PSEC-KEM 自己評価書

1　まえがき

本資料では、鍵共有を目的とする公開鍵暗号方式（暗号スキーム）「PSEC-KEM」の安全性および性能に関する自己評価について述べる。

2　設計方針および理論・技術

PSEC-KEM は、楕円 ElGamal 暗号関数（基本暗号プリミティブ）を [1] における変換方式で高い安全性を持つスキームに変換した（秘匿目的）暗号方式（PSEC-2 と呼んでいたスキーム）を鍵共有方式（Key Encapsulation Mechanism）に改良した方式 [4]（PSEC-KEM と呼ばれているもの）である。

PSEC-KEM は、実用上様々な特長を持つ楕円曲線暗号関数（楕円 ElGamal 暗号関数）の利点を引き継ぐと共に、最も高いレベルの安全性（適応的選択暗号文攻撃に対して強秘匿）をもつことが証明された暗号方式である。以下、安全性、性能についての概要を示す。

2.1　安全性の概要

PSEC-KEM は、楕円曲線上の Diffie-Hellman 計算 (CDH) 仮定とランダムオラクルモデルの下で、適応的選択暗号文攻撃に対して強秘匿である。

楕円曲線上の実用的かつ最強の意味の安全性の証明のついた他の公開鍵暗号（鍵共有／秘匿）としては、楕円 Cramer-Shoup 暗号や ECIES が知られている。楕円 Cramer-Shoup 暗号は、ランダム関数に対する仮定が現実的な汎用一方向性ハッシュ関数である点で優れているが、整数論的な仮定では（PSEC-KEM で仮定する CDH 仮定よりも強い仮定である）楕円 Diffie-Hellman 決定 (DDH) 仮定に基づいている。一方、PSEC-KEM は、理想的なランダム関数（ランダムオラクルモデル）に基づいているものの、整数論的な仮定では基本的な楕円曲線上の Diffie-Hellman 計算 (CDH) 仮定に基づいている点で楕円 Cramer-Shoup 暗号よりも優れている。また、ECIES は、ランダムオラクルモデルの下で、整数論的な仮定が楕円曲線上の Gap-Diffie-Hellman (GDH) 仮定（CDH 仮定よりも強い仮定である）である。（ECIES は、ランダムオラクルモデルを用いない証明も可能であるが、その代わりにオラクルハッシュという非常に特殊な仮定が必要となる。他との比較のためには、ランダムオラクルモデルによる証明を用いる方が適切である。）

表 1: 安全性の比較

方式	安全性 （最強の意味で）	整数論的 仮定	ランダム関数 仮定	帰着の効率
PSEC-KEM	安全性証明つき	楕円 CDH	真にランダム	**
楕円 Cramer-Shoup	安全性証明つき	楕円 DDH	UOWHF	*
ECIES	安全性証明つき	楕円 GDH	真にランダム	**
楕円 ElGamal	攻撃可	–	–	–

（注： UOWHF は、汎用一方向性ハッシュ関数を意味する。また、帰着の効率の欄で ** は、ほぼ optimal であること、 * はそれに比べて効率が悪いことを意味する。）

2.2 性能の概要

PSEC-KEM の性能を他の代表的な方式と比べる。鍵サイズ、暗号文サイズでは、楕円 Cramer-Shoup が他に比べてかなり長くなるが、その他の方式はほぼ同等である。以下の表では、速度に関する比較を示す。各方式とも、ハッシュ関数の処理量は楕円曲線上の乗算演算の処理量に比べほぼ無視できるため、ここでは楕円乗算演算の回数のみで比較を行なう。

表 2: 処理量の比較

方式	暗号化 （乗算演算回数）	復号化 （乗算演算回数）
PSEC-KEM	2	2
楕円 Cramer-Shoup	5	3 (4*)
ECIES	2	1 (2*)
楕円 ElGamal	2	1

（注： $*$ を付けたものは、安全性証明上の理由で、復号処理において暗号文がベースポイントのつくる部分群に入っていることを検証する必要がある場合の演算回数である。 PSEC-KEM では、このような検証は必要ないが、 Cramer-Shoup, ECIES では必要となる場合がある。）

