ACE暗号攻撃評価報告書

目 次

1 はじめに

本報告では, Advanced Cryptographic Engine に含まれる公開鍵暗号方式に対する安全性評価を行う. 評価は数論仮定と関数仮定のもとで示された適応的選択暗号文攻撃に対する安全性証明の検証と, 用いられた数論仮定について推奨パラメータ長での安全性の評価の 2 点を行った. 2 章で提案暗号の概要, 3 章では公開鍵暗号の安全性証明理論についての概略を述べ, 4 章, 提案暗号アルゴリズムの紹介, 5, 6 章, 数論仮定として用いられた Diffie-Hellman 問題, および一方向性ハッシュ関数についての簡単な説明, 7 章で, 提案暗号方式の安全性の解析, 最後に 8 章で結論を述べる.

2 ACE 暗号の概要

Advanced Cryptographic Engine(以下 ACE) は公開鍵暗号方式だけでなく電子署名方式も実装しているソフトウェア・ルーチンライブラリである. 以後本報告の対象となる ACE に含まれる公開鍵暗号方式を ACE 暗号と呼ぶ. ACE 暗号は, 次に示す 3 つの仮定のもとで安全性の証明ができることを特徴としている.

(1) Diffie-Hellman(DDH) 仮定.

(2) SHA-1 第 2 プレイメージ衝突耐性.

(3) MARS 累積/カウンタ・モードの疑似ランダム性.

またランダムオラクルモデルを仮定した場合 (1) は計算量的 Diffie-Hellman 仮定に置き換えることができる. ACE 暗号が達成している安全性について 2 章以降で説明する.

2.1 仕様書の概要

仕様書は次のような構成になっている.

1 章 はじめに
ACE 暗号の概略と仕様書の構成について述べられている.

2 章 安全性の目標
公開鍵暗号方式の証明可能安全性についての解説. 公開鍵暗号の安全性証明理論, ACE 暗号の安全性を証明する上で前提とされている, Diffie-Hellmann 問題, SHA-1 第 2 プレイメージ衝突耐性, ブロック暗号 MARS の累積/カウンタ・モード擬似ランダム性などについて述べられている.

3 章 用語と表記法

4 章 暗号方式
ACE 暗号のアルゴリズムの詳細 (ハッシュ関数や対称暗号の仕様も含む), および ACE 暗号の安全性の解析, 証明が記述されている.

5 章 ASN.1 鍵の構文

6 章 パフォーマンス

4 章で述べられている安全性の解析では, ACE 暗号についての安全性証明を, [5], [21] に述べられている基本となる暗号方式の解析方法などを引用しながら行っている. そこでは ACE 暗号の解読成功確率を, DDH の成功確率や, ハッシュ関数のプレイメージ探索成功確率, 擬似乱数生成器の出力と乱数との判別成功確率などを用いて評価しており, 攻撃ゲームを様々に変形しながら最終の評価式を導くという手法がとられている.

2.2 知られている攻撃法

現在のところ ACE 暗号に対して特化された攻撃は発表されていない.

3 公開鍵暗号の安全性

ACE 暗号の安全性を説明するために, まず一般に公開鍵暗号方式に対して構築されている安全性理論を紹介する.

公開鍵暗号への攻撃のタイプは, 暗号文のみから解読を試みる受動的攻撃と, 送信者に暗号文 (解読対象以外の) を送り復号結果を得ることが許され, そこで得た情報を利用して解読対象の暗号文を解読する能動的攻撃に分類される.

一方, 公開鍵暗号は通信内容の秘匿を目的に用いられるため, 通信内容の秘匿の度合いが重要になる. これを暗号の秘匿性と呼ぶ. さらに暗号文から平文の内容を知ることはできないが, 暗号文を操作することにより, 対応する平文の意図的な改ざんを行うことができる可能性がある. このような意図的な改ざんが一切できないことを頑強性 (non-malleable 性) と呼ぶ. これら秘匿性と頑強性をまとめて強度と呼ぶ. 以上の 2 つの観点より, 公開鍵暗号の安全性を以下のように分類する.

