10-2011

General construction of chameleon all-but-one trapdoor functions

Shengli LIU
Shanghai Jiaotong University

Junzuo LAI
Chinese Academy of Sciences

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

DOI: https://doi.org/10.1007/978-3-642-24316-5_18

Provable Security: 5th International Conference, ProvSec 2011, Xi'an, China, October 16-18: Proceedings
http://dx.doi.org/10.1007/978-3-642-24316-5_18

General Construction of Chameleon All-But-One Trapdoor Functions

Shengli Liu[1,2], Junzuo Lai[3,⋆], and Robert H. Deng[3]

[1] Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai 200240, China
slliu@sjtu.edu.cn
[2] State Key Laboratory of Information Security,
Institute of Software, Chinese Academy of Sciences
[3] 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 [12] 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 can obtain the first chameleon ABO-TDFs based on the Decisional Diffie-Hellman (DDH) assumption.

Keywords: Lossy Trapdoor Functions, Chameleon All-But-One Trapdoor Functions, Chameleon Hash Functions.

1 Introduction

Lossy trapdoor functions (LTDFs) were first introduced by Peikert and Waters [18] and further studied in [6,7,8,9,19,14]. LTDFs imply lots of fundamental cryptographic primitives, such as collision-resistant hash functions [18], oblivious transfer [17]. 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 [16]. 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

⋆ Corresponding author.

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 $\mathcal{B}$, and an ABO trapdoor function $g_b(\cdot)$ is uniquely determined by a function index g and a branch $b \in \mathcal{B}$. There exists a unique branch $b^* \in \mathcal{B}$ 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 many lossy branches instead of one. Let $\mathcal{B}^*$ be the set of lossy branches. Then, an ABO trapdoor function $g_b(\cdot)$ is injective if $b \in \mathcal{B}^*$ and lossy if $b \in \mathcal{B} \setminus \mathcal{B}^*$.

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 [12]. 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 [12]. 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. [12] 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 [7] showed that smooth homomorphic hash proof systems imply LTDFs, and homomorphic encryption over cyclic groups also imply LTDFs [8]. 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 [3] 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 [14] 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. [9] 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. [12] 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 [11], 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 [13].

1.2 Our Contribution

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 [15]. Let $\mathcal{Y}$ be the range of a collection of chameleon ABO-TDFs and $\mathcal{B}$ be the branch set of a collection of ABO-TDFs. With the help of a family $\mathcal{T}$ of collision-resistant hash functions from $\mathcal{Y}$ to $\mathcal{B}$, a collection of chameleon hash functions can be integrated into a collection of ABO-TDFs to result in a collection of chameleon ABO-TDFs.

Following our black-box construction of chameleon ABO-TDFs, we can obtain the first chameleon ABO-TDFs based on the DDH assumption, which is the integration of the DL-based chameleon hash function [11] proposed by Krawczyk and Rabin and the ABO-TDFs [6] based on the DDH assumption.

1.3 Organization of the Paper

The paper is organized as follows. In Section 2, we review the notion of chameleon hash functions. In Section 3, we review the notion of 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. Finally, Section 5 concludes the paper.

1.4 Notation

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

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

is negligible.

A family $\mathcal{H}$ of chameleon hash functions [13], mapping $\mathcal{U} \times \mathcal{V}$ to $\mathcal{Y}$ 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 Hgen(1^κ)**:** On input a security parameter 1^κ, the key generation algorithm outputs an index k of $\mathcal{H}$ and a trapdoor td. The index k determines a specific hash function $H_k : \mathcal{U} \times \mathcal{V} \to \mathcal{Y}$.

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

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

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

$$\begin{aligned}\Pr[H_k(u, v) = H_k(u', v') :{} & (k, td) \xleftarrow{\$} \mathbf{Hgen}(1^\kappa), u, u' \in \mathcal{U}, \\ & v \in \mathcal{V}, v' \xleftarrow{\$} H_k^{-1}(u, v, td, u')] = 1.\end{aligned} \tag{1}$$

Uniformity Property: There exists a distribution $\mathcal{D}_v$ over $\mathcal{V}$, such that for all $u \in \mathcal{U}$, 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 $\mathcal{V}$ according to distribution $\mathcal{D}_v$, and $b \xleftarrow{\$} \mathcal{Y}$.

Collision Resistance Property: For all $H_k \in \mathcal{H}$, 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 $\mathcal{A}$, its advantage $\mathbf{Adv}^{CR}_{\mathcal{A},\mathcal{H}}(1^\kappa)$, defined as

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

is negligible.

