Tight Security Bounds for Triple Encryption

Jooyoung Lee

Faculty of Mathematics and Statistics
Sejong University, Seoul, Korea 143-747
jlee05@sejong.ac.kr

Abstract. In this paper, we revisit the long-standing open problem asking the exact provable security of triple encryption in the ideal cipher model. For a blockcipher with key length κ and block size n, triple encryption is known to be provably secure up to $2^{\kappa+\frac{1}{2}\min\{\kappa,n\}}$ queries, while the best attack requires $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ query complexity. So there has been a gap between the upper and lower bounds for the security of triple encryption. We close this gap by proving the security up to $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ query complexity. With the DES parameters, triple encryption is secure up to $2^{82.4}$ queries, greater than the current bound of $2^{78.3}$ and comparable to $2^{83.5}$ for 2-XOR-cascade [10].

We also analyze the security of two-key triple encryption, where the first and the third keys are identical. We prove that two-key triple encryption is secure up to $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ blockcipher queries and $2^{\min\{\kappa,\frac{n}{2}\}}$ construction queries. For the DES parameters, this result is interpreted as the security of two-key triple encryption up to 2^{32} plaintext-ciphertext pairs and $2^{81.1}$ blockcipher encryptions.

1 Introduction

A blockcipher is said to be secure if there is no known attack faster than exhaustive key search. On the other hand, without utilizing any weakness of a blockcipher, one can recover its secret key simply by trying all possible keys over a small number of plaintext-ciphertext pairs. So the key length of a blockcipher can be viewed as the maximum level of security that the blockcipher is able to provide. However the key length providing a sufficient level of security might change over time. For example, the Data Encryption Standard (DES) [1] using 56-bit keys was one of the most predominant algorithms for encryption of data. No feasible attacks faster than exhaustive key search have been proposed (as most of them require a huge amount of data), while the availability of increasing computational power made the brute-force attack itself practical. As a result, DES was replaced by a new standard algorithm AES [4]. On the other hand, in order to protect legacy applications based on DES, there has been considerable research on constructing DES-based encryption schemes which employ longer keys. This approach is called *key-length extension*, for which Triple-DES [2, 3, 5] and DESX (due to Rivest) are the most popular constructions.

The Triple-DES approach transforms a κ-bit key n-bit blockcipher E into an encryption scheme that accepts three κ-bit keys $k_1, k_2, k_3 \in \{0,1\}^\kappa$ and encrypts an n-bit message block u as $v = E_{k_3}(E_{k_2}(E_{k_1}(u)))$ as seen in Figure 1.[1] Bellare and Rogaway [6] proved its security up to $2^{\kappa+\frac{1}{2}\min\{n,\kappa\}}$ query complexity assuming E is an ideal blockcipher, and later Gaži and Maurer [9] fixed some flaws of the original proof.

The DESX approach transforms a κ-bit key n-bit blockcipher E into an encryption scheme that accepts a κ-bit key $k \in \{0,1\}^\kappa$ and additional n-bit whitening keys $k_i, k_o \in \{0,1\}^n$ and

[1] In the standards, the second key is applied to the decryption algorithm, while it makes no difference in our security proof for triple encryption and its two-key variant.

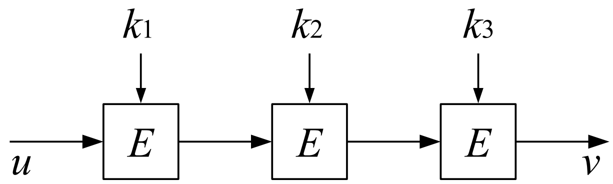


Fig. 1. Triple encryption

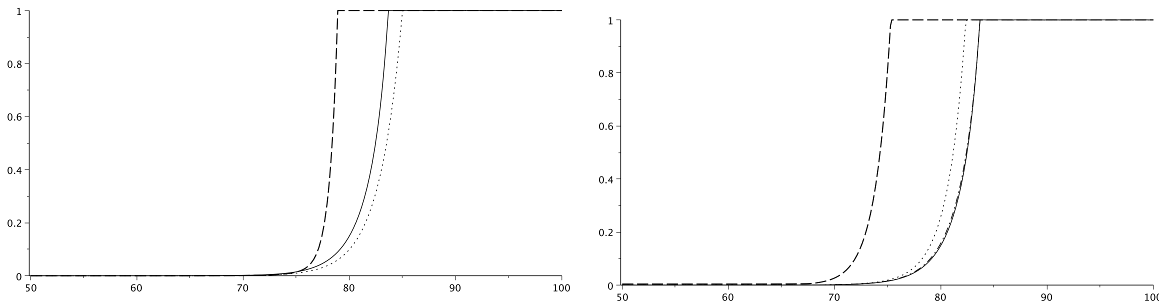
encrypts an n-bit message block u as $v = k_o \oplus E_k(k_i \oplus u)$. Killan and Rogaway [11] proved its security up to $2^{\frac{\kappa+n}{2}}$ query complexity. In order to improve the security, Gaži and Tessaro [10] proposed a cascade of two DESX schemes with some refinement (called 2-XOR-cascade), and proved its security up to $2^{\kappa+\frac{n}{2}}$ query complexity.

Our Contribution. In this paper, we revisit the long-standing open problem asking the exact provable security of triple encryption in the ideal cipher model. Since the best information theoretic attack requires $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ query complexity [12](see also Appendix A), there has been a gap between the upper and lower bounds for the security of triple encryption. We close the gap by proving the security up to $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ query complexity, improving over the currently known bound $2^{\kappa+\frac{1}{2}\min\{\kappa,n\}}$. With the DES parameters and the threshold distinguishing advantage 1/2, triple encryption is secure up to $2^{82.4}$ queries, greater than the current bound $2^{78.3}$ and comparable to $2^{83.5}$ for 2-XOR-cascade [10].

In order to save key materials, the standards define an alternative keying option: k_1 and k_2 are independent, and $k_3 = k_1$. However this variant, called *two-key triple encryption*, is vulnerable to the classic meet-in-the-middle attack making approximately 2^κ queries to the underlying blockcipher and 2^κ queries to the outer permutation. This attack was refined in [7] into a trade-off between time and data: given q_P plaintext-ciphertext pairs one can find the secret key by making $2^{\kappa+n}/q_P$ queries to the underlying blockcipher. So these attacks naturally raise the question if the two-key triple encryption is secure with data complexity limited to a certain bound. We answer this question affirmatively, proving that two-key triple encryption is secure up to $2^{\kappa+\min\{\kappa,\frac{n}{2}\}}$ blockcipher queries and $2^{\min\{\kappa,\frac{n}{2}\}}$ construction queries. For the DES parameters, this result is interpreted as the security of two-key triple encryption up to 2^{32} plaintext-ciphertext pairs and $2^{81.1}$ blockcipher encryptions. Table 2 compares upper bounds on distinguishing advantage for three-key and two-key triple encryption with the DES parameters $\kappa = 56$ and $n = 64$.

Proof Techniques. Our security proof is based on the combinatorial interpretation of Gaži and Maurer's random system framework. We define a transcript, the set of all query-response pairs that a distinguisher obtains by the interaction with the underlying blockcipher and the construction, and consider a directed graph defined by this information (see Section 3.2). The problem of security proof is reduced to limiting adversarial capability of constructing many number of long paths in this graph representation.

In the security proof of triple encryption, we need to restrict the number of directed paths of length 3. In [6, 9], they fixed the first and the third edges and probabilistically upper bounded the number of the second edges that connect them. Since each query generates a single edge in the graph, this estimation basically gives the upper bound on the number of 3-paths that is not smaller than q^2, where q denotes the number of blockcipher queries. In this paper, we take a different approach: we classify the set of 3-paths into two subsets according



(a) Left to right: (1) triple encryption [6, 9] (2) triple encryption (this paper) (3) 2-XOR-cascade [10]. The number of construction queries is set to be the maximum 2^n.

(b) Left to right: (1-3) two-key triple encryption with the number of construction queries $q_P = 2^{40}$, 2^{32}, 2^{24}, respectively (4) three-key triple encryption (this paper).