- 受動的攻撃 (passive attack)
  - 暗号文攻撃 (ciphertext-only attack) :暗号文だけを利用する攻撃.
  - 選択平文攻撃 (chosen-plaintext attack : CPA) ：任意の平文に対応する暗号文を，暗号化関数をオラクルとして用いることで得ることのできる状況下において，解読対象である暗号文を解読しようとする攻撃. (公開鍵暗号では暗号化鍵が公開されているため，常に実行可能) .
- 能動的攻撃 (active attack)
  - 非適応的選択暗号文攻撃 (non-adaptive chosen-ciphertext attack : CCA1) ：攻撃対象の暗号文が与えられる前の段階においてのみ，任意の暗号文 (攻撃対象を除く) に対する平文を，復号化関数をオラクルとして用いる (**decryption oracle**) ことで得ることのできる状況下において，解読対象である暗号文を解読しようとする攻撃.
  - 適応的選択暗号文攻撃 (adaptive chosen-ciphertext attack : CCA2) ：ターゲットとなる暗号文が与えられる前後に関わらず，任意の暗号文 (攻撃対象を除く) に対する平文を，復号化関数をオラクルとして用いる (**decryption oracle**) ことで得ることのできる状況下において，解読対象である暗号文を解読しようとする攻撃.

この定義から，CPA, CCA1, CCA2 の順でより強力な攻撃法であることが分かる.
さらに，強度の観点から次のように分類される.

- 秘匿性
  - 完全解読困難 ： 一方向性 (one-way : OW) ：任意の暗号文の平文を完全に求めることが困難であること.
  - 部分解読困難 ： 暗号文から平文の部分情報を求めることが困難であること.
  - 強秘匿 (semantically secure / indistinguishable : IND) ：どのような部分情報も部分解読困難なこと.

- 頑強性 (non-malleability : NM) ：暗号文 $y = E(x)$ に対して，$R(x, x_1, x_2, \ldots, x_k)$ となる関係 R と暗号文 $y_i = E(x_i)$ $(1 \leq i \leq k)$ を作成することが困難であること (但し，E は暗号化関数) .

以上の定義から，公開鍵暗号の安全性として，{ 強度 }-{ 攻撃法 } の組み合わせを考える．例えば，NM-CCA2 は適応的選択暗号文攻撃に対して頑強であることを意味する．これらの安全性の定義間の関係を図 1 に示す．($A \to B$ とは，ある公開鍵暗号が A であれば必ず B であることを意味する．$A \not\to B$ はその否定である.)．ここで，特に重要な結果としては，IND-CCA2 と NM-CCA2 の等価性が示されていることである．このことから，IND-CCA2 または NM-CCA2 である公開鍵暗号が最も安全な公開鍵暗号であると考えられている．これらに属し, かつ実用的である公開鍵暗号としては OAEP, EPOC 等が知られている.

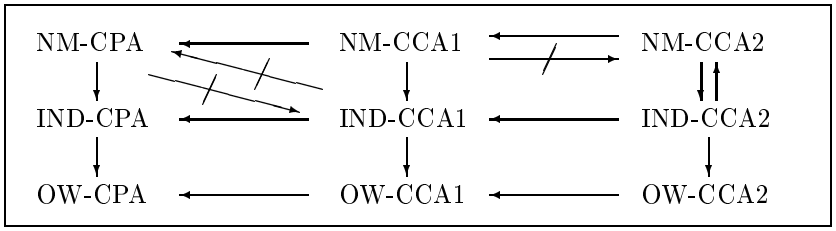

図 1: 公開鍵暗号の安全性の定義間の関係

後の考察のため, IND-CCA2 の概念をより詳しく説明する.

攻撃者は encryption oracle にメッセージ m_0, m_1 を与え, encryption oracle はコイントスにより $b = 0$ or 1 を選択, m_b を暗号化したのち攻撃者に送り返す．攻撃者がこの暗号文が m_0, m_1 いずれの暗号文であるかを当てるゲームを考える．その際, 攻撃者は decryption oracle にターゲットである暗号文以外の任意の暗号文を復号してもらうことができる．しかも, それは問題となる暗号文を手にする前でも構わない．ある公開鍵暗号が IND-CCA2 であるとは, 任意の確率的多項式時間アルゴリズム (攻撃者) に対しても, このようなゲームにおいて攻撃者が 1/2 よりよい確率で正解しないことをいう.

