8-2011

General Construction of Chameleon All-but-one Trapdoor Functions

Shengli LIU

Junzuo LAI
Singapore Management University, ctlai.2010@smu.edu.sg

Robert H. DENG
Singapore Management University, robertdeng@smu.edu.sg

General Construction of Chameleon All-But-One Trapdoor Functions

Shengli Liu*
Department of Computer Science and Engineering
Shanghai Jiao Tong University
Shanghai 200240, China
slliu@sjtu.edu.cn

Junzuo Lai†, Robert H. Deng†
School of Information Systems,
Singapore Management University
Singapore 178902
{junzuolai,robertdeng}@smu.edu.sg

Abstract

Lossy trapdoor functions enable black-box construction of public key encryption (PKE) schemes secure against chosen-ciphertext attack [18]. Recently, a more efficient black-box construction of public key encryption was given in [13] with the help of chameleon all-but-one trapdoor functions (ABO-TDFs). In this paper, we propose a black-box construction for transforming any ABO-TDFs into chameleon ABO-TDFs with the help of chameleon hash functions. Instantiating the proposed general black-box construction of chameleon ABO-TDFs, we obtain the first chameleon ABO-TDFs based on the Decisional Diffie-Hellman (DDH) assumption.

Keywords: Lossy trapdoor functions, chameleon ABO-TDFs, Decisional Diffie-Hellman (DDH) assumption

1 Introduction

Lossy trapdoor functions (LTDFs) were first introduced by Peikert and Waters [18] and further studied in [6, 8, 7, 11, 19, 15]. LTDFs imply lots of fundamental cryptographic primitives, such as collision-resistant hash functions, oblivious transfer. LTDFs can be used to construct many cryptographic schemes, such as deterministic public-key encryption [2], encryption and commitments secure against selective opening attacks [1], non-interactive string commitments [17]. Most important of all, LTDFs enable black-box construction of public key encryption (PKE) schemes secure against chosen-ciphertext attack (CCA-secure PKE in short) [18].

A lossy trapdoor function is a public function f which works in two computationally indistinguishable modes, i.e., there is no efficient adversary who can tell which working mode f is in, given only the function description. In the first mode, it behaves like an injective trapdoor function and the input x can be recovered from $f(x)$ with the help of a trapdoor. In the second mode, f turns into a many-to-one function and it loses a significant amount of information about the input x. Hence, f in the latter mode is called a lossy function.

LTDFs were further extended to a richer abstraction called all-but-one trapdoor functions (ABO-TDFs), which can be constructed from LTDFs [18]. A collection of ABO-TDFs is associated with a branch set $\mathscr{B}$, and an ABO trapdoor function $g_b(\cdot)$ is uniquely determined by a function index g and a branch $b \in \mathscr{B}$. There exists a unique branch $b^* \in \mathscr{B}$ such that $g_{b^*}(\cdot)$ is a lossy function, while all $g_b(\cdot)$, $b \neq b^*$, are injective ones. However, the lossy branch b^* is computationally hidden by description of the function g. Freeman et al. [6] generalized the definition of ABO trapdoor functions by allowing possibly

Journal of Internet Services and Information Security, volume: 1, number: 2/3, pp. 74-88
*An extended abstract of this paper has been published in ProvSec 2011, LNCS 6890, Springer, 2011. This work is Funded by National Natural Science Foundation of China (No. 60873229, 61170229)
†Supported by A*STAR SERC Grant No. 102 101 0027 in Singapore.

many lossy branches instead of one. Let $\mathscr{B}^*$ be the set of lossy branches. Then, an ABO trapdoor function $g_b(\cdot)$ is injective if $b \in \mathscr{B}^*$ and lossy if $b \in \mathscr{B} \setminus \mathscr{B}^*$.

The black-box construction of CCA-secure PKE from LTDFs in [18] needs a collection of LTDFs, a collection of ABO-TDFs, a pair-wise independent family of hash functions, and a strongly unforgeable one-time signature scheme, where the set of verification keys is a subset of the branch set of the ABO collection.

The black-box construction of CCA-secure PKE from LTDFs was further improved in [13]. The improved construction is free of the strongly unforgeable one-time signature scheme, and employs a collision-resistant hash function instead. This results in ciphertexts of shorter length and encryption/decryption of greater efficiency. The price is that the collection of ABO-TDFs is replaced by a special kind of ABO-TDFs, namely chameleon ABO-TDFs. The notion of chameleon ABO-TDFs was first proposed in [13]. Chameleon ABO-TDFs behave just like ABO-TDFs except the following specific properties. Chameleon ABO-TDFs have two variables (u,v) to represent a branch. The chameleon property requires that given any half branch u, there exists an efficient algorithm to compute the other half branch v with a trapdoor such that (u,v) is a lossy branch.

Lai et al. [13] proposed a general construction of chameleon ABO-TDFs based on any CPA-secure homomorphic PKE scheme with some additional property, like the Damgård-Jurik encryption scheme [5]. This paper will further explore a more general construction of chameleon ABO-TDFs, which combines ABO-TDFs with chameleon hash functions.

1.1 Related Works

Since this paper focuses on the general construction of chameleon ABO-TDFs, we review here the existing constructions of LTDFs in the literature.

Peikert and Waters [18] showed how to construct LTDFs and ABO-TDFs based on the Decisional Diffie-Hellman (DDH) assumption and the worst-case hardness of lattice problem. Freeman et al. [6] presented LTDFs and ABO-TDFs based on the Quadratic Residuosity (QR) assumption, the Decisional Composite Residuosity (DCR) assumption and the d-Linear assumption. Hemenway and Ostrovsky [8] showed that smooth homomorphic hash proof systems imply LTDFs, and homomorphic encryption over cyclic groups also imply LTDFs [7]. Kiltz et al. [10] showed that the RSA trapdoor function is lossy under the ϕ-Hiding assumption of Cachin et al. [4]. Recently, Boyen and Waters [9] proposed two new discrete-log-type LTDFs based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption.

Rosen and Segev [19] showed that any collection of injective trapdoor functions that is secure under very natural correlated products can be used to construct a CCA-secure PKE scheme, and demonstrated that any collection of LTDFs with sufficient lossiness yields a collection of injective trapdoor functions that is secure under natural correlated products.

Mol and Yilek [15] extended the results of [18] and [19] and showed that only a non-negligible fraction of a single bit of lossiness is sufficient for building CCA-secure PKE schemes.

Recently, Kiltz et al. [11] introduced the notion of adaptive trapdoor functions (ATDFs) and tag-based adaptive trapdoor functions (TB-ATDFs). They showed that ATDFs and TB-ATDFs can be constructed directly by combining LTDFs and ABO-TDFs.

Lai et al. [13] introduced the notion of chameleon ABO-TDFs, presented a construction using CPA-secure homomorphic PKE schemes with some additional property and instantiated it with the Damgård-Jurik encryption scheme [5].

Our work is also related to chameleon hash functions, which are randomized collision-resistant hash functions with the additional property that given a trapdoor, one can efficiently generate collisions. Chameleon hash functions found various applications in chameleon signatures [12], online/offline signatures [20], transformations for strongly unforgeable signatures [21], etc. Recently, Mohassel presented a general construction of one-time signatures from chameleon hash functions [14].

1.2 Our Contribution

We design a black-box construction of chameleon ABO-TDFs and give some instantiations. Specifically,

1. We propose a black-box construction of chameleon ABO-TDFs by combining chameleon hash functions with ABO-TDFs with the help of a collision-resistant hash function family [16]. Let $\mathscr{Y}$ be the range of a collection of chameleon ABO-TDFs and $\mathscr{B}$ be the branch set of a collection of ABO-TDFs. With the help of a family $\mathscr{T}$ of collision-resistant hash functions from $\mathscr{Y}$ to $\mathscr{B}$, a collection of chameleon hash functions can be integrated into a collection of ABO-TDFs to result in a collection of chameleon ABO-TDFs.

