摘　要

密码学是信息安全的核心技术。如果要在不同的密码系统之间进行通信，就需要支持异构通信的密码体制。

签密能够同时实现信息通信过程中的机密性和不可伪造性，大多数密码方案只考虑同类公钥密码系统之间的签密问题，而在现实生活中，不同的应用平台采用不同密码技术。因此对于签密方案的设计、分析问题，必须考虑异构密码体制如何实现的。

本文主要是从以下三方面进行研究，具体如下:

(1) 提出了一个从传统的公钥密码(Tradational Public Key Cryptography，TPKC)到基于身份的公钥密码 (Identity Public Key Cryptography ，IDPKC)的异构签密方案。该方案实现了通信过程中机密性和不可伪造性两个目标。而且，进行签密和解签密运算时，只需要 1 个对运算，效率有了一定的提高。在安全性方面，该方案满足通信过程中的内部安全性。在随机预言模型下，证明该方案满足 IND-SC-CCA2 安全性和 EUF-SC-CMA 安全性。

(2)提出了一个从无证书公钥密码(Certificateless Public Key Cryptography，CLPKC)到传统公钥密码(TPKC)的异构签密方案。该方案在签密过程中仅一个幂运算，在解签密的过程中仅需要一个对运算和幂运算，具有较高的效。同时也满足通信过程中的内部安全性。

(3)提出了一个匿名的从身份公钥密码系统(IDPKC) 到传统公钥密码系统（TPKC）异构签密方案。该方案有以下特点：首先，不仅保证在安全通信过程中信息的机密性和不可伪造性，而且在随机预言模型下，该文方案满足签密过程中的内部安全性；其次，与同类方案相比效率有了一定的提高。同时，方案实现了密文的匿名性，有效保护了接收方与发送方的隐私；最后，方案中 IDPKC 和 TPKC 密码环境中应用了不同的系统参数，从而更加有效的模拟了实际的应用场景。

关键词：异构签密，匿名性，传统公钥密码体制，无证书公钥密码体制，身份公钥密码体制

Abstract

Cryptography is the core technology of information security. If you want to communicate between different cryptographic systems, you need to support heterogeneous communication password system.

Signcryption can be achieved at the same time in the process of information communication confidentiality and unforgeability, most of the password scheme only consider the same type of public key cryptosystem between the signcryption problem, and in real life, different application platform using different cryptography. Therefore, for the design of signcryption program, analysis of the problem, we must consider how to achieve heterogeneous cryptography.

This paper is mainly from the following three aspects of research, as follows:

(1) Propose a heterogeneous signcryption scheme from the traditional public key cryptography (TPKC) to the identity public key cryptography (IDPKC). The scheme achieves two goals of confidentiality and unforgeability in the communication process. Moreover, the signcryption and decryption signed operation, only one of the operation, the efficiency has been improved. In terms of security, the program meets the internal security in the communication process. In the stochastic prediction model, it is proved that the scheme meets the safety of IND-SC-CCA2 and EUF-SC-CMA.

(2) Propose a heterogeneous signcryption scheme from certificateless public key cryptography (CLPKC) to traditional public key cryptography (TPKC). The scheme has only one exponentiation in the process of signcryption, and only one pair of operations and exponential operations are required in the process of deconvolution. But also to meet the internal security of the communication process.

(3) Propose an anonymous from the identity of the public key cryptography (IDPKC) to the traditional public key cryptosystem (TPKC) heterogeneous signcryption program. The scheme has the following characteristics: First, not only ensure the confidentiality and unforgeability of information in the process of secure communication, but also in the random prediction model, the program satisfies the internal security in the process of signcryption; secondly, Than the efficiency has been improved. At the same time, the scheme realizes the anonymity of the ciphertext and

effectively protects the privacy of the receiver and the sender. Finally, the IDPKC and TPKC cryptographic environments use different system parameters to simulate the practical application scene more effectively

Keywords: Heterogeneous signcryption, Anonymity, Traditional public key cryptography, Certificateless public key cryptography, Identity public key cryptography

1 绪论

1.1 研究背景和意义

签密能够同时实现信息通信过程中的机密性和不可伪造性，大多数密码方案只考虑同类公钥密码系统之间的签密问题，而在现实生活中，不同的应用平台采用不同密码技术。因此对于签密方案的设计、分析问题，必须考虑，在异构密码体制下是如何实现的。

目前，大多数的签密方案都是基于同一种密码系统，即接收方与发送方属于同一种密码系统。随着大数据[1]、云计算[2]、5G 异构网络[3]等实际应用场景的变化，跨平台的操作将会越来越频繁。例如，大型电子邮件系统[7]、无线移动网络[8]的跨系统访问等。对于签密[5]而言，我们要考虑异构密码环境[4]下方案的设计、分析以及实现。

不同的计算机和通信系统可能采用不同的安全技术，在异构网络中自然也需要考虑异构密码环境（即发送方和接收方具有不同的公钥密码环境）。因此，研究异构密码环境之间的签密问题将是最值得研究的问题之一。

1.2 研究现状

基于传统的公钥密码体制[6] (TPKC)不仅能够提供不可否认性、机密性和完整性[9]。在随机语言模型下基于传统的签密体制已经比较成熟，但是在标准模型下安全的基于 TPKC 的签密体制还比较少，这也是未来的一个重要研究方向。2008 年，Tan[10]利用了 Boneh 和 Boyen 设计了基本身份的加密方案[11]以及 Boneh 和 Boyen 的短签名方案[12]设计了一个 在标准模型下安全签密方案。但是，这个方案的发送者密钥生成方法和接收者密钥生成方法是不同的。如果一个用户既是发送者又是接收者，那么他需要拥有两对密钥，同时需要申请两个公钥证书，这对于实际的应用是不利的。

为了解决基于身份的密码体制的密钥托管问题，Al-Riyami 和 Paterson[13]提出了无证书密码体制(CLPKC)的概念。实际上，无证书密码体制并不一定要使用双线性对来构造，如文献[14]和[15]中构造的无证书加密方案就没有使用双线性对 ，他们运用非对称的签名方案来生成部分私钥。另外一个重要的方向是构造标准模型下无证书签密体制。2010 年，Liu[16]等人在无证书模型下建立设计无证书的签密体制。

现代计算机和通信系统形成了一个全球覆盖的基础设施。2010 年，Sun 和 Li[17]等人提出了从多接收者的 IDPKC-TPKC 双向异构签密方案，但是他的方案仅仅满足外部安全性，不能满足内部安全性，更没有提供否认性。2011 年，Huang、Wong 和 Yang 提出了一个具有密钥隐私的异构签密方案[18]。2013 年，Fu 等[19]人在 Sun 的基础上提出了一个 IDPKC → TPKIC 多接收者的异构签密方案，2013 年，Li、Zhang 和 Takagi 提出一个从 IDPKC-TPKC 双向异构签密方案[20]。

1.3 本文的主要工作和内容安排

第一章：研究背景和意义。主要介绍了密码学的发展[21]、公钥密码体制、及异构签密[22]的发展现状。

第二章：基础知识。主要介绍包括双线性对、随机预言模型、哈希函数、公钥密码体制等密码学知识和数学知识。

第三章：设计了一个高效的 TPKC→IDPKC 的异构签密方案。本方案相对于文献[26]效率方面总体上有了一定的提高，同时也满足通信过程中的内部安全性，但是解签密操作在效率上仍有可以改进的空间。

第四章：提出了基于双线性对 CLPKC→TPKC 的异构签密方案。本方案在进行签密操作的时候不需要对运算，而进行解签密操作的时候只需要一个对运算，同时本方案也满足通信过程中内部安全性[23]；但是本方案在基于无证书的密码体制与传统的密码体制所采用的系统参数都是一致的，不能更好的模拟实际的应用场景。

第五章：匿名 IDPKC→TPKC 异构签密方案。IDPKC→TPKC 异构签密方案不仅保证在安全通信过程中信息的机密性和不可伪造性，而且在随机预言模型下，该文方案满足签密过程中的内部安全性；其次，与文献[33]相比效率有了一定的提高；最后，方案中 IDPKC 和 TPKC 密码环境中使用了不同的系统参数，从而更加有效的模拟了实际的应用场景；同时方案满足匿名性[24]，可以有效保护发送方和接收方的隐私。

第六章：总结展望。对几类异构签密方案的效率、匿名性以及安全性研究进行一个总结，并对未来的研究工作进行展望。

2 基础知识

2.1 双线性对

设G_1为P生成的循环加法群，阶为，G_2为具有相同阶q的循环乘法群，a和b是Z_q^*中的元素，存在两个群G_1和G_2上的双线性对映射[22]$e:G_1\times G_1\to G_2$，且满足以下性质：

⑴双线性：对于任意的$P,Q\in G_1$和$a,b\in Z_p^*$，$e(aP,bQ)=e(P,Q)^{ab}$成立。

⑵非退化性：$P,Q\in G_1$，使得$e(P,Q)\neq 1$。

⑶可计算性：对于所有的P，$Q\in G_1$，存在有效的计算$e(P,Q)$

2.2 相关困难问题

(1) 计算 Diffie-Hellman 问题(Computable Diffie-Hellman Problem):给定一个阶为q的循环加法群G_1和一个生成元P，输入(p,aP,bP)，计算abP。这里$a,b\in Z_q^*$是未知的整数。

(2) 双线性对 Diffie-Hellman 问题(Bilinear Diffie-Hellman Problem):给定两个阶都为q的循环加法群G_1和循环乘法群G_2、双线对映射$e:G_1\times G_1\to G_2$和群G_1的生成元P，输入(p,aP,bP,cP)，计算$e(P,Q)^{abc}\in G_2$，这里$a,b,c\in Z_q^*$是未知的整数。

(3) q强 Diffie-Hellman 问题(q-Strong Diffie-Hellman Problem)：给定两个阶都为 q 的循环加法群G_1和循环乘法群G_2，双线性对映射$e:G_1\times G_1\to G_2$，输入$(P_1,\alpha P_1,\alpha^2,...\alpha^n P_1)$，计算$(\omega,\frac{1}{\omega+\alpha})\in Z_p^*\times G_1$。

2.3 随机预言模型

1986 年，Fiat 和 Shamir[25]首次提出了随机预言的概念。后来，Rogaway 和 Bellare 于 1993 年从文献[25]的思想中的到启迪，将随机预言机的概念转换成了随预言模型。在随机预言模型下进行安全证明：假设有一个能够让各方共同使用的公开的随机参数，即就是随机预言机，且随机预言机需要满足下列下列三个性质：