暗号化関数が決定性の関数 (同じ平文に対して常に同じ暗号文を出力する) である場合にはこの概念は全く意味をなさない．IND-CCA2 は確率暗号に対しての概念である.

上記の定義をより正確に定式化することもできるがここでは省略する.

4 ACE 暗号のアルゴリズム概要

ACE 暗号は, 公開鍵暗号を用いて秘密鍵暗号である MARS 暗号の鍵共有を行い, 実際の平文の暗号化は, ブロック暗号 MARS を擬似乱数生成器として用い, 平文と MARS の出力との排他的論理和を暗号文とするハイブリッド方式を採用している．その際, MARS は累積/カウンタ・モードで用いられ, 同時にメッセージ認証も行う．また暗復号化の過程で用いられるハッシュ関数として SHA-1 を用いて構成された関数も合わせて提案されている．本報告では, 特に Diffie-Hellman 判定問題との関連を中心に評価を行うため, ハッシュ関数, MARS の使用モード 等の詳細には触れず, ACE 暗号を簡略化したものを評価対象とする.

4.1 鍵生成

入力: サイズ・パラメータ $1024 \leq m \leq 16,384$.

出力: 公開鍵 $(P, q, g_1, g_2 j_1, k_2)$, 秘密鍵 (w, x, y, z_1, z_2).

1. $|q| = 256$ なる素数 q をランダムに生成する.
2. $P \equiv 1 \pmod{q}$, $|P| = m$ なる素数 P をランダムに生成する.
3. $g_1^q \equiv 1 \pmod{P}$, $g_1 \in \mathbb{Z}_P^*$, $g_1 \neq 1$ なる整数 g_1 をランダムに生成する.
4. $w, x, y, z_1, z_2 \in \mathbb{Z}_q$ なる整数 w, x, y, z_1, z_2 をランダムに生成する.
5. 次の整数を計算する.
$$g_2 = g_1^w, \quad c = g_1^x, \quad d = g_1^y, \quad h_1 = g_1^{z_1}, \quad h_2 = g_1^{z_2}.$$
6. ハッシュ鍵 k_1, k_2 をランダムに選択する.
7. 公開鍵/秘密鍵のペア $(P, q, g_1, g_2, c, d, h_1, h_2, k_1, k_2)$, (w, x, y, z_1, z_2) を出力して終了.

4.2 暗号化

入力: 平文 m, 公開鍵 $(P, q, g_1, g_2, c, d, h_1, h_2, k_1, k_2)$.

出力: 暗号文 (s, u_1, u_2, v, e).

1. 128 ビット列 s を任意に生成する.
2. $r \in \mathbb{Z}_q$ なる r をランダムに生成する.
3. 暗号文のプリアンブル (u_1, u_2, v) を生成する.
   3.1. $u_1 = g_1^r$, $u_2 = g_2^r$ をそれぞれ計算する.
   3.2. 160 ビットの出力を持つ汎用一方向性ハッシュ関数 H を用いて $0 \leq \alpha < 2^{160}$, $\alpha = H(k_1, s, u_1, u_2)$ なる整数 α を計算する.
   3.3. $v = c^r d^{\alpha r}$ を計算する.
4. 対称暗号の鍵 k を計算する.
   4.1. $\tilde{h}_1 = h_1^r$, $\tilde{h}_2 = h_2^r$ をそれぞれ計算する.
   4.2. 256 ビットの出力を持つ, 汎用一方向性ハッシュ関数 H' を用いて $0 \leq k < 2^{256}$, $k = H'(k_2, s, u_1, \tilde{h}_1, \tilde{h}_2)$ なる整数 k を生成する.
5. 鍵長が 256 ビットの対称暗号 C 及び鍵 k を用いて, $e = C_k(m)$ を計算する.
6. 暗号文 (s, u_1, u_2, v, e) を出力して終了.

