White-Box Implementation of the Identity-Based Signature Scheme in the IEEE P1363 Standard for Public Key Cryptography

Yudi Zhang[1], Debiao He[1], Xinyi Huang[2], Ding Wang[3] and Kim-Kwang Raymond Choo[4]

[1]Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, School of Cyber Science and Engineering, Wuhan University, Wuhan, China.
hedebiao@163.com
[2]School of Mathematics and Computer Science, Fujian Normal University, China
[3]School of Electronics Engineering and Computer Science, Peking University, Beijing, China.
[4]Department of Information Systems and Cyber Security and the Department of Electrical and Computer Engineering, The University of Texas at San Antonio, San Antonio, TX 78249, USA.

Abstract. Unlike black-box cryptography, an adversary in a white-box security model has full access to the implementation of the cryptographic algorithm. Thus, white-box implementation of cryptographic algorithms is more practical. Nevertheless, in recent years, there is no white-box implementation for public key cryptography. In this paper, we propose the first white-box implementation of the identity-based signature scheme in the IEEE P1363 standard. Our main idea is to hide the private key to multiple lookup tables, so that the private key cannot be leaked during the algorithm executed in the untrusted environment. We prove its security in both black-box and white-box models. We also evaluate the performance of our white-box implementations, in order to demonstrate utility for real-world applications.

Keywords: White-box implementation, White-box security, IEEE P1363, Identity-based signature, Key extraction

1 Introduction

White-Box cryptography was first introduced by Chow et al. [CEJVO02, CEJv03] in 2002, and is designed to prevent software implementation of cryptographic algorithm from being attacked in untrusted envrionments. Specifically, the key purpose of white-box cryptography is to ensure the *confidentiality of secret keys*. Since the first white-box implementations of DES and AES algorithms [CEJVO02, CEJv03], a number of other white-box implementations have been proposed in the literature [BCD06, Kar11].

In the trusted environment, an adversary knows the algorithm of the cryptographic system. The adversary can also require a number of inputs and obtain outputs from the program. However, the adversary does not have the permission to access the internal process of the program's execution. In practice, an adversary can also observe and modify the algorithm's implementation to obtain the internal details, such as the secret key. Many side-channel attacks have been proposed in recent, most of them can be mounted on the existing cryptographic system, such as timing, power, and fault analysis attacks. For example, the digital rights management (DRM) is commonly used to restrict the use of proprietary hardware and copyrighted works. In the example shown in Fig 1, a broadcasting company wishes to distribute their digital content (e.g., music and movies) on the Internet, and set different permissions to the users such that only paying users

can access the purchased content. However, these users should not be able to copy or re-distribute the content. Therefore, the provider should encrypt the content before distributing the content on the public network. If the user has the license to access the content, then the Rights Expression Manager can parse the user's authentication and decrypt the encrypted content using the corresponding decryption program D. However, iTunes DRM has been reportedly cracked by Johansen [Orl], where the vulnerability can be exploited to re-distribute the content without authentication. Similar vulnerabilities in iOS DRM applications have been revealed by D'Orazio and Choo [DC16], which can be exploited to gain access to copyrighted materials for free.

Figure 1: A Typical DRM Architecture

As more DRM services are offered via mobile devices / applications (apps), it is vital to ensure the security of DRM and other services/apps, for example via the use of cryptographic tools such as encryption and digital signature schemes. The latter is indispensable in the Internet especially in e-commerce, due to its capability to demonstrate the validity of user's message and identity. Formally, a valid digital signature ensures that the message was generated by a known signer, the signer cannot deny his/her signature, and that the integrity of the message has not been compromised. To avoid the limitations inherent in public key-based digital signature schemes, such as those of [ElG84, AR00, JMV01], Shamir introduced the first identity-based cryptography [Sha84]. Since the seminal work of Shamir, many other identity-based signature (IBS) schemes, such as those of [Hes03, CC03, BLMQ05], have been proposed in the literature.

While identity-based digital signature is a topic that has been extensively studied, there is not any known white-box implementation of identity-based signature scheme. Thus, this is the focus and contribution of the work in this paper. Specifically, in our study, we focus on the white-box implementation of identity-based signature scheme in the IEEE P1363 standard for public key cryptography [Gro]. As far as we know, this is the first white-box implementation of identity-based signature scheme (in the IEEE P1363 standard). Our method is lightweight, and meets the white-box security requirement. As shown in Fig 2, our method can be implemented in an untrusted wireless environment, including on mobile devices such as Android or iOS device. Specifically, the Sign algorithm is implemented on some user devices in the white-box model. Therefore, the malicious applications or hackers obtaining user's private key is impossible. Moreover, even if the device is lost, no one else can get the user's private key.

In Section 2, we introduce related white-box cryptography literature, prior to presenting the notations, the identity-based signature scheme in the IEEE P1363 standard, mathematical assumptions and the definitions of white-box security in Section 3. In Section 4, we propose our white-box implementation of the identity-based signature scheme in the IEEE P1363 standard, and give the description of the detailed construction. In Section 5, we lay special stress on analyzing the black-box and white-box security. We implement our proposed method on a personal computer (PC), then we show and evaluate the implementation performance in Section 6. We point that our method is efficient and secure in the industrial area and the real-world applications. In the last section, we conclude the paper.