(1) 一致性，针对同样的输入，得出的回答一定是一样的；

(2) 有效性，对于所有询问输入，输出的回答结果是可以在多项式时间内计

算完成的。

(3) 随机性，对于所有询问，预言机的输出呈现出均匀分布，无碰撞。

2.4 哈希函数

定义 2.2 哈希函数[26]，又可以叫做杂凑函数，他可以把任意长度的消息压缩转换程固定长度的比特串，它在消息完整性检验、数字签密[4]等领域被广泛应用。

为了实现对消息的认证，哈希函数 h 必须具备以下性质：

(1) 散列性：对于任意的输入 x，$h(x)$ 在 $[0,2^k]$ 中均匀分布。

(2) 有效性：对于任意输入 x，可以在低阶多项式时间内计算 $h(x)$，即可计算 $h(x)$。

(3) 抗弱碰撞攻击性：已知 x，找到 $x \neq y$ 满足 $h(x)=h(y)$ 在计算上是不可行的。

3 高效的 TPKC→IDPKC 的异构签密方案

3.1 高效的 TPKC→IDPKC 异构签密方案形式化定义

⑴TPKC 系统建立算法

该算法通过 CA 来实现，输入参数1^k，并公开产生系统参数 $params$，其中 k 是安全参数。

⑵TPKC 密钥生成算法

TPKC 系统中的用户使用该算法生成公钥 pk_s 和私钥 sk_s。

⑶IDPKC 密钥提取算法

用户提交身份给 PKG，计算私钥 sk_r 并发送给这个用户。用户的公钥是身份 ID_U。这种公钥不需要数字证书。

⑷签密算法

该算法输入系统参数 $params$、TPKC 环境下发送者的私钥 sk_s、IDPKC 境下接收者的公钥 pk_r 和消息 m，输出密文 σ，该算法可以表示为 $\sigma = Signcrypt(sk_s, pk_r, m)$。

⑸解签密算法

该算法输入系统参数、TPKC 环境下发送者的公钥 pk_s, IDPKC 环境下的接收者的私钥 sk_r 和签密密文 σ, 输出消息 m 或者“⊥”。

上述算法需要满足签密一致性[27]要求，即如果 $\sigma = Signcrypt(sk_s, pk_r, m)$，那么 $m = Unsigncrypt(pk_s, sk_r, \sigma)$。

3.2 高效的 TPKC→IDPKC 的异构签密安全模型

机密性

游戏 3.1

(1) 初始阶段：用户运行“TPKC 密钥生成算法”，可以得到发送者的公钥 pk_s 和发送者的私钥 sk_s，并将其发送给 A。

(2) 挑战阶段： A 输出两个长度相同的明文 m_0、m_1 和接收者身份 ID_u 并将它们发送者 C。C 首先对 ID_u 调用“TPKC 密钥生成算法”，获得发送者的私钥 sk_s。随机选择一个比特 $\gamma \in \{0,1\}$ 并计算 $\sigma^* = Signcrypt(sk_s, ID_u, m_\gamma)$。

(3)猜测阶段： A 输出一个比特 γ。如果 $\gamma` = \gamma$，那么 A 赢得这个游戏。

A 赢得上述游戏的优势被定义为 $Adv\ (A) = \left|\Pr(\gamma` = \gamma) - 1/2\right|$，其中 $\Pr(\gamma` = \gamma)$

表示 $\gamma^{`}=\gamma$ 的概率。

定义 3.1 如果没有任何多项式有界的敌手在 t 时间内，以至少 ε 的优势赢得游戏 3.1，那么称本文方案是 $(\varepsilon,t,q_k,q_u)-IND-HSC-CCA2$ (Indistinguishability against adaptive-Hetero geneous Selected Chosen-Chosen Ciphertext Attack)安全的。

不可伪造性

游戏 3.2

初始阶段：C 同时运行“IDPKC 密钥提取算法”以获得接收者的公钥 pk_r 和接收者私钥 sk_r，并将 pk_r 发送给 F。

攻击阶段：在一个签密询问中，F 提交一个接收者的公钥 pk_w 和一个消息给 C。如果 pk_r 不等于 pk_w，并且 pk_w 是一个合法的公钥，那么 C 运行“签密算法”并返回密文 $\sigma=Signcrypt(sk_r,pk_w,m)$ 给 F；否则返回错误符号“⊥”。在一个解签密询问中，F 提交一个密文 σ 给 C，输出解签密密文 $Unsigncrypt(pk_w,sk_r,\sigma)$。

伪造阶段：F 产生新的密文 σ^* 和密钥对 (pk_t,pk_t)。当满足下列两个条件，F 赢得这个游戏。

(1) $Unsigncrypt(sk_t,\sigma^*)$ 输出 (m,s,pk_r) 且满足 $\perp=Verrify(pk_r,m,s)$。

(2）F 没有询问过涉及消息 m 和接收者公钥 pk_w 的签密询问，这个询问会返回密文 σ 并且 (pk_r,m,s) 成立。这里的 pk_w 可能不等于 pk_r。

定义 3.2 如果没有任何多项式有界的敌手在 t 时间内以至少 ε 的优势赢得了游戏 3.2，那么就称这个具有密钥隐私性质的 $(\varepsilon,t,q_s,q_u)-EUF-HSC-CMA$ 安全的。

3.3 高效的 TPKC→IDPKC 的异构签密方案详细描述

(1)系统建立算法

设 G_1 为 P 生成的循环加法群，阶为 q（q 为 k 特素数，k 为安全参数)，定义生成元 $P\in G_1$，G_2 为具有相同阶的 q 的循环乘法群，存在一个双线性映射 $e:G_1\times G_1\to G_2$。定义三个 $Hash$ 函数 $H_1:\{0,1\}^{l_m}\times G_1^3\to Z_q^*$、$H_2:G_2\to\{0,1\}^{l_m}$、$H_3:G_2\to\{0,1\}^{l_m}$，$l_m$ 是签密消息的长度,， PKG 随机选择一个主密钥 $P_{pub}=sP$，计算 $P_{pub}=sP$。设 $g=e(P,P)$，PKG 公开系统参数 $\{G_1,G_2,e,P,P_{pub},l_m,g,H_1,H_2,H_3\}$ 保密主密钥 s。

(2) TPKC 密钥生成算法

TPKC 系统中的用户随机选择 $u\in Z_q^*$，作为自己的私钥，并且设置公钥 $pk_s=sk_sp$。

(3) IDPKC 密钥提取算法

IDPKC 系统中的用户提交身份 ID_r 给 PKG，PKG 计算用户的私钥 $sk_r = \frac{1}{H_1(ID_r)+s}P$。

(4)签密算法

假设用户 Alice 属于 TPKC 系统，作为发送者。它的公钥为 $pk_s = sk_sP$，私钥为 sk_s；用户 Bob 属于 IDPKC 系统，作为接收者。它的公钥就是身份信息 $pk_r = H_2(ID_r)$ 和私钥 sk_r。当 *Alice* 发送消息 m 交给 *Bob* 时，执行以下步骤：

① 随机选择 $r \in Z_q^*$，并计算 $W = rP$ 和 $pk_r = H_2(ID_r)$。

② 计算 $V = sk_s H_1(m, W, pk_r)$。

③ 计算 $x = g^r$ 和 $c = m \oplus H_3(x)$。

④ 计算 $T = r[H_1(ID_r)P + P_{pub}]$。

⑤ 消息 m 的签密密文：$\sigma = (c, T, W, V)$。

(5)解签密算法

① 计算 $x = e(T, sk_r)$ 和 $m = c \oplus H_3(x)$

② 计算 $N = H_1(m, W, pk_r)$ 和 $pk_r = H_2(ID_u)$

6)验证算法

① 如果 $pk_r \notin G_1$，那么输出错误符号“⊥”，检查等式 $e(pk_r, N) = e(P, V)$,是否成立，如果成立，那么输出 $(m, (W, Q_r, V), pk_r)$，否则输出错误符号“⊥”。

3.4 高效的 TPKC→IDPKC 的异构签密方案的性质

3.4.1 正确性

本方案满足一致性要求:接收者 Bob 可以正确的解密，密文也可以被正确的验证。

(1) 当 Bob 获得密文 $\sigma = (c, T, W, V)$ 时，可以被正确的解密。

$$
\begin{aligned}
&e(T, sk_r)\\
&= e(r(H_1(ID_u)P + P_{pub}), sk_r)\\
&= e(r(H_1(ID_u)P + P_{pub}), \frac{1}{H_1(ID_r)+s}P)\\
&= e(rP, P)\\
&= g^r = x
\end{aligned}
$$

(2) 正确的验证：当 Bob 收到密文 $\sigma = (c, T, W, V)$ 和 *Alice* 的公钥，可以通过

验证等式： $e(pk_u, N) = e(P, V)$ 。

$$
\begin{aligned}
&e(pk_s, N)\\
&= e(sk_s P, H_1(m, W, pk_r))\\
&= e(P, sk_u H_1(m, W, pk_r))\\
&= e(P, V)
\end{aligned}
$$

3.4.2 机密性

定理 3.1 在随机预言模型中，若存在一个敌手 A 够在 t 时间内，以 ε 优势解决游戏 3.1，则存在一个算法 C，能够在 $t' \le t + O(q_u)t_p + O(q^2 H_1)t_m + O(q_u q_{H_2})t_e$ 时间内，以 $\varepsilon' \ge \frac{\varepsilon}{q_{H_1}(2q_{H_2} + q_{H_3})}(1 - \frac{q_u}{2^k})$ 的优势解决 n-BDHI 问题(这里 $n = qH_1$)。