4.3 復号化

入力: 暗号文 (s, u_1, u_2, v, e), 公開鍵 $(P, q, g_1, g_2, c, d, h_1, h_2, k_1, k_2)$, 秘密鍵 (w, x, y, z_1, z_2).

出力: 復号文 m.

1. 汎用一方向性ハッシュ関数 H を用いて $\alpha = H(k_1, s, u_1, u_2)$ なる整数 α を計算する.
2. $v \neq u_1^{x+\alpha y}$ の場合, Reject を出力して終了.
3. 対称暗号の鍵を計算する.
   3.1. $\tilde{h}_1 = u_1^{z_1}$, $\tilde{h}_2 = u_2^{z_2}$ をそれぞれ計算する.
   3.2. 汎用一方向性関数 H' を用いて $0 \leq k < 2^{256}$, $k = H'(k_2, s, u_1, \tilde{h}_1, \tilde{h}_2)$ なる整数 k を生成する.
4. $m = C_k(e)$ を計算する.
5. m を出力して終了.

5 Diffie-Hellman 問題

G を素数位数 q の群とし $g \in G$ を生成元とする．このとき計算量的 Diffie-Hellman(CDH) 問題とは g^x と g^y から g^{xy} を計算する問題のことであり, 計算量的 Diffie-Hellman 仮定とは計算量的 Diffie-Hellman 問題が計算量的に困難であるという仮定である.

与えられた $g_1, g_2, u_1, u_2 \in G$ に対して, $u_1 = g_1^x, u_2 = g_2^x$ を満たす x が存在するか否かを判定する問題を決定性 Diffie-Hellman 問題 (DDH) という. この問題が計算量的に困難であるという仮定を決定性 Diffie-Hellman 仮定という.

これらの問題の困難性は, 任意の確率的多項式時間アルゴリズムに対し, 問題に正解する確率が十分小さいという形で定式化することができるが, ここでは詳細には触れない.

Diffie-Hellman 問題の持つ重要な性質として自己ランダム帰着性がある.

明らかに G 上の離散対数問題 (DL) が解ければ CDH と解くことができ, また CDH が解ければ DDH を解くことができる. しかしながら現在のところ, DDH, CDH と DL との計算量的同値性は, 特殊な条件下での結果が若干知られているのみであり, 一般的には未解決である.

6 汎用一方向性ハッシュ関数

汎用一方向性ハッシュ関数は, Naor と Yung([17]) により導入された概念である. 鍵付きハッシュ関数 H が, 平文 x が選択され, さらにランダムな鍵 k 与えられた場合において, $H_k(x) = H_k(y)$ となるような $y \neq x$ を求めることが難しいという条件を満たしてとき H を汎用一方向性関数と呼ぶ.

明らかに分かるように, ハッシュ関数に対する要求条件としては, 汎用一方向性ハッシュ関数の方が衝突困難ハッシュ関数よりも弱い.

7 安全性解析

ACE 暗号はいくつかの仮定のもとで安全性が証明できる暗号方式であり, 安全性の評価は (1) 安全性証明の検証, (2) 安全性証明のために用いられた仮定についての安全性検討 の 2 点について行う必要がある. この章では安全性の証明を説明し, 次に数論仮定 (DDH) の安全性について考察する.

7.1 安全性の証明

ACE 暗号は次の定理により安全性を保障する.

定理 7.1. *汎用一方向性ハッシュ関数と, 擬似ランダム性を持つ対称暗号の存在, および決定性 Diffie-Hellman 問題 (DDH) の困難性を前提として, ACE 暗号は適応的選択暗号文攻撃に対して安全 (IND-CCA2) である.*

以下, 定理 7.1 の証明の概略を説明する．特に断ることなく前章までに用いた記号をそのまま用いる.