2. Following our black-box construction of chameleon ABO-TDFs, we present the first chameleon ABO-TDFs based on the DDH assumption, which is the integration of the DL-based chameleon hash function [12] proposed by Krawczyk and Rabin and the ABO-TDFs [6] based on the DDH assumption. Recall that Lai et al. [13] instantiated their black-box construction of chameleon ABO-TDFs with the Damgård-Jurik (DJ) encryption scheme [5] to only obtain a collection of *almost-always* chameleon ABO-TDFs, based on the Decisional Composite Residuosity (DCR) problem. In the mean time, we can also get chameleon hash functions from the Damgård-Jurik encryption, which can convert the ABO-TDFs based on the DJ scheme into an *almost-always* chameleon ABO-TDFs, and the security of chameleon ABO-TDFs is also based on the DCR problem.

1.3 Organization of the Paper

The paper is organized as follows. In Section 2, we review the notion of chameleon hash functions and introduce the DL-based construction of chameleon hash functions proposed by Krawczyk and Rabin [12]. In Section 3, we review the notions of LTDFs, ABO-TDFs and chameleon ABO-TDFs. In Section 4, we present a black-box construction of chameleon ABO-TDFs by combining any chameleon hash function with ABO-TDFs with the help of a collision-resistant hash function family. In Section 5, we instantiate our black-box construction of chameleon ABO-TDFs to obtain the first chameleon ABO-TDFs based on the DDH assumption. Finally, Section 6 concludes the paper. Appendix shows how instantiate our black-box construction to obtain chameleon ABO-TDFs based on the DCR assumption.

1.4 Notation

Let $\mathscr{H}$ denote a set, $|\mathscr{H}|$ denote the cardinality of the set $\mathscr{H}$, and $h \stackrel{\$}{\leftarrow} \mathscr{H}$ denote sampling uniformly from the uniform distribution on set $\mathscr{H}$. If $A(\cdot)$ is an algorithm, then $a \stackrel{\$}{\leftarrow} A(\cdot)$ denotes running the algorithm and obtaining a as an output, which is distributed according to the internal randomness of $A(\cdot)$. A function $f(\lambda)$ is *negligible* if for every $c > 0$ there exists an λ_c such that $f(\lambda) < 1/\lambda^c$ for all $\lambda > \lambda_c$.

2 Chameleon Hash Functions

A family of chameleon hash functions is a set of randomized collision-resistant (CR) hash functions with an additional property that one can efficiently generate collisions with the help of a trapdoor.

Let $\mathscr{H}$ be a set of hash functions, with each function mapping $\mathscr{X}$ to $\mathscr{Y}$. Let $k \xleftarrow{\$} \mathbf{Hindex}(1^\kappa)$ denote the index generation algorithm. Each index $k \in \{1, 2, \cdots, |\mathscr{H}|\}$ determines a hash function $H_k \in \mathscr{H}$. Then, $\mathscr{H}$ is collision-resistant if for any polynomial-time adversary $\mathscr{A}$, its advantage $\mathbf{Adv}^{CR}_{\mathscr{H},\mathscr{A}}(1^\kappa)$, defined as

$$\mathbf{Adv}^{CR}_{\mathscr{H},\mathscr{A}}(1^\kappa) = \Pr\left[H_k(x_1) = H_k(x_2) : k \xleftarrow{\$} \mathbf{Hindex}(1^\kappa); x_1, x_2 \xleftarrow{\$} \mathscr{A}(H_k)\right],$$

is negligible.

A family $\mathscr{H}$ of chameleon hash functions [14], mapping $\mathscr{U} \times \mathscr{V}$ to $\mathscr{Y}$ consists of three (probabilistic) polynomial-time algorithms: the index generating algorithm, the evaluation algorithm and the inversion algorithm, satisfying *chameleon*, *uniformity* and *collision resistance* properties.

Index generation $\mathbf{Hgen}(1^\kappa)$: On input a security parameter 1^κ, the key generation algorithm outputs an index k of $\mathscr{H}$ and a trapdoor td. The index k determines a specific hash function $H_k : \mathscr{U} \times \mathscr{V} \to \mathscr{Y}$.

Evaluation $H_k(u, v)$: Each hash function $H_k \in \mathscr{H}$, takes $u \in \mathscr{U}$ and $v \in \mathscr{V}$ as inputs, and outputs a hash value in $\mathscr{Y}$.

Inversion $H_k^{-1}(u, v, td, u')$: On input $(u, v) \in \mathscr{U} \times \mathscr{V}$, the trapdoor td and $u' \in \mathscr{U}$, where $(k, td) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa)$, the algorithm H_k^{-1} outputs $v' \in \mathscr{V}$.

Chameleon property: Given a hash input (u, v) of H_k, the trapdoor td of H_k, and $u' \in \mathscr{U}$, the algorithm H_k^{-1} computes $v' \in \mathscr{V}$ such that $H_k(u, v) = H_k(u', v')$. More precisely,

$$\Pr\left[H_k(u, v) = H_k(u', v') : (k, td) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa), u, u' \in \mathscr{U}, v \in \mathscr{V}, v' \xleftarrow{\$} H_k^{-1}(u, v, td, u')\right] = 1. \quad (1)$$

Uniformity property: There exists a distribution $\mathscr{D}_v$ over $\mathscr{V}$, such that for all $u \in \mathscr{U}$, the distributions $(k, H_k(u, v))$ and (k, b) are computationally indistinguishable, where $(k, td) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa)$, v is chosen from $\mathscr{V}$ according to distribution $\mathscr{D}_v$, and $b \xleftarrow{\$} \mathscr{Y}$.

Collision resistance property: For all $H_k \in \mathscr{H}$, without the knowledge of the corresponding trapdoor, it is hard to find a collision, i.e., it is hard to compute two different pairs (u, v) and (u', v') such that $H_k(u, v) = H_k(u', v')$. More precisely, for any polynomial-time adversary $\mathscr{A}$, its advantage $\mathbf{Adv}^{CR}_{\mathscr{A},\mathscr{H}}(1^\kappa)$, defined as

$$\mathbf{Adv}^{CR}_{\mathscr{A},\mathscr{H}}(1^\kappa) = \Pr\left[H_k(u, v) = H_k(u', v') : (k, td) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa); (u, v, u', v') \xleftarrow{\$} \mathscr{A}(H_k)\right],$$

is negligible.

We generalize the definition of chameleon hash functions by allowing that Eq.(1) holds with overwhelming probability. Then, $\mathscr{H}$ is called a family of *almost-always* chameleon hash functions.

Below we introduce the Krawczyk and Rabin's construction [12] of chameleon hash functions based on the Discrete Logarithm (DL) assumption, which followed the chameleon commitment [3][9].

Construction 1. *[12] The DL-based chameleon hash functions.*

- **Index generation:** *The algorithm generates a group G of prime order p and picks a generator g of G. Randomly choose $x \in \mathbb{Z}_p^*$ and compute $y = g^x$. Return (G, p, g, y) as the hash index and $td = x$ as the trapdoor.*

- **Evaluation:** *Given a hash index input* (G, p, g, y) *and* $(u, v) \in \mathbb{Z}_p \times \mathbb{Z}_p$*, return*

$$H(u, v) = g^u \cdot y^v.$$

- **Inversion:** *Given a hash index* (G, p, g, y)*, a hash input* $(u, v) \in \mathbb{Z}_p \times \mathbb{Z}_p$*, the trapdoor* x*, and* $u' \in \mathbb{Z}_p$*, return* $v' = v + (u - u')x^{-1} \mod p$.

In the Appendix, we describe a construction of chameleon hash functions based on the Damgård-Jurik encryption scheme. The construction takes advantage of a cyclic group of the ciphertexts.

3 LTDFs, ABO-TDFs and Chameleon ABO-TDFs

In this section, we review the notions of LTDFs, ABO-TDFs and chameleon ABO-TDFs.

3.1 Lossy Trapdoor Functions

Informally, a collection of LTDFs [18] is a collection of functions with two computationally indistinguishable branches: an injective branch with a trapdoor and a lossy branch losing information about its input.