3 PSEC-KEM の安全性証明

本節では、 PSEC-KEM の安全性に関する結果（楕円曲線上の Diffie-Hellman 計算 (CDH) 仮定とランダムオラクルモデルの下で、 PSEC-KEM が適応的選択暗号文攻撃に対して強秘匿であること）を示す（[4] 参照）。

ここで、 MGF をランダムオラクルと考える。つまり、以下のような 2 つの独立なランダムオラクルを想定する。

$$G : \mathbf{B}_{hLen} \to \mathbf{B}_{pLen+128+KeyLen},$$

$$H : \mathbf{B}_{32+2\cdot qmLen} \to \mathbf{B}_{hLen}.$$

また、簡単のため、 EC で $\mathrm{ECP2OSP}(C_1, qmLen)$ を、また EQ で $\mathrm{ECP2OSP}(Q, qmLen)$ を表すものとする。

定理 3.1 $\mathcal{A}$ を、復号オラクル、ランダムオラクル G, H にそれぞれ q_D 回、 q_G 回、 q_H 回の質問をし、アドバンテージが ε で実行時間が t であるような *PSEC-KEM* $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ に対する *IND-CCA2*（適応的選択暗号文攻撃に対して強秘匿）における攻撃者とする。このとき、成功確率が ε' で実行時間が t' の（$\mathcal{K}$ に関する）楕円 *Diffie-Hellman* 問題の回答を含む $(q_H + q_D)$ 個の値のリストを出力するようなアルゴリズムが存在して、

$$\varepsilon' \geq \frac{\varepsilon}{2(1+2^{-128})} - \frac{(q_G + 3q_D)(1+2^{-128})}{p} - \frac{q_D + q_G}{2^{hLen}}.$$

$$t' \leq t + q_H \cdot (T + \mathcal{O}(1)).$$

ここで T は $\mathcal{K}$ に関する楕円曲線上の乗算 *2* 回を行う計算時間である。

注： Diffie-Hellman 問題の回答を含む $(q_H + q_D)$ 個の値のリストを出力するようなアルゴリズムが存在すれば、そのアルゴリズムを用いて効率的に Diffie-Hellman 問題の回答を抽出することができる [3, 4]（Diffie-Hellman 問題の持つランダム帰着性の性質を用いる）。

以降の評価において、下記の補題をしばしば用いる。

補題 3.2 任意の事象 E, F *and* G に対して、以下が成立する。

$$\Pr[\mathsf{E} \wedge \mathsf{F} \mid \mathsf{G}] \leq \begin{cases} \Pr[\mathsf{E} \mid \mathsf{F} \wedge \mathsf{G}] \\ \Pr[\mathsf{F} \mid \mathsf{G}]. \end{cases}$$

定理 3.1 を以下のような 3 段階の手順で証明する。まず最初に、（計算的） Diffie-Hellman (CDH) 問題を破る問題を PSEC-KEM を IND-CCA2の意味で破る敵に帰着させるアルゴリズムを示す。その次に、この帰着における復号オラクルのシミュレーションが圧倒的確率で効率的に実行できることを示す。最後に、上で述べた復号オラクルシミュレーションを含めて帰着の全体の成功確率ならびに実行時間を評価する。

注：鍵配送機能（key encapsulation mechanism: KEM）における IND（つまり、強秘匿）の定義は、（公開鍵）暗号における INDの定義と若干違っている。この定義については、文献 [4] の 2.2 節を参照されたい。

3.1 帰着アルゴリズム

$\mathcal{A}$ を PSEC-KEM $(\mathcal{K}, \mathcal{E}, \mathcal{D})$ の IND-CCA2攻撃者とする。$\mathcal{A}$ は、時間 t 以内で、$\mathcal{A}$ は、復号オラクル、G- オラクル、H- オラクルにそれぞれ q_D 回、q_G 回、q_H 回の質問を行ない、鍵 K が正しいものかランダムに選ばれたものかの判断を ε 以上のアドバンテージで行なう。帰着アルゴリズム $\mathcal{B}$ は以下のとおりである。

3.1.1 トップレベルの記述

1. $\mathcal{B}$ は、曲線 E 上の点 P、W を含む公開鍵 PK と別の点 C_1^* を与えられる。帰着 $\mathcal{B}$ の目的は、(P, W, C_1^*) が与えられたとき以下を満足するような CDH 問題の回答 Q^* を含む値のリストを求めることである。$\log_P W = \log_{C_1^*} Q^*\ (= s)$.

