IMES DISCUSSION PAPER SERIES

公開鍵暗号の理論研究における最近の動向

宇根正志・岡本龍明

Discussion Paper No. 98-J-28

IMES

INSTITUTE FOR MONETARY AND ECONOMIC STUDIES
BANK OF JAPAN

日本銀行金融研究所

〒100-8630 東京中央郵便局私書箱 203 号

IMES Discussion Paper Series 98-J-28
1998 年 11 月

公開鍵暗号の理論研究における最近の動向

宇根正志*1・岡本龍明*2

要 旨

公開鍵暗号は、暗号化と復号に異なる鍵を用いる暗号であり、復号に用いる鍵を秘密にする一方、暗号化に用いる鍵を公開することができる。公開鍵暗号は、公開する鍵の真正性を確保する仕組みが必要となるものの、①事前に通信相手に鍵を配送する必要がない、②通信相手確認や受信データの真正性確認等を可能にするディジタル署名を実現できる、等の利点を有することから、インターネット等オープンなネットワークにおける情報セキュリティ技術として幅広く利用されている。

公開鍵暗号の理論研究は、1976 年に Diffie と Hellman によって公開鍵暗号のアイデアが提案されて以来、多くの暗号学者によって進められてきた。これまでに RSA 暗号や ElGamal 暗号をはじめとする様々な暗号方式が提案されている。

最近の公開鍵暗号の理論研究において特に注目されているのは、証明可能な安全性と実用性を兼ね備えた暗号方式に関する研究である。現在実用化されている暗号方式の中には、これまでに効率的な解読法が見つかっていないものの、安全性が厳密な意味で証明されていないものが多い。したがって、それらの暗号方式について、効率的な解読法が存在する可能性を否定することはできない。これに対し、最近では、OAEP や EPOC 等、既存の方式に改良を加えることによって証明可能な安全性と実用性を両立させる暗号方式が提案されている。OAEP 等これらの方式の一部は、既に実用化されている暗号プロトコルの安全性を高める目的から、PKCS#1 等いくつかの業界標準で採用されている。

公開鍵暗号の安全性証明に関する研究は、公開鍵暗号を利用した様々なシステムにおける信頼性を高める上で、今後一層重要になると考えられる。本稿では、こうした最近の安全性証明に関する研究の動向を中心に、これまでの主要な公開鍵暗号に関する研究成果について説明する。

キーワード：安全性証明、公開鍵暗号、実用性、ディジタル署名

JEL Classification：L86、L96、Z00

*1 日本銀行金融研究所研究第 2 課（E-mail: masashi.une@boj.or.jp）
*2 日本電信電話株式会社情報通信研究所（E-mail: okamoto@sucaba.isl.ntt.co.jp）

本論文は、1998 年 11 月 4 日に日本銀行で開催された「金融分野における情報セキュリティ技術に関するシンポジウム」への提出論文に加筆・修正を加えたものである。

目　次

頁

Ⅰ. はじめに

公開鍵暗号は、復号に用いられる鍵を秘密に管理する必要がある一方、暗号化に用いられる鍵は公開することができる暗号方式である。公開鍵暗号には、公開する鍵の真正性を確保する仕組みが必要となるものの、①事前に通信相手に鍵を配送する必要がない、②各利用者は自分の秘密鍵と公開鍵のみを管理すればよい、③通信相手の確認や受信データの真正性確認等を可能にするディジタル署名を実現できる、等の利点が存在する。このため、インターネット等不特定多数の利用者が参加するオープンなネットワークにおいては、情報セキュリティを確保するための必要不可欠な技術として位置付けられている。

公開鍵暗号の理論研究は、1976 年に Diffie と Hellman によって公開鍵暗号のアイデアが提案されて以来、多くの暗号学者によって進められてきた。これまでに RSA 暗号や ElGamal 暗号をはじめとする様々な暗号方式が提案されているほか、各暗号方式の安全性に関する研究成果が数多く発表されている。現在実用化されている公開鍵暗号方式の多くは、これまでに効率的な解読法が見つかっていないものの、安全性が厳密な意味で証明されているわけではない。したがって、そのような暗号方式について、効率的な解読法が存在する可能性を否定することはできない。

最近の公開鍵暗号の理論研究において特に注目されているのは、証明可能な安全性と実用性を兼ね備えた暗号方式に関する研究である。RSA 暗号等既存の方式に改良を加えることによって、証明可能な安全性と実用性を両立させる暗号方式がいくつか提案されており、データ守秘専用の方式としては、OAEP、EPOC、Cramer-Shoup 暗号が挙げられるほか、ディジタル署名専用の方式としては、PSS 署名、改良 ElGamal 署名、TDH-ESIGN 署名が挙げられる。これらの方式は、攻撃者が暗号文しか解読に利用できない受動的攻撃だけではなく、より強力な能動的攻撃に対しても、一定の仮定の下で安全であることが証明されている。

こうした証明可能な安全性を有する暗号方式を業界標準に採用し、既に実用化されている暗号プロトコルのセキュリティ水準を高める動きもみられている。RSA 社が開発した暗号通信データ形式の規格 PKCS#1 Version 1（RSA 暗号を利用）は、SSL 等の暗号プロトコルに組み込まれており、幅広い分野で利用されているが、PKCS#1 Version 1 をある一定の条件の下で実装した場合には能動的攻撃によって効率的に解読することが可能となるとの研究が発表された。これを受けて、RSA 社は、能動的攻撃に対する安全性が証明されている OAEP を利用して PKCS#1 Version 1 に改良を加え、新たに PKCS#1 Version 2 を発表している。