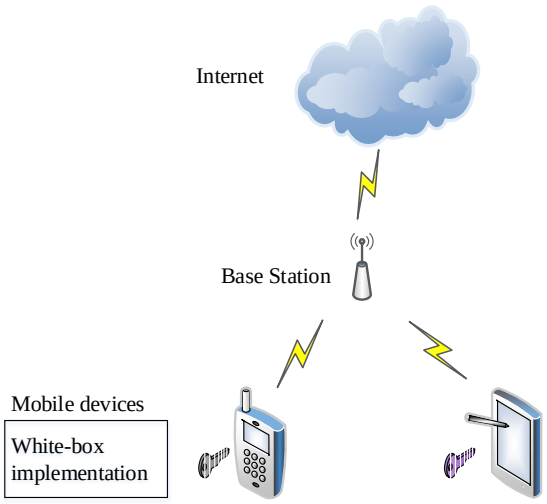

Figure 2: A Use Case for White-Box Implementation in Wireless Environment

2 Related Work

White-box cryptography (WBC) is designed to protect software implementations of cryptographic algorithms when the software is running on untrusted environment, in the sense that the adversary has full access to the implementation [CEJVO02, CEJv03]. Chow *et al.* [CEJVO02, CEJv03] proposed the first white-box implementation for both DES and AES algorithms, and the authors also introduced the White-Box Attack Context (WBAC). In WBAC, the adversary seeks to extract the secret keys from the implementation. In recent years, many other white-box implementations have also been put forward, such as white-box implementations of DES and AES [BCD06, Kar11], although many of these schemes have been shown to be vulnerable to practical key extraction or table-decomposition attacks [BGEC04, WMGP07, LRM+14]. For example, using linear algebra, Lepoint *et al.* [LRM+14] demonstrated how Lepoint's construction can be broken. Biryukov *et al.* [BBK14] also broke Chow *et al.*'s construction in 2014.

Billet *et al.* [BGEC04] proposed an effective cryptanalysis for white-box implementations of AES algorithm in 2004. They used algebraic cryptanalysis to analyze specific lookup tables, and removed the non-linear parts of the internal implementation. In a later work, Michiels *et al.* [MGH09] proposed an improved cryptanalysis, which can be used to analyze a generic class of white-box implementations.

Biryukov *et al.* [BBK14] also showed that the white-box implementations of AES and DES [CEJVO02, CEJv03] can be identified as a 3-layer ASA (affine-substitution-affine) structure, and they proposed a more secure structure (i.e., a 5-layer ASASA construction). Since the work of Biryukov *et al.* [BBK14], other researchers [BS10, BKLT13] have studied the decomposition of secret nonlinear and linear layers. Theoretically, the more layers that are employed, the more secure the construction is. However, Biryukov *et al.* [BK15] also showed that even a 9-layer construction SASASASAS is vulnerable.

Delerabl *et al.* [DLPR14] proposed a notion of *incompressibility*: an adversary has full access to the white-box implementation, but generate a program with the same function and dramatically small size is impossible. Such a notion is also referred to as *weak white-box* [BBK14] or *space hardness* [BI15] in the literature. In this notion, the adversary cannot extract the key from the white-box implementation of the cryptographic algorithm, if the implementation is large and incompressible.

More recently in 2016, Bellare *et al.* [BKR16] utilized a large encryption key to protect the key, which is called the bounded-retrieval model (BRM), and Fouque *et al.* [FKKM16] proposed the first construction with provable security guarantee. They also introduced a new definition of incompressibility (i.e. weak and Fouque *et al.* incompressibility).

A number of differential cryptanalysis techniques can be used to crack white-box implementations [CEJVO02, LN05], especially on white-box implementations for DES. For example, Chow *et al.* [CEJVO02] showed that their white-box implementations of DES is vulnerable. They then introduced an attack similar to differential power analysis, i.e. statistical bucketing attack. The statistical bucketing attack method has been improved subsequently by Link and Neumann [LN05].

While there are numerous identity-based signature schemes, they are generally not white-box attack resilience, and we are not aware of any white-box implementation for identity-based signature scheme. Hence, our research has filled the gap in white-box cryptography.

3 Preliminaries

We let S denote a set or distribution, and $a \xleftarrow{r} S$ denote that a is randomly selected from S. In this paper, n is the security parameter, for any polynomial p, if the equation $\mu(n) = O(1/p(n))$ holds, then the function $\mu(n)$ is negligible. The trusted key generation center denotes in KGC, the probabilistic polynomial time algorithm denotes in $\mathsf{P.P.T.}$ H_1 and H_2 are two secure hash functions, such that $H_1 : \{0,1\}^* \to \mathbb{Z}_p$, and $H_2 : \{0,1\}^* \to \mathbb{Z}_q$.

Let Setup be the algorithm that, given the security parameter n, it outputs the bilinear map parameters $(g_1, g_2, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3, e)$. $\mathbb{G}_1$ and $\mathbb{G}_2$ are two cyclic additive groups, g_1, g_2 are generators of $\mathbb{G}_1, \mathbb{G}_2$ respectively, $\mathbb{G}_3$ is a multiplicative group, there exists an efficient bilinear map such that $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$, which contains properties as follows:

1. For all $x_1, x_2 \in \mathbb{G}_1$ and $y_1, y_2 \in \mathbb{G}_2$, $e(x_1 + x_2, y_1) = e(x_1, y_1)e(x_2, y_1)$ and $e(x_1, y_1 + y_2) = e(x_1, y_1)e(x_1, y_2)$.
2. For all $0 \neq x \in \mathbb{G}_1$, there exists $y \in \mathbb{G}_2$ such that $e(x, y) \neq 1$.
3. For all $0 \neq x \in \mathbb{G}_2$, there exists $y \in \mathbb{G}_1$ such that $e(x, y) \neq 1$.