2. $\mathcal{B}$ はランダムなビット b と二つのランダムなビット列 $c_2^* \in \mathbf{B}_{hLen}$ と $K \in \mathbf{B}_{keyLen}$ を生成する。$\mathcal{B}$ は A にこの公開鍵と暗号文 $c^* = (EC^*, c_2^*)$ （ここで、$EC^* = \mathrm{ECP2OSP}(C_1^*, qmLen)$）ならびに K を入力として与え、実行させる。$\mathcal{B}$ は、復号オラクルやランダムオラクル G, H への A からの質問に対する回答をシミュレーションする。これらシミュレーションの方法は下で述べる。

3. A は最終的に K に対する判断 b' を出力する。$\mathcal{B}$ は、H への質問の（特に EQ の部分の）全リストを出力する（その中に正しい EQ^* が含まれていることが期待される）。

3.1.2 ランダムオラクル G、H のシミュレーション

ランダムオラクルシミュレーションは、G、H の回答をシミュレーションするとともにその質問／回答をそれぞれ G-List、H-List として以下のように管理する。それらの初期値

は空リストである。（同じ値が質問された場合は、このリストに従い同じ回答を返す。）

- H オラクルに対して、$EC \neq EC^*$ であるような最初に聞かれた質問 $\delta = (EC, EQ)$ に対しては、シミュレータはランダムな値 $H(\delta)$ を回答し、その対 $(\delta, H(\delta))$ が H-List に追加される。G オラクルに対して、最初に聞かれた質問 r に対しては、シミュレータはランダムな値 $G(r)$ を回答し、その対 $(r, G(r))$ が G-List に追加される。
- H オラクルに対して、最初に聞かれた質問 $\delta^* = (EC|^*, EQ)$ に対しては、シミュレータはランダムな値 $H(\delta^*)$ を回答し、その対 $(\delta^*, H(\delta^*))$ が H-List に追加される。次に、$r = c_2^* \oplus H(\delta^*)$ を求め、$GList$ にその値があるかどうかを探す。もし無ければ、$t = [G(r)]^{pLen+128}$ として $t \in \mathbf{B}_{pLen+128}$ をランダムに定める。それを用いて、$\alpha = \mathrm{OS2IP}(t) \bmod p$ を計算し、$C_1^* = \alpha P$ を満足するかどうかを確認する。もし満足すれば、$Q^* = \alpha W$ を計算する（これが、求めたい CDH 問題 (P, W, C_1^*) の回答であり、出力する）。もし、$EQ = \mathrm{ECP2OSP}(Q^*, qmLen)$ ならば、$[G(r)]_{keyLen} = K$ （$b = 0$ のとき）とするか、$[G(r)]_{keyLen}$ をランダムに定める（$b = 1$ のとき）。この場合、(EC^*, c_2^*) は、正しい暗号文（暗号オラクルの正しい出力）である。$C_1^* \neq \alpha P$ のときは、$[G(r)]_{keyLen}$ をランダムに定める。

3.1.3 復号オラクルのシミュレーション

復号オラクルへの質問 $c = (EC, c_2)$ に対して、シミュレータ $\mathcal{DS}$ は $\delta = (EC, EQ)$（EQ は任意）が質問であるような対 $(\delta, H(\delta)) \in$ H-List を探す。もしそのような対が H-List にない場合、“Reject” を返す。もしそのような対が H-List にある場合、$r = c_2 \oplus H(\delta)$ を計算し、$GList$ に r があるかどうかを調べる。もしなければ、G の r への回答 $G(r)$ をランダムに選び、$GList$ に追加する。次に、$t = [G(r)]^{pLen+128}$ として、$\alpha = \mathrm{OS2IP}(t) \bmod p$ を計算し、$EC = \mathrm{ECP2OSP}(\alpha P)$ および $EQ = \mathrm{ECP2OSP}(\alpha W)$ を満足するかどうかをチェックする。もしいずれかが満足されなければ、“Reject” を返す。いずれも満足されたならば、$\mathcal{DS}$ は、$[G(r)]_{keyLen}$ を鍵 k として出力する。