公開鍵暗号の安全性証明に関する研究は、公開鍵暗号を利用した様々なシステムにおける信頼性を高める上で、今後一層重要になってくると考えられる。本稿では、こうした公開鍵暗号の安全性証明に関する最近の理論研究の動向を中心に、これまでの主要な公開鍵暗号に関する研究成果について説明する。

Ⅱ．公開鍵暗号の概要と機能

公開鍵暗号は、暗号化に利用される鍵と復号に利用される鍵が異なり、一方の鍵から他方の鍵を算出することが計算量的に困難である[1]ため、どちらか一方の鍵を公開することが可能となる。通常、暗号化に利用される鍵が公開されることから公開鍵と呼ばれ、復号に利用される鍵が秘匿されることから秘密鍵と呼ばれる。

1. 公開鍵暗号の特徴

公開鍵暗号は、共通鍵暗号と比べて次のような特徴を持っている。

①鍵配送が容易である

共通鍵暗号では、送信者は受信者に鍵を安全に配送しなければならない。一方、公開鍵暗号では、送信者の公開鍵の正当性を確認する仕組みが必要となる[2]ものの、暗号化に利用する鍵を公開することができるため、受信者に鍵を配送する必要はなくなる。

②各利用者は自分の秘密鍵・公開鍵のみを管理すればよい

n 人の利用者が互いに暗号通信を行う場合、共通鍵暗号においては、各利用者が n-1 個の鍵を秘密に管理する必要があるため、利用者が増加すると鍵の管理が煩瑣になる。これに対して、公開鍵暗号では、各利用者は自分の秘密鍵と公開鍵を管理するだけでよい。このため、利用者が膨大となるオープンなネットワークにおける利便性が高い。ただし、各利用者の公開鍵が正当であることを確認するための仕組み（公開鍵インフラ）が別途必要となる。

③ディジタル署名に利用できる

小切手や契約書等紙ベースでの署名（サイン）は、その署名特有の形状から署名者を一意に特定することを可能にし、署名が付された文書の作成者等を確定させる役割を有している。公開鍵暗号を利用したディジタル署名では、署名の対象が紙ベースの文書からディジタルデータに置き換わり、（i）署名者がデータを秘密鍵で変換する署名生成の

---

[1] 計算量的に困難であるとは、その計算を行うことは理論的には可能であるものの、実際にその計算を実行するには計算量が非常に大量となり、膨大な費用と時間を必要とすることから、事実上不可能であることを意味する。どの程度の計算量が「事実上不可能」であるかは、その時々の技術条件等によって左右される。

[2] 一般的には、公開鍵の所有者や有効期限等の属性情報やその真正性は、公開鍵証明書と呼ばれるデータによって確認される仕組みが利用されるケースが多い。公開鍵証明書には、証明の対象となる公開鍵やその所有者・有効期間等の情報のほか、それらの情報の真正性を確保するためのディジタル署名（後述）が含まれる。ディジタル署名は、認証機関と呼ばれる信頼できる第三者機関によって生成・発行される。このような公開鍵暗号を実用化するために必要となる仕組みは、公開鍵インフラと呼ばれている。

ステップと、（ii）不特定多数の署名検証者が公開鍵を利用して署名者固有の変換の正当性を確認する署名検証のステップによって構成される。秘密鍵を所有する者は唯一人であるため、秘密鍵による変換は署名者固有の変換となり、署名検証者は署名者の公開鍵で署名の正当性を検証できる。さらに、署名者はディジタル署名付きデータを作成した事実を後で否定できない（否認防止）。なお、共通鍵暗号では、鍵が通信当事者間で共有されているため、暗号文がその鍵を所有する二人のどちらによって作成されたかを特定することができない。

④処理速度が遅い

一般的に、公開鍵暗号による暗号化・復号処理には、べき乗剰余演算等計算時間が比較的多く必要となる演算が利用されるケースが多く、加算や乗算等高速処理が可能な演算が主に利用される共通鍵暗号に比べて、処理速度が遅くなる傾向にある。

これらの特徴により、膨大なデータの処理が要求されるデータの暗号化には共通鍵暗号が利用され、ディジタル署名や共通鍵暗号に用いられる鍵の配送には公開鍵暗号が利用される場合が一般的である。

2. 公開鍵暗号の原理

データ M に対して公開鍵 e を用いた公開鍵暗号方式による変換を E(e, M)とし、秘密鍵 d を用いた変換を D(d, M)とする（図 1 参照）と、公開鍵暗号方式は、まず次の 3 つの条件を満たす必要がある。

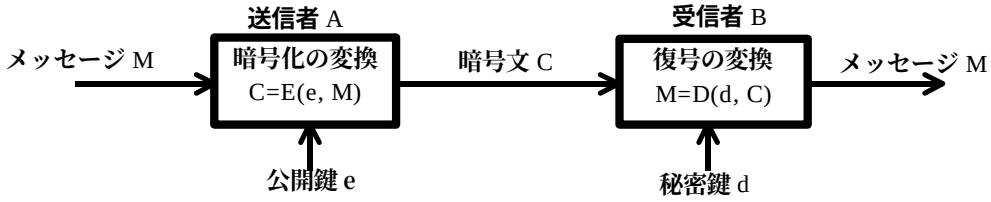

図 1 公開鍵暗号による暗号通信の一般的な手順

① 公開鍵 e を知っている場合、E(e, M)の計算は容易である。秘密鍵 d を知っている場合には、D(d, M)の計算は容易である。
② 秘密鍵 d を知らない場合、公開鍵 e、変換アルゴリズム E、暗号文 C = E(e, M)を知っていたとしても、秘密鍵 d を求めることは計算量的に困難である。
③ 秘密鍵 d を知らない場合、公開鍵 e、変換アルゴリズム E、暗号文 C = E(e, M)を知っていたとしても、元の平文 M を求めることは計算量的に困難である。