3.1 The Identity-based Signature Scheme in IEEE P1363 Standard

In this section, we review the identity-based signature in the IEEE P1363 standard [BLMQ05] briefly. The detailed algorithms are described below:

1. **Setup:** Taken as input the security parameter n, the KGC outputs the system parameters params as follows:
   (a) Chooses $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3$ and a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$.
   (b) Picks a random generator Q_2 of $\mathbb{G}_2$, and calculates $Q_1 = \phi(Q_2) \in \mathbb{G}_1$.
   (c) Randomly selects $s \xleftarrow{r} \mathbb{Z}_p$, sets s as the master secret key, then calculates $R = sQ_2$ and $g = e(Q_1, Q_2)$.
   (d) Sets and outputs the system parameters $\mathsf{params} = (R, g, Q_1, Q_2, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3, e)$ available.
2. **Extract:** Taken as input a user's identity ID, the KGC outputs the user's private key as follows:
   (a) Computes the identity element $h_{ID} = H_1(ID)$ where $h_{ID} \in \mathbb{Z}_p$.
   (b) Outputs $K_{ID} = (h_{ID} + s)^{-1}Q_1$.

3. **Sign:** Taken as input a message m, the user's identity ID, the signer outputs the signature σ as follows:
   (a) Randomly selects $r \xleftarrow{r} \mathbb{Z}_q$, computes $u = g^r$.
   (b) Computes $h = H_2(m, u)$ and $S = (r + h)K_{ID}$.
   (c) Outputs the signature $\sigma = (h, S)$.
4. **Verify:** Taken as input the signature σ, the corresponding message m, and the identity ID , this algorithm should check the validation of the signature. The verifier executes the steps as follows:
   (a) Computes $h_{ID} = H_1(ID)$.
   (b) Computes $u = \frac{e(S, h_{ID}Q_2 + R)}{e(Q_1, Q_2)^h}$.
   (c) If $h = H_2(m, u)$, then outputs 1; otherwise, outputs 0.

3.2 Mathematical Assumptions

Definition 1. We assume that there exists bilinear map groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_3$, P is the generator of $\mathbb{G}_1$, Q is the generator of $\mathbb{G}_2$. The q-Strong Diffie-Hellman problem (q-SDHP) in $(\mathbb{G}_1, \mathbb{G}_2)$ is described as follows: taken as input $(q + 2)$-tuple $(P, Q, xQ, x^2Q, \ldots, x^qQ)$, and output that $(c, \frac{1}{c+x}P)$ where $c \in \mathbb{Z}_q^*$. A PPT algorithm $\mathcal{A}$ solves q-SDHP in $(\mathbb{G}_1, \mathbb{G}_2)$ with the advantage ϵ if

$$\Pr[\mathcal{A}(P, Q, xQ, x^2Q, \ldots, x^qQ) = (c, \frac{1}{c+x}P)] \geq \epsilon.$$

We say that q-SDHP in $(\mathbb{G}_1, \mathbb{G}_2)$ is infeasible if all PPT algorithms can solve q-SDHP in $(\mathbb{G}_1, \mathbb{G}_2)$ with a negligible advantage ϵ.

Definition 2. Let $\mathbb{G}$ be a cyclic group of prime order q. The DL problem in $\mathbb{G}$ is to compute $a \in \mathbb{Z}_q$ for given (P, Y) where $Y = aP \in \mathbb{G}$. A PPT algorithm $\mathcal{A}$ sovles DL problem in $\mathbb{G}$ with the advantage ϵ if

$$\Pr[\mathcal{A}(P, Y) = a : a \in \mathbb{Z}_q, Y = aP] \geq \epsilon.$$

We say that the DL problem in $\mathbb{G}$ is infeasible if all PPT algorithm can solve the DL problem in $\mathbb{G}$ with a negligible advantage ϵ.

3.3 White-Box Security

We adapt existing definitions of white-box security [BBK14, BI15], presented below.

Definition 3. White-Box Attack Context (WBAC) [CEJv03]:

- an attack software has full privileges and shares a host with the cryptographic software, and it has full access to the implementation of the algorithm;
- cryptographic software can be executed dynamically and observed (i.e. the instantiated cryptographic keys);
- the internal details of the algorithm are completely visible and alterable.

Definition 4. Strong White-Box Security: We assume that the pair of algorithm (E, D) is a symmetric key scheme, and K is the secret key. Let $\mathcal{O}_{E_K}$ be a function that computes E_K. Given full access to $\mathcal{O}_{E_K}$, if obtain a function $\mathcal{D}'$ equivalent to D_K is computationally hard, then we say that $\mathcal{O}_{E_K}$ is a secure strong white-box implementation for E_K.

Based on the definition in [BBK14] and our proposed white-box implementation of identity-based signature scheme, we give the definition of weak white-box security.

