3-5, the state
$$\frac{1}{\sqrt{2^{t_2}}}\sum_{j=0}^{2^{t_2}-1}(-1)^{H(Q_j)}|Q_j\rangle_1 \quad (13)$$
is obtained.

In this way, the oracle H can be efficiently implemented on a quantum computer, and the corresponding query complexity is $\mathcal{O}(m)$.

4. COMPLEXITY ANALYSIS

In this section, the complexity analysis of proposed quantum slide attack on 1K-AES is given. Since the complexity analysis of quantum related-key attack on PRINCE is similar, so we focus our analysis on the quantum slide attack on 1K-AES (the complexity analysis of quantum related-key attack on PRINCE is supplied in the Appendix).

In the classical slide attack on 1K-AES [31], the data set contains $2^{n/2}\cdot(2^{n/2}-1) \approx 2^n$ pairs since the plaintext/ciphertext pairs (P_j, C_j) and (P_i, C_i) belong to the same set. Therefore, the probability that the data set contains at least one slide pair is $1-(1-2^{-n})^{2^n} \approx 1-1/e \approx 0.63$. On the other hand, the probability that a random pair (P_i, Q_j) satisfies the condition $P_i \oplus \tilde{C}_i = \tilde{Q}_j \oplus D_j$ is 2^{-n}. Thus, each slide pair indicates a collision in **Algorithm 1** which suggests the right key candidate. Hence, the right key can be found by checking all output collisions in **Algorithm 1**. In this way, the query complexity is $\mathcal{O}(2^{n/2})$ and the success probability is about 63%. In addition, its time and memory complexities are about $\mathcal{O}(2^{n/2})$ operations.

For the presented quantum slide attack on 1K-AES, we can easily obtain the quantum query complexity is $\mathcal{O}(2^{t_1}+\sqrt{2^{t_2}/s})$, where s is the number of the output collisions. Thus, the data set contains $2^{(t_1+t_2)}$ pairs, the probability that the data set contains at least one slide pair is
$$\begin{aligned} P_Q &= 1-(1-2^{-n})^{2^{t_1+t_2}} \\ &= 1-((1-2^{-n})^{2^n})^{\frac{t_1+t_2}{n}} \\ &\approx 1-(e^{-1})^{\frac{t_1+t_2}{n}} \\ &= 1-e^{-\frac{t_1+t_2}{n}}. \end{aligned} \quad (14)$$

Without loss of generality, we can assume $s = 1$. Therefore, the quantum query complexity is $\mathcal{O}(2^{t_1}+\sqrt{2^{t_2}})$. Most particularly, the optimal quantum query complexity is $\mathcal{O}(2^{t_1})$ when $t_2 = 2t_1$. In this case, the success probability of the quantum slide attack is $P_Q = 1-e^{-\frac{3t_1}{n}}$. Hence, it can be easily calculated that if $t_1 = \frac{n}{3}$, i.e. the quantum query complexity is $\mathcal{O}(2^{n/3})$, the success probability is about 63%. Under this success probability, we

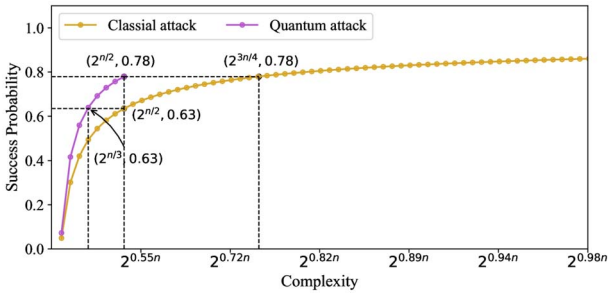

FIGURE 4. The comparison of the relationship between the complexity and success probability of classical slide attack and quantum slide attack on 1K-AES.

can obtain the memory complexity is $\Theta(2^{n/3})$ since the size of table L is $2^{n/3}$. For the cost of queries to the oracles and computations in this attack, we assume that it takes 1 unit of operation to query the oracle and compute values. Therefore, the time complexity of the proposed attack is $2^{n/3} + 2^{n/3} + \log 2^{n/3} \cdot 2^{n/3} + \sqrt{2^{2n/3}} \cdot (1 + \log 2^{n/3}) = \widetilde{\mathcal{O}}(2^{n/3})$ operations, where $\widetilde{\mathcal{O}}$ means ignoring logarithmic factors.