これらの条件を満足する関数を構築するために、「一方向性関数」と呼ばれる関数が利

用される。一方向性関数 F においては、関数 F の計算は容易であるが、逆関数 F^{-1} の計算が計算量的に困難となる。この一方向性関数に、逆関数の計算を容易にするためのパラメーター（「落し戸」と呼ばれる）を組み込み、「落し戸付き一方向性関数」と呼ばれる特殊な関数を構築する。上記の例の場合、E(e, M)に対応する逆関数 D(d, M) (=E^{-1}(e, M))におけるデータ d が落し戸となる。このような関数において、公開鍵を e、秘密鍵を d とすれば、以上の 3 つの条件が満足され、公開鍵暗号方式を実現させることができる。

3. 公開鍵暗号の機能

公開鍵暗号の主要な機能として、（1）守秘、（2）ディジタル署名、（3）ブラインド署名、（4）鍵配送が挙げられる。

(1) 守秘

公開鍵暗号を利用することによって、次のような手順で通信データの守秘を実現することができる（図 1 参照）。

（ステップ 1）データの送信者は、受信者の公開鍵によって平文を暗号化し、受信者に送信する。
（ステップ 2）受信者は、暗号文を自分の秘密鍵で復号する。

各受信者の公開鍵は不特定多数に公開されているため、誰もが同じ公開鍵によって暗号通信が可能となる。一方、秘密鍵を所有していない者が暗号文を解読することは計算量的に困難となる。

(2) ディジタル署名

ディジタル署名の方法としては、認証子照合法が利用されるケースが多い[3]。認証子照合法では、以下の手順によってディジタル署名の生成・検証が行われる（図 2 参照）。

（ステップ 1）署名作成者は、データ M を「ハッシュ関数」と呼ばれる特殊な関数 h により一定長のデータ h(M)に圧縮し、これをディジタル署名方式により自分の秘密鍵 d で変換してディジタル署名 S=D(d, h(M))を生成する。

---

[3] ディジタル署名の方法として、データ自身を秘密鍵で変換してディジタル署名を作成する方法（通信文復元法と呼ばれる）も存在する。しかし、この方法では、より多くのデータを処理する必要から計算効率が低下するほか、データ x に対して公開鍵 e で暗号化した場合、データ x が暗号化されたデータ E(e, x)に対するディジタル署名になるという問題がある。これは、E(e, x)を秘密鍵 d で変換すると、D(d, E(e, x)) = E^{-1}(e, E(e, x)) = x が成立するためである。さらに、RSA 暗号のように、公開鍵暗号が乗法的である場合（E(e, xy) = E(e, x)E(e, y)が成立）には、x、y に対する署名 D(d, x)、D(d, y)を入手した者は xy に対する署名 D(d, xy)を偽造できるという問題がある。このような問題点が存在するため、認証子照合法が一般的である。

（ステップ 2）署名作成者は署名付きデータ(M, S)を署名検証者に送信する。
（ステップ 3）署名検証者は、署名 S をディジタル署名方式によって署名作成者の公開鍵 e で変換した結果 E(e, S)と、文書 M をハッシュ関数 h で圧縮した結果 h(M)を突合して一致するかどうかを検証する。

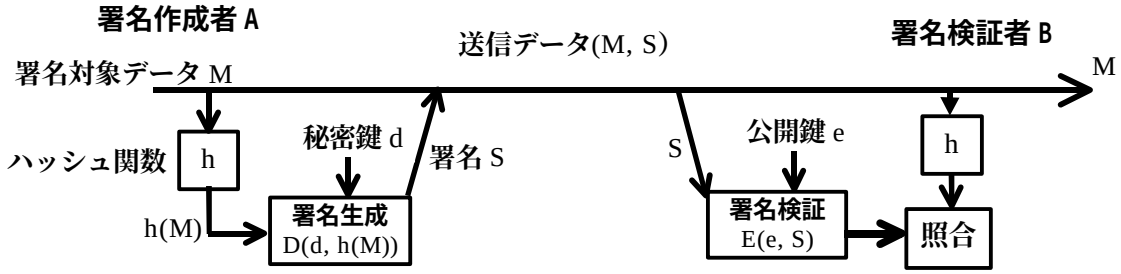

図 2　公開鍵暗号によるディジタル署名の一般的な手順

ディジタル署名を実現する場合、署名作成者と署名検証者以外の信頼できる第三者を介する「調停署名方式」が利用されるケースが多い。調停署名方式の代表例は、認証機関による公開鍵証明書を利用した方式であり、その手順は以下の通り（図 3 参照）。