Definition 5. Weak White-Box Security: Let the pair of algorithms (S, V) be a signature scheme, and K is the private key. We generate an equivalent key set $\mathfrak{F}(K)$ for K, and it is trivial to obtain an algorithm from $\mathfrak{F}(K)$ which is equivalent to S_K. The function $\mathcal{O}_{S_K}$ is a T-secure weak white-box implementation for S_K, if given the full access to $\mathcal{O}_{S_K}$, to obtain the K of length less than T from $\mathfrak{F}(K)$ is computationally hard.

That is, when an adversary is given the full access to the secure weak white-box implementation to find out any compact equivalent function smaller than T, it is computationally hard.

4 White-Box Implementation of the Identity-Based Signature Scheme in IEEE P1363

In this section, we propose our white-box implementation of the identity-based signature scheme in the IEEE P1363 standard.

Our proposed method consists of the following five algorithms, namely: Setup, Extract, WhiteBoxKeyGen, Sign and Verify.

1. **Setup:** Taken as input the security parameter n, the KGC outputs the system parameters params as follows:
   (a) Chooses $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3$ and a pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_3$.
   (b) Picks a random generator Q_2 of $\mathbb{G}_2$, and calculates $Q_1 = \phi(Q_2) \in \mathbb{G}_1$.
   (c) Randomly selects $s \in \mathbb{Z}_p$, sets s as the master secret key, and calculates $R = sQ_2$ and $g = e(Q_1, Q_2)$.
   (d) Sets params $= (R, g, Q_1, Q_2, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_3, e)$.
2. **Extract:** Taken as input a user's identity ID, the KGC outputs the user's private key as follows :
   (a) Computes the identity element $h_{ID} = H_1(ID)$ where $h_{ID} \in \mathbb{Z}_p$.
   (b) Outputs $K_{ID} = (h_{ID} + s)^{-1} Q_1$.
3. **WhiteBoxKeyGen:** Taken as input the user with the identity ID and params, the KGC outputs white-box keys as follows:
   (a) Randomly selects $x_1, x_2 \cdots, x_k \xleftarrow{r} \mathbb{Z}_p$ where k is a number greater than or equal to 256.
   (b) Computes $\{u_1 = g^{x_1}, u_2 = g^{x_2}, \cdots, u_k = g^{x_k}\}$ and $\{X_1 = x_1 K_{ID}, X_2 = x_2 K_{ID}, \cdots, X_k = x_k K_{ID}\}$.
   (c) Randomly selects $y_1, y_2 \cdots, y_{256} \xleftarrow{r} \mathbb{Z}_p$.
   (d) Computes $\{v_1 = g^{y_1}, v_2 = g^{y_2}, \cdots, v_{256} = g^{y_{256}}\}$ and $\{Y_1 = K_{ID} + y_1 K_{ID}, Y_2 = 2K_{ID} + y_2 K_{ID}, \cdots, Y_{256} = 2^{255} K_{ID} + y_{256} K_{ID}\}$.
   (e) Deletes $\{x_1, x_2 \cdots, x_k\}$ and $\{y_1, y_2 \cdots, y_{256}\}$.
4. **Sign:** Taken as input a message m, the user's identity ID, the signer outputs the signature σ as follows:
   (a) Generates a k bits number r randomly, where r is a binary and represented as $r_k \cdots r_2 r_1$. Computes $u' = \prod_{i:r_i=1} u_i$, and $S_1 = \sum_{i:r_i=1} X_i$.

(b) Computes $h = H_2(m, u')$, where h is a binary and represented as $h_{256} \cdots h_2 h_1$.

(c) Computes $S_2 = \sum_{i:h_i=1} Y_i$, and sets $S' = S_1 + S_2$.

(d) Outputs $\sigma = (h, S')$.

5. **Verify:** Taken as input the signature σ, the corresponding message m, and the identity ID, this algorithm should check the validation of the signature. The verifier executes the steps as follows:

(a) Binary h is represented as $h_{256} \cdots h_2 h_1$, and computes $t_1 = \prod_{i:h_i=1} v_i$.

(b) Computes $t_2 = \frac{e(S', h_{ID}Q_2+R)}{e(Q_1,Q_2)^h}$, and sets $u = \frac{t_2}{t_1}$.

(c) Computes $h' = H_2(m, u)$. If $h = h'$, then outputs 1; otherwise, outputs 0.

5 Security Analysis

We show the black-box and white-box security analysis separately in this section.

5.1 Black-Box Security Analysis

First, we prove that our proposed method achieves the security requirement in the black-box model. According to existing security model [BNN04, CC03, BLMQ05] for the identity-based signatures, an identity-based signature scheme is existentially unforgeable under adaptive chosen-message attacks.

Definition 6. If an IBS scheme is existentially unforgeable under adaptive chosen message and identity attacks, then for any P.P.T adversary $\mathcal{A}$ who interacts with a challenger $\mathcal{C}$ will play the game as follows:

1. $\mathcal{C}$ executes Setup algorithm to produce the system parameters, then returns it to $\mathcal{A}$.

2. $\mathcal{A}$ performs the two queries as follows:

(a) Query on Extract oracle. On input an identity ID, $\mathcal{C}$ outputs a private key which corresponds to the identity ID.

(b) Query on Sign oracle. On input an identity ID and a message m, $\mathcal{C}$ outputs a signature which corresponds to the ID's private key.

3. $\mathcal{A}$ outputs the tuple (ID^*, m^*, σ^*), where such ID^* have never been queried to Extract oracle, and (ID^*, m^*) have never been queried to Sign oracle. If Verify accepts (ID^*, m^*, σ^*), then $\mathcal{A}$ wins the game.