Definition 1. *(Lossy Trapdoor Functions). A collection of* (n, k)*-lossy trapdoor functions is a 3-tuple of (possibly probabilistic) polynomial-time algorithms* $(\mathsf{G}, \mathsf{F}, \mathsf{F}^{-1})$ *such that:*

1. **Sampling an injective function:** $\mathsf{G}(1^\kappa, \mathsf{injective})$ *outputs* (s, td) *where* s *is a function index and* td *is its trapdoor. The algorithm* $\mathsf{F}(s, \cdot)$ *computes a (deterministic) injective function* $f_s(\cdot)$ *over the domain* $\{0,1\}^n$*, and* $\mathsf{F}^{-1}(s, td, \cdot)$ *computes* $f_s^{-1}(\cdot)$.

2. **Sampling a lossy function:** $\mathsf{G}(1^\kappa, \mathsf{lossy})$ *outputs* s *where* s *is a function index. The algorithm* $\mathsf{F}(s, \cdot)$ *computes a (deterministic) function* $f_s(\cdot)$ *over the domain* $\{0,1\}^n$ *whose image has size at most* 2^{n-k}.

3. **Hard to distinguish injective from lossy:** *The ensembles* $\{s : (s, td) \leftarrow \mathsf{G}(1^\kappa, \mathsf{injective})\}_{\kappa \in \mathbb{N}}$ *and* $\{s : s \leftarrow \mathsf{G}(1^\kappa, \mathsf{lossy})\}_{\kappa \in \mathbb{N}}$ *are computationally indistinguishable.*

3.2 All-But-One Trapdoor Functions

The notion of ABO-TDFs, introduced by Peikert and Waters [18], is generalized by Freeman et al. [6]. In an ABO collection, each function has a branch set $\mathscr{B}$. There exists a subset $\mathscr{B}^* \subset \mathscr{B}$ such that all the branches in $\mathscr{B} \setminus \mathscr{B}^*$ make the function injective, while all branches in $\mathscr{B}^*$ make the function lossy. The set $\mathscr{B}^*$ is called the lossy branch set.

Definition 2. *(All-But-One Trapdoor Functions). A collection of* (n, k)*-all-but-one trapdoor functions is a 3-tuple of (possibly probabilistic) polynomial-time algorithms* $(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1})$ *such that:*

1. **Sampling a function:** *For any* $\kappa \in \mathbb{N}$ *and* $b^* \in \mathscr{B}$*,* $\mathsf{G}_{abo}(1^\kappa, b^*)$ *outputs* $(i', td, \mathscr{B}^*)$*, where* i' *is a function index,* td *is a trapdoor and* $\mathscr{B}^*$ *is a set of lossy branches with* $b^* \in \mathscr{B}^* \subset \mathscr{B}$.

2. **Evaluation of injective functions:** *Given* $(i', td, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$*, for all* $b \notin \mathscr{B}^*$*,* $\mathsf{F}_{abo}(i', b, \cdot)$ *computes a (deterministic) injective function* $f_{i',b}(\cdot)$ *over the domain* $\{0,1\}^n$*, and* $\mathsf{F}_{abo}^{-1}(i', b, td, \cdot)$ *computes* $f_{i',b}^{-1}(\cdot)$.

3. **Evaluation of lossy functions:** *Given* $(i', td, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$*, for all* $b \in \mathscr{B}^*$*,* $\mathsf{F}_{abo}(i', b, \cdot)$ *computes a (deterministic) function* $f_{i',b}(\cdot)$ *over the domain* $\{0,1\}^n$ *whose image has size at most* 2^{n-k}*.*

4. **Security:** *The ensembles* $\{i' : (i', td, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b_0^*)\}_{\kappa \in \mathbb{N}, b_0^* \in \mathscr{B}}$ *and* $\{i' : (i', td, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b_1^*)\}_{\kappa \in \mathbb{N}, b_1^* \in \mathscr{B}}$ *are computationally indistinguishable. Formally, Let* $\mathscr{A}$ *be a distinguisher and define its advantage as*

$$\mathsf{Adv}_{\mathscr{A}}^{\mathsf{ABO}}(1^\kappa) = \left| Pr\left[\beta = \beta' : \begin{array}{l} (b_0^*, b_1^*) \leftarrow \mathscr{A}(1^\kappa); \\ (i_0', td_0, B_0^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b_0^*); \\ (i_1', td_1, B_1^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b_1^*); \\ \beta \xleftarrow{\$} \{0,1\}; \beta' \leftarrow \mathscr{A}(i_\beta', b_0^*, b_1^*) \end{array} \right] - \frac{1}{2} \right|.$$

A collection of all-but-one trapdoor functions is secure, if $\mathsf{Adv}_{\mathscr{A}}^{\mathsf{CH\text{-}LI}}(1^\kappa)$ *is negligible for every PPT distinguisher* $\mathscr{A}$*.*

5. **Hidden lossy branches:** *This property means it is hard to find one-more lossy branch. More precisely, any probabilistic polynomial-time algorithm* $\mathscr{A}$ *that receives* (i', b) *as input, where* $(i', td, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$ *and* $b \xleftarrow{\$} \mathscr{B}^*$*, has only a negligible probability of outputting another lossy branch* $b' \in \mathscr{B}^* \backslash \{b\}$*.*

3.3 Chameleon ABO-TDFs

Chameleon ABO-TDFs is a specific kind of ABO-TDFs with two variable (u, v) as a branch [13]. The chameleon property requires that given any u, it is easy to compute a unique lossy branch (u, v) with the help of a trapdoor. The security requires that without the trapdoor, any lossy branch (u, v_0) and any branch (u, v_1) from the injective branch set are computationally indistinguishable. Meanwhile, given a lossy branch (u, v), it is impossible to generate another lossy branch (u', v') without the trapdoor.

Let $\mathbb{U} \times \mathbb{V} = \{\mathscr{U}_\kappa \times \mathscr{V}_\kappa\}_{\kappa \in \mathbb{N}}$ be a collection of sets whose elements represent the branches.

Definition 4 (Chameleon All-But-One Trapdoor Functions). A collection of (n, k)-chameleon all-but-one trapdoor functions is a 4-tuple of (possibly probabilistic) polynomial-time algorithms $(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}_{ch}^{-1}, \mathsf{CLB}_{ch})$ such that:

1. **Sampling a function:** For any $\kappa \in \mathbb{N}$, $\mathsf{G}_{ch}(1^\kappa)$ outputs (i, td, S) where i is a function index, td is the trapdoor and $S \subset \mathscr{U}_\kappa \times \mathscr{V}_\kappa$ is a set of lossy branches. Hereafter we will use $\mathscr{U} \times \mathscr{V}$ instead of $\mathscr{U}_\kappa \times \mathscr{V}_\kappa$ for simplicity.

2. **Evaluation of injective functions:** For any $(u, v) \in \mathscr{U} \times \mathscr{V}$, if $(u, v) \notin S$, where $(i, td, S) \leftarrow \mathsf{G}_{ch}(1^\kappa)$, then $\mathsf{F}_{ch}(i, u, v, \cdot)$ computes a (deterministic) injective function $g_{i,u,v}(\cdot)$ over the domain $\{0,1\}^n$, and $\mathsf{F}_{ch}^{-1}(i, u, v, td, \cdot)$ computes $g_{i,u,v}^{-1}(\cdot)$.

3. **Evaluation of lossy functions:** For any $(u, v) \in \mathscr{U} \times \mathscr{V}$, if $(u, v) \in S$, where $(i, td, S) \leftarrow \mathsf{G}_{ch}(1^\kappa)$, then $\mathsf{F}_{ch}(i, u, v, \cdot)$ computes a (deterministic) function $g_{i,u,v}(\cdot)$ over the domain $\{0,1\}^n$ whose image has size at most 2^{n-k}.

4. **Chameleon property:** there exists an algorithm CLB_{ch} which, on input the function index i, the trapdoor td and any $u \in \mathscr{U}$, computes a unique $v \in \mathscr{V}$ to result in a lossy branch (u, v). In formula, $v \leftarrow \mathsf{CLB}_{ch}(i, td, u)$ such that $(u, v) \in \mathscr{B}^*$.