（ステップ 1）署名作成者 A は自分の ID（ID_A）と公開鍵 K_{PA} を認証機関に送信する。
（ステップ 2）認証機関は、秘密鍵 K_S を利用して(ID_A ,K_{PA})に対する署名 S_{AC} を生成して公開鍵証明書とし、A に返信する。
（ステップ 3）A は、署名付きデータ（M,S_A）に公開鍵証明書（ID_A ,K_{PA},S_{AC}）を添付し、署名検証者 B に送信する。
（ステップ 4）B は、A の公開鍵証明書（ID_A ,K_{PA},S_{AC}）の真正性を認証機関の公開鍵 K_P で検証した後、A から送信された署名付きデータ（M,S_A）の真正性を A の公開鍵 K_{PA} によって検証する。

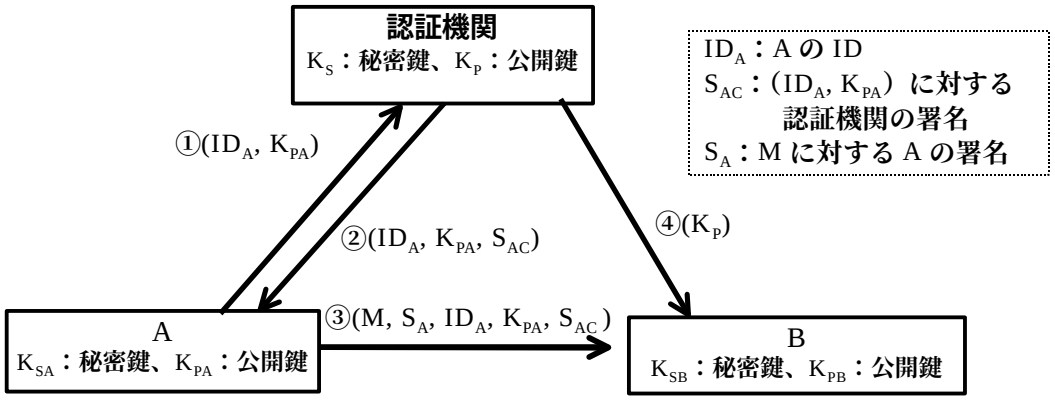

図 3　調停署名方式によるディジタル署名

(3) ブラインド署名

ブラインド署名は、署名依頼者が署名者にデータの内容を知られることなく署名を受ける方法で、Chaum[1983]によって提案された。ブラインド署名は、電子現金や電子投票等に利用される技術である。例えば、署名を受ける文書とカーボン紙を封筒に入れ、その上から署名をしてもらうという紙ベースの方法を電子的に実現する方法といえる。具体的な方法としては、Chaum[1983]による RSA 暗号を利用する方式や、ElGamal 署名を利用する方式（Camenisch et al.[1994]）等が提案されている。

RSA 暗号によるブラインド署名の手順は、以下の通り。署名者 B の RSA 暗号の公開鍵、秘密鍵はそれぞれ(e, n)、d とする。

（ステップ 1） 署名依頼者 A は、秘密の乱数 r を生成した後、署名対象データ M、署名者 B の公開鍵(e, n)を用いて以下の計算を行い、B に X を送信する。

$$X = Mr^e \bmod n$$

（ステップ 2） B は、自分の秘密鍵 d を用いて以下の計算を行い、Y を A に送信する。

$$Y = X^d \bmod n = M^d r \bmod n$$

（ステップ 3） A は、Y と乱数の逆数 r^{-1} から、署名 $S = M^d \bmod n$ を得る。

（ステップ 4） A は、署名 S の正当性を以下の等式によって検証する。

$$M = S^e \bmod n$$

上記の方法では、A が B にとって都合の悪いデータに署名させることが可能となる。Chaum は、これを防ぐ方法として「cut-and-choose-methodology」と呼ばれる手順を組み込んだ方式を提案している。この方式は、署名者が署名対象データの中からいくつかのデータを抜取検査し、抜き取ったデータに問題がなければ署名を行う、というものである。

(4) 鍵配送

公開鍵暗号は、データの暗号化に利用される共通鍵暗号の暗号鍵を通信相手に配送する手段としても利用される。最も単純な鍵配送の方法は、送信者が共通鍵暗号の暗号鍵を受信者の公開鍵で暗号化し、通信相手に送信する、という方法である。ただし、この方法では、受信者は送られてきた鍵が確かに送信者のものであるか否かを確認することはできない。そこで、送信者が暗号鍵にディジタル署名を添付して配送することが必要となる。

幅広く利用されている鍵配送の方法として、Diffie-Hellman 方式（Diffie-Hellman[1976]）が挙げられる。Diffie-Hellman 法による鍵配送の手順は、下記の通り。素数 p とその原始根 α が公開鍵となる。

（ステップ 1） A は p-1 以下の自然数 X_A をランダムに選び、

$$Y_A = \alpha^{X_A} \bmod p$$

を計算して Y_A を B に送信する。

（ステップ 2）　B も、同様に p-1 以下の自然数 X_B をランダムに選び、

$$Y_B = \alpha^{X_B} \bmod p$$

を計算して Y_B を A に送信する。

（ステップ 3）　A は、共通鍵暗号の暗号鍵 K を次のようにして計算する。

$$K = Y_B^{X_A} \bmod p = \alpha^{X_A X_B} \bmod p$$

（ステップ 4）　B も同様にして鍵 K を計算する。

この方式では、攻撃者は p、α のほか、通信経路の傍受によって Y を入手できる。ところが、p、α、Y から X_A や X_B を入手するためには、「離散対数問題」（詳細は後述）を解く必要がある。p、α が十分に大きい場合、離散対数問題を解くことは計算量的に困難とされている。