IND-CCA2 の定義で想定しているゲームにおいて，攻撃者の攻撃実行時間が高々t, decryption oracle への質問回数が高々 κ, テストメッセージ m_0, m_1 の長さの上限が l の場合の攻撃者の優位性 (正解確率-1/2) を

$$\mathsf{AdvEnc}(t, \kappa, l)$$

と書く.

同様に実行時間が高々 t の DDH に対する, 出力 0, 1 の統計テスト全体の集合を $\mathbf{T}$ とするとき

$$\mathsf{AdvDDH}(t) = \max_{T \in \mathbf{T}}\{|\Pr[T(\mathbf{R}) = 1] - \Pr[T(\mathbf{D}) = 1]|\}$$

($\Pr[E]$ は事象 E が生じる確率) とおく.

hash 関数 H に対し, 実行時間が高々 t の, 第 2 プレイメージを見つけるアルゴリズムを全て動かしたときの最大確率を

$$\mathsf{AdvH}(t)$$

とかく.

最後に擬似乱数を生成するアルゴリズム C に対し, 実行時間が高々 t, C の l-bit の出力分布 $\mathbf{P}_l$ と, l-bit のランダムな系列の分布 $\mathbf{R}_l$ とを区別する全ての統計テストの集合を $\mathbf{T}$ とするとき

$$\mathsf{AdvC}(t, l) = \max_{T \in \mathbf{T}}\{|\Pr[T(\mathbf{R}_l) = 1] - \Pr[T(\mathbf{P}_l) = 1]|\}$$

とおく.

定理 7.1 は次の定理から導かれる：

定理 7.2. l' を P のバイト長とするとき

$$\begin{aligned}\mathsf{AdvEnc}(t, \kappa, l) \quad \leq \quad & \mathsf{AdvDDH}(O(t)) + \\ & \mathsf{AdvH}(O(t))(\lceil l/64 \rceil + \lceil (2\lceil l'/4 \rceil + 4)/16 \rceil) + \\ & \mathsf{AdvC}(O(t), 65\lceil l/1024 \rceil + 7) \cdot 2 + \\ & (2\kappa + 3)/q + (\kappa + 2)/2^{128}.\end{aligned}$$

ここで定義をする：

定義 7.1. 暗号文が **vaild** とは, $\log_{g_1}(u_1) = \log_{g_2}(u_2)$ であることをいう. そうでないとき **invalid** という.

実行時間 t, 長さ高々 l のテストメッセージについて decryption oracle への高々 κ 回の質問をする攻撃ゲームを考える. もともとのスキームに対する攻撃ゲームを G_0 とし, S_0 を攻撃者がゲーム G_0 の中で秘匿ビット b をあてる事象とする. (以下, 同様にゲーム G_i 内で秘匿ビットをあてる事象を S_i とおく)

このとき定義により, $\mathsf{AdvEnc}(t, \kappa, l) = |\Pr[S_0] - 1/2|$ である. 定理の証明は攻撃ゲームを変形していきながらこの値の上限評価をしていくことで行われる.

Game G_1. スキームを次のように変形する: プライベート鍵を x_1, x_2, y_1, y_2, z_{11}, z_{12}, z_{21}, z_{22}, 公開鍵を

$$c = g_1^{x_1} g_2^{x_2}, \quad d = g_1^{y_1} g_2^{y_2}, \quad h_1 = g_1^{z_{11}} g_2^{z_{12}}, \quad h_2 = g_1^{z_{21}} g_2^{z_{22}}$$

とし, さらに復号化において,

$$u_1^q \equiv u_2^q \equiv 1 \pmod{P}, \quad u_1^{x_1 + y_1\alpha} u_2^{x_2 + y_2\alpha} \equiv v \pmod{P}$$

を確認するステップを挿入する. 最後に対称暗号 (C) の鍵の導出において,

$$\tilde{h}_1 = u_1^{z_{11}} u_2^{z_{12}}, \quad \tilde{h}_2 = u_1^{z_{21}} u_2^{z_{22}}$$

とする. [5] と同様の議論 (攻撃者に公開されている情報から reject されない invalid な暗号文をつくるためには, あるいくつかの超平面の共通部分として与えられる曲線 (line) 上の点をあてなければならい...), および簡単な確率計算 (*cf.* 仕様書 Lemma 4.10.1) により

$$|\Pr[S_1] - \Pr[S_0]| \leq \frac{2\kappa}{q} \tag{1}$$

が得られる.