5. **Security (1): Indistinguishability between lossy branches and injective branches.** It is hard to distinguish a lossy branch from an injective branch. Any probabilistic polynomial-time algorithm $\mathscr{A}$ that receives i as input, where $(i,td,S) \leftarrow \mathsf{G}_{ch}(1^\kappa)$, has only a negligible probability of distinguishing a pair $(u,v_0) \in S$ from $(u,v_1) \notin S$, even u is chosen by $\mathscr{A}$. Formally, Let $\mathscr{A}$ be a $\mathsf{CH\text{-}LI}$ distinguisher and define its advantage as

$$\mathsf{Adv}_{\mathscr{A}}^{\mathsf{CH\text{-}LI}}(1^\kappa) = \left| \Pr\left[\beta = \beta' : \begin{array}{l} (i,td,S) \leftarrow \mathsf{G}_{ch}(1^\kappa); u \leftarrow \mathscr{A}(i); \\ v_0 = \mathsf{CLB}_{ch}(i,td,u); v_1 \xleftarrow{\$} \mathscr{V}; \\ \beta \xleftarrow{\$} \{0,1\}; \beta' \leftarrow \mathscr{A}(i,u,v_\beta) \end{array} \right] - \frac{1}{2} \right|.$$

Given a collection of chameleon all-but-one trapdoor functions, it is hard to distinguish a lossy branch from an injective branch, if $\mathsf{Adv}_{\mathscr{A}}^{\mathsf{CH\text{-}LI}}(\cdot)$ is negligible for every PPT distinguisher $\mathscr{A}$.

6. **Security (2): Hidden lossy branches.** It is hard to find one-more lossy branch. Any probabilistic polynomial-time algorithm $\mathscr{A}$ that receives (i,u,v) as input, where $(i,td,S) \leftarrow \mathsf{G}_{ch}(1^\kappa)$ and $(u,v) \xleftarrow{\$} S$, has only a negligible probability of outputting a pair $(u',v') \in S \backslash \{(u,v)\}$.

In the above definition, if $\mathsf{F}_{ch}^{-1}(s,td,u,v,\cdot)$ inverts correctly on all values in the image of $g_{s,u,v}(\cdot)$ with $(u,v) \notin S$, and $\mathsf{CLB}_{ch}(s,td,u)$ outputs v such that $(u,v) \in S$, both *with overwhelming probability*, the collection is called *almost-always* chameleon ABO-TDFs.

4 General Construction of Chameleon ABO-TDFs

Given a family of ABO-TDFs $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$, we show how to transform it into a family of chameleon ABO-TDFs $\left(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}_{ch}^{-1}, \mathsf{CLB}_{ch}\right)$ with the help of a family of chameleon hash functions $\left(\mathsf{HGen}, H_k, H_k^{-1}\right)$ and possibly a family $\mathscr{T}$ of collision-resistant hash functions. The idea is the integration of the chameleon hash functions into the ABO-TDFs by replacing each branch of an ABO-TDFs with the branch's pre-image in the chameleon hash function. Let $\mathscr{Y}$ be the range of the chameleon hash functions, and $\mathscr{B}$ the branch set of the family of ABO-TDFs. When $\mathscr{Y} \nsubseteq \mathscr{B}$ we still need a family $\mathscr{T}$ of collision-resistant hash functions to map $\mathscr{Y}$ to $\mathscr{B}$.

In the construction of chameleon ABO-TDFs from ABO-TDFs, a family of chameleon hash functions is needed and their input (u,v) serves as the branches of the chameleon ABO-TDFs. With the help of a family of chameleon hash functions $\mathscr{H}$ and a family $\mathscr{T}$ of collision-resistant hash functions, all (u,v) are mapped into branches of an ABO-TDF i.e., $b = T\left(H_k(u,v)\right) \in \mathscr{B}$ and $H_k \in \mathscr{H}, T \xleftarrow{\$} \mathscr{T}$. The evaluation of the chameleon ABO-TDF behaves exactly as the ABO-TDF with $b = T\left(H_k(u,v)\right)$ as its branch input. Consequently, the set of lossy branches of the chameleon ABO-TDF is made up of the pre-images of all lossy branches of the ABO-TDF, i.e., $\{(u,v) : T\left(H_k(u,v)\right) = b^*, b^* \in \mathscr{B}^*\}$, with $\mathscr{B}^*$ the set of lossy branches of the ABO-TDFs. The chameleon property of the chameleon ABO-TDFs inherits from that of chameleon hash functions and the security of the chameleon ABO-TDFs inherits mainly from the security and the property of "hidden lossy branches" of the ABO-TDFs.

Construction 2. *Let $\left(\mathsf{HGen}, H_k, H_k^{-1}\right)$ describe a family of chameleon hash functions with $H_k : \mathscr{U} \times \mathscr{V} \to \mathscr{Y}$, and $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$ describe a family of (n,k)-ABO-TDFs with $\mathscr{B}$ the set of branches. Let $\mathscr{T}$ describe a family of collision-resistant hash functions mapping $\mathscr{Y}$ to $\mathscr{B}$. Then, a family of (n,k)-chameleon ABO-TDFs with branch set $\mathscr{U} \times \mathscr{V}$ can be constructed with the following algorithms $\left(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}_{ch}^{-1}, \mathsf{CLB}_{ch}\right)$.*

Sampling a function $\mathsf{G}_{ch}(1^\kappa)$: *Given a security parameter $\kappa \in \mathbb{N}$, $T \xleftarrow{\$} \mathscr{T}$, $(k, td_1) \xleftarrow{\$}$ **Hgen**(1^κ), $u^* \xleftarrow{\$} \mathscr{U}$, $v^* \xleftarrow{\$} \mathscr{V}$, compute $b^* = T(H_k(u^*, v^*))$. Sample a function from the ABO-TDFs with $(i', td_2, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$. Let $\mathscr{S} = \{(u, v) : T(H_k(u, v)) = b^*, b^* \in \mathscr{B}^*\}$. Return $i = (i', H_k, T)$ as the function index, $td = (td_1, (u^*, v^*), td_2)$ as the trapdoor, and $\mathscr{S}$ as the set of lossy branches.*

Evaluation of functions: *For all injective branch (u, v), define*

$$\mathsf{F}_{ch}(i, u, v, \cdot) := \mathsf{F}_{abo}(i', T(H_k(u, v)), \cdot).$$

Then, $\mathsf{F}_{ch}(i, u, v, \cdot)$ computes an injective function if $T(H_k(u, v)) \notin \mathscr{B}^*$, and a lossy function if $T(H_k(u, v)) \in \mathscr{B}^*$.

Inversion of injective functions: *On input a function index i, a branch $(u, v) \notin \mathscr{S}$, the trapdoor $td = (td_1, (u^*, v^*), td_2)$, and $z = \mathsf{F}_{ch}(i, u, v, x)$, the inverse function returns*

$$\mathsf{F}_{ch}^{-1}(i, u, v, td, z) := \mathsf{F}_{abo}^{-1}(i', T(H_k(u, v)), td_2, z).$$

Chameleon property(Computing a lossy branch): *On input the trapdoor $td = (td_1, (u^*, v^*), td_2)$, and $u' \xleftarrow{\$} \mathscr{U}$, CLB_{ch} computes $v' = H_k^{-1}(u^*, v^*, td_1, u')$, and return (u', v'). In formula,*

$$\mathsf{CLB}_{ch}(i, td, u') := H_k^{-1}(u^*, v^*, td_1, u').$$

Theorem 1. *The above general construction of chameleon ABO-TDFs satisfies (1) indistinguishability between lossy branches and injective branches; (2) hidden lossy branches.*

Proof. (1) Indistinguishability between lossy branches and injective branches: This property holds due to the uniformity property of the chameleon hash functions and the security of the ABO-TDFs. Suppose that there is an adversary $\mathscr{A}$ who is able to distinguish a lossy branch from an injective branch, then we can build another algorithm $\mathscr{E}$ who can break the security of the ABO-TDFs as follows.