证明：

初始阶段： C 随机选择 $\tau \in \{1, ..., q_{H_1}\}$ ， $e \in Z_p^*$ 和 ω_1 ， … ， $\omega_{\tau-1}, \omega_{\tau+1}, \omega_n \in Z_p^*$ 。对于 $i = 1$ ， …, $\tau - 1, \tau + 1, ...,$ n ， C 计算 $e_i = e_\tau - \omega_i$,并使用它的输入计算一个生成元 $Q \in G_1$,以至于它知道 $n-1$ 对 $(\omega_i, Vi = \frac{1}{\alpha + \omega_i} Q)$ ， $i \in \{1, ..., n\} \setminus \{\tau\}$ 。 C 为了获得这样的元素， C 展开多项式: $f(z) = \prod_{i=1, i \ne \tau}^{n} (z + \omega_i) = \sum_{j=0}^{n-1} c_j z^j$ ， 生成元 Q 和元素 X 可以分别通过 $Q = \sum_{j=0}^{n-1} c_j (\alpha^j P) = f(\alpha) P$ 和 $X = \sum_{j=1}^{n} c_{j-1} (\alpha^j P) = \alpha f(\alpha) = \alpha Q$ 得到。同时 (ω_i, V_i) 可以通过 $f_i(z) = \frac{f(z)}{z + \omega_i} = \sum_{j=0}^{n-2} d_j z^j$ 并计算 $V_i = \sum_{j=0}^{n-2} d_j (\alpha^j P) = fi(\alpha) P = \frac{f(\alpha)}{\alpha + \omega_i} P$ $= \frac{1}{\alpha + \omega_i} Q$ 得到。PKG 的公钥设为 $Q_{pub} = -X - e_\tau Q = (-\alpha - e_\tau) Q$ ，相应的私钥为 $s = -\alpha - e_\tau \in Z_p^*$ 。对于 $i \in \{1, ..., n\} \setminus \{\tau\}$ ，这里有 $(e_i, -V_i) = (e_i, \frac{1}{e_i + s} Q)$ 。

C 将系统参数[包括 Q 、 $Q_{pub} = (-\alpha - e_\tau) Q$]和 $g = e(Q, Q)$]发送给 A 。 C 同时运行密钥生成算法获得发送者的公钥/私钥对 (pk_u, sk_u) 并发送给 A 。

阶段 1： C 模拟游戏 1 中 A 的挑战者。 C 维护 L_1 、 L_2 、 L_3 三张列表，分别用于跟踪 A 对预言机 H_1 、 H_2 、 H_3 的询问。这里假设每次 H_1 询问是不同的，目标身份 ID_v 在某个时候被询问过 H_1 和身份 ID_v 在被使用到其它询问之前已经询

问过 H_1 预言机。

H_1 询问：对于 H_1 询问，C 首先检查元组 (m, P_1, P_2, P_3) 是否已经存在于列表 L_1 中。如果已经存在，那么返回结果。如果不存在并且等式 $e(P_1, P_2) = e(P, P_3)$ 成立和 $(P_1, P_2, \perp)$ 已经存在列表 L_1 中，C 用 P_3 代替“$\perp$”并将存在的结果返回给 F 。对其他的情况，C 随机选择 $t \in Z_p$ 并返回 tP 给 F 。询问的元素和返回的值都将存储在列表 L_1 中。

H_2 询问：设计计数器 v，初始值为 1。对于一个 $H_1(ID_v)$ 询问，C 返回 e_v 作为回答，将 (ID_v, e_v) 存进列表 L_2 并将计数器 v 的值的加 1。

密钥提取询问：当 A 询问 ID_i 的私钥时，如果 $i = \tau$，那么 C 失败并停止；否则 C 知道 $H_1(ID_i) = e_i$，并返回 $-V_i = \frac{1}{e_i + s} Q$ 给 A。

解签密询问：A 提交一个接收者身份 ID_j 和一个密文给挑战者 C。如果 $j = \tau$，那么 C 知道接收者的私钥 $S_j = -V_j$，可以按照正常的解签密步骤来回答这个询问。如果 $j = \tau$，那么对于所有合法的密文，有 $\log_{sk_u}(V - hsk_u) = \log_{e_j Q_u + Q_{pub}} T$，这里这里的 $h = H_2(m, x)$。因此，等式 $e(T, sk_u) = e(e_j Q_u + Q_{pub}, V - hsk_u)$ 成立。C 首先计算 $\xi = e(V, e_j Q_u + Q_{pub})$ 并在 L_2 中查找形式为 $\{m_i, x_i, h_{2,i}, c, \xi\}$ 的条目，这里 $i \in \{1, ..., q_{H_2}\}$。如果没有这样的条目，那么 C 拒绝回答 σ 的解签密询问；否则 C 进一步检查等式 $\frac{e(T, sk_u)}{e(e_j Q_u + Q_{pub}, V)} = e(e_j Q_u + Q_{pub}, V - hsk_u)$。如果唯一满足上述方程的 $i \in \{1, ..., q_{H_2}\}$ 已找到，那么 C 返回匹配的消息 m_i。

挑战阶段：A 产生两个一样长度的明文 m_0 和 m_1 和接收者身份 $ID_B \neq ID_\tau$ 并将他们发送给 C。如果 $ID_B \neq ID_\tau$，那么 C 终止；否则 C 随机选择 $c^* \in \{0,1\}^{l_m}$、$V^* \in G_1$ 和 $T^* = -\lambda Q$ 并计算 $T^* = -\lambda Q$。C 返回密文 $\sigma^* = (c^*, V^*, W^*)$ 给 A。如果定义 H_2 且 $s = -\alpha - e_\tau$，那么就有 $T^* = -\lambda Q = -\rho\alpha Q = \rho e_\tau Q + \rho Q_{pub}$ 除非 A 对进行 σ^* 和 H_3 询问，否则他（她）不能辨别出 σ^* 是一个不合法的密文。

猜测阶段：

C 从列表 L_2 或者 L_3 中提取随机的条目 $(m_i, x_i, h_{2,i} c_i, \xi_i)$ 或者 $(x_i, h_{3,i})$。既然 L_3 包含的条目至多为 $2q_{H_2} + q_{H_3}$，那么随机选择的条目包含了正确元素

$x_i = e(Q,Q)^{\rho} = e(P.P)^{f(\alpha)^2 \lambda / \alpha}$ 的概率为 $\frac{1}{2q_{H_2} + q_{H_3}}$。如果 $\xi^* = e(P, P)^{1/\alpha}$，那么 n-BDHI 问题可以通过下式解决:

$$e(Q,Q)^{1/\alpha} = \xi^{*(c_2^0)} e(\sum_{j=0}^{n-2} c_{j+1}(\alpha^j P), c_0 P) e(Q, \sum_{j=0}^{n-2} c_{j+1}(\alpha_j) P)$$

下面分析 C 的优势，定义以下三个事件

$E_1 : A$ 在挑战阶段没有选择 ID_τ 为接收者的身份。

$E_2 : A$ 对身份 ID_τ 进行了秘钥提取询问。

$E_3 : A$ 由于在解签密询问中拒绝了一个合法的的密文而终止。

由上面的分析来看：不终止的概率为 $Pr[\neg end]=Pr[\neg E_1 \wedge \neg E_2 \wedge \neg E_3]$，$Pr[\neg E_1] = 1/q_{H_1}$ 和 $Pr[E_3] \le q_u / 2^k$，此外，$\neg E_1$ 意味着 $\neg E_2$。所以，有 $Pr[\neg end] \ge \frac{1}{q_{H_1}}(1 - \frac{q_u}{2^k})$。另外，$C$ 从 L_2 或者 L_3 中选择正确元素的概率为 $\frac{1}{2q_{H_2} + q_{H_3}}$。

因此有 $\varepsilon' \ge \frac{\varepsilon}{q_{H_1}(2q_{H_2} + q_{H_3})}(1 - \frac{q_u}{2^k})$。

3.4.3 不可伪造性

定理 3.2 在随机预言模型中，若存在一个敌手 F 能够以 ε 的优势攻破本方案 EUF-HSC-CMA 的安全性，则存在一个算法 C，以 $(1 - q_u 2^{-ploy(k)})(1 - q_H q_s 2^{-ploy(k)})$ 的优势解决 CDH 问题。其中，q_u 表示最大的解签密询问次数；q_{H_1} 表示最大的 H_1 询问次数；q_s 表示最大的签密询问次数，$ploy$ 表示一个多项式；k 为安全参数。

证明：

初始阶段：C 运行“IDPKC 系统建立算法”，将将主密钥 s 也发送给 F，同时运行“TPKC 密钥生成算法”获得发送者的公钥/私钥 (pk_s, sk_s) 发送给 F 并且设 $pk_v = bP$ 为挑战公钥。

H_1 询问：对于一个 $(P_1, P_2, \perp)$ 询问，C 首先检查元组 (m, P_1, P_2, P_3) 是否已经存在于列表 L_1 中。如果已经存在，那么返回结果。如果不存在并且等式 $e(P_1, P_2) = e(P, P_3)$ 成立和 $(P_1, P_2, \perp)$ 已经存在列表 L_1 中，C 用 P_3 代替“$\perp$”并将存在的结果返回给 F。

H_2 询问：对于一个 $H_1(ID_v)$ 询问，C 返回 e_v 作为回答，将 (ID_v, e_v) 存进列表 L_2

并将计数器 v 的值的加 1。

签密询问：F 提交接收者的公钥 pk_w 和消息 m 给 C。如果 pk_w 不合法或者等于 pk_u，那么返回一个错误符号“⊥”；否则 C 随机选择 $r \in Z_p$，计算 $W = rP$。如果 $(m, W, pk_\omega, rpk_\omega)$ 经存在于 L_1 中且相应的 $Hash$ 值是 taP，那么 C 失败并停止。否则，C 随机选择 $t' \in Z_p$ 并返回 $t'P$ 作为 $H_1(m, W, pk_\omega, rpk_\omega)$ 的值，询问元组、返回值和 t' 都被存进列表 L_1 中。C 计算 $(W, Psk\lambda$，同理可以获得 $H_2(m, W, pk_\omega, rpk_\omega)$ 并计算 $T = (m \| pk_u \ t \| pk_u \oplus H_2 \ W \ R(k_\omega \ rPk_\omega$，得到密文 $\sigma = (c, T, W, \backslash$。

解签密询问：F 提交一个签密密文 $\sigma = (T, W)$ 给挑战者 C，得出解签密密文 $Unsigncrypt(sk_u, \sigma)$。执行下面的步骤：

(1)在列表 L_2 中查找元组 (W, pk_u, λ) 使得 $e(P, \lambda) = e(U, pk_u)$ 成立或 $\lambda = \top$。如果列表 L_2 中不存在元组 $\lambda = \perp$，那么在 L_2 中插入一个新的条目，这个条目以 $(U, pk_s, \perp)$ 为询问元组。符号“⊥”表示 (P, U, pk_s) 的 CDH 问题的解。这一步确保了 $(P, \lambda) = (W, pk_s)$，在 σ 解签密前确定下来。如果元组 (W, pk_s, λ) 已经存在于 L_2 中，那么存在的结果将为 $H_2(W, pk_s, \lambda)$ 的值。