3.1.4 補足

上記のシミュレーション過程で Q^* を発見したとき、それを出力して以降の処理を終了することもできるが、ここでの解析では、帰着処理を継続して行ない、最後に、$\mathcal{B}$ は、その値を出力するか、H への質問のリストを出力する。

シミュレーション環境下での α の分布は実環境での α の分布とは異なる。つまり、シミュレーション環境下での α の分布は $\{0, 1, \ldots, p-1\}$ 上で一様であるが、実環境での α

の分布は、t が $\mathbf{B}_{pLen+128}$ 上で一様であるとき $\alpha = \mathrm{OS2IP}(t) \bmod p$ であり、一様分布から少しだけ偏っている。

ここで用いる楕円曲線上の点の符号化は全単射写像なので、$C_1 = C_1'$ ならばかつそのときに限り $EC = EC'$ であり、$Q = Q'$ ならばかつそのときに限り $EQ = EQ'$ である。

3.2 記法

まず最初に、星印 $(^*)$ のついた変数はすべて、暗号オラクルから出力された解読対象暗号文 $c^* = (EC^*, c_2^*)$ に関連するものである。それ以外の変数は、敵から復号オラクルに質問された暗号文に関連するものである。

- AskH は、(EC^*, EQ^*) が H に質問される事象を意味し、AskG は、r^* が G に質問される事象を意味する。
- GBad は、r^* $(= c_2^* \oplus H(EC^*, EQ^*))$ が G に質問され、かつ $EC^* \neq \mathrm{ECP2OSP}(\alpha P)$, もしくは $EC^* = \mathrm{ECP2OSP}(\alpha P)$ であるが $[G(r^*)]_{keyLen} \neq K$ （$b = 0$ のとき）であるような事象のことを意味する。ここで、$t^* = [G(r^*)]^{pLen+128}$、$\alpha = \mathrm{OS2IP}(t^*) \bmod p$ である。ここで、GBad ならば AskG であることに注意されたい。上で見たように、GBad は、選択平文攻撃において、シミュレーションが失敗する唯一の事象である。
- Fail は、上記の復号オラクルシミュレータが q_D 個の回答のうち少なくとも 1 個の間違った復号結果を返す事象である。
- Bad = GBad ∨ Fail.
- CBad = RBad ∨ EBad。ここで、
  - EBad は、$EC = EC^*$ となる事象;
  - RBad は、$r = r^*$ となる事象;
- AskRE = AskR ∧ AskE。ここで、
  - AskR は、r $(= c_2 \oplus H(EC, EQ))$ が G に質問される事象;
  - AskE は、(EC, EQ) が H に質問される事象;

Fail という事象は、$\mathcal{DS}$ が暗号文を不正としたにもかかわらず復号オラクルが正しい暗号文と判断する場合に限定されることに注意。実際、$\mathcal{DS}$ が暗号文を正しいものと判断したときは、復号オラクルが必ず正しい暗号文とする。

3.3 復号オラクルシミュレーションの解析

3.3.1 安全性の命題

補題 3.3 最大 *1* 個の暗号文 $c^* = (EC^*, c_2^*)$ を暗号オラクルから受け取るとき、復号オラクルシミュレータ $\mathcal{DS}$ は、q_D 個の質問（暗号文; $c = (EC, c_2)$, $c \neq c^*$）に対して復号オラクルと同じ回答を成功確率 ε_1 で、実行時間 t_1 で出力することができる。ここで、

$$\varepsilon_1 \geq 1 - \left(\frac{(q_G + 3q_D)(1 + 2^{-128})}{p} + \frac{q_D}{2^{hLen}} \right),$$

$$t_1 \leq q_H \cdot (T + \mathcal{O}(1)),$$

であり、T は αP と αW を計算する計算時間である。

解析を始める前に、もう一度復号オラクルシミュレータの動作を確認しておこう。シミュレータは、暗号オラクルから暗号文 c^* ならびにランダムオラクルシミュレータとのやりとりの結果 G-List and H-List を受け取り、（敵から受け取った）暗号文 c を復号する。もし、暗号文が敵により正しく作られていれば（つまり、r が G に問い合わされ、(EC, EQ) が H に問い合わされる）、シミュレーションは正しく復号できる。それ以外の場合には、“Reject”を出力するが、敵は、ランダムオラクル G、H へ質問せずに正しい暗号文を作ることができるかもしれない。