Fig. 2. Upper bounds on distinguishing advantage for three-key and two-key triple encryption. Given as functions of $\log_2 q$ where q is the number of queries made to the underlying blockcipher.

to the direction of the query by which the second edge has been obtained and upper bound the size of each subset. For example, in order to upper bound the number of 3-paths whose second edge has been obtained by a forward query, we fix the third edge from q possibilities. For each edge, we can probabilistically upper bound the number of edges coming into it by forward queries approximately by $\frac{q}{2^n}$. Since each of the possible second edges has again 2^κ possible edges coming into it, we can upper bound the number of 3-paths by $2^{\kappa-n}q^2$, which is smaller than q^2 when $\kappa < n$.

2 Preliminaries

2.1 General Notation

For an integer $n \geq 1$, let $I_n = \{0,1\}^n$ be the set of binary strings of length n. The set of all permutations on I_n will be denoted $\mathcal{P}_n$. For integers $1 \leq s \leq t$, we will write $(t)_s = t(t-1)\cdots(t-s+1)$ and $(t)_0 = 1$ by convention.

2.2 The Ideal Cipher Model

A blockcipher is a function family $E : \{0,1\}^\kappa \times \{0,1\}^n \to \{0,1\}^n$ such that for all $k \in \{0,1\}^\kappa$ the mapping $E(k,\cdot)$ is a permutation on $\{0,1\}^n$. We write $BC(\kappa, n)$ to mean the set of all such blockciphers. In the ideal cipher model, a blockcipher E is chosen from $BC(\kappa, n)$ uniformly at random. It allows for two types of oracle queries $E(k, x)$ and $E^{-1}(k, y)$ for $x, y \in \{0,1\}^n$ and $k \in \{0,1\}^\kappa$.[2] The response to an inverse query $E^{-1}(k, y)$ is $x \in \{0,1\}^n$ such that $E(k, x) = y$. Throughout this paper, we will write $K = 2^\kappa$ and $N = 2^n$.

2.3 Indistinguishability

Let C be an n-bit encryption scheme that employs λ-bit keys and makes oracle queries to a blockcipher $E \in BC(\kappa, n)$. So each key $\mathbf{k} \in \{0,1\}^\lambda$ and a blockcipher $E \in BC(\kappa, n)$ define a permutation $\mathsf{C}_{\mathbf{k}}[E]$ on I_n. In the *indistinguishability* framework (in the ideal cipher model),

[2] We interchangeably use both representations $E(k, x)$ and $E_k(x)$, and similarly $E^{-1}(k, y)$ and $E_k^{-1}(y)$.

$\mathsf{C}_{\mathbf{k}}[E]$ uses a random secret key $\mathbf{k}$ and makes oracle queries to an ideal blockcipher E, while a permutation P is chosen uniformly at random from $\mathcal{P}_n$. A distinguisher $\mathcal{D}$ would like to tell apart two worlds $(\mathsf{C}_{\mathbf{k}}[E], E)$ and (P, E) by adaptively making forward and backward queries to the permutation and the blockcipher. Formally, $\mathcal{D}$'s distinguishing advantage is defined by

$$\begin{aligned}\mathbf{Adv}_{\mathsf{C}}^{\mathsf{PRP}}(\mathcal{D}) = \mathbf{Pr}&\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[P, E] = 1\right] \\ &- \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} \{0,1\}^{\lambda}, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[\mathsf{C}_{\mathbf{k}}[E], E] = 1\right].\end{aligned}$$

For q_P, $q_E > 0$, we define

$$\mathbf{Adv}_{\mathsf{C}}^{\mathsf{PRP}}(q_P, q_E) = \max_{\mathcal{D}} \mathbf{Adv}_{\mathsf{C}}^{\mathsf{PRP}}(\mathcal{D})$$

where the maximum is taken over all distinguishers $\mathcal{D}$ making exactly q_P queries to the outer permutation and exactly q_E queries to the underlying blockcipher.

COMBINATORIAL FRAMEWORK. We assume that a distinguisher $\mathcal{D}$ makes q_P forward and/or backward queries to the permutation oracle and records a query history

$$\mathcal{Q}_P = (u^j, v^j)_{1 \le j \le q_P}$$

where (u^j, v^j) represents the evaluation obtained by the j-th query to the permutation oracle. So according to the instantiation, it implies either $\mathsf{C}_{\mathbf{k}}[E](u^j) = v^j$ or $P(u^j) = v^j$. By making q_E queries to the underlying blockcipher E, $\mathcal{D}$ also records the second query history

$$\mathcal{Q}_E = (x^j, k^j, y^j)_{1 \le j \le q_E}$$

where (x^j, k^j, y^j) represents the evaluation $E(k^j, x^j) = y^j$ obtained by the j-th query to the blockcipher. Sometimes we need to record the direction in which a blockcipher query has been made. If the j-th query has been made in a forward direction, the evaluation might be denoted as $(x^j, k^j, y^j, +)$. If it is obtained by a backward query, it is denoted as $(x^j, k^j, y^j, -)$.[3] The pair of the query histories

$$\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$$

is called the *transcript* of the attack; it contains all the information that $\mathcal{D}$ has obtained at the end of the attack. In this work, we will only consider information theoretic distinguishers. Therefore we can assume that a distinguisher is deterministic without making any redundant queries, and hence the output of $\mathcal{D}$ can be regarded as a function of $\mathcal{T}$, denoted $\mathcal{D}(\mathcal{T})$ or $\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)$.

If a permutation $\mathsf{C}_{\mathbf{k}}[E]$(resp. P) is consistent with $\mathcal{Q}_P$, i.e., $\mathsf{C}_{\mathbf{k}}[E](u^j) = v^j$(resp. $P(u^j) = v^j$) for every $j = 1, \ldots, q_P$, then we will write $\mathsf{C}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P$(resp. $P \vdash \mathcal{Q}_P$). Similarly, if a blockcipher $E \in BC(\kappa, n)$ is consistent with $\mathcal{Q}_E$ (i.e., $E(k^j, x^j) = y^j$ for $j = 1, \ldots, q_E$), then we will write $E \vdash \mathcal{Q}_E$. Using these notations, we have

$$\begin{aligned}\mathbf{Adv}_{\mathsf{C}}^{\mathsf{PRP}}(\mathcal{D}) = &\sum_{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1} \mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \\ &- \sum_{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1} \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} \{0,1\}^{\lambda}, E \xleftarrow{\$} BC(\kappa, n) : \mathsf{C}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right]\end{aligned}$$

[3] The sign of each query in $\mathcal{Q}_E$ is uniquely defined assuming that $\mathcal{D}$ is deterministic.

where the sum is taken over all the possible transcripts $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$ such that $\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E) = 1$. Here we only consider "valid" transcripts that $\mathcal{D}$ might produce by communicating with a permutation $P \in \mathcal{P}_n$ and a blockcpher $E \in BC(\kappa, n)$. Precisely, a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$ is called *valid* if and only if

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \neq 0.$$

2.4 Main Lemma

Let C be an n-bit encryption scheme that employs λ-bit keys and makes oracle queries to a blockcipher $E \in BC(\kappa, n)$. We will call C *perfect secure against construction queries* if for each key $\mathbf{k} \in I_\lambda$, $\mathsf{C}_{\mathbf{k}}[E]$ becomes a truly random permutation on I_n over a random choice of $E \in BC(\kappa, n)$. For example, triple encryption and its two-key variant are all perfect secure against construction queries. In this section, we give a combinatorial lemma that can be applied to any encryption scheme that is perfect secure against construction queries.

In order to state the lemma, we need to define a certain set of *bad transcripts*, denoted BadT. The probability that a distinguisher obtains a bad transcript in the ideal world is assumed to be small. Specifically, for any distinguisher $\mathcal{D}$ making q_P queries to the outer permutation and q_E queries to the underlying blockcipher, let

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D} \text{ produces } (\mathcal{Q}_P, \mathcal{Q}_E) \in \mathsf{BadT}\right] \leq \epsilon_1$$

for a small $\epsilon_1 > 0$. For each transcript $(\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}$, we also define a certain (small) set of *bad keys*, denoted BadK.[4] Suppose that

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda : \mathbf{k} \in \mathsf{BadK}\right] \leq \epsilon_2$$

for any transcript $(\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}$. With this setting, we can state the following lemma.