Now, we generalize the success probability to the general case and compare the complexities of these two attacks. For the classical slide attack on 1K-AES, the success probability is $P_C = 1 - (1 - 2^{-n})^{2^{2r_1}} \approx 1 - e^{-\frac{2r_1}{n}}$ when its complexity is $\mathcal{O}(2^{r_1})$, where $r_1 \in (0, n]$. On the other hand, the complexity of the quantum slide attack on 1K-AES is $\mathcal{O}(2^{r_2})$ with the success probability $P_Q = 1-(1-2^{-n})^{2^{3r_2}} \approx 1-e^{-\frac{3r_2}{n}}$, here $r_2 \in (0, \frac{n}{2}]$. At this point, with the same success probability $P_C = P_Q$, i.e. $2r_1 = 3r_2$, we can obtain the complexities of classical slide attack and quantum slide attack on 1K-AES are $\mathcal{O}(2^{r_1})$ and $\mathcal{O}(2^{\frac{2r_1}{3}})$ respectively (see Fig. 4). Thus, the quantum slide attack on 1K-AES can achieve subquadratic speed-up compared with the classical attack no matter on the query complexity, time complexity or memory complexity under the same success probability. Of course, the quantum slide attack equips preferable success probability if the complexities of these two attacks are the same.

5. CONCLUSIONS

In this study, we consider the quantum slide attack on 1K-AES and quantum related-key attack on PRINCE. The quantum slide attack on 1K-AES indicates that the quantum slide attack could also be applied on SPN construction. Our attacks require $\mathcal{O}(2^{n/3})$ quantum queries when the success probability is about 63%. Furthermore, the query complexity of the proposed quantum slide attack on 1K-AES is less than Grover search on 1K-AES by a factor of $2^{n/6}$. When compared with the Grover search on PRINCE, the query complexity of the presented quantum attack on PRINCE is reduced from $\mathcal{O}(2^n)$ to $\mathcal{O}(2^{n/2})$. When compared with the combination of Grover and Simon's algorithms on PRINCE, the query complexity of our attack on PRINCE is reduced from $\mathcal{O}(n \cdot 2^{n/2})$ to $\mathcal{O}(2^{n/2})$.

TABLE 1. Comparison of query complexities of several key-recovery attacks on PRINCE.

	Query complexity
The related-key attack [35]	$\mathcal{O}(2^n)$
Grover search [5]	$\mathcal{O}(2^n)$
The combination of Grover and Simon's algorithms [21]	$\mathcal{O}(n \cdot 2^{n/2})$
The presented quantum attack	$\mathcal{O}(2^{n/2})$

DATA AVAILABILITY

No new data were generated or analysed in support of this research.

FUNDING

National Natural Science Foundation of China (Grant Numbers 61972048, 61976024); BUPT Excellent Ph.D. Students Foundation (Grant Number CX2019207); China Scholarship Council (Grant Number 202006470082); Fundamental Research Funds for the Central Universities (Grant Number 2019XD-A01); the 111 Project (Grant Number B21049).

REFERENCES

[1] Nielsen, M. and Chuang, I. (2010) *Quantum computation and quantum information*. Cambridge University Press, Cambridge.
[2] Montanaro, A. (2016) Quantum algorithms: an overview. *NPJ Quantum Inf.*, 2, 15023:1–15023:8.
[3] Low, G.H. and Chuang, I. (2017) Optimal Hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118, 010501-1-010501-5.
[4] Shor, P.W. (1997) Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26, 1484–1509.
[5] Grover, L.K. (1996) *A fast quantum mechanical algorithm for database search*. Proceedings of STOC 1996, Philadelphia PA, 1 July 1996, pp. 212–219. ACM, New York.
[6] Brassard, G., Høyer, P., Mosca, M. and Tapp, A. (2002) Quantum amplitude amplification and estimation. *Contemp. Math.*, 305, 53–74.
[7] Harrow, A.W., Hassidim, A. and Lloyd, S. (2009) Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103, 150502-1 - 150502-4.
[8] Wan, L.C., Yu, C.H., Pan, S.J., Gao, F., Wen, Q.Y. and Qin, S.J. (2018) Asymptotic quantum algorithm for the Toeplitz systems. *Phys. Rev. A*, 97, 062322-1 - 062322-9.