(2)计算 $(m \| pk_s \ \| V) = T \oplus H_2 \ W, pk_s, \lambda$ 并在列表 L_1 中查找 (m, W, pk_s, λ) 使得 $(P, \lambda) = (W, pk_s)$ 成立或者 $\lambda = \perp$。如果 L_1 中不存在 $H_1(m, W, pk_s, \lambda)$，那么 L_1 中插入一个新的条目，这个条目以 (m, W, pk_s, λ) 为询问元组，以一个随机值 taP 为返回值。这里 $t \in Z_p$。值得注意的是，λ 可能已经从上面的 L_2 获得，等式 $e(P, \lambda) = e(W, pk_s)$ 可能成立。如果元组 $(m, (W, pk_s, \lambda, V)$ 已经存在于 L_1 中并且等式 $e(P, \lambda) = e(W, pk_s)$ 成立，那么存在的值作 $H_1(m, W, pk_s, \lambda)$ 的值，λ 也用于更新 L_2 中相应的条目。如果 $abP = t^{-1}V$ 存在于 L_1 中。如果 λ 可以从 L_2 中获得，那么 $(m, W, pk_s, \perp)$ 也应该被更新为 (m, W, pk_s, λ)。

(3)检查等式 $e(pk_s, H_1(m, W, pk_s, \lambda)) = e(P, V)$ 是否成立。若不成立，则输出错误符号“⊥”否则进一步检查等式 $e(P, \lambda) = e(W, pk_s)$ 是否成立。若成立，则输出消息/签名 $(m, (W, pk_s, \lambda, V)$ 发送者的公钥 pk_s。若 $\lambda = \perp$，则 C 失败并停止。

伪造阶段：当 F 产生一个密文 σ 和一个密钥对 (sk_v, pk_v) 时，$V = taP$ 按照上述解签密模拟的步骤对密文进行解签密。如果这个伪造是合法的，那么有 $e(pk_s, H_1(U, pk_s, \lambda, V)) = e(P, V) = e(bP, taP)$。$C$ 可以得到 $V = taP$ 并解决 CDH 问题 $abP = t^{-1}V$，C 输出 abP 并停止。

由上面的解签密模拟可以看出，L_1 中必定存在一个针对 $H_1(m, W, pk_s, \lambda)$ 的条

目且相应的返回值一定是taP形式。如果说$H_1(m,W,pk_v,\lambda)$的值为$t'P$，即从签密询问中生成，那么也应该在签密询问阶段生成。这于游戏 2 中的限制条件（σ不是由签密预言机输出的）相矛盾。

C失败的情况有两种。第一种在签密询问中，选择了$r \in Z_p$，$(m,W=rp,pk_\omega)$已经询问过了H_1了，这种事件发生的概率为$q_{H_1}q_s/l_1$。C不失败的概率至少为abP。第二种在解签密询问中，同理也可以得出C在解签密询问中不失败的概率至少为$1-q_u 2^{-poly(k)}$。如果C不失败并且F赢得了这个游戏，那么C能够解决 CDH 问题。因此，$\Pr[win] \geq (1-q_u 2^{-poly(k)})(1-q_{H_1}q_u 2^{-poly(k)})\varepsilon$。

3.5 效率分析

表 1 列出了该文方案与其他方案所需要的运算量，其中p表示对运算，e表示幂运算。

表 1 该文方案与其他方案所需的运算量

方案	签密操作	解签密操作	内部安全性
文献[19]	1p+1e	2p+1e	不满足
文献[17]	1p	2p+1e	满足
本文方案	1e	1p	满足

3.6 本章小结

本章提出了高效的 TPKC→IDPKC 的异构签密方案，虽然基于不同的公钥密码系统实现的，但是两种系统所采用的参数都是相同的，得出的效果并不是很理想，所以在以后异构签密的研究中，不同的密码系统采用不同的参数，满足现实生活中的应用场景；虽然该文方案效率上有了一定的提高，但是仍有可以改进的空间。

4 改进的 CLPKC→TPKC 的异构签密方案

4.1 CLPKC→TPKC 异构签密方案的定义及安全模型

4.1.1 形式化定义

(1) CLPKC 系统建立算法：获取安全参数 l，选择系统主密钥 s，得到系统密钥 P_{pub} 和参数 $Params$。

(2) CLPKC 部分私钥提取算法：获取用户 ID_i、主密钥 s 和系统参数 $Params$，获取部分私钥 D_i，并将 D_i 以安全的方式发送给用户。

(3) CLPKC 私钥生成算法：获取秘密值 x_i、系统参数 $Params$、部分私钥 ID_i，得到用户完整私钥 $S_i = (x_i, D_i)$ 和公钥 P_i。

(4) TPKC 密钥提取算法：运行算法产生 $TPKC$ 系统中该用户的公私钥对 sk_r / pk_r。

(5) 签密算法：输入消息 m、$CLPKC$ 系统中发送者的公私钥对 S_i / P_i、$TPKC$ 系统中接收方的公钥 pk_r 和系统的参数 $Params$，获得消息 m 的密文 σ。

(6) 解签密算法：输入签密的密文 σ、发送者身份和公钥 ID_i / P_i 和系统参数 $Params$ 和接收者的私钥 sk_r，输出消息 m 或者符号“⊥”。

以上的算法必须满足一致性约束，即如果 $\sigma = signcrypt(m, ID_i, P_i, S_i, sk_r)$，则必有 $m = Unsigncrypt(\sigma, ID_i, P_i, sk_r)$ 成立。

4.1.2 安全模型

机密性

游戏 4.1

(1) 初始阶段

F 运行“CLPKC 系统建立算法”，提供安全参数 l 和主密钥 s 和系统参数 $Params$，并发送给攻击者 A。

(2) 阶段 1

解签密询问：攻击者 A 提交一个密文 σ 给 F，F 获取接收方的私钥 sk_r，运行“解签密算法”，并返回给 A。

(3) 挑战阶段

攻击者 A 决定在什么时候终止“阶段 1”，并进入“挑战阶段”。 A 选择消息 m_0 和 m_1，并且长度相同，发送方 ID_i、接收方公钥 pk_r 作为要挑战的信息，并提交给 F 。

(4) 猜测阶段

A 输出一个比特 b，如果 $b^{'}=b$，那么 A 赢得游戏。 A 的优势被定义为 $Adv(A)=|\Pr[b^{'}=b]-1/2|$，其中 $\Pr[b^{'}=b]$ 表示 $b^{'}=b$ 的概率。

定义 4.1 如果在 t 时间内没有任何多项式有界攻击者那么称这个 CLPKC→TPKC 异构签密方案是 (ε,t,q_u)-IND-CLPKC -TPKC-HSC-CCA2 安全的。

不可伪造性

游戏 4.2

我们必须考虑 CLPKC→TPKC 异构签密方案在适应性选择消息攻击下具有不可伪造性。

(1)初始阶段

用户运行“TPKC 密钥生成算法”计算接收者的公钥/私钥对 (pk_r,sk_r)。发送系统参数 $Params$、系统密钥 P_{pub}、接收者公钥/私钥对 (pk_r,sk_r) 给 A_I 。

(2)攻击阶段

① 部分私钥询问：攻击者 A_I 提交用户身份 ID_i，若列表中存在对应部分秘密值信息，则直接返回，否则，C 运行 “CLPKC 用户部分私钥生成算法”获得 ID_i，对应的部分私钥 D_i，返回 D_i 给 A。

② 签密询问：攻击者 A_I 提交发送者的身份 ID_i、发送者的公钥 $P_i^{`}$、接收者公钥 pk_r 和消息 m_i，挑战者 C 首先调用“CLPKC 用户秘钥生成算法”获得 ID_i 的私钥 S_i，然后运行“签密算法”并返回密文 σ^* 返回给 A_I 。

(3) 伪造阶段

攻击者 A_I 提交 $(m^*,\ ID_i,\ P_i,\ pk_r,\ \sigma^*)$，当下列两个条件成立时，则 A_I 赢得这个游戏。

① σ^* 对于 P_i 和 pk_r 是合法的密文，即 $Unsigncrypt=(\sigma^*,ID_i,sk_r)$。

② ID_A 不是被 A_I 执行过秘密值询问的身份。

③ 对于 (ID_i,m^*,pk_r) 的签密询问， A_I 没有执行过。

定义 4.2：如果在 t 时间内没有任何多项式有界攻击者，至少 ε 的优势赢得游戏 4.2，那么称这个 CLPKC→TPKC 异构签密方案是 $(\varepsilon,t,q_{pr},q_{sv},q_{pk},q_s)$-EUF-CLPKC -TPKC-HSC-CMA-I 安全的。

游戏 4.3

假定 F 为挑战者， CLPKC→TPKC 异构签密方案由三个阶段组成。

(1)初始阶段

用户运行“TPKC 密钥生成算法”计算接收者公钥/私钥对 (pk_r,sk_r)。F 将发送者系统参数 Params、系统密钥 s、接收者公钥/私钥对 (pk_r,sk_r) 给 A_{II}。

(2)攻击阶段

签密询问：攻击者 A_{II} 提交接收者公钥 pk_r、发送者的公钥 P_i、发送者的身份 ID_i、和消息 m_i，F 首先调用“CLPKC 用户公钥生成算法”和“CLPKC 用户私钥生成算法”获得 ID_i 对应私钥 S_i，然后得到密文 $\sigma=Signcrypt(m_i,S_i,pk_r)$ 给 A_{II}。

(3)伪造阶段

攻击者 A_I 提交 $(m^*,\ ID_A,\ P_i,\ pk_r,\ \sigma^*)$，当下列三个条件成立时，则 A_I 赢得这个游戏。

① σ^* 对于 P_i 和 pk_r 是合法的密文，即 $Unsigncrypt=(\sigma^*,ID_A,sk_r)$。② ID_A 不是被 A_I 执行过秘密值询问的身份。③对于 (ID_A,m^*,pk_r) 的签密询问，A_I 没有执行过。

定义 4.3：如果在 t 时间内没有任何多项式有界攻击者至少 ε 的优势赢得游戏 4.3，那么称这个 CLPKC→TPKC 异构签密方案是 $(\varepsilon,t,q_{sv},q_{pk},q_s)$-EUF-CLPKC -TPKC-HSC-CMA-II 安全的。