Lemma 1. *Let q_P, q_E, $\delta > 0$. Assume that for any transcript $(\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}$ such that $|\mathcal{Q}_P| = q_P$ and $|\mathcal{Q}_E| = q_E$,*

$$\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}) \geq (1-\delta)\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK})$$

where

$$\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}) = \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E \;\middle|\; \mathsf{C}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P \wedge \mathbf{k} \notin \mathsf{BadK}\right],$$

$$\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}) = \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E \;\middle|\; P \vdash \mathcal{Q}_P \wedge \mathbf{k} \notin \mathsf{BadK}\right].$$

Then we have

$$\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{C}}(q_P, q_E) \leq \delta + \epsilon_1 + \epsilon_2.$$

[4] This set might depend on the transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$, but we will hide the parameter in the notation.

Proof. For a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}$, define

$$\begin{aligned}
\mathsf{p}_1(\mathcal{Q}_P \wedge \neg\mathsf{BadK}) &= \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, E \xleftarrow{\$} BC(\kappa, n) : \mathsf{C}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P \wedge \mathbf{k} \notin \mathsf{BadK}\right], \\
\mathsf{p}_2(\mathcal{Q}_P \wedge \neg\mathsf{BadK}) &= \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n : P \vdash \mathcal{Q}_P \wedge \mathbf{k} \notin \mathsf{BadK}\right], \\
\mathsf{p}_1(\mathcal{Q}_P \wedge \mathcal{Q}_E \wedge \neg\mathsf{BadK}) &= \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, E \xleftarrow{\$} BC(\kappa, n) : \mathsf{C}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E \wedge \mathbf{k} \notin \mathsf{BadK}\right] \\
&= \mathsf{p}_1(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg\mathsf{BadK})\mathsf{p}_1(\mathcal{Q}_P \wedge \neg\mathsf{BadK}), \\
\mathsf{p}_2(\mathcal{Q}_P \wedge \mathcal{Q}_E \wedge \neg\mathsf{BadK}) &= \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E \wedge \mathbf{k} \notin \mathsf{BadK}\right] \\
&= \mathsf{p}_2(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg\mathsf{BadK})\mathsf{p}_2(\mathcal{Q}_P \wedge \neg\mathsf{BadK}).
\end{aligned}$$

Since C is perfect secure against construction queries, we have

$$\mathsf{p}_1(\mathcal{Q}_P \wedge \neg\mathsf{BadK}) = \mathsf{p}_2(\mathcal{Q}_P \wedge \neg\mathsf{BadK}).$$

In the following estimation, we will also use inequalities

$$\begin{aligned}
&\sum_{\substack{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1 \\ (\mathcal{Q}_P, \mathcal{Q}_E) \in \mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \\
&\qquad \leq \mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D} \text{ produces } (\mathcal{Q}_P, \mathcal{Q}_E) \in \mathsf{BadT}\right] \leq \epsilon_1
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{\substack{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1 \\ (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E \wedge \mathbf{k} \in \mathsf{BadK}\right] \\
&\leq \sum_{\substack{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1 \\ (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\lambda, P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathbf{k} \in \mathsf{BadK} \,\middle|\, P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \\
&\qquad \times \mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \\
&\qquad \leq \epsilon_2 \sum_{\substack{\mathcal{D}(\mathcal{Q}_P, \mathcal{Q}_E)=1 \\ (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}}} \mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : P \vdash \mathcal{Q}_P \wedge E \vdash \mathcal{Q}_E\right] \leq \epsilon_2
\end{aligned}$$

that hold for any distinguisher $\mathcal{D}$ making q_P queries to the outer permutation and q_E queries to the underlying blockcipher. Then for any such distinguisher $\mathcal{D}$, we have

$$\begin{aligned}
\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{C}}(\mathcal{D}) &\leq \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k}\xleftarrow{\$} I_\lambda, P\xleftarrow{\$}\mathcal{P}_n, E\xleftarrow{\$} BC(\kappa,n) : P\vdash\mathcal{Q}_P\wedge E\vdash\mathcal{Q}_E\right]\\
&- \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k}\xleftarrow{\$} I_\lambda, E\xleftarrow{\$} BC(\kappa,n) : \mathsf{C}_{\mathbf{k}}[E]\vdash\mathcal{Q}_P\wedge E\vdash\mathcal{Q}_E\right]\\
&+ \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\in\mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k}\xleftarrow{\$} I_\lambda, P\xleftarrow{\$}\mathcal{P}_n, E\xleftarrow{\$} BC(\kappa,n) : P\vdash\mathcal{Q}_P\wedge E\vdash\mathcal{Q}_E\right]\\
&\leq \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_2(\mathcal{Q}_P\wedge\mathcal{Q}_E\wedge\neg\mathsf{BadK}) - \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_1(\mathcal{Q}_P\wedge\mathcal{Q}_E\wedge\neg\mathsf{BadK})\\
&+ \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathbf{Pr}\left[\mathbf{k}\xleftarrow{\$} I_\lambda, P\xleftarrow{\$}\mathcal{P}_n, E\xleftarrow{\$} BC(\kappa,n) : P\vdash\mathcal{Q}_P\wedge E\vdash\mathcal{Q}_E\wedge\mathbf{k}\in\mathsf{BadK}\right]+\epsilon_1\\
&\leq \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P\wedge\neg\mathsf{BadK})\mathsf{p}_2(\mathcal{Q}_P\wedge\neg\mathsf{BadK})\\
&- \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P\wedge\neg\mathsf{BadK})\mathsf{p}_1(\mathcal{Q}_P\wedge\neg\mathsf{BadK})+\epsilon_2+\epsilon_1\\
&\leq \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P\wedge\neg\mathsf{BadK})\mathsf{p}_2(\mathcal{Q}_P\wedge\neg\mathsf{BadK})\\
&-(1-\delta) \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P\wedge\neg\mathsf{BadK})\mathsf{p}_2(\mathcal{Q}_P\wedge\neg\mathsf{BadK})+\epsilon_2+\epsilon_1\\
&\leq \delta \sum_{\substack{\mathcal{D}(\mathcal{Q}_P,\mathcal{Q}_E)=1\\(\mathcal{Q}_P,\mathcal{Q}_E)\notin\mathsf{BadT}}} \mathsf{p}_2(\mathcal{Q}_P\wedge\mathcal{Q}_E\wedge\neg\mathsf{BadK})+\epsilon_2+\epsilon_1\leq\delta+\epsilon_1+\epsilon_2. \qquad \square
\end{aligned}$$

3 Security of Triple Encryption

In this section, we prove the security of triple encryption using a κ-bit key n-bit blockcipher. The triple encryption will be denoted as TE. So given the underlying blockcipher $E\in BC(\kappa,n)$ and a key $\mathbf{k}=(k_1,k_2,k_3)\in I_\kappa^3$, then

$$\mathsf{TE}_{\mathbf{k}}[E](u)=E_{k_3}(E_{k_2}(E_{k_1}(u)))$$

for each $u\in I_n$. Our goal is to prove the security of TE far beyond N queries, so we will assume that a distinguisher makes all possible N queries to the outer permutation. Let q denote the number of queries made to the underlying blockcipher.

3.1 Graph Representation

When we define a certain type of bad keys, we will use a graph representation of a transcript. Given a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$, we will define a graph $\mathcal{G}$ on I_n as follows.

1. If $(u, v) \in \mathcal{Q}_P$, then $\mathcal{G}$ contains an edge $v \to u$ (with no label and the direction inversed).
2. If $(x, k, y, \sigma) \in \mathcal{Q}_E$, then $\mathcal{G}$ contains an edge $x \xrightarrow{(k,\sigma)} y$, where $\sigma \in \{+, -\}$ denotes the sign.

Sometimes we will drop the sign for simplicity.

3.2 Bad Transcripts

In order to apply Lemma 1, we define a set of bad transcripts $\mathsf{BadT}(L)$ parameterized by a certain parameter $L > 0$. Specifically, a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$ is defined to be *bad* if either

$$\max_{y^* \in I_n} |\{(x, k, y^*, +) \in \mathcal{Q}_E\}| > L$$

or

$$\max_{x^* \in I_n} |\{(x^*, k, y, -) \in \mathcal{Q}_E\}| > L.$$

So a bad transcript means an L-multi-collision on the blockcipher obtained by only forward queries or only backward queries. We can upper bound the probability that a distinguisher obtains a bad transcript in the ideal world as follows.