3.3.2 成功確率

ここでの目標は、事象 Fail の生起確率を求めることである。¬CBad ∧ AskRE を認めれば、シミュレーションは完全であり、Fail の生起確率は 0 である。従って、その補事象を考える：

$$\Pr[\mathsf{Fail}] = \Pr[\mathsf{Fail} \wedge \mathsf{CBad}] + \Pr[\mathsf{Fail} \wedge \neg\mathsf{CBad} \wedge \neg\mathsf{AskRE}].$$

まず、最初の項 $\Pr[\mathsf{Fail} \wedge \mathsf{CBad}]$ について考えよう。$\mathsf{CBad} = \mathsf{RBad} \vee \mathsf{EBad}$ であるが、ここで RBad は決して起こらない。なぜなら、$r = r^*$ ならば $\alpha = \alpha^*$, $C_1 = C_1^*$ (i.e., $EC = EC^*$)、$Q = Q^*$、$H(EC, EQ) = H(EC^*, EQ^*)$ となる。従って、$c_2 = c_2^*$、つまり $c = (EC, c_2)$ は $c^* = (EC^*, c_2^*)$ と等しい。そのような c は、復号オラクルへの質問としては許されない。よって、$\mathsf{CBad} = \mathsf{EBad}$。そこで、$\Pr[\mathsf{Fail} \wedge \mathsf{EBad}]$ を評価しよう。これは、$c_2 \neq c_2^*$ (i.e., $r \neq r^*$)、$\mathrm{OS2IP}([G(r)]^{pLen+128}) \equiv \mathrm{OS2IP}([G(r^*)]^{pLen+128}) \pmod{p}$ であるような $c = (EC^*, c_2)$ が q_D 個の質問に含まれている確率である。それぞ

れの c がこの関係を満足する確率は、高々 $\frac{1+2^{-128}}{p}$ である。敵は、q_G 回の確認する（潜在的）機会と q_D 回の復号オラクルへの質問の機会がある。従って、以上より

$$\Pr[\mathsf{Fail} \wedge \mathsf{EBad}] \leq \frac{(q_G + q_D)(1 + 2^{-128})}{p}.$$

次に、後の項、$\Pr[\mathsf{Fail} \wedge \neg\mathsf{CBad} \wedge \neg\mathsf{AskRE}]$、の評価を行う。$\mathsf{CBad} = EBad$ and $\mathsf{AskRE} = \mathsf{AskR} \wedge \mathsf{AskE}$ なので、この項の値は、以下となる。

$$\Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge (\neg\mathsf{AskR} \vee \neg\mathsf{AskE})]$$

$$\leq \Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge \neg\mathsf{AskR}] + \Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge \mathsf{AskR} \wedge \neg\mathsf{AskE}].$$

この前の項に対して、r は G に質問されなく r^* から独立なので、c が正しい（つまり、$\alpha = \mathrm{OS2IP}([G(r)]^{pLen+128}) \pmod p$ かつ $EC = \mathrm{ECP2OSP}(\alpha P)$) 確率は高々 $\frac{1+2^{-128}}{p}$ である。敵は、q_D 回の質問を復号オラクルにする機会があるので、

$$\Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge \neg\mathsf{AskR}] \leq \frac{q_D(1 + 2^{-128})}{p}.$$

この後の項、$\Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge \mathsf{AskR} \wedge \neg\mathsf{AskE}]$、に対して、$r$ は G に質問されるが、(EC, EQ) は H に質問されず (EC^*, EQ^*) から独立である。そこで、$r' = c_2 \oplus H(EC, EQ)$ が r と等しいか $G(r')$ が C_1 と整合が取れている確率は、高々 $\frac{1}{2^{hLen}} + \frac{1+2^{-128}}{p}$ である。敵は q_D 回の質問を復号オラクルにする機会があるので、

$$\Pr[\mathsf{Fail} \wedge \neg\mathsf{EBad} \wedge \neg\mathsf{AskR}] \leq q_D(\frac{1}{2^{hLen}} + \frac{1 + 2^{-128}}{p}).$$