[9] Yu, C.H., Gao, F., Wang, Q.L. and Wen, Q.Y. (2016) Quantum algorithm for association rules mining. *Phys. Rev. A*, 94, 042311-1 - 042311-8.
[10] Yu, C.H., Gao, F., Liu, C.H., Huynh, D., Reynolds, M. and Wang, J.B. (2019) Quantum algorithm for visual tracking. *Phys. Rev. A*, 99, 022301-1 - 022301-10.
[11] Yu, C.H., Gao, F., Lin, S. and Wang, J.B. (2019) Quantum data compression by principal component analysis. *Quantum Inf. Process*, 18, 249:1-249:20.
[12] Yu, C.H., Gao, F. and Wen, Q.Y. (2021) An improved quantum algorithm for ridge regression. *IEEE Trans. Knowl. Data Eng.*, 33, 858–866.
[13] Pan, S.J., Wan, L.C., Liu, H.L., Wang, Q.L., Qin, S.J., Wen, Q.Y. and Gao, F. (2020) Improved quantum algorithm for A-optimal projection. *Phys. Rev. A*, 102, 052402-1 - 052402-11.
[14] Liu, H.L., Wu, Y.S., Wan, L.C., Pan, S.J., Qin, S.J., Gao, F. and Wen, Q.Y. (2021) Variational quantum algorithm for the Poisson equation. *Phys. Rev. A*, 104, 022418-1 - 022418-12.
[15] Jordan, S.P. and Liu, Y.K. (2018) Quantum cryptanalysis: shor, grover, and beyond. *IEEE Secur. Privacy*, 16, 14–21.
[16] Simon, D.R. (1997) On the power of quantum computation. *SIAM J. Comput.*, 26, 1474–1483.
[17] Kuwakado, H. and Morii, M. (2010) *Quantum distinguisher between the 3-round Feistel cipher and the random permutation*, pp. 2682–2685. IEEE International Symposium on Information Theory, IEEE, Austin, TX13-18 June, Piscataway, NJ.
[18] Kuwakado, H. and Morii, M. (2012) Security on the quantum-type Even-Mansour cipher. In *International Symposium on Information Theory and its Applications*. Honolulu, HI, pp. 312–316. IEEE, Piscataway, NJ 28-31 October.
[19] Kaplan, M., Leurent, G., Leverrier, A. and Naya-Plasencia, M. (2016) Breaking symmetric cryptosystems using quantum period finding. In *Advances in Cryptology - CRYPTO 2016*. Santa Barbara, CA, 14-18 August, pp. 207–237. Springer, Berlin, Heidelberg.
[20] Dong, X.Y., Dong, B.Y. and Wang, X.Y. (2020) Quantum attacks on some feistel block ciphers. *Designs Codes Cryptograph.*, 88, 1179–1203.
[21] Leander, G. and May, A. (2017) Grover Meets Simon-Quantumly Attacking the FX-construction. In *Advances in Cryptology - ASIACRYPT 2017*. Hong Kong, China, 3-7 December, pp. 161–178. Springer, Cham.
[22] Dong, X.Y. and Wang, X.Y. (2018) Quantum key-recovery attack on Feistel structures. *SCIENCE CHINA Inf. Sci.*, 61, 102501: 1–102501:7.
[23] Dong, X.Y., Li, Z. and Wang, X.Y. (2019) Quantum cryptanalysis on some generalized Feistel schemes. *SCIENCE CHINA Inf. Sci.*, 62, 22501:1-22501:12.
[24] Ni, B.Y., Ito, G., Dong, X.Y. and Iwata, T. (2019) Quantum Attacks Against Type-1 Generalized Feistel Ciphers and Applications to CAST-256. In *Progress in Cryptology - INDOCRYPT 2019*. Hyderabad, India, 15-18 December, pp. 433–455. Springer, Cham.
[25] Ni, B.Y. and Dong, X.Y. (2020) Improved quantum attack on type-1 generalized Feistel schemes and its application to CAST-256. *J. Electron. Inf. Technol.*, 42, 295–306.
[26] Bonnetain, X. and Naya-Plasencia, M. (2018) Hidden shift quantum cryptanalysis and implications. In *Advances in Cryptology - ASIACRYPT 2018*. Brisbane, QLD, 2-6 December, pp. 560–592. Springer, Cham.
[27] Kuperberg, G. (2005) A subexponential-time quantum algorithm for the dihedral hidden subgroup problem. *SIAM J. Comput.*, 35, 170–188.
[28] Hao, X.X., Zhang, F.R., Wei, Y.Z. and Zhou, Y. (2020) Quantum period finding based on the Bernstein-Vazirani algorithm. *Quantum Inf. Comput.*, 20, 65–84.
[29] Bernstein, E. and Vazirani, U. (1997) Quantum complexity theory. *SIAM J. Comput.*, 26, 1411–1473.
[30] Brassard, G., Høyer, P. and Tapp, A. (1998) Quantum cryptanalysis of hash and claw-free functions. In *Theoretical Informatics*. Brazil, 20-24 April, pp. 163–169. Springer, Berlin, Heidelberg.
[31] Bar-On, A., Biham, E., Dunkelman, O. and Keller, N. (2018) Efficient slide attacks. *J. Cryptol.*, 31, 641–670.
[32] Boyer, M., Brassard, G., Høyer, P. and Tapp, A. (1996) Tight bounds on quantum searching. *Fortschritte Der Physik (Prog. Phy.)*, 46, 493–505.
[33] Daemen, J. and Rijmen, V. (2002) *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer-Verlag, Berlin.
[34] Borghoff, J. *et al.* (2012) PRINCE - A low-latency block cipher for pervasive computing applications. In *Advances in Cryptology - ASIACRYPT 2012*Beijing, China, 2-6 December, pp. 208–225. Springer, Berlin, Heidelberg.
[35] Jean, J., Nikolić, I., Peyri, T., Wang, L. and Wang, S. (2013) Security Analysis of PRINCE. In *Fast Software Encryption 2013*Singapore, 11-13 March, pp. 92–111. Springer, Berlin, Heidelberg.
[36] Zhou, S.S., Loke, T., Izaac, J.A. and Wang, J.B. (2017) Quantum Fourier transform in computational basis. *Quantum Inf. Process*, 16, 82:1-82:19.
[37] Giovannetti, V., Lloyd, S. and Maccone, L. (2008) Quantum random access memory. *Phys. Rev. Lett.*, 100, 160501-1 - 160501-4.