Game G_2. [5] で用いられた議論と同様に (*cf.* [5] §4, Lemma 2, Claim 2, Case 3), encryption oracle の動きを次のように変形する:

encryption oracle がつくるターゲットの暗号文において u'_1, u'_2 を (位数 q の) ランダムな値とする. また encryption oracle は $v' = (u'_1)^{x_1+y_1\alpha'}(u'_2)^{x_2+y_2\alpha'}$ を計算する. このとき

$$|\Pr[S_2] - \Pr[S_1]| \leq \frac{2\kappa}{q} \tag{2}$$

が成り立つ.

Game G_3. G_2 において攻撃者が decryption oracle に $(s, u_1, u_2) \neq (s', u'_1, u'_2)$ であって, $\alpha = \alpha'$ なる暗号文を submit する事象を V_2 とおく. G_3 では $(s', u'_1, u'_2, \alpha')$ の計算を攻撃に先駆けて行うこととして, V_2 が生じた場合は攻撃を終了させることとする. このとき [21] と同様の議論により,

$$|\Pr[S_3] - \Pr[S_2]| \leq \mathsf{AdvH}(O(t))\lceil(2\lceil l'/4\rceil + 4)/16\rceil \tag{3}$$

が成り立つ.

Game G_4. G_4 ではさらに encryption oracle において $\tilde{h}'_1$, $\tilde{h}'_2$ を (位数 q の) ランダムな値とする. [5] と同様の議論により, invalid な暗号文が reject されない確率は高々 $1/q$ となり, これにより

$$|\Pr[S_4] - \Pr[S_3]| \leq \frac{\kappa + 1}{q} \tag{4}$$

が得られる.

Game G_5. G_5 では対称暗号の鍵をランダムなものに置き換える. このとき entoropy smoothing theorem ([15], [12]), および $\tilde{h}'_1$, $\tilde{h}'_2$ が少なくとも 2^a, $a = 256 + 2 \times 127$ 個の元を持つ集合からランダムに選ばれていることから

$$|\Pr[S_5] - \Pr[S_4]| \leq \frac{2}{2^{128}} \tag{5}$$

が得られる.

Game G_6. decryption oracle に質問された暗号文のプリアンブル (s, u_1, u_2, v) がターゲットの暗号文のプリアンブルと等しい場合には reject するよう decryption oracle を変更する. このとき次が示される：

$$\begin{aligned} |\Pr[S_6] - \Pr[S_5]| \quad \leq \quad & \mathsf{AdvC}(O(t), 65\lceil l/1024\rceil + 7) \\ & + \mathsf{AdvH}(O(t))\lceil l/64\rceil + \kappa/2^{128}. \end{aligned} \tag{6}$$

Game G_7. G_6 の encryption oracle における擬似乱数生成アルゴリズムの出力を乱数に置き換える. このときゲーム G_7 はランダムな出力をあてるゲームとなり,

$$|\Pr[S_7]| = \frac{1}{2} \tag{7}$$

であり, また

$$|\Pr[S_7] - \Pr[S_6]| \leq \mathsf{Adv}\mathsf{C}(O(t), 65\lceil l/1024 \rceil + 7) \tag{8}$$

がわかる.

これらのゲーム間における攻撃成功確率の差の評価 (2)∼(10) をまとめることで定理を証明することができる.

以上の証明において, 考察の不足, 理論的ギャップ等は特に認められず, ACE 暗号の安全性証明には問題はないと考える.

7.2 数論仮定 (DDH) の安全性

現在のところ, DDH, CDH を解くためには離散対数問題 (DL) を解くことが必要であるため, ここでは推奨パラメータ長のうち最小のものについて DL を考える.