以上より、

$$\Pr[\mathsf{Fail}] \leq \frac{(q_G + 3q_D)(1 + 2^{-128})}{p} + \frac{q_D}{2^{hLen}}.$$

このシミュレーションの実行時間を評価する。ここでは、復号オラクルへの質問暗号文 $c = (EC, c_2)$, それに対応する H-List に含まれる (EC, EQ) の値ならびに対応する G-List の r の値から得られる α のすべての値に対して、αP and αW の計算を行う。従って、このシミュレーションの実行時間は高々

$$q_H \cdot (T + \mathcal{O}(1)),$$

であり、T は αP と αW の実行時間である。

3.4 帰着の成功確率

ここでは、この方式に対する IND-CCA2の敵のアドバンテージに関して帰着の成功確率を求める。帰着の目標は、楕円曲線のパラメータと (P, W, C_1^*) を受け取って、Q^* を含む q_H からなる値のリストを出力することである。つまり、この帰着の成功確率は、事象 AskH が起こる確率により得られる。

そこでまず、事象 AskH を事象 Bad に関して分解して考える。

$$\Pr[\mathsf{AskH}] = \Pr[\mathsf{AskH} \wedge \mathsf{Bad}] + \Pr[\mathsf{AskH} \wedge \neg\mathsf{Bad}].$$

最初に、前の項を評価する。

$$\begin{aligned}
\Pr[\mathsf{AskH} \wedge \mathsf{Bad}] &= \Pr[\mathsf{Bad}] - \Pr[\neg\mathsf{AskH} \wedge \mathsf{Bad}] \\
&\geq \Pr[\mathsf{Bad}] - \Pr[\neg\mathsf{AskH} \wedge \mathsf{GBad}] - \Pr[\neg\mathsf{AskH} \wedge \mathsf{Fail}] \\
&\geq \Pr[\mathsf{Bad}] - \Pr[\mathsf{GBad} \,|\, \neg\mathsf{AskH}] - \Pr[\mathsf{Fail}] \\
&\geq \Pr[\mathsf{Bad}] - \Pr[\mathsf{AskG} \,|\, \neg\mathsf{AskH}] - \Pr[\mathsf{Fail}] \\
&\geq \Pr[\mathsf{Bad}] - \frac{(q_G + 3q_D)(1 + 2^{-128})}{p} - \frac{2q_D + q_G}{2^{hLen}}.
\end{aligned}$$

ここで、$\Pr[\mathsf{Fail}] \leq \frac{(q_G+3q_D)(1+2^{-128})}{p} + \frac{q_D}{2^{hLen}}$ は、補題 3.3 より得られ、$\Pr[\mathsf{GBad} \,|\, \neg\mathsf{AskH}] \leq \Pr[\mathsf{AskG} \,|\, \neg\mathsf{AskH}]$ は、事象 GBad が起こるならば必ず事象 AskG が起こることより得られる。¬AskH ならば、$H(EC^*, EQ^*)$ は予測不可能であり、$r^* = c_2^* \oplus H(EC^*, EQ^*)$ もまた予測不可能である。従って、$\Pr[\mathsf{AskG} \,|\, \neg\mathsf{AskH}] \leq \frac{q_G+q_D}{2^{hLen}}$.