一方、Diffie-Hellman 法には、離散対数問題を解く以外の攻撃法「中間侵入攻撃」が有効であることが知られている。中間侵入攻撃は、攻撃者 C が通信路上でデータ Y_A、Y_B を奪取し、自分が作成したデータ $Y_C = \alpha^{X_C} \bmod p$ に置き換えることで、自分と A、自分と B の間で鍵を共有し、A に対しては B に、B に対しては A に成り済ますという攻撃である。この攻撃法は、Diffie-Hellman 法において、通信相手を相互に確認する仕組みが存在しないために有効となっていることから、Diffie-Hellman 法に相手確認の手段を導入した方式も提案されている（Diffie,Oorschot and Wiener[1992]）。

Ⅲ．主要な公開鍵暗号方式

主要な公開鍵暗号方式は、機能面から、①データ守秘・ディジタル署名の両方の機能を有する方式、②データ守秘専用の方式、③ディジタル署名専用の方式の 3 種類に大別されるほか、依拠している数学の問題によって、①素因数分解問題に基づく方式、②有限体上の乗法群における離散対数問題に基づく方式、③楕円曲線によって定義された離散対数問題に基づく方式の 3 つに分類することができる。

また、既存の公開鍵暗号方式を若干改良し、より高い安全性を確保する方式も提案されている。データ守秘専用の方式としては、RSA 暗号の改良版の OAEP、Uchiyama-Okamoto 暗号の改良版の EPOC、ElGamal 暗号の改良版の Cramer-Shoup 暗号が挙げられる。これらの暗号方式は、一定の仮定の下で能動的攻撃に対する安全性が証明されている。また、ディジタル署名専用の方式についても、既存のディジタル署名方式に改良を施すことによって安全性が証明可能な方式が提案されており、比較的実用性が高い方式として、RSA 暗号の改良版の PSS 署名、ElGamal 署名の改良版の改良 ElGamal 署名、ESIGN 署名の改良版の TDH-ESIGN 署名等が挙げられる。これらの主要な公開鍵暗号を分類すると、以下の表 1 の通り。

表 1　公開鍵暗号方式の分類

	データ守秘・ディジタル署名	データ守秘専用	ディジタル署名専用
素因数分解問題に基づく方式	RSA 暗号 Rabin 暗号	Okamoto-Uchiyama 暗号	ESIGN 署名
改良方式		EPOC OAEP	PSS 署名 TDH-ESIGN 署名
有限体上の乗法群における離散対数問題に基づく方式		ElGamal 暗号	ElGamal 署名 DSA 署名
改良方式		Cramer-Shoup 暗号	改良 ElGamal 署名
楕円曲線によって定義された離散対数問題に基づく方式		EC-ElGamal 暗号	EC-ElGamal 署名 EC-DSA 署名
改良方式		EC-Cramer-Shoup 暗号	改良 EC-ElGamal 署名

1. データ守秘・ディジタル署名の両方の機能を有する方式

（1）RSA 暗号

RSA 暗号は、Rivest, Shamir, and Adleman[1978]によって考案された最初の本格的な公開鍵暗号である。RSA 暗号は、大きな合成数の素因数を求める問題（素因数分解問題）を解くことが計算量的に困難であることに依拠している。RSA 暗号の安全性に関する数学的に厳密な証明（例えば、RSA 暗号の安全性と素因数分解問題の困難性との等価性）はこれまで示されていないものの、RSA 暗号が発表されてから 20 年近くもの間、多くの暗号研究者によって様々な角度から研究が進められてきた結果、素因数分解よりも有効な解読

法は報告されていない[4]。このため、RSA 暗号は、最も信頼性の高い公開鍵暗号方式として幅広い分野において実用化されており、事実上の標準と目されている[5]。RSA 暗号による暗号化・復号および署名生成・検証の手順は、以下の通り。

＜RSA 暗号＞

【鍵生成】最初に 2 つの大きな素数 p と q を選び、これらの積 n=pq を計算する。次に、p-1 と q-1 の最小公倍数 L = LCM(p-1,q-1)を計算する。最後に、最大公約数 GCD(e, L) = 1 を満足する自然数 e を選び、ed = 1 (mod L)を満たす d を求める。

【秘密鍵】(d, p, q)

【公開鍵】(e, n)

（1）データ守秘機能

【暗号化】$C = M^e \bmod n$（ただし、M は平文、C は暗号文）

【復号】$M = C^d \bmod n$

（2）ディジタル署名機能（ハッシュ関数を利用するケース）

【署名生成】平文 M のハッシュ値 m = H(M)を生成した後（H はハッシュ関数）、以下の計算によって M に対するディジタル署名 S を生成する。

$S = m^d \bmod n$

【署名検証】(M, S)を受け取った受信者は、以下の等式が成立するか否かを検証する。

$H(M) = S^e \bmod n$

RSA 暗号は、現時点では安全性に関する大きな欠陥が示されていない暗号であるが、素数 p と q の選び方によっては、素因数分解問題の高速解法が適用可能となるほか、安全性と関連の深い暗号化関数の周期[6]に影響を与える。したがって、鍵生成の際には、素数 p と q を注意深く選ぶ必要がある。素数 p、q の満たすべき主な条件として、次の 3 つが挙げられる。

---

[4] ただし、最近では、RSA 暗号の安全性と素因数分解問題の困難性が等価にはならない可能性があるとの研究成果も発表されている（Boneh and Venkatesan[1998]）。