4.2 具体的 CLPKC→TPKC 异构签密方案

(1)CLPKC 系统建立算法。设参数 k 为大素数，定义阶为 q 的群 G_1 和 G_2，双线性映射 $e:G_1\times G_1\to G_2$，选择哈希函数 $H_1:\{0,1\}^*\to Z_q^*$，$H_2:\{0,1\}^*\times\{0,1\}^*\times G_1\times G_1\to Z_p^*$，$H_3:\{0,1\}^*\to\{0,1\}^{l_m+2l_1}$，$l_m$ 表示消息的长度，l_1 表示群 G_1 中一个元素的比特长度。KGC 选取 $s\in Z_q^*$ 为主秘钥，计算 $P_{pub}=sP,g=(P,P)$，发布系统参数 $Params=\{G_1,G_2,e,q,g,P_{pub},H_1,H_2,H_3\}$，消息空间 $M=\{0,1\}^*$，保存主秘钥 s，系统密钥为 $P_{pub}=sP$。

(2) CLPKC 部分私钥生成算法。KGC 计算 $Q_i=H_1(ID_i)$、$D_i=\dfrac{1}{s+Q_i}P$，发送 D_i 给用户。

(3) CLPKC 用户秘钥生成算法。用户选择秘密值 $x_i \in Z_q^*$，计算公钥 $P_i = x_i(P_{pub} + Q_i P) = x_i(s + Q_i)$，当用户收到部分私钥 D_i，产生用户的完全私钥 $S_i = (x_i, D_i)$。

(4) TPKC 密钥生成算法。TPKC 中用户选择 $x_r \in Z_q^*$，计算自己的私钥 $sk_r = x_r$ 和公钥 $pk_r = x_r P$。

(5) 签密算法

如果 CLPKC 系统中 Alice 为发送方，其身份为 ID_i、对应的公钥和私钥为 $P_i = x_i(s + Q_i P)$ 和 $S_i = (x_i, D_i)$；TPKC 系统中的用户 Bob 为接收者，其公钥 $sk_r = x_r P$ 和私钥 sk_r。当 Alice 给 Bob 发送消息 m 时，Alice 执行以下步骤:

① 随机选择 $u \in Z_q^*$，计算 $Q_i = H_1(ID_i), R = g^u, V = uP$。

② $T = upk_r$，$h = H_2(m \| ID_i \| P_i \| R)$，$W = \frac{u+h}{x_i} D_i = \frac{u+h}{x_i} \frac{1}{s+Q_i} P$。

③ $C = (m \| ID_i \| P_i \| W) \oplus H_3(V, pk_r, T)$，则消息 m 的签密密文为 $\sigma = (C, V, R)$。

(5)解签密算法

当收到密文 $\sigma = (C, V, R)$ 时，Bob 执行以下操作：

① $T' = x_r V$，$(m \| ID_i \| P_i \| W) = C \oplus H_3(V, pk_r, T')$。

② $Q_A = H_1(ID_i)$，$h = h_2(m \| ID_i \| P_i \| R)$。

③ 验证 $e(W, P_i) g^{-h} = R$ 是否成立。

4.3 CLPKC→TPKC 异构签密具体方案安全性和效率分析

4.3.1 正确性

本方案必须满足一致性要求：接收者 Bob 可以正确地解密，密文也可以被正确验证。

(1) 当 Bob 收到密文 $\sigma = (C, V, R)$，可以正确的解密。

$$
\begin{aligned}
&(m \| ID_i \| P_i \| W) = C \oplus H_3(V, pk_r, T') \\
&= (m \| ID_i \| P_i \| W) \oplus H_3(V, pk_r, upk_r) \oplus H_3(V, pk_r, x_r V) \\
&= (m \| ID_i \| P_i \| W) \oplus H_3(V, pk_r, ux_r P) \oplus H_3(V, pk_r, x_r uP) \\
&= (m \| ID_i \| P_i \| W)
\end{aligned}
$$

(2)正确验证。当 Bob 解密密文 $\sigma = (C, V, R)$ 后，可以被正确的验证。

$$e(W,P_i)g^{-h}=e(\frac{u+h}{x_i}D_i,P_i)g^{-h}=e(\frac{u+h}{x_i}\frac{1}{s+Q_i}P,x(P_{pub}+Q_iP))g^{-h}$$

$$=e(\frac{u+h}{x_i}\frac{1}{s+Q_i}P,x_i(s+Q_i)P)g^{-h}=e((u+h)P,P)=\ g^{u+h}=g^{-h}=g^{u}=R$$

4.3.2 机密性

定理 4.1 在随机预言机模型下，如果 CLPKC→TPKC 异构签密方案 IND-CLPKC -TPKC-HSC-CCA2 安全性，被攻击者 A 能够以ε的优势攻破，那么存在算法F能以$\frac{\varepsilon}{q_{H_3}+q_{H_2}}(1-\frac{q_u}{2^l})$的优势解 CDH 问题。

证明：

初始阶段：F设接收者公钥$pk_r=ap$，F 运行“CLPKC 系统建立算法”，输入安全参数l，获得系统参数 *Params* 和主密钥s。

H_1询问：F保持列表$L_1=(ID_i,t_i)$,初始为空。A询问H_1预言机，若L_1中存在询问项则直接返回，否则，F选择S^*,计算t_iP，返回t_iP并将(ID_i,t_i)增加到L_1中。

H_2询问：F保持一个列表$L=\{m,ID,P,R,h\}$，初始为空，L_1提出对(m,ID,P,R)的询问，若列表L_2中存在询问项，F将h的值返回给A；否则，F随机选择$h\in Z_q^*$,令(ID_i,Q_i,D_i),F将h返回给A，并将(m,ID_i,P_i,R_i)添加到列表L_2。

签密询问：攻击者A提交消息 m 和一个接收者公钥pk_w。如果pk_w等于pk_r或者不合法，则返回错误符号“⊥”，否则执行正常的签密操作，即调用“签密算法”生成并返回密文信息$\sigma=(V,C,W)$。

挑战阶段

攻击者 A 决定在什么时间终止“阶段 1”同时进入“挑战阶段”。A 产生两个长度一样的明文m_0与m_1、发送者和接收者公私钥S^*和pk_r。输出密文，C 随机选择比特$b\in(0,1)$,并计算m_b在发送者私钥S^*和被攻击者的接收者公钥pk_r，返回密文$\sigma_b^*=Signcrypt(m_b,S^*,pk_r)$。F 随机选择$C^*\in\{0,1\}^{lm}$和$\lambda^*\in G_1$，令$V^*=bP$、$Q_i^*=H_1(ID_i)$、$h^*=h_2(m\|x_i\|P_i\|R)$、$W^*=\frac{r+h^*}{x}\frac{1}{s+Q_i^*}P$，F 返回$\sigma^*=(C^*,V^*,R^*)$给$A$。

猜测阶段

F 从 L_3 中提取一个随机的条目 (V_i, pk_r, λ) 或 $(m, U_i, P_i, V_i, pk_r, \lambda)$。$L_3$ 中包含 q_{H_3} 项条目，随机选择的条目满足 $e(bP, pk_r) = e(P, \lambda)$ 的概率为 $\frac{1}{q_{H_3}}$，CDH 问题得以解决。

4.3.3 不可伪造性

定理 4.2 在随机预言模型下，假设 q-SDH 问题困难，则提出 CLPKC→TPKC 异构签密方案在适应选择消息攻击下是存在性不可伪造的，即 EUF-CLPKC-TPKC-HSC-CMA 安全的。

证明:我们构造一个算法 B 利用 A 解决 q-SDH 问题，保证攻击者 A 能够以一定的优势攻破改方案。假设 B 是 q-SDH 问题的挑战者，对随机预言机 H_1、H_2、H_3 进行询问，B 的目标是根据身份 ID^* 和消息 M^*，得到有效的签名，从而解决 q-SDH 问题。

假设在群 G_1 上，获得 $q+1$ 元组 $(P, aP, a^2P, \ldots, a^q p)$，当作 q-SDH 问题的输入，目标是让 B 得到对 $(c, \frac{1}{a+c}P)$,其中 $a,\ c,\ q \in Z_q^*, P \in G_1$。

首先，获得生成元 $P^{`} \in G_1$，依照下列步骤计算对 $(y_{i,} \frac{1}{a+y_i}P^{'})$，其中 $y_1,\ y_2 \ldots,\ y_{q-1} \in Z_p^*, p' \in G$。

随机选择 $y_1,\ y_2 \ldots,\ y_{q-1} \in Z_p^*$,有展开式 $f(x) = \prod_{i=1}^{q-1}(x+y_i), c_0, c_1 \ldots, c_{q+1} \in Z_p^*,$，若 $c_0, c_1 \ldots, c_{q+1} \in Z_p^*$，则有 $f(x) = \sum_{i=0}^{q-1} c_i x_i$，设 $P^{'} = \sum_{i=0}^{q-1} c_i(a_i P) = f(a)P$，系统公钥 $Pub^{'} = \sum_{i=1}^{q} c_{i-1}(a_i P) = aP^{'}$,a 是主密钥。$B$ 展开 $f(x) = \frac{f(x)}{(x+y_i)} = \sum_{i=0}^{q-2} c_i x^i$,有 A_{II},可以计算对 $(y_i, \frac{1}{a+y_i}P^{'})$。

B 设 $g^{'} = e(P^{'}, P^{'})$，$P_{pub}{}^{'} = aP^{'}$，a 为系统密钥，对 B 保密，系统参数 $^{'}Params = (G_1, G_2, e, q, P, P_{pub}, g, H_1, H_2)$，$q_H$ 为 H_1 询问的最大次数。

H_1 询问：保持一个列表 $L_1 = \{ID_i,\ Q_i\}$，初始为空，设 ID_i 是对 H_1 的第 i 次询问，若 ID_i 在 L_1 列表中，返回对应 Q_i 值。否则，执行下列步骤：如果 $ID_i = ID^*$，

B 选择 $Q^* \in Z_p^*$，且 $Q^* \notin \{Q_1, Q_2, ..., Q_{q_h}\}$，将 Q^* 返回，并将 $(ID_i,\ Q_i)$ 添加到表 L_1 中；否则，从 $\{Q_1, Q_2, ..., Q_{q_h}\}$ 中选择一个值，返回给 A 并将 $(ID_i,\ Q_i)$ 添加到表 L_1 中。