$\mathcal{A}$ can win this game with a negligible advantage.

Lemma 1. ***[BLMQ05]*** *Given an adaptively chosen message and the identity to the attacker $\mathcal{A}$, $\mathcal{A}$ can make q_{h_i} queries to the oracle H_1 and the oracle H_2, q_s queries to the signing oracle. Within the time bound t, if $\mathcal{A}$ can produce a forgery with the advantage $\epsilon \geq 10(q_s+1)(q_s+q_{h_2})/2^n$, then there exists another algorithm $\mathcal{B}$ which can solve the q-SDHP problem with the advantage $t' \leq 120686_{q_{h_2}} t/\epsilon$.*

Proof. In order to apply the forking lemma [PS00], for the P.P.T algorithm $\mathcal{B}$, on input $(P, Q, xQ, x^2Q, \ldots, x^qQ)$, it finds a pair $(c, \frac{1}{c+x}P)$. Similar to the proof in [BLMQ05], $\mathcal{B}$ computes $\sum_{i=0}^{q-2} d_i \psi(x^i Q) = \frac{1}{x+\omega_i} G$, where G is the generator of $\mathbb{G}_1$.

Firstly, $\mathcal{B}$ initializes l a counter, and sets $l = 1$, then executes $\mathcal{A}$ on the input (H_{pub}, ID^*), where $H_{pub} \in \mathbb{G}_2$ is the public key.

- H_1-queries: On input an identity $ID \in \{0,1\}$, if $ID = ID^*$, then $\mathcal{B}$ selects $w^* \xleftarrow{r} \mathbb{Z}_p^*$ randomly and returns it. Otherwise, $\mathcal{B}$ selects $w_l \xleftarrow{r} \mathbb{Z}_p^*$ randomly, sets $w = w_l$, and answers w together with the increments l. Then, $\mathcal{B}$ stores (ID, w) in the list L_1.
- Key extraction queries: For an input $ID \neq ID^*$, $\mathcal{B}$ searches the list L_1 and recovers the pair (ID, w), then computes $(1/(x+w))G$ and returns it.
- Sign queries: For an input tuple (m, ID), $\mathcal{B}$ randomly selects $S \xleftarrow{r} \mathbb{G}_1$, $h \xleftarrow{r} \mathbb{Z}_p^*$, and computes $u = e(S, H_1(ID)H + H_{pub})e(G, H)^{-h}$, then sets $H_2(m, u) = h$, if $H_2(m, u)$ is already set, then $\mathcal{B}$ aborts.

If the adversary $\mathcal{A}$ has no knowledge of the private key, but he still can simulate the tuple (u, h, S), then there exists a P.P.T algorithm $\mathcal{A}'$ which can employ $\mathcal{A}$ to generate two signatures (m, u, h_1, S_1) and (m, u, h_2, S_2) where $h_1 \neq h_2$ with the time $t' \leq 120686_{q_{h_2}} t/\epsilon$. Both signatures can pass the Verify algorithm.

Algorithm $\mathcal{B}$ executes $\mathcal{A}'$ and to generate two different forgeries (m^*, u, h_1, S_1) and (m^*, u, h_2, S_2), the two messages m^*, u are the same. $\mathcal{B}$ searches the list L_1 and gets the pair (ID^*, w^*). Note that, $w^* \notin \{w_1, \ldots, w_{q-1}\}$, and the probability is at least $1 - q/2^n$. If the two forgeries can pass the Verify algorithm, then we have

$$e((h_1 - h_2)^{-1}(S_1 - S_2), (w^* + x)H) = e(G, H),$$

due to $(h_1 - h_2)^{-1}(S_1 - S_2) = \frac{1}{w^*+x}G$, then $\mathcal{B}$ can extract $\sigma^* = \frac{1}{w^*+x}G$.

Therefore, if $\mathcal{A}$ can forge a signature with the advantage $\epsilon \geq 10(q_s + 1)(q_s + q_{h_2})/2^n$ in time t, then, $\mathcal{B}$ can solve q-SDHP within time t'. □

5.2 White-Box Security Analysis

We proved that our proposed method is existentially unforgeable under chosen-message attacks in the previous subsection. Now, we analyze the white-box security of our white-box implementation of identity-based signature in the IEEE P1363 standard.

Lemma 2. *If a P.P.T algorithm can compute and obtain the private key K_{ID} from the public parameter, then it can solve the DL problem.*

Proof. In our proposed method, the public parameter includes both params and white-box key – see Table 1, Table 2, Table 3 and Table 4.

In the **WhiteBoxKeyGen** phase, KGC deletes $\{x_1, x_2, \ldots, x_k\}$ and $\{y_1, y_2, \ldots, y_{256}\}$. If a P.P.T adversary $\mathcal{A}$ can compute K_{ID} from Table 2 and Table 3, then it means that there exists a P.P.T algorithm $\mathcal{A}'$ which can solve the DL problem in a non-negligible advantage.