[5] RSA 暗号は、金融業務における公開鍵暗号を利用した鍵管理の方法に関する標準規格 ISO 11166（Banking ― Key management by means of asymmetric algorithms）の Part 2（Approved algorithms using the RSA cryptosystem）において規定されているほか、公開鍵証明書を利用したディジタル署名の利用方法に関する標準規格案 ISO/IEC FDIS 14888-3（Information technology – Security techniques – Digital signature with appendix – Part 3: Certificate-based mechanism）において、利用可能な公開鍵暗号方式の 1 つとして記載されている。なお、ISO/IEC FDIS 14888-3 に記載されている公開鍵方式は、RSA 暗号のほか、ESIGN 署名、DSA 署名、EC-DSA 署名、Pointcheval-Vaudenay 方式である。

[6] 暗号化関数を $f(x) = x^e \bmod n$ と表わす。このとき、$f^m(x) = x$ を満たす最少の正整数 m が「x に対する周期」と呼ばれる。周期の値は e,n,x の値に依存している。周期が短いと一般的解読や全面的解読が比較的容易になることが示されている（Rivest[1978]、Williams and Schmid[1979]、勝野[1983]）。

① |p-q|を大きくする。
② (p±1)と(q±1)は、それぞれ大きな素因数 p_+, p_-, q_+, q_-を持つ。
③ (p_+±1)と(p_-±1)、および(q_+±1)と(q_-±1)は、それぞれ大きな素因数を持つ。

上記 3 条件のうち、条件①と②はそれぞれ Fermat 法、p-1 法と p+1 法 という素因数分解の高速解法（後述）を適用不可能にするための対策である。条件②と③は解読に繋がる短周期の防止策となっている。なお、これらの素数の条件は、RSA 暗号に限らず、素因数分解問題の困難性に依拠している暗号方式すべてに当てはまる。

また、RSA 暗号の利用方法によっては安全性が損なわれる可能性がある。例えば、（i）データ全体に対して署名を生成し、そのデータを復元する形で検証が行われる場合（通信文復元型ディジタル署名）、（ii）利用者全員が共通の法 n を用いる場合、（iii）平文の暗号化処理やディジタル署名検証の高速化を図るために、公開鍵 e としてサイズの小さな数値を利用する場合には、安全性に問題が生じることが示されている（Simmons[1983]、Coppersmith et al.[1996]）。

（2）Rabin 暗号

Rabin 暗号は、Rabin[1979]によって提案された方式であり、RSA 暗号に改良を加えることによって、「受動的攻撃に対する完全解読（後述）の困難性が素因数分解問題の困難性と等価である」ことを証明可能にした方式である。しかし、Rabin 暗号には、①暗号文を一意に復号することができず、復号によって 4 通りのデータが生成される、②一部のデータに対しては、ディジタル署名を生成することができない、③能動的攻撃（後述）によって完全解読が可能になる、といった問題点がある。Rabin 暗号の暗号化・復号および署名生成・検証手順は以下の通り。

＜Rabin 暗号＞
【鍵生成】2 つの大きな素数 p と q を選び、これらの積 n=pq を計算する。0≦b＜n を満足する自然数 b を定める。
【秘密鍵】(p, q)
【公開鍵】(n, b)
（1）データ守秘機能
【暗号化】C = M(M+b) mod n（ただし、M は平文、C は暗号文）
【復号】次の連立合同式を解いて平文 M を算出する。
$M^2+Mb-C = 0$（mod p）
$M^2+Mb-C = 0$（mod q）
（2）ディジタル署名機能
【署名生成】平文 M のハッシュ値 m = h(M)を利用して以下の連立合同式を満足するディジタル署名 S を生成する（ただし、こうしたディジタル署名 S が生

成される確率は約 1/4 である）。連立同号式を満足する S は 4 つ存在するが、そのうちいずれかの値をディジタル署名として M に添付して送信する。

$S^2 + Sb - m = 0 \pmod p$

$S^2 + Sb - m = 0 \pmod q$

【署名検証】受け取った M と S を用いて、以下の等式が成立するか否かを検証する。

$h(M) = S(S+b) \bmod n$

暗号文を一意に復号できない、任意の文書に署名できない、という Rabin 暗号における問題点は、暗号化・復号関数が全単射でないことに起因している。これに対し、Williams[1980]は、素数 p と q に制限を加えることによって暗号化・復号関数が全単射となる公開鍵暗号方式 Williams 暗号を考案している。また、黒澤・伊東・竹内[1987] は、Williams 暗号を改良した方式として「逆数暗号」を提案している。しかし、Williams 暗号、逆数暗号ともに、Rabin 暗号における「能動的攻撃によって解読される」という問題が残されている。

2. データ守秘専用の方式

(1) ElGamal 暗号

ElGamal 暗号は、ElGamal[1985a]によって発表された有限体上の乗法群における離散対数問題の困難性に基づく公開鍵暗号である。有限体上の乗法群における離散対数問題は、次のような数学の問題である。

【有限体上の乗法群における離散対数問題】

素数 p、p を法とする乗法群の原始根 a を選び、ある整数 x に対して $b = a^x \bmod p$ を計算する。このとき、p, a, b を所与として、$b = a^x \bmod p$ を満足する x を求めよ。