Lemma 2. *Let $L = L' + 2q/N$ for $L' > 0$ and let $\mathcal{D}$ be a distinguisher making all possible N queries to the outer permutation and exactly q queries to the underlying blockcipher. Then we have*

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D} \text{ produces } (\mathcal{Q}_P, \mathcal{Q}_E) \in \mathsf{BadT}(L)\right] \leq \frac{N}{2}\left(\frac{2eq}{L'N}\right)^{L'}. \quad (1)$$

Proof. We will say a blockcipher query is a *super* query if $\mathcal{D}$ has already made $N/2$ queries with the same key before the blockcipher query. Otherwise, the blockcipher query is called *normal*. During the interaction, $\mathcal{D}$ would make at most $2q/N$ super queries. Therefore in order for $\mathcal{D}$ to produce a transcript in $\mathsf{BadT}(L'+2q/N)$, $\mathcal{D}$ would have to obtain an L'-multi-collision by using only normal queries. Since the response to each normal query is chosen from more than $N/2$ possibilities, we have

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D} \text{ produces } (\mathcal{Q}_P, \mathcal{Q}_E) \in \mathsf{BadT}(L)\right]$$
$$\leq \binom{q}{L'}\left(\frac{2}{N}\right)^{L'-1} \leq \frac{N}{2}\left(\frac{2eq}{L'N}\right)^{L'}. \qquad \square$$

3.3 Bad Keys

Given a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}(L)$, we define three types of bad keys.

COLLIDING KEYS. Let

$$\mathsf{Co} = \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{either } k_1 = k_2 \text{ or } k_1 = k_3 \text{ or } k_2 = k_3\}$$

denote the set of "colliding" keys. We have

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3 : \mathbf{k} \in \mathsf{Co}\right] \leq \frac{3}{K}.$$

Heavy Keys. For a fixed parameter $M > 0$, we say a key $\mathbf{k} = (k_1, k_2, k_3) \in I_\kappa^3$ is *heavy* if

$$|\{k_i : (x, k_i, y) \in \mathcal{Q}_E\}| > M,$$

for some $i = 1, 2, 3$. Let $\mathsf{He}(M)$ denote the set of heavy keys. Since the number of keys that are queried more than M times is at most q/M, we have

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3 : \mathbf{k} \in \mathsf{He}(M)\right] \leq \frac{3q}{KM}.$$

Keys Making Bad Paths. We will define keys producing paths of length 3 or 4 in $\mathcal{G}$ to be bad. Specifically, let

$$\begin{aligned}
\mathsf{Pa}_{(0,+)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_3} v \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(0,-)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } u \xrightarrow{k_1} x \xrightarrow{(k_2,-)} y \xrightarrow{k_3} v \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(1,+)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } x \xrightarrow{k_2} y \xrightarrow{(k_3,+)} v \longrightarrow u \xrightarrow{k_1} z \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(1,-)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } x \xrightarrow{k_2} y \xrightarrow{(k_3,-)} v \longrightarrow u \xrightarrow{k_1} z \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(2,+)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } y \xrightarrow{k_3} v \longrightarrow u \xrightarrow{(k_1,+)} z \xrightarrow{k_2} w \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(2,-)} &= \{(k_1, k_2, k_3) \in I_\kappa^3 : \text{ there is a path } y \xrightarrow{k_3} v \longrightarrow u \xrightarrow{(k_1,-)} z \xrightarrow{k_2} w \text{ in } \mathcal{G}\}
\end{aligned}$$

and let

$$\mathsf{Pa} = \mathsf{Pa}_{(0,+)} \cup \mathsf{Pa}_{(0,-)} \cup \mathsf{Pa}_{(1,+)} \cup \mathsf{Pa}_{(1,-)} \cup \mathsf{Pa}_{(2,+)} \cup \mathsf{Pa}_{(2,-)}.$$

We can upper bound the size of $\mathsf{Pa}_{(0,+)}$ by the number of paths of form $u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_3} v$. For a node $x \in I_n$, let $d_{in}(x)$ and $d_{out}(x)$ denote the in-degree and the out-degree of x, respectively, with respect to the edges defined by $\mathcal{Q}_E$. If a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$ is not contained in $\mathsf{BadT}(L)$, then the number of paths of form $u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_3} v$, and hence the size of $\mathsf{Pa}_{(0,+)}$ is upper bounded by

$$KL \sum_{y \in I_n} d_{out}(y) \leq KLq$$

since for each $y \in I_n$ the number of $(k_2, +)$-labeled edges coming into y is at most L, and for each $x \in I_n$ such that there exists an edge $x \xrightarrow{(k_2,+)} y$ in $\mathcal{G}$, we have $d_{in}(x) \leq K$. Applying similar arguments to the other types of paths, we have $|\mathsf{Pa}| \leq 6KLq$, and hence

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3 : \mathbf{k} \in \mathsf{Pa}\right] \leq \frac{6Lq}{K^2}.$$

Summary. We define the total set of bad keys $\mathsf{BadK}(M) = \mathsf{Co} \cup \mathsf{He}(M) \cup \mathsf{Pa}$. Then

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3 : \mathbf{k} \in \mathsf{BadK}(M)\right] \leq \frac{3}{K} + \frac{3q}{KM} + \frac{6Lq}{K^2}. \tag{2}$$

3.4 Comparing $\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$ and $\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$

In this section, we compute a small δ satisfying the condition of Lemma 1. First, we fix a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}(L)$. Then for each key $\mathbf{k} = (k_1, k_2, k_3) \notin \mathsf{BadK}(M)$, we decompose the blockcipher query history $\mathcal{Q}_E$ as

$$\mathcal{Q}_E = \mathcal{Q}_E^{k_1} \cup \mathcal{Q}_E^{k_2} \cup \mathcal{Q}_E^{k_3} \cup \mathcal{Q}_E^{*}$$

where

$$\mathcal{Q}_E^{k_i} = \{(x, k, y) \in \mathcal{Q}_E : k = k_i\}$$

for $i = 1, 2, 3$, and $\mathcal{Q}_E^{*}$ is the set of the remaining queries. Let

$$\mathsf{p}^*(\mathbf{k}) = \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E^{*}\right]$$

and let $h_i = |\mathcal{Q}_E^{k_i}|$ for $i = 1, 2, 3$. Then we have

$$\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) = \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E\right] = \frac{\mathsf{p}^*(\mathbf{k})}{(N)_{h_1}(N)_{h_2}(N)_{h_3}} \tag{3}$$

for any key $\mathbf{k} \notin \mathsf{BadK}(M)$ since the choice of the key and a random permutation P is independent of E. On the other hand, let

$$\mathsf{p}_1(\mathbf{k}) = \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E \;\middle|\; \mathsf{TE}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P\right]$$

for each $\mathbf{k} \notin \mathsf{BadK}(M)$. Since $\mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : \mathsf{TE}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P\right]$ is the same for every $\mathbf{k} \notin \mathsf{BadK}(M)$, we have

$$\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) = \frac{1}{|I_\kappa^3 \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k} \notin \mathsf{BadK}(M)} \mathsf{p}_1(\mathbf{k}). \tag{4}$$

Since each key defines an independent random permutation in the ideal cipher model, we have

$$\begin{aligned}
\mathsf{p}_1(\mathbf{k}) &= \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E^{k_1} \cup \mathcal{Q}_E^{k_2} \cup \mathcal{Q}_E^{k_3} \;\middle|\; \mathsf{TE}_{\mathbf{k}}[E] \vdash \mathcal{Q}_P\right] \\
&= \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[P_1, P_2, P_3 \xleftarrow{\$} \mathcal{P}_n : P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P_2 \vdash \overline{\mathcal{Q}}_E^{k_2} \wedge P_3 \vdash \overline{\mathcal{Q}}_E^{k_3} \;\middle|\; P_3 \circ P_2 \circ P_1 \vdash \mathcal{Q}_P\right] \\
&= \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[P_1, P_2, P \xleftarrow{\$} \mathcal{P}_n : P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P_2 \vdash \overline{\mathcal{Q}}_E^{k_2} \wedge P \circ P_1^{-1} \circ P_2^{-1} \vdash \overline{\mathcal{Q}}_E^{k_3} \;\middle|\; P \vdash \mathcal{Q}_P\right]
\end{aligned}$$

where $\overline{\mathcal{Q}}_E^{k_i} = \left\{(x, y) : (x, k_i, y) \in \mathcal{Q}_E^{k_i}\right\}$ for $i = 1, 2, 3$. The conditional probability appearing in the last line is the probability of event

$$\mathsf{E}: \; P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P_2 \vdash \overline{\mathcal{Q}}_E^{k_2} \wedge P_* \circ P_1^{-1} \circ P_2^{-1} \vdash \overline{\mathcal{Q}}_E^{k_3}$$

over random choice of P_1 and P_2, where P_* is the unique permutation that is consistent with $\mathcal{Q}_P$. Let

$$\begin{aligned}
V &= \left\{v \in I_n : \text{there exists } y \xrightarrow{k_3} v \text{ in } \mathcal{G}\right\} \\
V' &= \left\{v \in I_n : \text{there exists } x \xrightarrow{k_2} y \xrightarrow{k_3} v \text{ in } \mathcal{G}\right\}.
\end{aligned}$$

Then event E requires that P_1 satisfy the following.