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

3 Chameleon ABO-TDFs

Chameleon ABO-TDFs is a specific kind of ABO-TDFs with two variable (u,v) as a branch [12]. 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 $\mathbb{U} \times \mathbb{V} = \{\mathcal{U}_\kappa \times \mathcal{V}_\kappa\}_{\kappa \in \mathbb{N}}$ 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 $(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}^{-1}_{ch}, \mathsf{CLB}_{ch})$ 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 $S \subset \mathcal{U}_\kappa \times \mathcal{V}_\kappa$ is a set of lossy branches. Hereafter we will use $\mathcal{U} \times \mathcal{V}$ instead of $\mathcal{U}_\kappa \times \mathcal{V}_\kappa$ for simplicity.
2. **Evaluation of Injective Functions:** For any $(u,v) \in \mathcal{U} \times \mathcal{V}$, 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 $\mathsf{F}^{-1}_{ch}(i,u,v,td,\cdot)$ computes $g^{-1}_{i,u,v}(\cdot)$.
3. **Evaluation of Lossy Functions:** For any $(u,v) \in \mathcal{U} \times \mathcal{V}$, 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 $u \in \mathcal{U}$, computes a unique $v \in \mathcal{V}$ to result in a lossy branch (u, v). In formula, $v \leftarrow \mathsf{CLB}_{ch}(i, td, u)$ such that $(u, v) \in \mathcal{B}^*$.
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 $\mathcal{A}$ 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 $\mathcal{A}$. Formally, Let $\mathcal{A}$ be a CH-LI distinguisher and define its advantage as

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

   Given a collection of chameleon all-but-one trapdoor functions, it is hard to distinguish a lossy branch from an injective branch, if $\mathsf{Adv}^{\mathsf{CH\text{-}LI}}_{\mathcal{A}}(\cdot)$ is negligible for every PPT distinguisher $\mathcal{A}$.
6. **Security (2): Hidden Lossy Branches.** It is hard to find one-more lossy branch. Any probabilistic polynomial-time algorithm $\mathcal{A}$ 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 $\mathsf{F}^{-1}_{ch}(s, td, u, v, \cdot)$ 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 $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}^{-1}_{abo}\right)$, we show how to transform it into a family of chameleon ABO-TDFs $\left(\mathsf{G}_{ch}, \mathsf{F}_{ch}, \mathsf{F}^{-1}_{ch}, \mathsf{CLB}_{ch}\right)$ with the help of a family of chameleon hash functions $\left(\mathsf{HGen}, H_k, H^{-1}_k\right)$ and possibly a family $\mathcal{T}$ 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 $\mathcal{Y}$ be the range of the chameleon hash functions, and $\mathcal{B}$ the branch set of the family of ABO-TDFs. When $\mathcal{Y} \nsubseteq \mathcal{B}$ we still need a family $\mathcal{T}$ of collision-resistant hash functions to map $\mathcal{Y}$ to $\mathcal{B}$.

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 $\mathcal{H}$ and a family $\mathcal{T}$ of collision-resistant hash functions, all (u, v) are mapped into branches of an ABO-TDF i.e., $b = T\left(H_k(u, v)\right) \in \mathcal{B}$ and $H_k \in \mathcal{H}, T \xleftarrow{\$} \mathcal{T}$. The evaluation of the chameleon ABO-TDF behaves exactly as the

ABO-TDF with $b = T(H_k(u,v))$ 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., $\{(u,v) : T(H_k(u,v)) = b^*, b^* \in \mathcal{B}^*\}$, with $\mathcal{B}^*$ 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 1. *Let* $\left(\mathsf{HGen}, H_k, H_k^{-1}\right)$ *describe a family of chameleon hash functions with* $H_k : \mathcal{U} \times \mathcal{V} \to \mathcal{Y}$*, and* $\left(\mathsf{G}_{abo}, \mathsf{F}_{abo}, \mathsf{F}_{abo}^{-1}\right)$ *describe a family of* (n,k)*-ABO-TDFs with* $\mathcal{B}$ *the set of branches. Let* $\mathcal{T}$ *describe a family of collision-resistant hash functions mapping* $\mathcal{Y}$ *to* $\mathcal{B}$*. Then, a family of* (n,k)*-chameleon ABO-TDFs with branch set* $\mathcal{U} \times \mathcal{V}$ *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 $\mathbf{G}_{ch}(1^\kappa)$**:** *Given a security parameter* $\kappa \in \mathbb{N}$*,* $T \xleftarrow{\$} \mathcal{T}$*,* $(k, td_1) \xleftarrow{\$} \boldsymbol{Hgen}(1^\kappa)$*,* $u^* \xleftarrow{\$} \mathcal{U}$*,* $v^* \xleftarrow{\$} \mathcal{V}$*, compute* $b^* = T(H_k(u^*, v^*))$*. Sample a function from the ABO-TDFs with* $(i', td_2, \mathcal{B}^*) \leftarrow \mathsf{G}_{abo}(1^\kappa, b^*)$*. Let* $\mathcal{S} = \{(u,v) : T(H_k(u,v)) = b^*, b^* \in \mathcal{B}^*\}$*. Return* $i = (i', H_k, T)$ *as the function index,* $td = (td_1, (u^*, v^*), td_2)$ *as the trapdoor, and* $\mathcal{S}$ *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* $T(H_k(u,v)) \notin \mathcal{B}^*$*, and a lossy function if* $T(H_k(u,v)) \in \mathcal{B}^*$*.*

Inversion of injective functions: *On input a function index* i*, a branch* $(u,v) \notin \mathcal{S}$*, 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* $u' \xleftarrow{\$} \mathcal{U}$*,* CLB_{ch} *computes* $v' = H_k^{-1}(u^*, v^*, td_1, u')$*, and return* (u', v')*. In formula,*

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

When the range of the chameleon hash functions falls into the branch set of the ABO-TDFs, i.e, $\mathcal{Y} \subseteq \mathcal{B}$, the family $\mathcal{T}$ of collision-resistant hash functions can be omitted in the construction. We now state the security theorem of the above chameleon ABO-TDFs. The proofs will be given in the full version of the paper.