ElGamal 暗号の解読困難性と有限体上の乗法群における離散対数問題の困難性との関係については、これまで厳密な理論的結果が示されていない。ElGamal 暗号では、暗号化の際に秘密の乱数が利用されており、同じ平文を暗号化しても異なる暗号文が生成される。このような性質を有する暗号は確率暗号と呼ばれている[7]。ElGamal 暗号の暗号化・復号の手順は以下の通り。

＜ElGamal 暗号＞

【鍵生成】大きな素数 p と乱数 x を生成し、法 p の乗法群の原始根 α を選ぶ。これらを用いて $y = \alpha^x \bmod p$ を計算する。

[7] これに対し、同一の鍵の下で同じ平文を暗号化すると常に同じ暗号文が生成される暗号は、確定暗号と呼ばれている。

【秘密鍵】x
【公開鍵】(y, p, α)
【暗号化】送信者は乱数 k を生成し、受信者の公開鍵 y を用いて暗号文（C_1,C_2）を以下のように作成する。ただし、平文を M とする。

$C_1 = \alpha^k \bmod p$

$C_2 = My^k \bmod p$

【復号】受信者は以下の等式から M を得る。

$MC_1^x = C_2 \bmod p$

ElGamal 暗号を利用する場合には、安全性の観点から、通信の都度乱数 k を生成する必要がある。これは、同一の乱数 k によって異なる 2 つの平文（M',M''）を暗号化した場合、対応する暗号文（C_2', C_2''）との間に、$M' / M'' = C_2' / C_2''$という関係が成立し、M'と C_2'が分かると C_2''から M''を知ることができるためである。

（2）Okamoto-Uchiyama 暗号

Okamoto-Uchiyama 暗号は、Okamoto and Uchiyama[1998]によって提案された公開鍵暗号方式であり、素因数分解問題の困難性に依拠した方式である。Okamoto-Uchiyama 暗号は、①受動的攻撃に対する一方向性（後述）が素因数分解問題の困難性と等価であることが証明されている、②暗号化・復号に必要な計算量が RSA 暗号と同程度である、③確率暗号である、といった特徴を有している。ただし、能動的攻撃に対しては安全ではないが、Okamoto-Uchiyama 暗号に改良を加えた方式 EPOC において、能動的攻撃に対する安全性が証明されている。

Okamoto-Uchiyama 暗号の暗号化・復号手順は以下の通り。

＜Okamoto-Uchiyama 暗号＞

【鍵生成】大きな素数 p と q を生成し、$n = p^2q$ となる自然数 n と、n を法とする乗法群の原始根 g を選ぶ。

【秘密鍵】（p, q）

【公開鍵】（n, g）

【暗号化】送信者は乱数 r（$0 < r < n$）を生成し、次のように平文 m に対応する暗号文 C を生成する。ただし、平文 m のデータは、$0 < m < p$ を満足する。

$C = g^{m+nr} \bmod n$

【復号】受信者は、$C_p = C^{p-1} \bmod p^2$ と $g_p = g^{p-1} \bmod p^2$ を計算した上で、関数 $L(x) = (x-1)/p$ を利用して以下の計算によって平文 m を復号する。

$$m = \frac{L(C_p)}{L(g_p)} \bmod p$$

なお、Okamoto-Uchiyama 暗号は、P 部分群問題の困難性を仮定することによって、受動的攻撃に対して強秘匿（後述）であることが証明されている。P 部分群問題の概要は、以下の通り。

【P 部分群問題】
平文 M、乱数 r の下での Okamoto-Uchiyama 暗号の暗号化関数を E(M,r)とする。2 つの乱数 t と s を生成し、以下の計算を行う。
$E(0, t) = h^t \bmod n$、$E(1, s) = gh^s \bmod n$
このとき、$(h^t \bmod n)$と$(gh^s \bmod n)$の値のみを所与として、それぞれが E(0, t)と E(1, s)のどちらの値かを判定せよ。

（3）EC-ElGamal 暗号

楕円曲線によって定義された離散対数問題に基く公開鍵暗号方式は、有限体によって定義された楕円曲線上の点によって構成される有限可換群上での離散対数問題の困難性を安全性の根拠とするもので、Koblitz[1987]と Miller[1986]によって発表された。楕円曲線および楕円曲線によって定義された離散対数問題の概要は、以下の通り。

【楕円曲線】
楕円曲線は、3 次曲線（関数 F(x,y)の次数が 3 となる代数方程式 F(x,y) = 0 の解の集合）のうち、特異点（F(x,y)の x および y に関する偏微分係数が 0 となる(x,y)）を含まない点の集合である。

【有限体上の楕円曲線】
有限体上の楕円曲線は、要素の個数 p（p > 3 の素数）である有限素体 F_p において、
$$\left\{(x, y) \mid y^2 = x^3 + ax + b (\bmod p)\right\} (4a^3 + 27b^2 \neq 0, a, b \in F_p)$$
を満足する点(x,y)（x,y は F_p の要素）の集合、と定義される。

【楕円曲線によって定義された離散対数問題】
素数 p に対して有限体 F_p 上の楕円曲線に無限遠点 O を加えた集合を $E(F_p)$とし、$E(F_p)$上の 2 点間の加法演算を定義すると、$E(F_p)$は有限可換群となる（無限遠点 O はこの有限可換群の零元となる）。有限可換群 $E(F_p)$における要素の個数は、p + 1 − t（$-2\sqrt{p} \le t \le 2\sqrt{p}$）と表され、t の値はトレースと呼ばれる。