現在知られている有限体の乗法群における DL への攻撃アルゴリズムは, 次の 2 種類に大別される:
(1) 対象となっている部分群の位数のサイズに依存したもの (ACE 暗号では q のサイズに依存).
(2) 有限体の元の個数に依存したもの (ACE 暗号では P のサイズに依存).

(1) に属するものとしては Shanks, Pohlig-Hellman, Pollard のアルゴリズムなどがあり, (2) としては指数計算法として, Adleman, Coppersmith, Gordon のアルゴリズムなどが知られている.

有限素体の乗法群 $\mathbf{Z}_p^*$ における離散対数計算法として, 現在比較的有効とされているアルゴリズムは Gordon のアルゴリズム ([11]) で, これは素因数分解のために考案された数体ふるい法に基づいており, 一般数体ふるい法によるものと特殊数体ふるい法によるものに分けられる. 一般と特殊は p の性質の違いによる. この Gordon のアルゴリズムは計算量評価に用いられる関数

$$L_p[a, b] = \exp((b + o(1))(\log p)^a (\log\log p)^{1-a})$$

を用いると"一般"の場合, $L_p[1/3, 2.08008]$ であり, "特殊"では $L_p[2/5, 1.00475]$ と評価されている (素因数分解に一般数体ふるい法を用いた場合, $L_p[1/3, 1.901]$). この評価からは漸近的に"一般"の方が高速に動くことがわかるが, 実は p が極めて巨大にならなければ"特殊"の方がやはり早いことが知られている.

攻撃アルゴリズム (1) に属する代表的なものとして Pohlig-Hellman のアルゴリズムがあるが, これは $p - 1$ の素因子が大きければ有効に働かない.

ACE 暗号で用いられている群は, P は 1024 ビット以上, q は 256 ビットとされている.

この群に対して, 上記攻撃法の適応を考えると, まず Pohlig-Hellman 法に対しては $P - 1$ が大きな素因子として q を含むので適応は難しいと考えられる.

次に Gordon 法を適応した場合を考えると, "特殊"版の条件 (ここでは詳細な記述は避けるが, 大雑把には, 整数係数多項式 f であって, $\mathbf{Z}_P$ において解を持ち, かつ係数が適当に小さくなるものが存在するような素数 P のことを"特殊"といっている) に当てはまらない P を選択しさえすれば, "一般"版を用いることになり, その場合, 現在の計算機能力では困難とされる 1024 ビット RSA タイプ合成数の素因数分解と同等以上の強度を持つことがわかる.

ACE 暗号の仕様書には上記 Gordon のアルゴリズムなどに対して耐性を持つ素数 (鍵) の生成について特に記述がなく, 注意が必要と思われるが, 素数サイズの設定には全く問題はないと考える.

8 結論

本報告では ACE 暗号に対して以下の観点から安全性の評価を行った.

(1) 汎用一方向性ハッシュ関数と, 擬似ランダム性を持つ擬似乱数生成器の存在, および決定性 Diffie-Hellman 問題の困難性を前提としての, 適応的選択暗号文攻撃に対しての安全性.
(2) 推奨パラメータ長での Diffie-Hellman 仮定の安全性.

(1) については [5], [21] で構築された理論に基づいた形で仕様書に記載されている安全性の証明を検証した. 結果, 証明の不足, 誤り等は特に認められず, 主張通り ACE 暗号はいくつかの仮定のもとで適応的選択暗号文攻撃に対して強秘匿性 (または頑強性) を持つことがわかった. 従って ACE 暗号は現在の公開鍵暗号方式に対する安全性の概念の中で, 最強暗号に属すことがわかった.

安全性証明のための仮定には汎用一方向性ハッシュ関数の存在が含まれるが, この仮定は, 安全性証明可能を主張する他の暗号の多くが前提とするいわゆる理想的ハッシュ関数の存在 (ランダムオラクルモデル) に比べると弱い (現実的) 仮定と考えられ, 比較的現実的な仮定のもとで安全性が証明され得る希少な公開鍵暗号方式であるといえる.