1. $P_1(u) = x$ for $(u, x) \in \overline{\mathcal{Q}}_E^{k_1}$.
2. $P_1(P_*^{-1}(v)) = x$ for $v \in V'$ and x such that $x \xrightarrow{k_2} y \xrightarrow{k_3} v$ in $\mathcal{G}$.
3. $P_1(P_*^{-1}(v)) \neq x$ for $v \in V \setminus V'$ and x such that $x \xrightarrow{k_2} y$ in $\mathcal{G}$.

In order to lower bound the probability that a random permutation P_1 satisfies the above three conditions, we need to note the following properties.

- For any $v \in V'$ and x such that $x \xrightarrow{k_2} y \xrightarrow{k_3} v$ in $\mathcal{G}$, neither $P_1(P_*^{-1}(v))$ nor $P_1^{-1}(x)$ is determined by $\overline{\mathcal{Q}}_E^{k_1}$ since $\mathbf{k} \notin \mathsf{Pa}_{(0,+)} \cup \mathsf{Pa}_{(0,-)} \cup \mathsf{Pa}_{(1,+)} \cup \mathsf{Pa}_{(1,-)}$.
- For any $v \in V \setminus V'$, if $x = P_1(P_*^{-1}(v))$ is determined by $\overline{\mathcal{Q}}_E^{k_1}$, then there is no edge $x \xrightarrow{k_2} y$ in $\mathcal{G}$ since $\mathbf{k} \notin \mathsf{Pa}_{(2,+)} \cup \mathsf{Pa}_{(2,-)}$.

By these properties, the probability of a random permutation P_1 satisfying the three conditions is lower bounded by

$$\left(1 - \frac{\alpha_2 h_2}{N - h_1 - \alpha_1}\right) \frac{1}{(N)_{h_1+\alpha_1}}$$

where $\alpha_1 = |V'|$ and $\alpha_2 = |V \setminus V'| = h_3 - \alpha_1$. Once P_1 is determined, E requires that P_2 satisfy the following.

1. $P_2(x) = y$ for $(x, y) \in \overline{\mathcal{Q}}_E^{k_2}$.
2. $P_2(P_1(P_*^{-1}(v))) = y$ for $v \in V \setminus V'$ and y such that $y \xrightarrow{k_3} v$ in $\mathcal{G}$.

For any $v \in V \setminus V'$, $P_2(P_1(P_*^{-1}(v)))$ is not determined by $\overline{\mathcal{Q}}_E^{k_2}$ since $\mathbf{k} \notin \mathsf{Pa}_{(2,+)} \cup \mathsf{Pa}_{(2,-)}$ and by the third condition on the choice of P_1. Therefore the probability that a random permutation P_2 satisfies the above two conditions is given by $\frac{1}{(N)_{h_2+\alpha_2}}$. Since h_1, h_2, $h_3 \leq M$ for each $\mathbf{k} \notin \mathsf{BadK}(M)$, we have

$$\mathbf{Pr}\left[\mathsf{E}\right] \geq \left(1 - \frac{\alpha_2 h_2}{N - h_1 - \alpha_1}\right) \frac{1}{(N)_{h_1+\alpha_1}(N)_{h_2+\alpha_2}} \geq \left(1 - \frac{M^2}{N - 2M}\right) \frac{1}{(N)_{h_1+\alpha_1}(N)_{h_2+\alpha_2}}.$$

Furthermore, since

$$\begin{aligned}
\frac{(N)_{h_1}(N)_{h_2}(N)_{h_3}}{(N)_{h_1+\alpha_1} \cdot (N)_{h_2+\alpha_2}} &= \frac{(N)_{h_3}}{(N-h_1)_{\alpha_1} \cdot (N-h_2)_{\alpha_2}} = \frac{(N)_{\alpha_1}}{(N-h_1)_{\alpha_1}} \cdot \frac{(N-\alpha_1)_{\alpha_2}}{(N-h_2)_{\alpha_2}} \\
&\geq \frac{(N-\alpha_1)_{\alpha_2}}{(N)_{\alpha_2}} \geq \left(1 - \frac{\alpha_1}{N - \alpha_2 + 1}\right)^{\alpha_2} \geq 1 - \frac{M^2}{N - M + 1}
\end{aligned}$$

and by (3) and (4), we have

$$\begin{aligned}
\mathsf{p}_1(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg \mathsf{BadK}(M)) &= \frac{1}{|I_\kappa^3 \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k} \notin \mathsf{BadK}(M)} \mathsf{p}_1(\mathbf{k}) \\
&= \frac{1}{|I_\kappa^3 \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k} \notin \mathsf{BadK}(M)} \mathsf{p}^*(\mathbf{k}) \mathbf{Pr}\left[\mathsf{E}\right] \\
&\geq \left(1 - \frac{M^2}{N - 2M}\right) \frac{\mathsf{p}_2(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg \mathsf{BadK}(M))}{|I_\kappa^3 \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k} \notin \mathsf{BadK}(M)} \frac{(N)_{h_1}(N)_{h_2}(N)_{h_3}}{(N)_{h_1+\alpha_1}(N)_{h_2+\alpha_2}} \\
&\geq \left(1 - \frac{M^2}{N - 2M}\right) \left(1 - \frac{M^2}{N - M + 1}\right) \mathsf{p}_2(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg \mathsf{BadK}(M)) \\
&\geq \left(1 - \frac{2M^2}{N - 2M}\right) \mathsf{p}_2(\mathcal{Q}_E | \mathcal{Q}_P \wedge \neg \mathsf{BadK}(M)).
\end{aligned}$$

3.5 Putting the Pieces Together

By applying Lemma 1 with

$$\delta = \frac{2M^2}{N-2M}$$

and (1) and (2) for ϵ_1 and ϵ_2, we obtain the following theorem.

Theorem 1. *For q, L',$M > 0$, we have*

$$\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{TE}}(N,q) \le \frac{2M^2}{N-2M} + \frac{N}{2}\left(\frac{2eq}{L'N}\right)^{L'} + \frac{3}{K} + \frac{3q}{KM} + \frac{6L'q}{K^2} + \frac{12q^2}{K^2N}.$$

Optimizing Parameters. By setting $\frac{M^2}{N} = \frac{q}{KM}$, let

$$M = \left(\frac{Nq}{K}\right)^{\frac{1}{3}}.$$

Let $L' = \max\{\frac{4eq}{N}, 2n\}$. Then we have

$$\frac{N}{2}\left(\frac{2eq}{LN}\right)^{L} \le \frac{1}{2N}.$$

Assuming $N - 2M \ge \frac{2N}{3}$ or equivalently $q \le \frac{KN^2}{216}$, we have our final result.