In other words, our proposed method meets the requirement of weak white-box security. □

Table 1: Lookup Table for u_i

Index	
1	g^{x_1}
2	g^{x_2}
$\cdots$	$\cdots$
i	g^{x_i}
$\cdots$	$\cdots$
k	g^{x_k}

Table 2: Lookup Table for X_i

Index	
1	$x_1 K_{ID}$
2	$x_2 K_{ID}$
$\cdots$	$\cdots$
i	$x_i K_{ID}$
$\cdots$	$\cdots$
k	$x_k K_{ID}$

Table 3: Lookup Table for Y_i

Index	
1	$K_{ID} + y_1 K_{ID}$
2	$2K_{ID} + y_2 K_{ID}$
$\cdots$	$\cdots$
i	$2^{i-1} K_{ID} + y_i K_{ID}$
$\cdots$	$\cdots$
256	$2^{255} K_{ID} + y_{256} K_{ID}$

Table 4: Lookup Table for v_i

Index	
1	g^{y_1}
2	g^{y_2}
$\cdots$	$\cdots$
i	g^{y_i}
$\cdots$	$\cdots$
256	$g^{y_{256}}$

Definition 7. White-Box Diversity [CEJv03]: If we encode the scheme implementation steps, and can count the possible encoded steps, then this is the white-box diversity. The greater the diversity value, the safer the scheme.

In our white-box implementation of the identity-based signature scheme in IEEE P1363 standard, the diversity is $2^{n+log_2 k}$.

6 Performance Evaluation

In this section, we implement our proposed method using MIRACL Cryptographic SDK [Mir], then we show the show and evaluate the implementation performance. In addition, we compare our proposed method with the original IEEE P1363 signature scheme. The implementation of our method is deployed on a PC (with an Intel Xeon E3-1230 v5 processor, 12GB memory and the Microsoft Windows 10 operating system). The curve we used to evaluate is BN curve which achieves the AES-128 security.

The comparative summary between our method and the original IEEE P1363 scheme is presented in Fig 3, where the black-box denotes the original scheme and the white-box is our method. Note that only WhiteBoxKeyGen algorithm is employed in our method, and it is executed by the KGC. Setup and Extract algorithms are same for the both schemes; thus, they are omitted from the comparative summary. The time costs for both Sign and Verify algorithms in the proposed and original schemes are similar, with the exception of the time costs for the WhiteBoxKeyGen algorithm. However, WhiteBoxKeyGen is executed

by the KGC, so it has little effect on the user.

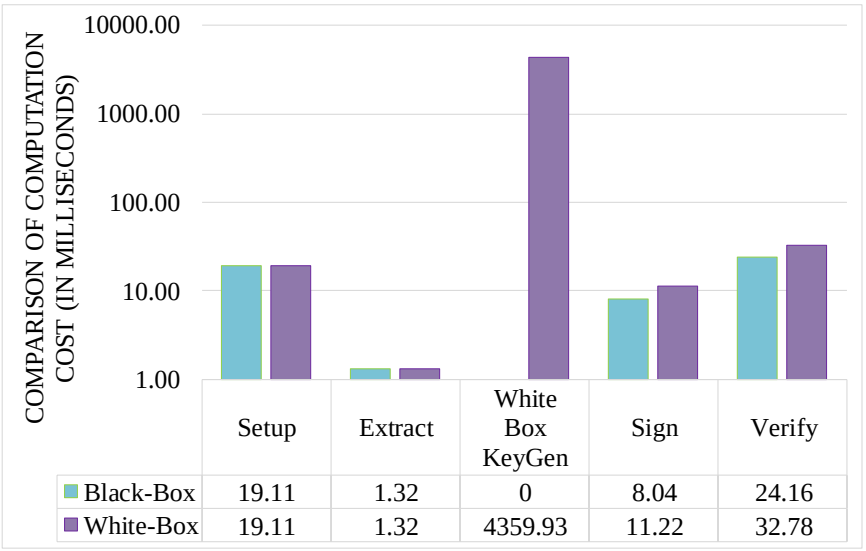

Figure 3: Computation Cost (in milliseconds): Comparative Summary

We also evaluate using messages of different lengths in both Sign and Verify algorithms. As shown in Fig 4, the lengths of the messages used are 1byte, 32bytes, 1K bytes, 10K bytes, 100K bytes and 1M bytes. With the exception of the message of 1M-byte in length, the messages are signed for approximately 11 ms and 8 ms in white-box and black-box implementation, respectively. It takes about 19 ms in white-box implementation and 15 ms in black-box implementation, respectively, when the length of the message is 1M bytes.

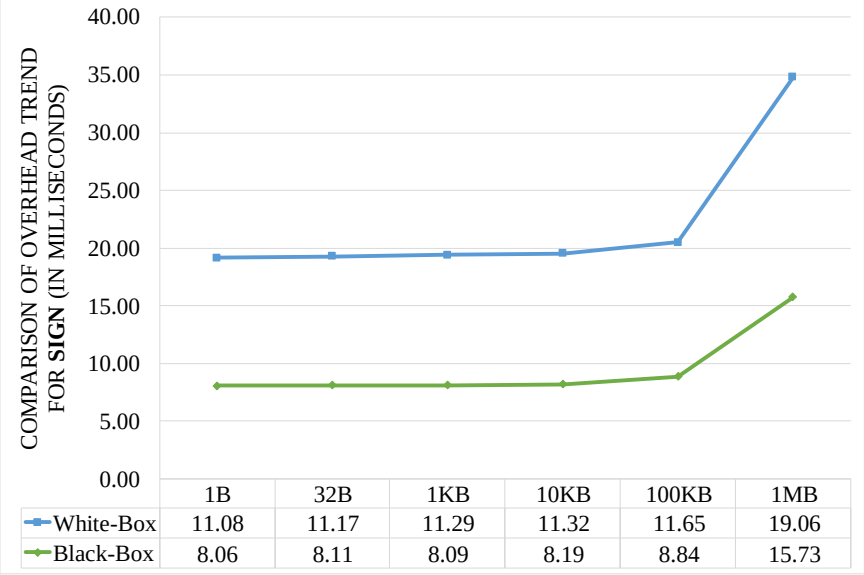

Figure 4: Time Costs for Messages of Different Sizes in the Sign Algorithm (in milliseconds)

Similarly, shown in Fig 5 is the time costs for the Verify algorithm. For messages less

than or equal to 100 bytes, it takes almost 32 ms and 24 ms in white-box and black-box implementation, respectively. However, when the message reaches 1M bytes, the time costs for white-box and black-box implementations are respectively 41 ms and 32 ms.