H_2 询问：B 保持一个列表 $L=\{m, ID, P_i, R, h\}$，初始为空，A 提出对 (m, ID_i, P_i, R_i) 的询问，若列表 L_2 中存在询问项，B 将 h 的值返回给 A；否则，B 随机选择 $h \in Z_q^*$，令 $h = h_2(m \| ID_i \| P_i \| R_i)$，$B$ 将 h 输出给 A，然后把 (m, ID_i, P_i, R_i) 添加到列表 L_2。

部分私钥询问：A_I 对 ID_i 进行询问，B 维持列表 $E=\{ID_i, Q_i, D_i\}$，初始化为空。对于身份 ID_i，B 可以从列表 L_1 中得到元组 $(ID_i,\ Q_i)$，如果 $ID_i = ID^*$，B 结束并返回“失败”。否则，计算 $D_i = \dfrac{1}{a+Q_i}$，返回 D_i 给 A，并将 (ID_i, Q_i, D_i) 添加到表 E。

私钥询问：当 A 对 ID_i 进行询问时，如果 $ID_i = ID^*$，B 结束并返回“失败”。反之，L 存在 (ID_i, x_i, P_i, t) 项、表 E 存在 (ID_i, Q_i, D_i) 项，B 将 (x_i, D_i) 返回 A；若不存在，B 对 ID_i 进行部分密钥询问和公钥询问获得 (x_i, D_i)，并将 (x_i, D_i) 返回给 A。

公钥替换询问：当 A 询问 (ID_i, P_i') 时，若 L 中存在 (ID_i, x_i, P_i, t) 项，B 将 P_i 改为 P_i'，$t=0$。并将 (ID_i, x_i', P_i', t) 添加到 L 表中。否则，B 作公钥询问获得 (ID_i, x_i, P_i, t)，然后设置 $P_i = P_i'$，假设 B 能够获得替换公钥 P_i' 对应的秘密值 x_i'，添加 (ID_i, x_i, P_i, t) 到表 $\sigma = (C_i, R_i, V_i)$。

签名询问：若 A_I 作 (m, ID_i, P_i, R_i) 签名询问，B 查表 L_1 和 L 获得 (ID_i, Q_i) 和 (ID_i, x_i, P_i, t).若 $ID_i = ID^*$，B 终止并返回“失败”。否则，执行下列过程：若 $t=1$，B 随机选择 $r_i \in Z_p^*$，计算 $R_i = g^{ri}$，然后查表 L_2 得到 h_i，计算 $V_i = \dfrac{r_i + h}{x_i} D_i = \dfrac{r_i + h}{x_i} \dfrac{1}{a+Q_i} P'$,则签名为 $\sigma = (C_i, R_i, V_i)$，B 返回 σ 给 A。若 $t=0, B$ 从 A 获得 x_i' 值。选取 $r_i \in Z_q^*$,计算 $R_i = g^{ri}$，计算 $V_i = \dfrac{r_i + h_i}{x_i} D_i = \dfrac{r_i + h_i}{x_i} \dfrac{1}{a+Q_i} P'$,则签名为 $\sigma = (C_i, R_i, V_i)$, B 返回 σ 给 A。

最后使用分叉技术[28]：假设 ID^* 是 A 攻击的目标，ID^* 的公钥是 P_{i^*}，则 A 对消息 m 伪造签名为 $\sigma = (C, R, V)$。通过重放技术，A_I 可以获得另一个有效签名 $\sigma' = (C, R, V')$,且有。σ 和 σ' 满足下列等式：

$$e(V,P_{ID^*})^{g'-h}=e(V',P_{ID^*})^{g'-h'}\Leftrightarrow e(V,P_{ID^*})e(V',P_{ID^*})^{-1}=g_1^{(h-h')}\Leftrightarrow$$
$$e(V-V',P_{ID^*})=g_1^{(h-h')}\Leftrightarrow e(V-V',x^*(P_{pub'}+Q^*P'))=e((h-h')P',P')\Leftrightarrow$$
$$e((V-V')x^*(a+Q^*),P')=e((h-h')P',P')$$

这样，B 能够成功计算 $(V-V')^{x^*}(a+Q^*)$，有 $\frac{1}{a+Q^*}P_i'=(V-V')^{x^*}$ 成立。对于 $Q^*\notin\{Q_1,Q_2,...,Q_{qh}\}$，输出一个对 $(Q^*,\frac{1}{a+Q^*}P_i)$

4.3.4 效率分析

在表 1 中，列出了本文方案与文献[29]所需的运算量。其中，p 代表对运算。分析表 1 可知，我们所提出的方案有更高的效率。

表 1 该文方案与其他方案所需的运算量

方案	签密	解签密	系统参数
文献[29]	*0p*	2*p*	不相同
本方案	*0p*	1*p*	相同

4.4 本章小结

本章提出基于双线性对的从 CLPKC 到 TPKC 的异构签密方案，这个方案是基于随机预言机模型下设计的。该方案在进行签密操作的时候不需要对运算，而进行解签密操作的时候只需要一个对运算，使方案的效率有了一定的提高。

5 匿名 IDPKC→TPKC 异构签密方案

5.1 匿名 IDPKC→TPKC 异构签密方案定义和安全性模型

5.1.1 形式化定义

(1)IDPKC 系统建立算法。输入参数 1^{k_1} ，输出主密钥 s_1 和系统参数 $parmas_1$ 。PKG 保密 s_1 ，公开参数 $parmas_1$ 。

(2)TPKC 系统建立算法。TPKC 系统中 CA（Certification authority）生成并发布参数 $parmas_2$ 。

(3)TPKC 密钥生成算法。该算法产生 TPKC 系统用户的私钥 sk_2 和公钥 pk_2 。

(4)IDPKC 密钥提取算法。IDPKC 系统中的用户使用这个算法获得自己的私钥。用户的公钥就是身份 $pk_1 = ID_1$ ，这种公钥不需要数字证书。

(5)签密算法。该算法输入系统参数 $parmas_1$ 及 $parmas_2$ ，发送者的私钥 sk_1 、接收者的公钥 pk_2 ，消息 m ，输出一个密文 σ 。

(6)解签密算法。该算法输入系统参数 $parmas_1$ 及 $parmas_2$ ，发送者的公钥 pk_1 、接收者的私钥 sk_2 ，密文 σ ，输出消息 m 或者错误符号“⊥”。

这些算法必须满足异构签密的一致性要求，即如果 $\sigma = Signcrypt(sk_1, pk_2, m)$ ，那么 $m = Unsigncrypt(pk_1, sk_2, \sigma)$ 。

5.1.2 安全性模型

机密性

游戏 5.1

初始阶段：

F 运行“IDPKC 系统建立算法”，获得参数 $parmas_1$ 和主密钥 s_1 。 F 运行“TPKC 秘钥生成算法”和“TPKC 系统建立算法”获得系统参数 $parmas_2$ 和接收者的公钥 pk_2 、接收者私钥 sk_2 ,发送参数 $parmas_1$ 、 $parmas_2$ 和接收者的公钥 pk_2 给 F 。

挑战阶段

攻击者 A 判断阶段 1 何时停止进入挑战阶段。产生两个长度相同 m_0 、 m_1 和发送者身份 ID_1 和接收者的公钥 pk_2 并将它们发送给 F 。 F 运行“IDPKC 密钥提取算法”获得发送者的私钥 sk_1 ，然后随机选择一个比特 $\gamma \in \{0,1\}$ 并计算 $\sigma^* = Signcrypt(sk_1, pk_2, m_\gamma)$ 。

猜测阶段

A 输出一个比特 γ，如果 $\gamma' = \gamma$，那么 A 赢得游戏。A 的优势被定义为 $Adv(A) = \Pr[\gamma' = \gamma] - 1/2$，其中 $\Pr[\gamma' = \gamma]$ 表示 $\gamma' = \gamma$ 的概率。

定义 5.1 如果在 t 时间内没有任何多项式有界攻击者，以至少 ε 的优势赢得游戏 5.1，那么称这个 IDPKC→TPKC 异构签密方案是 (ε, t, q_u)-IND-CLPKC-TPKC-HSC-CCA2 安全的。

不可伪造性

游戏 5.2

初始阶段：C 运行“IDPKC 系统建立算法”，获得参数 $parmas_1$。C 运行“TPKC 密钥生成算法”和“TPKC 系统建立算法”获得系统参数 $parmas_2$ 和接收者公钥/私钥 (pk_2, sk_2)，发送参数 $parmas_1$、$parmas_2$ 和 (pk_2, sk_2) 给 F。

攻击阶段：F 执行以下询问：

(1)密钥提取询问：F 选择身份 ID_1，C 运行“IDPKC 密钥提取算法”并将 ID_1 对应的私钥发送给 F。

(2)签密询问：F 提交发送者的身份 ID_1 和消息 m 给 C。C 首先运行“IDPKC 密钥提取算法”以获得发送者的私钥 sk_1，然后运行“签密算法”返回密文 $\sigma = Signcrypt(sk_1, pk_2, m)$ 给 F。

伪造阶段：F 产生发送者的身份 ID_1 和新的密文 σ^*。当满足以下两个条件时候，F 赢得这个游戏。

① σ^* 对于 ID_1 和 pk_2 是合法的密文，即 $Unsigncrypt(ID_1, sk_2, \sigma^*)$ 不会返回错误符号“⊥”。

② 没有询问过涉及 m^*、ID_1 和 pk_2 的签密询问。

③ 在攻击阶段没有询问过 ID_1 的私钥。

定义 5.2 如果没有任何多项式有界的敌手在 t 时间内，以至少 ε 的优势赢得游戏 5.2，那么 IDPKC→TPKC 异构签密方案是 EUF-HSC-CMA 安全的。

匿名性

游戏 5.3

初始阶段：F 运行了“TPKC 密钥生成算法”生成两个公钥和私钥，分别为 $(pk_{2,0}, sk_{2,0})$ 和 $(pk_{2,1}, sk_{2,1})$。我们把它作为密文的接收者，并将公钥和发送给 A。F 运行“IDPKC 密钥提取算法”生成两个公钥和私钥，分别为 $(pk_{1,0}, sk_{1,0})$ 和 $(pk_{1,1}, sk_{1,1})$。我们把它作为密文的发送者，并将私钥 $sk_{1,0}$ 和 $sk_{1,1}$ 发送给 A。

挑战阶段： A 输出消息 m 以及发送者的私钥 $sk_{1,0}$ 和 $sk_{1,1}$。F 选择 $c \in \{0,1\}$，并计算 $\sigma = Signcrypt(m, S_{U,c'}, pk_{r,c})$，获得 σ^* 给 A 作为挑战密文。