$\mathscr{E}$ samples a chameleon hash with $(k, td_1) \xleftarrow{\$}$ **Hgen**(1^κ), chooses $u_0^*, u_1^* \xleftarrow{\$} \mathscr{U}$, and $v_0^*, v_1^* \xleftarrow{\$} \mathscr{V}, T \xleftarrow{\$} \mathscr{T}$. With overwhelming probability, $T(H_k(u_0^*, v_0^*)) \neq T(H_k(u_1^*, v_1^*))$. Let $b_0^* = T(H_k(u_0^*, v_0^*)), b_1^* = T(H_k(u_1^*, v_1^*))$. $\mathscr{E}$ sends (b_0^*, b_1^*) to a challenger $\mathscr{C}$. The challenger $\mathscr{C}$ samples two ABO-TDF functions i_0' and i_1' with $\mathsf{G}_{abo}(1^\kappa, b_0^*)$ and $\mathsf{G}_{abo}(1^\kappa, b_1^*)$, where i_0' is the first output of $\mathsf{G}_{abo}(1^\kappa, b_0^*)$ and i_1' the first output of $\mathsf{G}_{abo}(1^\kappa, b_1^*)$. The challenger $\mathscr{C}$ randomly chooses $\beta \xleftarrow{\$} \{0, 1\}$, and sends i_β' to $\mathscr{E}$. $\mathscr{E}$ will guess the value of β.

Now $\mathscr{E}$ simulates the game between $\mathscr{A}$ and a challenger $\mathscr{C}'$ by playing the role of the challenger $\mathscr{C}'$. $\mathscr{E}$ sends a function index $i = (i_\beta', H_k, T)$ to $\mathscr{A}$. $\mathscr{A}$ chooses a $u \in \mathscr{U}$ and gives u to $\mathscr{E}$. $\mathscr{E}$ computes $v_0 = H_k^{-1}(u_0^*, v_0^*, td_1, u)$ and $v_1 = H_k^{-1}(u_1^*, v_1^*, td_1, u)$. $\mathscr{E}$ chooses $\beta' \xleftarrow{\$} \{0, 1\}$ and sends $v_{\beta'}$ to $\mathscr{A}$ as a challenge.

If $\mathscr{A}$ responds with 0, then $\mathscr{E}$ sets β' as its guess of β, otherwise $\mathscr{E}$ sets $1 - \beta'$ as its guess of β.

It is easy to see that $i = (i_\beta', H_k, T)$ is a function index of a chameleon ABO-TDF, both (u_β^*, v_β^*) and (u, v_β) being lossy branches. Since $u_{1-\beta}^*, v_{1-\beta}^*$ are randomly chosen, $H_k(u_{1-\beta}^*, v_{1-\beta}^*)$ is also randomly distributed in $\mathscr{Y}$ due to the uniformity property of the chameleon hash function. Consequently, $v_{1-\beta} = H_k^{-1}(u_{1-\beta}^*, v_{1-\beta}^*, td_1, u)$ is also uniformly distributed in $\mathscr{V}$. Therefore $\mathscr{E}$ simulates the challenger $\mathscr{C}'$ perfectly in the game.

If $\mathscr{A}$'s response is 0, which means $(u, v_{\beta'})$ is also a lossy branch, hence $\mathscr{E}$ will have β' as its guess of β. If $\mathscr{A}$'s response is 1, which means $(u, v_{1-\beta'})$ is a lossy branch, hence $\mathscr{E}$ will have $1 - \beta'$ as its guess of β.

Consequently, $\mathscr{E}$ will have the same advantage in distinguishing a lossy branch from an injective branch of the chameleon ABO-TDF as $\mathscr{A}$ in distinguishing i'_0 and i'_1, the first outputs of $\mathsf{G}_{abo}(1^\kappa, b_0^*)$ and $\mathsf{G}_{abo}(1^\kappa, b_1^*)$ of the ABO-TDF.

(2) Hidden lossy branches: This property holds due to the collision resistance property of the chameleon hash functions, the property "hidden lossy branches" of the ABO-TDFs and the collision-resistant property of the hash function T. Now we analyze the probability of an adversary $\mathscr{A}$ winning the following game.

A challenger $\mathscr{C}$ samples a chameleon hash function with $(k, td_1) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa)$, chooses $u^* \xleftarrow{\$} \mathscr{U}$, $v^* \xleftarrow{\$} \mathscr{V}$, $T \xleftarrow{\$} \mathscr{T}$, and computes $b^* = T(H_k(u^*, v^*))$. $\mathscr{C}$ samples a function from the ABO-TDFs with $(i', td_2, \mathscr{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$. $\mathscr{C}$ sends the function index $i = (i', H_k, T)$ and the lossy branch (u^*, v^*) of the chameleon ABO-TDF to $\mathscr{A}$, and $\mathscr{A}$ responds with another lossy branch (u, v). Let $a = H_k(u, v)$ and $a^* = H_k(u^*, v^*)$. There are three cases.

- $a = a^*$: $\mathscr{A}$ finds a collision $H_k(u, v) = H_k(u^*, v^*)$ for H_k. It happens with negligible probability due to the collision resistance property of H_k.
- $a \neq a^*$ but $T(a) = T(a^*)$: The uniformity property of the chameleon hash function H_k implies that $a^* = H_k(u^*, v^*)$ is randomly distributed in $\mathscr{Y}$. The collision-resistant property of the family $\mathscr{T}$ of hash functions guarantees that the probability of $T(a) = T(a^*)$ is negligible.
- $a \neq a^*$ and $T(a) \neq T(a^*)$: The branch (u^*, v^*) is lossy, hence $b^* = T(H_k(u^*, v^*)) = T(a^*)$ is a lossy branch of the ABO-TDF $F_{abo}(i', b^*, \cdot)$. If $\mathscr{A}$ finds another lossy branch (u, v) for the chameleon ABO-TDF, then $b = T(H_k(u, v)) = T(a)$ is also another lossy branch of the ABO-TDF $F_{abo}(i', b^*, \cdot)$. According to the property of "hidden lossy branches" of ABO-TDFs, this probability is negligible.

Consequently, $\mathscr{A}$ succeeds in outputing another lossy branch (u, v) with negligible probability. Q.E.D. $\square$

Note. When the range of the chameleon hash functions falls into the branch set of the ABO-TDFs, i.e, $\mathscr{Y} \subseteq \mathscr{B}$, the family $\mathscr{T}$ of collision-resistant hash functions can be omitted in the construction.

5 Instantiations of Chameleon ABO-TDFs Based on the DDH Assumption

In [6], Freeman et al. proposed a construction of ABO-TDFs $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$ based on the DDH assumption. Let G be a group of prime order p with g its generator. Let $\mathrm{Rk}_1(\mathbb{F}_p)$ be the set of $n \times n$ matrices over $\mathbb{F}_p$ of rank 1. Given a vector $\vec{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_p^n$, define $g^{\vec{x}} := (g^{x_1}, g^{x_2}, \ldots, g^{x_n}) \in G^n$. Given an $n \times n$ matrix $M = (a_{ij})$ over $\mathbb{F}_p$ and $g \in G$, define the $n \times n$ matrix $g^M := (g^{a_{ij}})$ over G. Given an $n \times n$ matrix $M = (a_{ij})$ over $\mathbb{F}_p$ and a column vector $\boldsymbol{g} = (g_1, g_2, \ldots, g_n) \in G^n$, define

$$\boldsymbol{g}^M = \left(\prod_{j=1}^{n} g_j^{a_{1j}}, \prod_{j=1}^{n} g_j^{a_{2j}}, \ldots, \prod_{j=1}^{n} g_j^{a_{nj}} \right).$$

Given a matrix $\boldsymbol{S} = (g_{ij}) \in G^{n \times n}$ and a column vector $\vec{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{F}_p^n$, define

$$\boldsymbol{S}^{\vec{x}} := \left(\prod_{j=1}^{n} g_{1j}^{x_j}, \prod_{j=1}^{n} g_{2j}^{x_j}, \ldots, \prod_{j=1}^{n} g_{nj}^{x_j} \right).$$

It follows that $\left(g^M\right)^{\vec{x}} = (g^{\vec{x}})^M = g^{(M\vec{x})}$.

Construction 3. *The ABO-TDFs based on the DDH assumption in [6] is defined as* $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$.