次に後の項を評価する。

$$\begin{aligned}
\Pr[\mathsf{AskH} \wedge \neg\mathsf{Bad}] &= \Pr[\neg\mathsf{Bad}] \cdot \Pr[\mathsf{AskH} \,|\, \neg\mathsf{Bad}] \\
&\geq \Pr[\neg\mathsf{Bad}] \cdot \Pr[\mathcal{A} = b \wedge \mathsf{AskH} \,|\, \neg\mathsf{Bad}] \\
&\geq \Pr[\neg\mathsf{Bad}] \cdot \left(\Pr[\mathcal{A} = b \,|\, \neg\mathsf{Bad}] - \Pr[\mathcal{A} = b \wedge \neg\mathsf{AskH} \,|\, \neg\mathsf{Bad}]\right).
\end{aligned}$$

ここで、¬AskH ならば、$H(EC^*, EQ^*)$ は予測不可能であるため、$r^* = c_2^* \oplus H(EC^*, EQ^*)$ も予測不可能であり、同様に b も予測不可能となる。このことは、¬Bad と独立な事象である。従って、$\Pr[\mathcal{A} = b \wedge \neg\mathsf{AskH} \,|\, \neg\mathsf{Bad}] \leq \Pr[\mathcal{A} = b \,|\, \neg\mathsf{AskH} \wedge \neg\mathsf{Bad}] = 1/2$. シミュレーションにおける $c^* = (EC^*, c_2^*)$ の分布は、実環境での c^* の分布と若干異なっている（$(1 + 2^{-128})$ 倍のバイアスがある）ため、

$$\frac{\varepsilon}{2(1 + 2^{-128})} + \frac{1}{2} \leq \Pr[\mathcal{A} = b] \leq \Pr[\mathcal{A} = b \,|\, \neg\mathsf{Bad}] \cdot \Pr[\neg\mathsf{Bad}] + \Pr[\mathsf{Bad}].$$

従って、

$$\Pr[\mathsf{AskH}\wedge\neg\mathsf{Bad}] \geq \left(\frac{\varepsilon}{2(1+2^{-128})} + \frac{1}{2} - \Pr[\mathsf{Bad}]\right) - \frac{\Pr[\neg\mathsf{Bad}]}{2} = \frac{\varepsilon/(1+2^{-128}) - \Pr[\mathsf{Bad}]}{2}.$$

前の項と後の項の評価を合わせ、また $\Pr[\mathsf{Bad}] \geq 0$ という事実を用いると、以下が得られる。

$$\Pr[\mathsf{AskH}] \geq \frac{\varepsilon}{2(1+2^{-128})} - \frac{(q_G+3q_D)(1+2^{-128})}{p} - \frac{2q_D+q_G}{2^{hLen}}.$$

3.5 帰着時間の評価

この帰着の実行時間のほぼすべては、復号オラクルのシミュレーション時間なので、帰着の全体の実行時間は、

$$t' = t + q_H \cdot (T + \mathcal{O}(1)),$$

であり、T は αP と αW の実行時間である。

4 実装評価

この章では、 PSEC-KEM を C 言語で実装した場合の速度およびメモリ使用量を示す。使用したパラメータは、技術仕様書で示した推奨パラメータである。

測定環境は以下の通りである。

CPU	Intel Pentium-III 600MHz
Memory	128 Mbytes
OS	Microsoft Windows2000 SP2
コンパイラ	Microsoft Visual Studio 6.0 Enterprise Edition

速度測定は、演算にかかった clock 数を測定することにより行なった。ランダムなデータを用いた 1000 回の平均をとっている。

実行速度は以下の通りである。

鍵生成	5.64 ms
暗号化	11.09 ms
復号化	10.97 ms

メモリ使用量は以下の通りである。

鍵生成	7.30 Kbytes
暗号化	2.64 Kbytes
復号化	2.39 Kbytes

参考文献

[1] Fujisaki, E. and Okamoto, T.: Secure Integration of Asymmetric and Symmetric Encryption Schemes, Proc. of Crypto'99, Springer-Verlag, LNCS 1666, pp. 535–554 (1999).

[2] Bailey, D. V. and Paar, C.: Optimal Extension Fields for Fast Arithmetic in Public-Key Algorithms, Proc. of Crypto'98, LNCS 1462, Springer-Verlag, pp.472-485 (1998).

[3] V. Shoup. Lower Bounds for Discrete Logarithms and Related Problems. In *Eurocrypt'97*, LNCS, Springer-Verlag, Berlin, 1997.

[4] V. Shoup. A Proposal for an ISO Standard for Public Key Encryption (v.2.0). ISO/IEC JTC1/SC27, N2563, http://shoup.net/papers/, 2001 Sep.