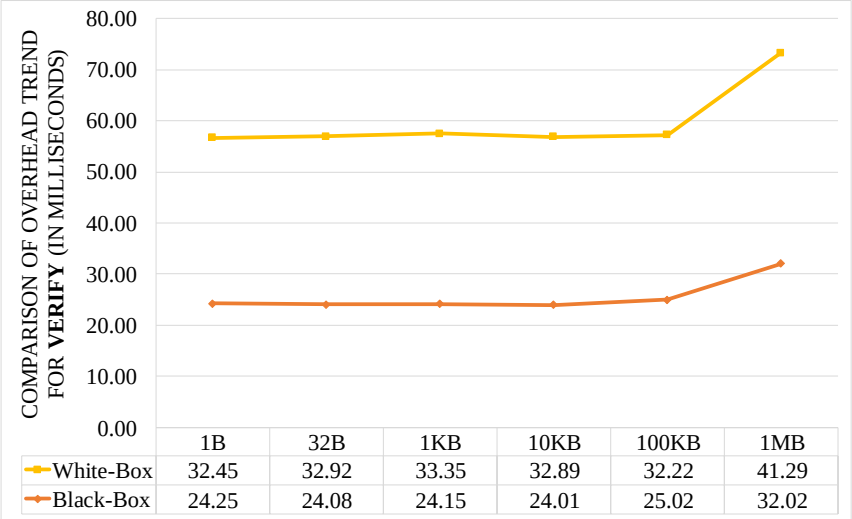

Figure 5: Time Costs for Messages of Different Sizes in the Verify Algorithm (in milliseconds)

7 Conclusion

White-box cryptanalysis and attacks are more crucial than black-box security in a real-world application, particularly in terms of ensuring the security of a user secret key.

In this paper, we proposed a novel white-box implementation for the identity-based signature scheme in the IEEE P1363 standard which is efficient and secure. Specifically, this allows us to produce a valid signature in a white-box model without leaking the private key. The security analysis demonstrated that our method can meets the white-box security requirement. According to the performance evaluation, our proposed method showed that it is potentially useful in the industrial area and the real world applications.

In the future, we intend to conduct a more comprehensive evaluation, for example using popular consumer devices (e.g., a broad range of Android, iOS, Windows Phones devices, as well as IoT devices).

References

[AR00] Michel Abdalla and Leonid Reyzin. A new forward-secure digital signature scheme. In Tatsuaki Okamoto, editor, *Advances in Cryptology – ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 116–129. Springer, Heidelberg, December 2000.

[BBK14] Alex Biryukov, Charles Bouillaguet, and Dmitry Khovratovich. Cryptographic schemes based on the ASASA structure: Black-box, white-box, and public-key (extended abstract). In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014, Part I*, volume 8873 of *Lecture Notes in Computer Science*, pages 63–84. Springer, Heidelberg, December 2014.

[BCD06] Julien Bringer, Herve Chabanne, and Emmanuelle Dottax. White box cryptography: Another attempt. Cryptology ePrint Archive, Report 2006/468, 2006. http://eprint.iacr.org/2006/468.

[BGEC04] Olivier Billet, Henri Gilbert, and Charaf Ech-Chatbi. Cryptanalysis of a white box AES implementation. In Helena Handschuh and Anwar Hasan, editors, *SAC 2004: 11th Annual International Workshop on Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 227–240. Springer, Heidelberg, August 2004.

[BI15] Andrey Bogdanov and Takanori Isobe. White-box cryptography revisited: Space-hard ciphers. In Indrajit Ray, Ninghui Li, and Christopher Kruegel:, editors, *ACM CCS 15: 22nd Conference on Computer and Communications Security*, pages 1058–1069. ACM Press, October 2015.

[BK15] Alex Biryukov and Dmitry Khovratovich. Decomposition attack on SASASASAS. Cryptology ePrint Archive, Report 2015/646, 2015. http://eprint.iacr.org/2015/646.

[BKLT13] Julia Borghoff, Lars R. Knudsen, Gregor Leander, and Søren S. Thomsen. Slender-set differential cryptanalysis. *Journal of Cryptology*, 26(1):11–38, January 2013.

[BKR16] Mihir Bellare, Daniel Kane, and Phillip Rogaway. Big-key symmetric encryption: Resisting key exfiltration. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016, Part I*, volume 9814 of *Lecture Notes in Computer Science*, pages 373–402. Springer, Heidelberg, August 2016.