- $\mathsf{G}_{abo}(1^\kappa, b^*)$: *On input the security parameter κ, choose $0 < \varepsilon < 1$. Let $n = \kappa$. Choose a random branch $b^* \in \mathscr{B} = \{0, 1, \cdots, 2^{\lfloor \varepsilon n \rfloor}\}$. Choose an $\lceil \varepsilon n \rceil$-bit prime number p and a group G of order p with generator g. Randomly choose a matrix $A \xleftarrow{\$} Rk_1(\mathbb{F}_p^{n\times n})$. Compute the matrix $M = A - b^* I_n \in \mathbb{F}_p^{n\times n}$ and $\boldsymbol{S} = g^M \in G^{n\times n}$. Return $(\boldsymbol{S}, g)$ as the function index, M as the trapdoor, and $\mathscr{B}^* = \{b^*, b^* - Tr(A)\}$ as the set of lossy branches.*

- $\mathsf{F}_{abo}(\boldsymbol{S}, g, b, \vec{x})$: *on input a function index $(\boldsymbol{S}, g)$, a branch $b \in \mathscr{B}$ and $\vec{x} = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$. Return $\boldsymbol{S}^{\vec{x}} \odot g^{b\vec{x}}$. Here $\odot$ denote the component-wise product of elements of G^n. If $b = b^*$ or $b^* - Tr(A)$, then function $\mathsf{F}_{abo}(\boldsymbol{S}, g, b, x) = \boldsymbol{S}^{\vec{x}} \odot g^{b\vec{x}} = g^{M + b^* I_n}$ or $g^{M+(b^* - Tr(A) I_n)}$, and the matrix $M + b^* I_n$ (with respect to $M + (b^* - Tr(A) I_n)$) is of rank 1. In this case, the image of the function is restricted in a subgroup of G^n of size $p < 2^{\varepsilon n}$, hence is lossy. Otherwise, A is of full rank and the function is injective.*

- $\mathsf{F}_{abo}^{-1}(\boldsymbol{S}, g, b, M, \boldsymbol{Z})$: *on input a function index $(\boldsymbol{S}, g)$, an injective branch b, the trapdoor M, an evaluation $\boldsymbol{Z} = \mathsf{F}_{abo}(\boldsymbol{S}, g, b, x) \in G^{n\times n}$, the inverse function computes $\boldsymbol{h} = (h_1, h_2, \cdots, h_n) = \boldsymbol{g}^{(M+bI_n)^{-1}}$ and $x_i = \log_g(h_i)$ with $i = 1, 2, \ldots, n$ and returns $\vec{x} = (x_1, x_2, \ldots, x_n)$.*

Now, using the DL-based chameleon hash function [12] proposed by Krawczyk and Rabin and Freeman et al.'s DDH-based ABO-TDFs, we instantiate our black-box construction of chameleon ABO-TDFs to obtain the first chameleon ABO-TDFs based on the DDH assumption.

Construction 4. *The integration of Construction 1 to Construction 3 gives a family of chameleon-ABO-TDFs with* $\left(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}_{ch}^{-1}, \mathsf{CLB}_{ch}\right)$.

- $\mathsf{G}_{ch}(1^\kappa)$: *On input the security parameter κ, choose $0 < \varepsilon < 1$. Let $n = \kappa$. Choose a $\lceil \varepsilon n \rceil$-bit prime number p and a group G of order p with its generator g. Choose $T \in \mathscr{T}$, with $\mathscr{T}$ a family of collision-resistant hash functions and $T : G \to \mathbb{Z}_p$.*

  Choose $x \xleftarrow{\$} \mathbb{Z}_p$ and compute $y = g^x$. A chameleon hash function $H : \mathbb{Z}_p \times \mathbb{Z}_p \to G$ is defined as $H(u, v) = g^u \cdot y^v$ with x being its trapdoor.

  Choose a random branch (u^*, v^*) from the branch set $\mathbb{Z}_p \times \mathbb{Z}_p$. Compute $b^* = T(H(u^*, v^*)) = T\left(g^{u^*} \cdot (g^x)^{v^*}\right)$.

  Randomly choose a matrix $A \xleftarrow{\$} \mathrm{Rk}_1(\mathbb{F}_p^{n\times n})$. Compute the matrix $M = A - b^* I_n \in \mathbb{F}_p^{n\times n}$ and $\boldsymbol{S} = g^M \in G^{n\times n}$.

  Return $(\boldsymbol{S}, g, y)$ as the function index, (M, x, u^*, v^*) as the trapdoor, and

  $\mathscr{S} = \{(u, v) : (u, v) \in \mathbb{Z}_p \times \mathbb{Z}_p, T(g^u \cdot y^v) = \{b^*, b^* - Tr(A)\}\}$ as the set of lossy branches.

- $\mathsf{F}_{ch}((\boldsymbol{S}, g, y), (u, v), x)$: *On input a function index $(\boldsymbol{S}, g, y)$, a branch $(u, v) \in \mathbb{Z}_p \times \mathbb{Z}_p$ and $x \in \{0,1\}^n$, compute $b = T(g^u \cdot y^v)$. Return $\boldsymbol{S}^{\vec{x}} \odot g^{b\vec{x}}$.*

  If $(u, v) \in \mathscr{S}$, the function is reduced to be a lossy function of the ABO-TDFs in Construction 3, otherwise it is just an injective function of the ABO-TDFs in Construction 3.

- $\mathsf{F}_{ch}^{-1}((\boldsymbol{S},g,y),(u,v),(M,x),\boldsymbol{Z})$: *On input a function index* $(\boldsymbol{S},g)$*, an injective branch* (u,v)*, the trapdoor* (M,x)*, and* $\boldsymbol{Z} = \mathsf{F}_{ch}((\boldsymbol{S},g),(u,v),x)$*, compute* $b = T(g^u \cdot y^v)$*, the inverse function returns* $\mathsf{F}_{abo}^{-1}(\boldsymbol{S},g,b,M,\boldsymbol{Z})$*, i.e., compute* $\boldsymbol{h} = (h_1,h_2,\cdots,h_n) = \boldsymbol{g}^{(M+bI_n)^{-1}}$ *and* $x_i = \log_g(h_i)$ *with* $i = 1,2,\ldots,n$ *and returns* $\vec{x} = (x_1,x_2,\ldots,x_n)$*.*

- $\mathsf{CLB}_{ch}((M,x,u^*,v^*),u')$: *On input the trapdoor* (M,x,u^*,v^*)*, and* $u' \xleftarrow{\$} \mathbb{Z}_p$*, return the output of the inverse function of the chameleon function, i.e.,*

$$v' = H^{-1}(x,u^*,v^*) = v^* + (u^* - u')x^{-1} \mod p.$$

Since Construction 1 is the DL-based chameleon hash function [12] and Construction 3 is the DDH-based ABO-TDFs, we have the following claim.

Claim 1. *Construction 4 gives a family of chameleon-ABO-TDFs based on the DDH assumption.*

Freeman et al. also proposed a construction of ABO-TDFs based on the DCR assumption in [6]. The chameleon hash functions of Construction 5 can help it change to chameleon ABO-TDFs, which performs as fast as the chameleon ABO-TDFs in [13], see the Appendix.

6 Conclusion

In this paper, we showed a black-box construction of chameleon ABO-TDFs, which can transform any ABO-TDFs into chameleon ABO-TDFs with the help of chameleon hash functions, and possibly some collision-resistant hash functions. We instantiated the construction with the existing ABO-TDFs and chameleon hash functions to obtain the first chameleon ABO-TDFs based on the DDH assumption. According to [13], these chameleon ABO-TDFs imply more efficient black-box construction of CCA-secure PKE in the standard model than that in [18].

References

[1] M. Bellare, D. Hofheinz, and S. Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *Proc. of the 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'09), Cologne, Germany, LNCS*, volume 5479, pages 1–35. Springer-Verlag, April 2009.

[2] A. Boldyreva, S. Fehr, and A. O'Neill. On notions of security for deterministic encryption, and efficient constructions without random oracles. In *Proc. of the 28th Annual International Cryptology Conference (CRYPTO'08), Santa Barbara, California, USA, LNCS*, volume 5157, pages 335–359. Springer-Verlag, August 2008.

[3] G. Brassard, D. Chaum, and C. Crepeau. Minimum disclosure proofs of knowledge. *JCSS*, 37(2):156–189, October 1988.

[4] C. Cachin, S. Micali, and M. Stadler. Computationally private information retrieval with polylogarithmic communication. In *Proc. of the 18th International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'99), Prague, Czech Republic, LNCS*, volume 1592, pages 402–414. Springer-Verlag, May 1999.