Corollary 1. *For $q > 0$, we have*

$$\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{TE}}(2^n,q) \le 6\left(\frac{q^2}{2^{2\kappa+n}}\right)^{\frac{1}{3}} + \frac{1}{2^{n+1}} + \frac{3}{2^{\kappa}} + \frac{12q^2}{2^{2\kappa+n}} + \max\left\{\frac{24eq^2}{2^{2\kappa+n}}, \frac{12nq}{2^{2\kappa}}\right\}.$$

In other words, triple encryption is secure if

$$q \ll \min\left\{\frac{2^{\kappa+\frac{n}{2}}}{\sqrt{24e}}, \frac{2^{2\kappa}}{12n}\right\}.$$

4 Security of Two-Key Triple Encryption

In this section, we prove the security of triple encryption where the first and the third keys are identical. We will denote the two-key triple encryption by TE^*. So given the underlying blockcipher $E \in BC(\kappa, n)$ and a key $\mathbf{k} = (k_1, k_2) \in I_\kappa^2$, then

$$\mathsf{TE}^*_{\mathbf{k}}[E](u) = E_{k_1}(E_{k_2}(E_{k_1}(u)))$$

for each $u \in I_n$. Suppose that a distinguisher $\mathcal{D}$ makes q_P queries to the outer permutation and q_E queries to the underlying blockcipher. The proof strategy is similar to the three-key triple encryption, based on the same graph representation $\mathcal{G}$ defined by a query history.

Bad Transcripts. Bad transcripts are defined as for the three-key triple encryption: a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E)$ is defined to be *bad* if either

$$\max_{y^* \in I_n} |\{(x,k,y^*,+) \in \mathcal{Q}_E\}| > L \text{ or } \max_{x^* \in I_n} |\{(x^*,k,y,-) \in \mathcal{Q}_E\}| > L$$

where $L = L' + \frac{2q}{N}$ for some $L' > 0$. The set of bad transcripts will be denoted as $\mathsf{BadT}(L)$.

BAD KEYS. Given a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}(L)$, sets of bad keys Co and $\mathsf{He}(M)$ are defined as similar to the three-key triple encryption.

$$\begin{aligned}\mathsf{Co} &= \{(k_1, k_2) \in I_\kappa^2 : k_1 = k_2\},\\ \mathsf{He}(M) &= \{(k_1, k_2) \in I_\kappa^2 : |\{k_1 : (x, k_1, y) \in \mathcal{Q}_E\}| > M \vee |\{k_2 : (x, k_2, y) \in \mathcal{Q}_E\}| > M\}\end{aligned}$$

for a parameter $M > 0$. We also define

$$\begin{aligned}
\mathsf{Pa}_{(0,+)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_1} v \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(0,-)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } u \xrightarrow{k_1} x \xrightarrow{(k_2,-)} y \xrightarrow{k_1} v \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(1,+)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } x \xrightarrow{k_2} y \xrightarrow{(k_1,+)} v \longrightarrow u \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(1,-)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } x \xrightarrow{k_2} y \xrightarrow{(k_1,-)} v \longrightarrow u \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(2,+)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } v \longrightarrow u \xrightarrow{(k_1,+)} x \xrightarrow{k_2} y \text{ in } \mathcal{G}\},\\
\mathsf{Pa}_{(2,-)} &= \{(k_1,k_2) \in I_\kappa^2 : \text{ there is a path } v \longrightarrow u \xrightarrow{(k_1,-)} x \xrightarrow{k_2} y \text{ in } \mathcal{G}\}
\end{aligned}$$

and

$$\mathsf{Pa} = \mathsf{Pa}_{(0,+)} \cup \mathsf{Pa}_{(0,-)} \cup \mathsf{Pa}_{(1,+)} \cup \mathsf{Pa}_{(1,-)} \cup \mathsf{Pa}_{(2,+)} \cup \mathsf{Pa}_{(2,-)}.$$

In order to upper bound the size of $\mathsf{Pa}_{(0,+)}$, consider the number of 2-paths of form $x \xrightarrow{(k_2,+)} y \xrightarrow{k_1} v$. This number is upper bounded by

$$L \sum_{y \in I_n} d_{out}(y) \le L q_E$$

since the number of nodes coming into y by forward queries is at most L. Each of such 2-paths is uniquely extended to a 3-path $u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_1} v$ since the first and the third keys are identical. Since the number of paths of form $u \xrightarrow{k_1} x \xrightarrow{(k_2,+)} y \xrightarrow{k_1} v$ upper bounds the size of $\mathsf{Pa}_{(0,+)}$, we have $|\mathsf{Pa}_{(0,+)}| \le Lq_E$. A similar analysis applies to $\mathsf{Pa}_{(0,-)}$, $\mathsf{Pa}_{(1,-)}$ and $\mathsf{Pa}_{(2,+)}$.

On the other hand, in order to restrict the size of $\mathsf{Pa}_{(1,+)}$, consider 2-paths of form $y \xrightarrow{(k_1,+)} v \longrightarrow u$. The number of 2-paths of this form is at most Lq_P. Each of these paths is extended to $x \xrightarrow{k_2} y \xrightarrow{(k_1,+)} v \longrightarrow u$ with K possible keys k_2. Therefore the size of $\mathsf{Pa}_{(1,+)}$ is upper bounded by Lq_PK, and a similar analysis applies to $\mathsf{Pa}_{(2,-)}$. Overall, the size of Pa is upper bounded by

$$4Lq_E + 2Lq_PK.$$

Finally, we define the total set of bad keys $\mathsf{BadK}(M) = \mathsf{Co} \cup \mathsf{He}(M) \cup \mathsf{Pa}$. Then we have

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^2 : \mathbf{k} \in \mathsf{BadK}(M)\right] \le \frac{1}{K} + \frac{2q_E}{KM} + \frac{4Lq_E}{K^2} + \frac{2Lq_P}{K}. \tag{5}$$

COMPARING $\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$ AND $\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$. In order to use Lemma 1, we need to lower bound the ratio of $\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$ to $\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M))$. First, we fix a transcript $\mathcal{T} = (\mathcal{Q}_P, \mathcal{Q}_E) \notin \mathsf{BadT}(L)$. Then for each key $\mathbf{k} = (k_1, k_2) \notin \mathsf{BadK}(M)$, we decompose the blockcipher query history $\mathcal{Q}_E$ as

$$\mathcal{Q}_E = \mathcal{Q}_E^{k_1} \cup \mathcal{Q}_E^{k_2} \cup \mathcal{Q}_E^*$$

where $\mathcal{Q}_E^{k_i} = \{(x, k, y) \in \mathcal{Q}_E : k = k_i\}$ for $i = 1, 2$, and $\mathcal{Q}_E^*$ is the set of the remaining queries. Then we have

$$\mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) = \frac{\mathsf{p}^*(\mathbf{k})}{(N)_{h_1}(N)_{h_2}} \tag{6}$$

where $h_1 = |\mathcal{Q}_E^{k_1}|$, $h_2 = |\mathcal{Q}_E^{k_2}|$ and $\mathsf{p}^*(\mathbf{k}) = \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E^*\right]$.

On the other hand, let

$$\mathsf{p}_1(\mathbf{k}) = \mathbf{Pr}\left[E \xleftarrow{\$} BC(\kappa, n) : E \vdash \mathcal{Q}_E \;\middle|\; \mathsf{TE}_{\mathbf{k}}^*[E] \vdash \mathcal{Q}_P\right]$$

for each $\mathbf{k} \notin \mathsf{BadK}(M)$. Then we have

$$\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) = \frac{1}{|I_\kappa^2 \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k} \notin \mathsf{BadK}(M)} \mathsf{p}_1(\mathbf{k}). \tag{7}$$

By replacing $\mathsf{TE}_{\mathbf{k}}^*[E]$ by a truly random permutation P, we have

$$\mathsf{p}_1(\mathbf{k}) = \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2} \;\middle|\; P \vdash \mathcal{Q}_P\right]$$

where $\overline{\mathcal{Q}}_E^{k_i} = \left\{(x, y) : (x, k_i, y) \in \mathcal{Q}_E^{k_i}\right\}$ for $i = 1, 2$. Let

$$X = \{x \in I_n : x \xrightarrow{k_2} y \in \mathcal{G} \text{ for some } y \in I_n\},$$
$$Y = \{y \in I_n : x \xrightarrow{k_2} y \in \mathcal{G} \text{ for some } x \in I_n\}$$

be the sets of end nodes of k_2-labeled edges. We decompose X as a disjoint union of X_1, X_2 and X_3, where

$$X_2 = \{x \in I_n : x \xrightarrow{k_2} y \xrightarrow{k_1} z \in \mathcal{G} \text{ for some } y, z \in I_n\}$$
$$X_3 = \{x \in I_n : w \xrightarrow{k_1} x \xrightarrow{k_2} y \in \mathcal{G} \text{ for some } w, y \in I_n\}$$

and $X_1 = X \setminus (X_2 \cup X_3)$. Accordingly, we define

$$Y_i = \{y \in I_n : x \xrightarrow{k_2} y \in \mathcal{G} \text{ for some } x \in X_i\}$$

for $i = 1, 2, 3$. Assuming $P_1 \vdash \overline{\mathcal{Q}}_E^{k_1}$ and $P \vdash \mathcal{Q}_P$, we will determine $v = P_1(y)$ for $y \in Y_1$ by lazy sampling, where we would like to avoid the following conditions on the value v.