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

5 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 can obtain the first chameleon ABO-TDFs based on the DDH assumption by instantiating the construction with the existing ABO-TDFs and chameleon hash functions. According to [12], these chameleon ABO-TDFs imply more efficient black-box construction of CCA-secure PKE in the standard model than that in [18].

Acknowledgement. We are grateful to the anonymous reviewers for their helpful comments. This work is partially funded by National Natural Science Foundation of China (No. 60873229) and Shanghai Rising-star Program (No. 09QA1403000), and also supported in part by A*STAR SERC Grant No. 102 101 0027 in Singapore.

References

1. Bellare, M., Hofheinz, D., Yilek, S.: Possibility and impossibility results for encryption and commitment secure under selective opening. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 1–35. Springer, Heidelberg (2009)
2. Boldyreva, A., Fehr, S., O'Neill, A.: On notions of security for deterministic encryption, and efficient constructions without random oracles. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 335–359. Springer, Heidelberg (2008)
3. Boyen, X., Waters, B.: Shrinking the Keys of Discrete-Log-Type Lossy Trapdoor Functions. In: Zhou, J., Yung, M. (eds.) ACNS 2010. LNCS, vol. 6123, pp. 35–52. Springer, Heidelberg (2010)
4. Cachin, C., Micali, S., Stadler, M.A.: Computationally private information retrieval with polylogarithmic communication. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 402–414. Springer, Heidelberg (1999)
5. Damgård, I., Jurik, M.: A generalisation, a simplification and some applications of Paillier's probabilistic public-key system. In: Kim, K.-c. (ed.) PKC 2001. LNCS, vol. 1992, pp. 119–136. Springer, Heidelberg (2001)
6. Freeman, D.M., Goldreich, O., Kiltz, E., Rosen, A., Segev, G.: More constructions of lossy and correlation-secure trapdoor functions. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 279–295. Springer, Heidelberg (2010)
7. Hemenway, B., Ostrovsky, R.: Lossy trapdoor functions from smooth homomorphic hash proof systems. In: ECCC, vol. 16(127) (2009)
8. Hemenway, B., Ostrovsky, R.: Homomorphic Encryption Over Cyclic Groups Implies Chosen-Ciphertext Security. Cryptology ePrint Archive, Report 2010/099 (2010)
9. Kiltz, E., Mohassel, P., O'Neill, A.: Adaptive trapdoor functions and chosen-ciphertext security. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 673–692. Springer, Heidelberg (2010)
10. Kiltz, E., O'Neill, A., Smith, A.: Lossiness of RSA and the chosen-plaintext security of OAEP without random oracles (2009) (manuscript)

11. Krawczyk, H., Rabin, T.: Chameleon signatures. In: NDSS 2000. The Internet Society (2000)
12. Lai, J., Deng, R.H., Liu, S.: Chameleon All-But-One TDFs and Their Application to Chosen-Ciphertext Security. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 228–245. Springer, Heidelberg (2011)
13. Mohassel, P.: One-time signatures and chameleon hash functions. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 302–319. Springer, Heidelberg (2011)
14. Mol, P., Yilek, S.: Chosen-ciphertext security from slightly lossy trapdoor functions. Cryptology ePrint Archive, Report 2009/524 (2009)
15. Naor, M., Yung, M.: Universal one-way hash functions and their cryptographic applications. In: Proc. of 21st ACM Symposium on the Theory of Computing, pp. 33–43 (1989)
16. Nishimaki, R., Fujisaki, E., Tanaka, K.: Efficient non-interactive universally composable string-commitment schemes. In: Pieprzyk, J., Zhang, F. (eds.) ProvSec 2009. LNCS, vol. 5848, pp. 3–18. Springer, Heidelberg (2009)
17. Peikert, C., Vaikuntanathan, V., Waters, B.: A framework for efficient and composable oblivious transfer. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 554–571. Springer, Heidelberg (2008)
18. Peikert, C., Waters, B.: Lossy Trapdoor Functions and Their Applications. In: STOC, pp. 187–196. ACM, New York (2008)
19. Rosen, A., Segev, G.: Chosen-ciphertext security via correlated products. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 419–436. Springer, Heidelberg (2009)
20. Shamir, A., Tauman, Y.: Improved online/Offline signature schemes. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 355–367. Springer, Heidelberg (2001)
21. Steinfeld, R., Pieprzyk, J., Wang, H.: How to strengthen any weakly unforgeable signature into a strongly unforgeable signature. In: Abe, M. (ed.) CT-RSA 2007. LNCS, vol. 4377, pp. 357–371. Springer, Heidelberg (2006)