[5] I. Damgård and M. Jurik. A generalisation, a simplification and some applications of paillier's probabilistic public-key system. In *Proc. of the 4th International Workshop on Practice and Theory in Public Key Cryptosystems (PKC'01), Cheju Island, Korea, LNCS*, volume 1992, pages 119–136. Springer-Verlag, February 2001.

[6] D. M. Freeman, O. Goldreich, E. Kiltz, A. Rosen, and G. Segev. More constructions of lossy and correlation-secure trapdoor functions. In *Proc. of the 13th International Workshop on Practice and Theory in Public Key Cryptosystems (PKC'10), Paris, France, LNCS*, volume 6056, pages 279–295. Springer-Verlag, May 2010.

[7] B. Hemenway and R. Ostrovsky. Homomorphic encryption over cyclic groups implies chosen-ciphertext security. Technical Report 99, Cryptology ePrint Archive, Report 2010/099, 2010.

[8] Brett Hemenway and Rafail Ostrovsky. Lossy trapdoor functions from smooth homomorphic hash proof systems. *Electronic Colloquium on Computational Complexity (ECCC)*, 16(127):127–127, November 2009.

[9] S. A. Kurtz J. Boyar and M. W. Krentel. A discrete logarithm implementation of perfiect zero-knowledge blobs. *Journal of Cryptology*, 2(2):63–76, January 1990.

[10] E. Kiltz, A. O'Neill, and A. Smith. Lossiness of RSA and the chosen-plaintext security of OAEP without random oracles. Manuscript, 2009.

[11] Eike Kiltz, Payman Mohassel, and Adam O'Neill. Adaptive trapdoor functions and chosen-ciphertext security. In *Proc. of the 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'10), French Riviera, LNCS*, volume 6110, pages 673–692. Springer-Verlag, May 2010.

[12] Hugo Krawczyk and Tal Rabin. Chameleon signatures. In *Proc. of the Network and Distributed System Security Symposium (NDSS'00), San Diego, California, USA*, pages 143–154, February 2000.

[13] Junzuo Lai, Robert H. Deng, and Shengli Liu. Chameleon all-but-one TDFs and their application to chosen-ciphertext security. In *Proc. of the 14th International Conference on Practice and Theory in Public Key Cryptography (PKC'11), Taormina, Italy, LNCS*, volume 6571, pages 228–245. Springer-Verlag, March 2011.

[14] Payman Mohassel. One-time signatures and chameleon hash functions. In *Proc. of the Selected Areas in Cryptography 17th International Workshop (SAC'10), Waterloo, Ontario, Canada, LNCS*, volume 6544, pages 302–319. Springer-Verlag, August 2010.

[15] P. Mol and S. Yilek. Chosen-ciphertext security from slightly lossy trapdoor functions. Technical Report 524, Cryptology ePrint Archive, Report 2009/524, 2009.

[16] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *Proc. of the 21st Annual ACM Symposium on Theory of Computing (STOC'89), Seattle, Washigton, USA*, pages 33–43. ACM Press, May 1989.

[17] Ryo Nishimaki, Eiichiro Fujisaki, and Keisuke Tanaka. Efficient non-interactive universally composable string-commitment schemes. In *Proc. of the 3rd International Conference on Provable Security (ProvSec'09), Guangzhou, China*, pages 3–18. IEEE, November 2009.

[18] Chris Peikert and Brent Waters. Lossy trapdoor functions and their applications. In *Proc. of the 40th Annual ACM Symposium on Theory of Computing (STOC'08), Victoria, British Columbia, Canada*, pages 187–196. ACM Press, May 2008.

[19] Alon Rosen and Gil Segev. Chosen-ciphertext security via correlated products. In *Proc. of the 6th Theory of Cryptography Conference (TCC'09), San Francisco, CA, USA, LNCS*, volume 5444, pages 419–436. Springer-Verlag, March 2009.

[20] Adi Shamir and Yael Tauman. Improved online/offline signature schemes. In *Proc. of the 21st Annual International Cryptology Conference (CRYPTO'01), Santa Barbara, California, USA, LNCS*, volume 2139, pages 355–367. Springer-Verlag, August 2001.

[21] Ron Steinfeld, Josef Pieprzyk, and Huaxiong Wang. How to strengthen any weakly unforgeable signature into a strongly unforgeable signature. In *Proc. of the Cryptographers' Track at the RSA Conference 2007 (CT-RSA'07), San Francisco, CA, USA, LNCS*, volume 4377, pages 357–371. Springer-Verlag, February 2007.

A Chameleon ABO-TDFs Based on the DCR Assumption

Here, we describe a construction of chameleon hash functions and a construction of ABO-TDFs $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$ proposed by Freeman et al. [6], both of which are based on the Damgård-Jurik (DJ) encryption scheme. Then, we will change the ABO-TDFs into chameleon ABO-TDFs, according to the black-box construction of chameleon ABO-TDFs.

We first describe the Damgård-Jurik (DJ) encryption scheme [5] which relies on the following theorem.

Theorem 2. *[5] For any admissible N such that $N = PQ$, P,Q odd primes and $\gcd(N,\phi(N)) = 1$, and $s < \min\{P,Q\}$, the map $\psi_s : \mathbb{Z}_{N^s} \times \mathbb{Z}_N^* \to \mathbb{Z}_{N^{s+1}}^*$ defined by $\psi_s(x,r) = (1+N)^x r^{N^s} \mod N^{s+1}$ is an isomorphism, where*

$$\psi_s(x_1 + x_2 \mod N^s, r_1 r_2 \mod N) = \psi_s(x_1,r_1) \cdot \psi_s(x_2,r_2).$$

Moreover, $\psi_s(x,r)$ can be inverted to recover (x,r) in polynomial time given $\lambda(N) = \mathrm{lcm}(P-1,Q-1)$.

Below describes the Damgård-Jurik encryption scheme.

DJKg(1^κ)**:** On input the security parameter κ, choose an admissible κ-bit modulus $N = PQ$, and $s < \min\{P,Q\}$ and return the public key $\mathsf{PK} = (N,s)$, and the secret key $\mathsf{SK} = \lambda(N)$.

DJEnc$(\mathbf{PK}, m)$**:** On input a plaintext $m \in \mathbb{Z}_{N^s}$ and the public key $\mathsf{PK} = (N,s)$, choose a random $r \in \mathbb{Z}_N^*$, and return $C = (1+N)^m r^{N^s} \mod N^{s+1}$.

DJDec$(C, \mathbf{SK})$**:** On input a ciphertext $C \in \mathbb{Z}_{N^{s+1}}^*$ and the secret key $\mathsf{SK} = \lambda(N)$, the inversion algorithm in Theorem 2 is used to compute $(m,r) \leftarrow \psi_s^{-1}(C)$. Return m.

The DJ encryption scheme is a homomorphic PKE scheme with CPA secuirty, based on the DCR assumption. We can construct chameleon hash functions from the DJ scheme, following the line of Construction 1.

Construction 5.
- **Index generation:** *Generate a public/private key pair $(PK, SK) \leftarrow DJKg(1^\kappa)$. Randomly choose $x \in \mathscr{M}$ and compute*

$$C_1 = DJEnc(PK, 1), C_2 = C_1^x.$$

Return (PK, C_1, C_2) as the hash index and $td = (SK, x)$ as the trapdoor.

- **Evaluation:** *Given a hash index (PK, C_1, C_2) and $(u,v) \in \mathscr{M} \times \mathscr{M}$, return*

$$H(u,v) = (C_1)^u \cdot (C_2)^v = C_1^{u+x\cdot v}.$$

- **Inversion:** *Given a hash index (PK, C_1, C_2), $(u,v) \in \mathscr{M} \times \mathscr{M}$, the trapdoor (SK, x), and $u' \in \mathscr{M}$, return*

$$v' = v + (u - u')x^{-1} \mod N^s.$$

Claim 2. *Construction 5 gives a family of almost-always chameleon hash functions.*

Proof. The plaintext space $\mathscr{M} = Z_{N^s}$ is a ring. The homomorphic property of the DJ scheme implies that $(\mathsf{DJEnc}(\mathsf{PK},1)^{\mathscr{M}}, \cdot)$ is a cyclic group of order $|\mathscr{M}|$ with $\mathsf{DJEnc}(\mathsf{PK},1)$ as a generator, and this group is a subgroup of $(Z_{N^{s+1}}^*, \cdot)$. The DL assumption applies to the cyclic group $(\mathsf{DJEnc}(\mathsf{PK},1)^{\mathscr{M}}, \cdot)$. The remaining proof follow that in [12].