1. $\mathcal{G}$ contains an edge $v \xrightarrow{k_2} y$ for some $y \in I_n$.
2. $\mathcal{G}$ contains a 2-path $v \longrightarrow u \xrightarrow{k_1} y$ for some $u, y \in I_n$.
3. $\mathcal{G}$ contains an edge $v \longrightarrow u$ for some $u \in I_n$ such that $x \xrightarrow{k_2} u$ for some $x \in I_n$.

Let E_1 denote the event that $v = P_1(y)$ satisfies one of the above three conditions for some $y \in Y_1$. Then the probability of E_1 under condition $P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P$ is upper bounded as follows.

$$\mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : \mathsf{E}_1 \;\middle|\; P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \le \frac{(h_1 + 2h_2)h_2}{N - h_1} \le \frac{3M^2}{N - M}. \tag{8}$$

By avoiding the first condition, each evaluation $P_1(y)$ does not generate an edge coming into $x \in X_1 \cup X_2$. By avoiding the second condition, $P_1(y)$ does not generate any 4-path. We allow the node v to be connected with some node u by $\mathcal{Q}_P$, while $P_1(y)$ will not determine $P_1(u)$ for any other node y in $Y_1 \cup Y_3$ since we exclude the third condition.

Assuming that $P_1(y)$ has been determined for every $y \in Y_1$ avoiding the above conditions, and under the conditions $P_1 \vdash \overline{\mathcal{Q}}_E^{k_1}$ and $P \vdash \mathcal{Q}_P$, we evaluate P^{-1} at $P_1(y)$ for $y \in Y_1 \cup Y_2$ if not determined, and evaluate P at $P_1^{-1}(x)$ for $x \in X_3$, where $P_1^{-1}(x)$ is determined by $\overline{\mathcal{Q}}_E^{k_1}$. In this evaluation, we would like to avoid the following conditions.

1. $P^{-1}(P_1(y)) \in Y_3$ for some $y \in Y_1 \cup Y_2$.
2. $P(P_1^{-1}(x)) \in X_1 \cup X_2$ for some $x \in X_3$.

Let E_2 denote the event that one of the two conditions holds for some $x \in X_3$ or $y \in Y_1 \cup Y_2$. Then the probability of E_2 under condition $\neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P$ is upper bounded as follows.

$$\mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : \mathsf{E}_2 \;\middle|\; \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \le \frac{h_2^2}{N - q_P} \le \frac{M^2}{N - q_P}. \tag{9}$$

Finally, under condition $\neg\mathsf{E}_2 \wedge \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P$, we would like to upper bound the probability of $P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2}$. Assume that $P^{-1}(P_1(y))$ and $P(P_1^{-1}(x))$ have been determined for $y \in Y_1 \cup Y_2$ and $x \in X_3$ respectively. Then the event $P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2}$ implies the evaluations

1. $P_1(P^{-1}(P_1(y)) = x$ for each $y \in Y_1 \cup Y_2$ and x such that $x \xrightarrow{k_2} y \in \mathcal{G}$,
2. $P_1(y) = P(P_1^{-1}(x))$ for each $y \in Y_3$ and x such that $x \xrightarrow{k_2} y \in \mathcal{G}$.

Since P_1-evaluations at the points $P^{-1}(P_1(y))$, $y \in Y_1 \cup Y_2$, and $y \in Y_3$ are all free and independent, we have

$$\mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2} \;\middle|\; \neg\mathsf{E}_2 \wedge \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \ge \frac{1}{(N)_{h_2}}. \tag{10}$$

By (8), (9), (10), we have

$$\begin{aligned}
\mathsf{p}_1(\mathbf{k}) &\ge \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : \neg\mathsf{E}_2 \wedge \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2} \;\middle|\; P \vdash \mathcal{Q}_P\right] \\
&= \mathsf{p}^*(\mathbf{k}) \cdot \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : P_1^{-1} \circ P \circ P_1^{-1} \vdash \overline{\mathcal{Q}}_E^{k_2} \;\middle|\; \neg\mathsf{E}_2 \wedge \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \\
&\quad \times \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : \neg\mathsf{E}_2 \;\middle|\; \neg\mathsf{E}_1 \wedge P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \\
&\quad \times \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : \neg\mathsf{E}_1 \;\middle|\; P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \wedge P \vdash \mathcal{Q}_P\right] \\
&\quad \times \mathbf{Pr}\left[P_1, P \xleftarrow{\$} \mathcal{P}_n : P_1 \vdash \overline{\mathcal{Q}}_E^{k_1} \;\middle|\; P \vdash \mathcal{Q}_P\right] \ge \frac{\mathsf{p}^*(\mathbf{k})}{(N)_{h_2}} \cdot \left(1 - \frac{M^2}{N - q_P}\right) \cdot \left(1 - \frac{3M^2}{N - M}\right) \cdot \frac{1}{(N)_{h_1}},
\end{aligned}$$

and then by (6) and (7)

$$
\begin{aligned}
\mathsf{p}_1(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) &= \frac{1}{|I^3_\kappa \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k}\notin\mathsf{BadK}(M)} \mathsf{p}_1(\mathbf{k}) \\
&\geq \frac{1}{|I^3_\kappa \setminus \mathsf{BadK}(M)|} \sum_{\mathbf{k}\notin\mathsf{BadK}(M)} \frac{\mathsf{p}^*(\mathbf{k})}{(N)_{h_1}(N)_{h_2}} \left(1-\frac{M^2}{N-q_P}\right)\left(1-\frac{3M^2}{N-M}\right) \\
&= \left(1-\frac{M^2}{N-q_P}\right)\left(1-\frac{3M^2}{N-M}\right) \mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)) \\
&\geq \left(1-\frac{M^2}{N-q_P}-\frac{3M^2}{N-M}\right) \mathsf{p}_2(\mathcal{Q}_E|\mathcal{Q}_P \wedge \neg\mathsf{BadK}(M)).
\end{aligned}
$$

Summary. Applying Lemma 1 with (1) and (5), we have the following theorem.

Theorem 2. *For q_P, q_E, L', $M > 0$, we have*

$$
\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{TE}^*}(q_P, q_E) \leq \frac{M^2}{N-q_P} + \frac{3M^2}{N-M} + \frac{N}{2}\left(\frac{2eq_E}{L'N}\right)^{L'} + \frac{1}{K} + \frac{2q_E}{KM} + \frac{4L'q_E}{K^2} + \frac{8q_E^2}{K^2N} + \frac{2L'q_P}{K} + \frac{4q_Pq_E}{KN}.
$$

Let $M = \left(\frac{Nq_E}{4K}\right)^{\frac{1}{3}}$ and let $L = \max\{\frac{4eq_E}{N}, 2n\}$. Assuming M, $q_P \leq \frac{N}{2}$, we also have the following corollary.

Corollary 2. *For q_P, $q_E > 0$, we have*

$$
\begin{aligned}
\mathbf{Adv}^{\mathsf{PRP}}_{\mathsf{TE}^*}(q_P, q_E) \leq 16\left(\frac{q_E^2}{16\cdot 2^{2\kappa+n}}\right)^{\frac{1}{3}} &+ \frac{1}{2^{n+1}} + \frac{1}{2^\kappa} + \frac{8q_E^2}{2^{2\kappa+n}} + \frac{4q_Pq_E}{2^{\kappa+n}} \\
&+ \max\left\{\frac{16eq_E^2}{2^{2\kappa+n}} + \frac{8eq_Pq_E}{2^{\kappa+n}}, \frac{8nq_E}{2^{2\kappa}} + \frac{4nq_P}{2^\kappa}\right\}.
\end{aligned}
$$

We can interpret this result in two ways.