(2) については決定性 Diffie-Hellman 問題の困難性を直接は評価せず, 対象となっている群上の離散対数問題の困難性について考察した. その結果, 指数計算法の適応に対して耐性を持つために素数 P の選択方法に注意が必要と思われるが, 適切な選択をするならば, パラメータ長は十分大きく取られているため現在の計算能力では離散対数を求めることは困難であり, 安全性には問題がないことがわかった. ただし, 離散対数問題の困難性と Diffie-Hellman 問題の困難性との等価性については未解決な部分があることに注意すべきと考える.

参考文献

[1] M. Bellare and P. Rogaway, Random oracles are practical: a paradigm for designing efficient protocols, In First ACM Conference on Computer and Communications Security, pp.62-73 (1993).

[2] M. Bellare and P. Rogaway. : Optimal asymmetric encryption – How to encrypt with RSA, *Advances in Cryptology – Eurocrypt'94*, LNCS 950, Springer-Verlag, pp.92-111 (1994)

[3] C. Burwick, D. Coppersmith, E. D'Avignon, R. Gennaro, S. Halevi, C. Jutla, S. Matyas Jr., L. O'Connor, M. Peyravian, D. Safford, and N. Zunic, MARS-a candidate cipher for AES, June 1998.

[4] D. Coppersmith, Modifications to the Number Field Sieve, J. Cryptology, 6, 3, pp.169-180 (1993).

[5] R. Crammer and V. Shoup, A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack, In Advances in Cryptology-Crypto '98, pp.13-25 (1998).

[6] C. Curry, The NFSNET Project, `http://orca.st.usm.edu/ cwcurry/nfs/nfs.html`

[7] I. Damgard, Collision free hash functions and public key signature schemes, In Advances in Cryptology-Eurocrypt (1987).

[8] W. Diffie and M. E. Hellman, New directions in cryptography, IEEE Trans. Info. Theory **22**, pp.644-654 (1976).

[9] T. ElGamal. : A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Trans. Information Theory*, IT-31, 4, pp.469-472(1985).

[10] S. Goldwasser and M. Bellare. : *Lecture Notes on Cryptography*, `http:/www-cse.ucsd.edu/users/mihir/` (1997).

[11] D. Gordon, Discrete logarithms in $GF(p)$ using the number field sieve, SIAM J. Discrete Math. **6**, pp.124-138 (1993).

[12] R. Impagliazzo and D. Zuckermann, How to recycle random bits, In 30th Annual Symposium on Foundations of Computer Science, pp.248-253 (1989).

[13] H. W. Lenstra Jr., Factoring Integers with Elliptic Curves, Annals of Math., 126 pp.649-673 (1987).

[14] A. K. Lenstra, H. W. Lenstra Jr., M. S. Manasse and J. M. Pollard, The Number Field Sieve, Proc. of STOC, pp.564-572 (1990).

[15] M. Luby, Pseudorandomness and Cryptographic Applications, Princeton University Press, 1996.

[16] A. Menesez, P. Van Oorschot, and S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

[17] M. Naor and M. Yung, Universal one-way hash functions and their cryptographic applications, In 21st Annual ACM Symposium on Thoery of Computing, 1989.

[18] T. Okamoto and S. Uchiyama, : A new public-key cryptosystem as secure as factoring, *Advances in Cryptology – Eurocrypt'98*, LNCS 1403, Springer-Verlag, pp.308-318 (1998).

[19] 岡本龍明，山本博資.：現代暗号，産業図書 (1997).

[20] R. L. Rivest, A. Shamir and L.Adleman. : A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, Vol.21, No.2, pp.120-126 (1978).

[21] V. Shoup, A composition theorem for universal one-way hash functions, In Advances in Cryptology-Eurocrypt2000 (2000).

[22] V. Shoup, Using hash functions as a hedge against chosen ciphertext attack, In Advances in Cryptology-Eurocrypt2000 (2000).

[23] D. Simon, Finding collisions on a one-way street: can secure hash functions be based on general assumptions?, In Advances in Cryptology-Eurocrypt '98, pp.334-345 (1998).

[24] P. Zimmerman, The ECMNET Project, `http://www.loria.fr/ zimmerma/records/ecm-net.html`