Since any element in Z_{N^s} has multiplicative inverse with overwhelming probability, the construction family is *almost-always* chameleon hash functions. Q.E.D. □

Now we introduce a construction of ABO-TDFs $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$ based on the DJ scheme proposed by Freeman et al. [6].

Construction 6. *The ABO-TDFs* $\left(G_{abo}, F_{abo}, F_{abo}^{-1}\right)$ *based on the DJ scheme is defined as follows.*

- $G_{abo}(1^\kappa, b^*)$: *Let* $n = \kappa$. $(PK, SK) \xleftarrow{\$} DJKg(1^\kappa)$ *with* $PK = (N,s)$ *and* $SK = \lambda(N)$. *Choose a random branch* $b^* \in \mathscr{B} = \{0,1\}^{n/2}$, *and compute* $C = DJEnc(PK, -b^*)$. *Return the function index* (PK, C), *the trapdoor* (SK, b^*), *and the lossy branch set* $\mathscr{B}^* = \{b^*\}$.
- $F_{abo}(PK, C, b, x)$: *On input a function index* (PK, C), *a branch* $b \in \mathscr{B}$ *and* $x \in \mathbb{Z}_{N^s}$. *Return* $C^x \cdot DJEnc(PK, bx)$. *Due to the homomorphic property of the DJ scheme,* $C^x \cdot DJEnc(PK, x\cdot b) = DJEnc(PK, x\cdot(b-b^*))$. *When* $b = b^*$, *then function is reduced to be* $DJEnc(PK, 0)$, *which is lossy. Otherwise, it is injective.*
- $F_{abo}^{-1}(PK, C, b, SK, b^*, z)$: *on input a function index* (PK, C), *the branch input* $b \neq b^*$, *the trapdoor* (SK, b^*), *and an evaluation* $z = F_{abo}(PK, C, b, x)$, *the inverse function returns* $(b-b^*)^{-1} \cdot DJDec(z, SK)$.

Both Construction 5 to Construction 6 are based on the DJ encryption scheme, then the integration of two constructions results in a family of chameleon ABO-TDFs according to Theorem 1.

Construction 7. *The combination of Construction 5 to Construction 6 also gives a family of almost-always chameleon ABO-TDFs given by* $\left(G_{ch}, F_{ch}, F_{ch}^{-1}, CLB_{ch}\right)$.

- $G_{ch}(1^\kappa)$: *Let* $n = \kappa$. $(PK, SK) \xleftarrow{\$} DJKg(1^\kappa)$ *with* $PK = (N,s)$ *and* $SK = \lambda(N)$, *and* $T \in \mathscr{T}$, *with* $\mathscr{T}$ *a family of collision-resistant hash functions and* $T : \mathbb{Z}_{N^{s+1}}^* \to \{0,1\}^{n/2}$.

  Randomly choose $x \in \mathbb{Z}_{N^s}$ *and compute*

  $$C_1 = DJEnc(PK, 1), \quad C_2 = C_1^x.$$

  The hash index (C_1, C_2) *uniquely determines a chameleon hash function defined as* $H(u,v) = (C_1)^u \cdot (C_2)^v$.

  Randomly choose $(u^*, v^*) \xleftarrow{\$} \mathbb{Z}_{N^s} \times \mathbb{Z}_{N^s}$ *and compute* $b^* = T(H(u^*, v^*)) = T\left((C_1)^{u^*} \cdot (C_2)^{v^*}\right)$. *Compute* $C = C_1^{-b^*}$.

  Return (PK, C_1, C_2, C) *as the function index,* $(SK, (u^*, v^*))$ *as the trapdoor, and* $\mathscr{S} = \{(u,v) : (u,v) \in \mathbb{Z}_{N^s} \times \mathbb{Z}_{N^s}, T((C_1)^u \cdot (C_2)^v) = b^*\}$ *as the set of lossy branches.*
- $F_{ch}((PK, C_1, C_2, C), (u,v), x)$: *on input a function index* (PK, C_1, C_2, C), *a branch* $(u,v) \in \mathbb{Z}_N \times \mathbb{Z}_N$ *and* $x \in \mathbb{Z}_{N^s}$, *compute* $b = T((C_1)^u \cdot (C_2)^v)$ *Return* $C^x \cdot DJEnc(PK, bx)$. *Due to the homomorphic property of the DJ scheme,* $C^x \cdot DJEnc(PK, bx) = DJEnc(PK, (b - b*)x)$. *When* $(u,v) \in \mathscr{S}$, *then function is reduced to be* $DJEnc(PK, 0)$, *which is lossy. Otherwise, it is injective.*
- $F_{ch}^{-1}((PK, C_1, C_2, C), SK, (u^*, v^*), (u,v), z)$: *on input a function index* (PK, C_1, C_2, C), *the trapdoor* $(SK, (u^*, v^*))$, *a branch* $(u,v) \notin \mathscr{S}$, *and* $z = F_{ch}((PK, C_1, C_2, C), (u,v), x)$, *the inverse function returns* $x = (b-b^*)^{-1} \cdot DJDec(z, SK) \mod N^s$, *where* $b = T((C_1)^u \cdot (C_2)^v)$ *and* $b^* = T(H(u^*, v^*)) = T\left((C_1)^{u^*} \cdot (C_2)^{v^*}\right)$. *Since* $b, b^* \in \{0,1\}^{n/2}$, *we know that* $\gcd(b-b^*, N^s) = 1$, *which ensures the existence of* $(b-b^*)^{-1}$.
- $CLB_{ch}(SK, (u^*, v^*), u')$: *On input the trapdoor* $(SK, (u^*, v^*))$, *and* $u' \xleftarrow{\$} \mathbb{Z}_{N^s}$, *return the output of the inverse function of the chameleon function, i.e.,* $v' = v^* + (u^* - u')x^{-1} \mod N^s$.

Claim 3. *Construction 7 gives a family of* almost always *chameleon-ABO-TDFs based on the DCR assumption.*

The family of chameleon ABO-TDFs from Construction 7 and the family proposed by Lai et al. are both based on the DJ scheme, hence based on the DCR assumption. The two families almost share the same efficiency.

Shengli Liu got her B.S., M.S. and Ph.D degrees from Xidian University in 1995, 1998, 2000 respectively. She got another Ph..D degree from Technische Universiteit Eindhoven on Feb 26, 2002. She joined Department of Computer Science and Engineering, Shanghai Jiao Tong University in 2002. She is now a professor and her research interests include Unconditional Security and provable security in public key cryptography.

Junzuo Lai received the B.S. and M.S. degrees in computer science and technology from Jingdezhen Ceramic Institute in 2002 and 2005, respectively. He got his Ph.D. degree fromShanghai Jiao Tong University in 2010. He is curently a research fellow at Singapore Management University. His research interests include cryptography and information security.

Robert H. Deng received his Bachelor's degree from National University of Defense Technology, China, M.Sc. and Ph.D. degrees from the Illinois Institute of Technology, USA. He has been with the Singapore Management University since 2004, and is currently professor, associate dean for Faculty & Research, School of Information Systems. Prior to this, he was principal scientist and manager of Infocomm Security Department, Institute for Infocomm Research, Singapore. He has 26 patents and more than 200 technical publications in international conferences and journals in the areas of computer networks, network security and information security. He has served as general chair, program committee chair and program committee member of numerous international conferences. He is an associate editor of the IEEE Transactions on Information Forensics and Security, associate editor of Security and Communication Networks Journal (John Wiley), and member of Editorial Board of Journal of Computer Science and Technology (the Chinese Academy of Sciences). He received the University Outstanding Researcher Award from the National University of Singapore in 1999 and the Lee Kuan Yew Fellow for Research Excellence from the Singapore Management University in 2006.