1. Two-key triple encryption is secure if $q_P \ll \frac{2^\kappa}{4n}$, $q_E \ll \min\left\{\frac{2^{\kappa+\frac{n}{2}}}{4\sqrt{e}}, \frac{2^{2\kappa}}{8n}\right\}$ and $q_Pq_E \ll \frac{2^{\kappa+n}}{8e}$.
2. Two-key triple encryption is secure if $q_P \ll \min\left\{\frac{2^\kappa}{4n}, \frac{2^{\frac{n}{2}}}{2\sqrt{e}}\right\}$ and $q_E \ll \min\left\{\frac{2^{\kappa+\frac{n}{2}}}{4\sqrt{e}}, \frac{2^{2\kappa}}{8n}\right\}$.

References

1. FIPS PUB 46: Data Encryption Standard (DES). National Institute of Standards and Technology (1977)
2. ANSI X9.52: Triple Data Encryption Algorithm Modes of Operation (1998)
3. FIPS PUB 46-3: Data Encryption Standard (DES). National Institute of Standards and Technology (1999)
4. FIPS PUB 197: Advanced Encryption Standard (AES). National Institute of Standards and Technology (2001)
5. NIST ST 800-67: Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher. National Institute of Standards and Technology (2004)
6. M. Bellare and P. Rogaway: The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. Eurocrypt 2006, LNCS 4004, pp. 409–426, Springer, Heidelberg (2006)
7. E. Biham, Y. Carmeli, I. Dinur, O. Dunkelman, N. Keller and A. Shamir: Cryptanalysis of Iterated Even-Mansour Schemes with Tow Keys. IACR Cryptology ePrint Archive, Report 2013/674, 2013. Available at http://eprint.iacr.org/2013/674

8. P. Gaži: Plain versus Randomized Cascading-Based Key-Length Extension for Block Ciphers. IACR Cryptology ePrint Archive, Report 2013/019, 2013. Available at http://eprint.iacr.org/2013/019
9. P. Gaži and U. Maurer: Cascade Encryption Revisited. Asiacrypt 2009, LNCS 5912, pp. 37–51, Springer, Heidelberg (2009)
10. P. Gaži and S. Tessaro: Efficient and Optimally Secure Key-Length Extension for Block Ciphers via Randomized Cascading. Eurocrypt 2012, LNCS 7237, pp. 63–80, Springer, Heidelberg (2012)
11. J. Kilian and P. Rogaway: How to Protect DES Against Exhaustive Key Search (an Analysis of DESX). Journal of Cryptology 14, pp. 17–35. Springer, Heidelberg (2001)
12. S. Lucks: Attacking Triple Encryption. FSE 1998, LNCS 1372, pp. 239–253. Springer, Heidelberg (1998)

A Matching Attacks on Triple Encryption

A.1 An Attack of $2^{\kappa+\frac{n}{2}}$ Query Complexity

This attack has been proposed by Lucks [12] and later extended by Gazi [8]. Let S denote the outer permutation instantiated with either $\mathsf{TE}_{\mathbf{k}}[E]$ using a random key $\mathbf{k} \in I_\kappa^3$ or a truly random permutation P. A distinguisher $\mathcal{D}$, parameterized by $r > 0$, executes the following steps.

1. Fix two sets S_0, $S_1 \subset I_n$ such that $|S_0| = |S_1| = rN^{\frac{1}{2}}$.
   (a) For each $k \in I_\kappa$ and $x \in S_0$, make a query $E_k(x)$.
   (b) For each $k' \in I_\kappa$ and $x \in S_1$, make a query $E_{k'}(x)$.
   (c) For each $x \in S_0$, make a query $\mathsf{S}(x)$.
   (d) For each $k'' \in I_\kappa$ and $x \in S_0$, make a query $E_{k''}^{-1}(\mathsf{S}(x))$.
2. For each key $k \in I_\kappa$, find a subset $U_k \subset S_0$ such that $|U_k| = \frac{r^2}{2}$ and $E_k(x) \in S_1$ for each $x \in U_k$. If there are a multiple number of such subsets, fix any of them. If U_k is not found for any key $k \in I_\kappa$, then output 1. Otherwise, proceed to the next step.
3. For each key k for which U_k exists, check if there are k', $k'' \in I_\kappa$ such that $E_{k'}(E_k(x)) = E_{k''}^{-1}(\mathsf{S}(x))$ for every $x \in U_k$. If there exists such a key, then output 0. Otherwise, output 1.

Analysis. Let $\mathsf{S} = \mathsf{TE}_{\mathbf{k}}[E]$ with a random key $\mathbf{k} = (k_1, k_2, k_3) \in I_\kappa^3$. In the ideal cipher model, $|E_{k_1}(S_0) \cap S_1|$ becomes a random variable that follows the hypergeometric distribution of mean r^2 and variance not greater than r^2. Therefore by Chevishev's inequality, the probability of $|E_{k_1}(S_0) \cap S_1| < \frac{r^2}{2}$ is at most $\frac{4}{r^2}$. Once $|E_{k_1}(S_0) \cap S_1| \geq \frac{r^2}{2}$, $\mathcal{D}$ moves to the next step, where $\mathcal{D}$ checks that $E_{k_2}(E_{k_1}(x)) = E_{k_3}^{-1}(\mathsf{S}(x))$ for every $x \in U_{k_1}$, and outputs 0. Therefore we have

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[\mathsf{TE}_{\mathbf{k}}[E], E] = 1\right] \leq \frac{4}{r^2}.$$

On the other hand, let $\mathsf{S} = P$ be a truly random permutation on I_n. For each key $\mathbf{k} = (k_1, k_2, k_3)$, the probability that $E_{k_2}(E_{k_1}(x)) = E_{k_3}^{-1}(\mathsf{S}(x))$ for every $x \in U_{k_1}$, assuming $|E_{k_1}(S_0) \cap S_1| \geq \frac{r^2}{2}$, is upper bounded by $1/(N)_{rN^{\frac{1}{2}}}$. Therefore we have

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[P, E] = 1\right] \geq 1 - \frac{1}{(N)_{rN^{\frac{1}{2}}}}.$$

We might set $S_0 = S_1$. Then $\mathcal{D}$ would make $2rKN^{\frac{1}{2}}$ queries to the underlying blockcipher and $rN^{\frac{1}{2}}$ queries to the outer permutation.

A.2 A Meet-in-the-Middle Attack of $2^{2\kappa}$ Query Complexity

A distinguisher $\mathcal{D}$, parameterized by $r > 0$, executes the following steps.

1. Fix a set $S_0 \subset I_n$ such that $|S_0| = r$.
   (a) For each $(k, k') \in I_\kappa^2$ and $x \in S_0$, make a query $E_{k'}(E_k(x))$.
   (b) For each $x \in S_0$, make a query $\mathsf{S}(x)$.
   (c) For each $k'' \in I_\kappa$ and $x \in S_0$, make a query $E_{k''}^{-1}(\mathsf{S}(x))$.
2. If there is a key $\mathbf{k} = (k, k', k'')$ such that $E_{k'}(E_k(x)) = E_{k''}^{-1}(\mathsf{S}(x))$ for every $x \in S_0$, then output 0. Otherwise, output 1.

Analysis. Suppose that $\mathsf{S} = \mathsf{TE}_{\mathbf{k}}[E]$ with a random key $\mathbf{k} = (k_1, k_2, k_3) \in I_\kappa^3$. Since $E_{k_2}(E_{k_1}(x)) = E_{k_3}^{-1}(\mathsf{S}(x))$ for every $x \in S_0$, we have

$$\mathbf{Pr}\left[\mathbf{k} \xleftarrow{\$} I_\kappa^3, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[\mathsf{TE}_{\mathbf{k}}[E], E] = 1\right] = 0.$$

On the other hand, let $\mathsf{S} = P$ be a truly random permutation on I_n. For each key $\mathbf{k} = (k, k', k'')$, the probability that $E_{k'}(E_k(x)) = E_{k''}^{-1}(\mathsf{S}(x))$ for every $x \in S_0$ is upper bounded by $1/(N)_r$. Therefore we have

$$\mathbf{Pr}\left[P \xleftarrow{\$} \mathcal{P}_n, E \xleftarrow{\$} BC(\kappa, n) : \mathcal{D}[P, E] = 1\right] \geq 1 - \frac{K^3}{(N)_r}.$$

In the first step, $\mathcal{D}$ makes $rK + rK^2$ queries to the underlying blockcipher and r queries to the outer permutation.