猜测阶段：A 随机选择 $d, d' \in \{0,1\}$，如果 $(d,d') = (c,c')$，则 A 赢得游戏 4.3.1 定义 A 赢得游戏的优势为： $Adv(A) = |\Pr[(d,d') = (c,c')] - 1/4|$

定义 5.3: 如果不存在任何多项式有界攻击者以不可忽略的优势赢得游戏 5.3, 则称 IDPKC→TPKC 异构签密方案在适应性选择攻击下具有密文匿名性[30]。

5.2 匿名 IDPKC➞TPKC 异构签密方案详细描述

(1) TPKC 系统建立算法

设 G_{T1} 为由 P_2 生成的循环加法群，阶为 q_2，G_{T2} 为具有相同阶的循环乘法群，$e': G_{T1} \times G_{T1} \to G_{T2}$ 为一个双线性映射。l_2 表示 G_{T1} 元素。其中 l_{2m} 是签密消息的长度， $a_2, b_2 \in Z_{q_2}^*$。发布系统参数 $Params = \{G_{T1}, G_{T2}, q_2, P_2, e', l_2, a_2, b_2\}$。

(2) TPKC 密钥提取算法

TPKC 系统中的用户随机选择 $x_2 \in Z_p^*$，计算自己的私钥 $sk_2 = x_2$ 和公钥 $pk_2 = x_2 P_2$。

(3) IDPKC 系统建立算法

设 G_1 由 P_1 生成循环加法群，阶为 q_1 （q_1 为 k_1 比特素数)，G_2 为具有相同阶 q_1 循环乘法群， $e: G_1 \times G_1 \to G_2$ 为一个双线性对映射。定义三个安全的 $Hash$ 函数[31] $H_1: \{0,1\}^* \to Z_{q_1}^*$、$H_2: \{0,1\} \times G_2 \to Z_{q_1}^*$、$H_3: \{0,1\}^* \to \{0,1\}^{l_{1m}}$ ，其中 l_{1m} 是签密消息的长度， l_1 是 G_1 元素长度。PKG 随机选择一个主密钥 $s_1 \in Z_{q_1}^*$，计算 $P_{pub} = s_1 P_1$，设 $g = e(P_1, P_1)$，PKG 公开系统参数 $Params_1 = \{G_1, G_2, q_1, l_1, P_1, P_{pub}, l_m, g, H_1, H_2, H_3\}$。

(4) IDPKC 密钥提取算法

IDPKC 系统中用户提交身份 ID_U 给 PKG，PKG 计算改用户的私钥 $sk_1 = \dfrac{1}{H_1(ID_1) + s_1} P_1$。

(5) 签密算法

假设发送者 Alice 属于 IDPKC 系统作为发送者，他的公钥为身份 ID_1，私钥为 $sk_1 = \dfrac{1}{H_1(ID_1) + s_1} P_1$；接收者 Bob 属于 TPKC 系统作为接收者，其公钥为

$pk_2 = x_2 P_2$，私钥为 $sk_2 = x_2$。当 Alice 希望通过签密的方式发送消息 m 给 Bob 时，他执行以下过程：

① 随机选择 $r_1 \in Z_{q1}^*, r_2 \in Z_{q2}^*,$，计算 $x = g^{n}$，$V = r_2 P_2$。

② 计算 $h = H_2(m, x)$ 和 $W = (r_1 + h) sk_1$。

③ 计算 $T = r_2 pk_2$ 和 $C = H_3(V, pk_2, T) \oplus (m \| ID_1 \| W \| h)$，则消息 m 的签密密文 $\sigma = (C, V)$。

(6)解签密算法：

① 计算 $\sigma = (C, V)$，$(m \| ID_1 \| W \| h) = C \oplus H_3(V, pk_2, T')$。

② 检查等式 $x = e(W, H_1(ID_1) P_1 + P_{pub}) g^{-h}$ 是否成立。

5.3 匿名 IDPKC→TPKC 异构签密方案安全型和效率分析

5.3.1 正确性

(1)正确解密：当 Bob 收到密文 $\sigma = (C, V)$ 时候，可以正确的解密。

$$
\begin{aligned}
&(m \| ID_1 \| pk_1 \| W \| h) = C \oplus H_3(V, pk_2, T`) \\
&= (m \| ID_1 \| pk_1 \| W \| h) \oplus H_3(V, pk_2, r_2 pk_2) \oplus H_3(V, pk_2, x_2 V) \\
&= (m \| ID_1 \| pk_1 \| W \| h) \oplus H_3(V, pk_2, r_2 x_2 P_2) \oplus H_3(V, pk_2, x_2 r_2 P_2) \\
&= (m \| ID_1 \| pk_1 \| W \| h)
\end{aligned}
$$

(2)正确验证：当 Bob 收到解密密文 $\sigma = (C, V)$ 时候，可以通过等式 $e(V, H_1(ID_1) P_1 + Pub) g^{-h}$

$$
\begin{aligned}
&e(V, H_1(ID_1) P_1 + Pub) g^{-h} \\
&= e((r_1 + h) sk_1, H_1(ID_1) P_1 + P_{pub}) g^{-h} \\
&= e((r_1 + h) \frac{1}{H_1(ID_1) + s_1} P_1, H_1(ID_1) P_1 + P_{pub}) g^{-h} \\
&= e((r_1 + h) \frac{1}{H_1(ID_1) + s_1} P_1, H_1(ID_1) P_1 + s_1 P_1) g^{-h} \\
&= e((r_1 + h) \frac{1}{H_1(ID_1) + s_1} P_1, (H_1(ID_1) + s_1) P_1) g^{-h} \\
&= e((r_1 + h) P_1, P_1) g^{-h} = g^{r_1}
\end{aligned}
$$

5.3.2 机密性

定理 5.1 在随机预言模型下，如果存在一个攻击者 A 能够以 ε 的优势攻破 IDPKC→TPKC 异构签密方案，那么存在一个算法 F 能以 $\frac{\varepsilon}{q_{H_3} + q_{H_2}} \geq \frac{\varepsilon}{q_t}$ 的优势解 CDH 问题。

证明：

初始阶段：C 运行“IDPKC 系统建立算法”，输入安全参数 l_1，产生系统参数 $Parmas_1$ 和主密钥 s_1，设 $P_{pub}=s_1P_1$。C 运行“TPKC 系统建立算法”和“ TPKC 秘钥生成算法”获得参数 $Parmas_2$ 和接收者公钥私钥对 (sk_2, pk_2)，C 发送 $Parmas_1$ 和 $Parmas_2$、挑战者身份 ID_i 和接收者公钥 pk_2 对发送给 A。

阶段 1：C 模拟游戏 1 中 A 的挑战者。H_1 维护 L_1、L_2 和 L_3 三张列表，分别用于跟踪 A 对预言机 H_1、H_2、H_3 的询问。

H_2 询问：当 A 询问 $H_2(m_i, x_i)$ 时，首先检查列表 L_2 是否已经存在这个询问的条目，如果存在，那么返回相同的回答；否则 C 从 Z_{q1}^* 中随机选取 $h_{2,i}$ 作为回答，最后，将元组 $(m_i, x_i, h_{2,i})$ 存入列表 L_2 中。

H_3 询问：C 保持列表 $L_3=(V_i, pk_2, T_i, \omega_i)$,初始为空，$A$ 询问 H_3 预言机，C 首先检查 L_3 中是否存在元组 (V_i, pk_2, T_i)。若存在相应项，则直接返回结果 ω_i；否则随机选择 $\omega_i \in \{0,1\}^{lm}$，返回 ω_i 给攻击者 A，并且将 $(V_i, pk_2, \Delta, \omega_i)$ 存入列表。如果 L_3 中已经存在 $(V_i, pk_2, \Delta, \omega_i)$ 并且有等式 $e(V_i, pk_2)=e(P_2, T_i)$ 成立，则用 T_i 来代替“Δ”。其中符号“Δ”表示 CDH 问题的一个解。

签密询问：攻击者 A 提交消息 m 和一个接收者公钥 pk_w。如果 pk_w 等于 pk_2 或者不合法，则返回错误符号“⊥”，否则执行正常的签密操作，即调用“签密算法”生成并返回密文信息 $\sigma=(V, C)$。

解签密询问：攻击者 A 提交密文 $\sigma=(V, C)$ 给 C，要求得到解密后的结果，F 执行以下操作过程：

C 首先计算 $sk_1=\dfrac{1}{H_1(ID_1)+s_1}P_1$、$V_i=r_2P_2$，然后检查以下等式是否成立：

$x=e(W, H_1(ID_1)P_1+P_{pub})g^{-h}$

I 如果不成立，则输出错误符号“⊥”；

II 如果成立则进一步检查以下等式；$e(V_i, pk_2)=e(V_i, \lambda)$。

①若成立，但 $\lambda \neq \Delta$，则输出消息/签名对 $(m, P_i, V_i, pk_{r,}, \lambda)$ 和发送者的公钥 P_i。

②若成立，且 $\lambda \neq \Delta$，C 失败并终止。

挑战阶段

A 生成两个相同长度的明文 m_0 与 m_1、发送方的私钥 S^* 和接收方的公钥 pk_r 请求 F 返回挑战密文，C 随机选择比特 $b \in (0,1)$,并计算 m_b 在发送者私钥 S^* 和被

攻击者的接收者公钥 pk_r，返回密文 $\sigma_b^* = Signcrypt(m_b, S^*, pk_2)$。$F$ 随机选择 $C^* \in \{0,1\}^{l_2 m}$ 和 $\lambda^* \in G_{T1}$、$r_1 \in Z_{q1}^*$，令 $V^* = bP_2$、$Q_i^* = H_1(ID_i)$、$h^* = H_2(m, x_i)$、$W = (r_1 + h^*)\frac{1}{Q_i^* + s_1}P_1$，$F$ 返回 $\sigma^* = (C^*, V^*)$ 给 A。

猜测阶段

F 从 L_3 中提取随机的条目 (V_i, pk_r, λ)。L_3 中包含 q_{H_3} 项条目，随机选择的条目满足 $e(\gamma pk_1, pk_2) = e(pk_1, \lambda)$ 的概率为 $\frac{1}{q_{H_3}}$，并且 CDH 问题得以解决。