[BLMQ05] Paulo S. L. M. Barreto, Benoît Libert, Noel McCullagh, and Jean-Jacques Quisquater. Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In Bimal K. Roy, editor, *Advances in Cryptology – ASIACRYPT 2005*, volume 3788 of *Lecture Notes in Computer Science*, pages 515–532. Springer, Heidelberg, December 2005.

[BNN04] Mihir Bellare, Chanathip Namprempre, and Gregory Neven. Security proofs for identity-based identification and signature schemes. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 268–286. Springer, Heidelberg, May 2004.

[BS10] Alex Biryukov and Adi Shamir. Structural cryptanalysis of SASAS. *Journal of Cryptology*, 23(4):505–518, October 2010.

[CC03] Jae Choon Cha and Jung Hee Cheon. An identity-based signature from gap Diffie-Hellman groups. In Yvo Desmedt, editor, *PKC 2003: 6th International Workshop on Theory and Practice in Public Key Cryptography*, volume 2567 of *Lecture Notes in Computer Science*, pages 18–30. Springer, Heidelberg, January 2003.

[CEJv03] Stanley Chow, Philip A. Eisen, Harold Johnson, and Paul C. van Oorschot. White-box cryptography and an AES implementation. In Kaisa Nyberg and Howard M. Heys, editors, *SAC 2002: 9th Annual International Workshop on Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 250–270. Springer, Heidelberg, August 2003.

[CEJVO02] Stanley Chow, Phil Eisen, Harold Johnson, and Paul C Van Oorschot. A white-box des implementation for drm applications. In *ACM Workshop on Digital Rights Management*, pages 1–15. Springer, 2002.

[DC16] Christian D'Orazio and Kim-Kwang Raymond Choo. An adversary model to evaluate drm protection of video contents on ios devices. 56:94–110, 2016.

[DLPR14] Cécile Delerablée, Tancrède Lepoint, Pascal Paillier, and Matthieu Rivain. White-box security notions for symmetric encryption schemes. In Tanja Lange, Kristin Lauter, and Petr Lisonek, editors, *SAC 2013: 20th Annual International Workshop on Selected Areas in Cryptography*, volume 8282 of *Lecture Notes in Computer Science*, pages 247–264. Springer, Heidelberg, August 2014.

[ElG84] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196 of *Lecture Notes in Computer Science*, pages 10–18. Springer, Heidelberg, August 1984.

[FKKM16] Pierre-Alain Fouque, Pierre Karpman, Paul Kirchner, and Brice Minaud. Efficient and provable white-box primitives. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology – ASIACRYPT 2016, Part I*, volume 10031 of *Lecture Notes in Computer Science*, pages 159–188. Springer, Heidelberg, December 2016.

[Gro] The P1363 Working Group. Ieee p1363 standard specifications for public key cryptography. http://grouper.ieee.org/groups/1363/. 2017.

[Hes03] Florian Hess. Efficient identity based signature schemes based on pairings. In Kaisa Nyberg and Howard M. Heys, editors, *SAC 2002: 9th Annual International Workshop on Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 310–324. Springer, Heidelberg, August 2003.

[JMV01] Don Johnson, Alfred Menezes, and Scott Vanstone. The elliptic curve digital signature algorithm (ecdsa). *International journal of information security*, 1(1):36–63, 2001.

[Kar11] Mohamed Karroumi. Protecting white-box AES with dual ciphers. In Kyung Hyune Rhee and DaeHun Nyang, editors, *ICISC 10: 13th International Conference on Information Security and Cryptology*, volume 6829 of *Lecture Notes in Computer Science*, pages 278–291. Springer, Heidelberg, December 2011.

[LN05] Hamilton E Link and William D Neumann. Clarifying obfuscation: improving the security of white-box des. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 1, pages 679–684. IEEE, 2005.

[LRM+14] Tancrède Lepoint, Matthieu Rivain, Yoni De Mulder, Peter Roelse, and Bart Preneel. Two attacks on a white-box AES implementation. In Tanja Lange, Kristin Lauter, and Petr Lisonek, editors, *SAC 2013: 20th Annual International Workshop on Selected Areas in Cryptography*, volume 8282 of *Lecture Notes in Computer Science*, pages 265–285. Springer, Heidelberg, August 2014.

[MGH09] Wil Michiels, Paul Gorissen, and Henk D. L. Hollmann. Cryptanalysis of a generic class of white-box implementations. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *SAC 2008: 15th Annual International Workshop on Selected Areas in Cryptography*, volume 5381 of *Lecture Notes in Computer Science*, pages 414–428. Springer, Heidelberg, August 2009.

[Mir] Miracl. Miracl library. https://www.miracl.com/. 2017.

[Orl] Andrew Orlowski. itunes drm cracked wide open for gnu/linux. seriously. https://www.theregister.co.uk/2004/01/05/itunes_drm_cracked_wide_open/. Jan 5, 2004.

[PS00] David Pointcheval and Jacques Stern. Security arguments for digital signatures and blind signatures. *Journal of Cryptology*, 13(3):361–396, 2000.

[Sha84] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer, Heidelberg, August 1984.

[WMGP07] Brecht Wyseur, Wil Michiels, Paul Gorissen, and Bart Preneel. Cryptanalysis of white-box DES implementations with arbitrary external encodings. In Carlisle M. Adams, Ali Miri, and Michael J. Wiener, editors, *SAC 2007: 14th Annual International Workshop on Selected Areas in Cryptography*, volume 4876 of *Lecture Notes in Computer Science*, pages 264–277. Springer, Heidelberg, August 2007.