A. APPENDIX

Here, the complexity analysis of quantum related-key attack on PRINCE is given.

In the related-key attack on PRINCE [35], the data set contains $2^{n/2} \cdot 2^{n/2} = 2^n$ pairs. Therefore, the probability that the data set contains at least one slide pair is $1-(1-2^{-n})^{2^n} \approx 1 - 1/e \approx 0.63$. Thus, the query complexity is $\mathcal{O}(2^{n/2})$ and the success probability is about 63%. Moreover, its time and memory complexities are about $\mathcal{O}(2^{n/2})$ operations. In the end, k_1 can be recovered by exhaustive search after retrieving k_0. Hence, the query complexity of the whole attack is $\mathcal{O}(2^n)$.

For the proposed quantum related-key attack on PRINCE, the quantum query complexity is $\mathcal{O}(2^{t_1} + \sqrt{2^{t_2}})$ and the probability $P_{Q'} = 1-e^{-\frac{t_1+t_2}{n}}$, here we assume the number of the output collisions is 1. Hence, the optimal quantum query complexity is $\mathcal{O}(2^{t_1})$ when $t_2 = 2t_1$ and $P_{Q'} = 1 - e^{-\frac{3t_1}{n}}$. Under 63% success probability, the quantum query complexity $\mathcal{O}(2^{n/3})$ can be obtained. Besides, the time complexity and memory complexity are $\widetilde{\mathcal{O}}(2^{n/3})$ and $\Theta(2^{n/3})$ operations, respectively. After retrieving k_0 by quantum related-key attack, the value of $k_0^{'}$ can be obtained immediately. Then, we can recover k_1 by Grover algorithm. Therefore, the query complexity of the whole quantum attack is $\mathcal{O}(2^{n/2})$.

Apart from that, it is easy to know that the query complexity of Grover search on PRINCE is $\mathcal{O}(2^n)$ for recovering k_0 and k_1. Besides, since the PRINCE cipher follows the FX construction and the combination of Grover and Simon's algorithms can be utilized to analyze FX construction directly. Thus, we can obtain the query complexity of the combination of Grover and Simon's algorithms on PRINCE is $\mathcal{O}(n \cdot 2^{n/2})$ from Theorem 2 of Ref. [21]. The comparison of query complexities of several key-recovery attacks on PRINCE is summarized in Table 1.