5.3.3 不可伪造性

定理 5.2 在随机预言模型中，若存在一个敌手 F 能够在 t 时间内，以 $\varepsilon \geq 10(q_s+1)(q_s+qH_2)/2^k$ 的优势解决游戏 5.2，则存在一个算法 C，能够在 $t' \leq 120686 q_{H_1} q_{H_2} \frac{t + O(q_s t_p)}{\varepsilon(1-1/2^k)(1-n/2^k)} + O(n^2 t_m)$ 时间内解决 n-SBDH 问题。

证明：

为了签密一个消息 m，生成的元组 (σ_1, h, σ_2) 对应着三阶段诚实验证者零知识认证协议[31]。$\sigma_1 = x$ 是证明者的承诺，$h = H_2(m, x)$ 是根据消息 m 和 σ_1 计算得到的 Hash 中，$\sigma_2 = W$ 是根据 σ_1、h 和私钥 sk_1 的回答。

下面显示 C 可以给 F 提供一个完美的模拟并通过与 F 的交互来解决 n-SDH 问题。C 输入 $(P_1, \alpha P_1, \alpha^2, ...\alpha^n P_1)$，目的在与找到一个对 $(\omega, \frac{1}{\omega+\alpha}P_1)$。

初始阶段：C 随机选择 ω_1，…，$\omega_{\tau-1}, \omega_{\tau+1}, \omega_n \in Z_p^*$。与文献[14]相同，$C$ 使用的输入 $(P_1, \alpha P_1, \alpha^2, ..\alpha^n P_1$，计算一个生成元 $Q_1 \in G_1$ 和 $Q_{pub} = \alpha, Q_1 \in G_1$ 以至于它知道 $n-1$ 对 $(\omega_i, V_i = \frac{1}{\alpha+\omega_i}Q_1)$，$i \in \{1,...,n-1\}$ 为了获得这样的元素，C 展开多项式: $f(z) = \prod_{i=1}^{n-1}(z+\omega_i)\sum_{j=0}^{n-1} c_j z^j$ 生成元 Q_1 和元素 Q_{pub} 可以分别通过 $Q_1 = \sum_{j=0}^{n-1} c_j(\alpha^j$ 1F $\alpha($ 和 $Q_{pub} = \sum_{j=1}^{n} c_{j-1}(\alpha^j P_1) = \alpha f(\alpha) = \alpha Q_1$ 得到。同时 (ω_i, V_i) 可

以通过 $f_i(z)=\frac{f(z)}{z+\omega_i}=\sum_{j=0}^{n-2}d_j z^j$ 并计算 $V_i=\sum_{j=0}^{n-2}d_j(\alpha^j P_1)=f_i(\alpha)P=\frac{f(\alpha)}{\alpha+\omega_i}P$

$=\frac{1}{\alpha+\omega_i}Q_1$得到。PKG 的公钥设为$Q_{pub}$，相应的私钥为$\alpha$

C将系统参数[包括Q_1、$Q_{pub}=\alpha Q_1$和$g=e(Q_1,Q_1)$]发送给F。同时C运行“IDPKC 系统建立算法”产生系统参数$Parmas_1$，设$P_{pub}=s_1P_1$。C运行“TPKC 系统建立算法”和“ TPKC 密钥生成算法”获得参数$Parmas_2$和接收者公私钥对(pk_2,sk_2)，C发送$Parmas_1$和$Parmas_2$ 挑战者身份ID_i和接收者公私钥(pk_2,sk_2)对发送给F。

攻击阶段：C模拟游戏 5.2 中 F 的挑战者。C维护L_1 、L_2、L_3三张列表，分别用于跟踪F对预言机H_1 、H_2、H_3的询问。这里假设每次H_1询问是不同的，目标身份ID_1在某个时候被询问过H_1和身份ID_1在被使用到其它询问之前已经询问过H_1预言机。

H_2询问：当A询问$H_2(m_i,x_i)$时，首先检查列表L_2是否已经存在这个询问的条目，如果存在，那么返回相同的回答；否则C从Z_{q1}^*中随机选取$h_{2,i}$作为回答，最后，将元组$(m_i,x_i,h_{2,i})$存入列表L_2中。

H_3询问：当F询问m时，初始化为空，C首先检查列表L_3是否已经存在元组(V_i,pk_2,T_i)，如果存在，那么直接返回结果β_i；否则，随机选择C从$\beta_i\in\{0,1\}^{l_m}$并返回β_i给攻击者F，并将$(V_i,pk_2,\Delta,\beta_i)$存入$L_3$中。

密钥提取询问：当F询问ID_i的私钥时，如果$ID_i=ID_1$，那么C失败并停止；否则C知道$H_1(ID_i)=\omega_i$，并返回$W_i=\frac{1}{\omega_i+\alpha}Q_1$给$F$。

签密询问：m提交发送者的身份ID_i和消息m给C。如果$ID_i=ID_1$，那么C知道发送者的私钥$S_i=V_i$，可以按照正常的签密步骤来询问。如果$ID_i=ID_1$，那么C执行下列的步骤：

① 随机选择$\theta,h\in Z_{q_1}^*$

② 计算$pk_2=\theta sk_2$和$V_i=r_2P_2$

③ 计算$T'=x_BV_i$

④ $(m\|ID_1\|W\|h)=C\oplus H_3(V_i,pk_2,T')$

⑤ 计算$T=r_2V_i$

⑥ 计算 $C = H_3(V, pk_2, T) \oplus (m \| ID_1 \| W \| h)$

⑦ 返回密文 $\sigma = (C, V)$ 给 F 。

由分叉引理[32]可得，如果 F 上述交互中是一个有效的伪造者，那么可以构造一个算法 F' 。 F' 可以输出同样承诺 x 的两个签名，即 $((ID_1, m), h, W$ 和 $((ID_1, m), h', W')$ ，这里 $h \neq h'$ 。

最后为了解决 n-SDH 问题，这里构造如下算法。

① C 通过运行 F' 得到两个不一样的签名 $((ID_1, m), h, W$ 和 $((ID_1, m), h', W')$ 。

② C 计算 $D^* = (h - h')^{-1}(W - W')$ $\frac{1}{\alpha + \omega_A} Q_1 = \frac{f(\alpha)}{\alpha + \omega_A} P_1$ 。

③ C 使用长除法将多项式 f 写为 $f(z) = \psi(z)(z + \omega_A) + \psi_{-1}$ ，这里，

$\psi(z) = \sum_{i=0}^{n-2} \psi_i z^i$ 和 $\psi_{-1} \in Z_{q1}^*$ 。 $\frac{f(z)}{z + \omega_A}$ 可以写为 $\frac{f(z)}{z + \omega_A} = \psi(z) + \frac{\psi_{-1}}{z + \omega_A}$

$= \sum_{i=0}^{n-2} \psi_i z^i + \frac{\psi_{-1}}{z + \omega_A}$ 。这样 C 就可以计算 $\frac{1}{\alpha + \omega_A} P_1 = \frac{1}{\psi_{-1}} [D^* - \sum_{i=0}^{n-2} \psi_i (\alpha^i P_1)]$ 。

③ C 输出 $(\omega_A, \frac{1}{\alpha + \omega_A} P_1)$ 作为 n-SDH 问题的解。

根据分叉引理，如果 F 能够在 t 时间内，以 $\varepsilon \geq 10(q_s + 1)(q_s + qH_2) / 2^k$ 的优势解决游戏 3.2.1，那么 C 能够在 $t' \leq 120686 q_{H_1} q_{H_2} \frac{t + O(q_s t_p)}{\varepsilon(1 - 1/2^k)(1 - n/2^k)} + O(n^2 t_m)$ 时间内解决 n-SBDHP 问题。

5.3.4 效率分析

在表 1 中，列出了该文方案与文章[33]中所需要的运算量，其中 p 表示对运算， e 表示幂运算。

表 1 本文方案与其他方案所需要的运算量

5.4 本章小结

本章所提出的 IDPKC→TPKC 异构签密方案不仅保证在安全通信过程中信息的机密性和不可伪造性；其次，与文献[20]相比效率有了一定的提高；最后，方案中 IDPKC 和 TPKC 密码环境中使用了不同的系统参数，从而更加有效的模

方案	签密	解签密	系统参数	匿名性
文献[20]	0*p*+1*e*	2*p*+1*e*	相同	不满足
本文方案	0*p*+1*e*	1*p*+1*e*	不相同	满足

拟了实际的应用场景；同时方案实现了通信过程中的匿名性。

6 总结与展望

6.1 总结

密码学是信息安全的核心。大部分发送方或者接收方都是在相同的密码环境进行通信(即同构密码环境)[34]。在现实生活中，不同的应用场景，所采用的密码环境是不一样的，如果要在采用不同密码环境的系统之间进行安全通信，就需要支持异构通信的签密体制。本文针对以上问题，在下面 3 个方面做了初步研究:

(1) 高效的 TPKC→IDPKC 的异构签密方案:在已有 Sun 等人方案的基础上提出一种高效的异构签密方案，适应于 TPKC 和 IDPKC 两种不同的密码环境，进行安全通信。

(2) 基于双线性对 CLPKC→TPKC 的异构签密方案：该方案的优点是在进行签密操作的时候不需要对运算，而进行解签密操作的时候只需要一个对运算，效率有了较大的提高；其次，该方案具有“密钥隐私”[35]的特性。

(3) 匿名 IDPKC→TPKC 异构签密方案：以上所提出的两种方案，虽然在效率上有了一定的提高，同时安全性有了很大的改进，相对于以前的方案，但是仍然存在一定的不足，不同密码环境所采用的参数是一样的，不能更好的模拟现实的应用场景，而且。因此，提出了一个匿名的异构签密方案，不仅在 IDPKC 和 TPKC 密码环境中采用了不同的系统参数，更好的模拟现实的应用场景；而且该方案实现了密文的匿名性，并且满足签密过程中的内部安全性[36]，从而保护了发送方和接收方的隐私；同时在效率上有了很大的提高。

6.2 展望

(1) 本文所提出的异构签密方案，不仅在效率有了一定的提高，而且，在安全性方面有了很大的改进，更重要的是在不同密码体制之间所采用的参数都是相同的，满足不同实际应用场景的需求。

(2)随着 5G 异构网络、大数据的推广，异构密码环境将逐渐普遍。未来工作可能设计标准模型[37]下安全的异构签密，或者设计处具有特殊性质的异构签密，门限异构签密[38]、代理异构签密体制[39]、盲异构签密体制[40]、环异构签密体制[41]等。还可以将异构签密应用于异构网络中，如云计算[42]和物联网[43] 等领域。