この $E(F_p)$上の点 A、B（A ≠ B）を選び、A と B がある自然数 x に対して A=xB という関係にある（定義された加法演算によって点 B を x 回加えると点 A になる）とする。このとき、p、$E(F_p)$、A、B を所与として、A = xB を満足する自然数 x を求めよ。

有限体上の乗法群における離散対数問題には、準指数関数時間の解法（指数計算法、詳細については後述）が存在する一方、楕円曲線によって定義された離散対数問題には、ある種の楕円曲線を除いて指数計算法の適用が困難であることから、安全性を維持しつつ、従来の方式に比べて鍵長を短縮できるとされている。楕円曲線によって定義される離散対数問題に基づく暗号方式は、有限体上の乗法群における要素を楕円曲線上の有限可換群の

数問題に基づく暗号方式は、有限体上の乗法群における要素を楕円曲線上の有限可換群の要素に、有限体上の乗法群において定義される乗法を楕円曲線上の有限可換群において定義される加法にそれぞれ対応させることによって構築される。有限体上の乗法群における要素の r 乗は楕円曲線上の有限可換群における要素の r 倍に対応し、通常乗法群における要素の r 乗を計算するのに用いられる高速演算も楕円曲線上の有限可換群における要素の r 倍を計算するのに用いられる。

楕円曲線によって定義される離散対数問題に基づく代表的な暗号方式として、ElGamal 暗号を利用した暗号方式（以下、EC-ElGamal 暗号とする）が挙げられる。EC-ElGamal 暗号の暗号化・復号の手順は、以下の通り。

＜EC-ElGamal 暗号＞

【鍵生成】まず、素数 p に対して有限素体 F_p 上に定義された楕円曲線を $E(F_p)$とし、$E(F_p)$ 上の点 G（ベースポイントと呼ばれる）を選ぶ。ただし、G は、位数 K が大きな素数となるように選ばれる[8]。整数 t を選び、Y=tG を計算する。

【秘密鍵】 t

【公開鍵】（$E(F_p)$, G, Y）

【暗号化】送信者は、法 K の乗法群から乱数 r を選び、$E(F_p)$上で以下の計算を行い、暗号文 M に対応する暗号文(C_1, C_3)を生成する。ただし、$x(C_2)$は、C_2 の x 座標の値を指す。

$C_1 = rG$

$C_2 = rY$

$C_3 = M \cdot x(C_2) \bmod p$

【復号】受信者は、秘密鍵 t を利用して $E(F_p)$上で以下の計算を行い、平文 M を得る。

$M = C_3 / x(tC_1) \bmod p$

EC-ElGamal 暗号のプロトコルの安全性については、ElGamal 暗号に関する議論がそのまま当てはまる。このため、同じ乱数 r を 2 回以上用いると、1 対の暗号文・平文ペアから同じ乱数の暗号文は全て解読される。

3. ディジタル署名専用の方式

（1）ElGamal 署名

ElGamal 署名は、ElGamal[1985a]によって提案された署名方式であり、有限体上の乗法群における離散対数問題の困難性に基づく初めての方式である。ElGamal 署名には、（i）署名生成の都度秘密の乱数を生成する必要がある、（ii）署名が法のサイズの 2 倍となる、という問題が存在する。（ii）の問題については、ElGamal 署名の改良版である

[8] 有限可換群 $E(F_p)$上で G を K 回加える（換言すると、$E(F_p)$上で定義された加法"+"を K 回行い、G + G +…＋G = KG を計算する）という演算を行う。このとき、KG = O（O は無限遠点）となる最小の自然数 K は、G の位数と呼ばれる。

Schnorr 署名や DSA 署名において改善されている。ElGamal 署名の署名生成・検証手順は以下の通り。

＜ElGamal 署名＞
【鍵生成】大きな素数 p と、p を法とする乗法群の要素 x を選び、$y = \alpha^x \bmod p$（α は p を法とする乗法群の原始根）を計算する。
【秘密鍵】 x
【公開鍵】（y, p, α）
【署名生成】送信者は乱数 k を生成し、以下の計算によって署名（r, t）を生成する。ただし、平文を M、ハッシュ関数を h とする。
$r = \alpha^k \bmod p$
$t = (h(M)-xr)/k \bmod (p-1)$
【署名検証】受信者は、公開鍵 y を利用して、以下の等式が成立するか否かを検証する。
$\alpha^{h(M)} = y^r r^t \bmod p$

ElGamal 署名を利用する場合には、ElGamal 暗号と同様の理由により、毎回異なる乱数 k を利用する必要がある。また、公開鍵である p と α の値がある一定の条件を満足する場合には、秘密鍵を知らなくても容易にディジタル署名の偽造が可能となることが示されており（Bleichenbacher[1996]）、他者が生成した公開鍵 p と α を利用するべきではないといわれている。なお、後述する DSA 署名には、この攻撃法は適用できないことが示されている。

（2）DSA 署名

DSA 署名は ElGamal 署名の改良方式であり、NIST[9][1991]によって提案され、1994 年に米国連邦政府のディジタル署名標準（FIPS 186）となっている[10]。DSA 署名は、Schnorr[1990]によって提案された「法 p に対して p-1 の約数 q を法とする有限体上で署名を構成する」という技法を参考にして、ElGamal 署名の署名長を 2 p から 2 q に短縮することを可能にした。q を法とする有限体上での離散対数問題に対しては、指数計算法が直接適用できないため、q を p よりかなり小さい値に設定しても安全性は損なわれないとされている。DSA 署名では、署名生成は高速に実行することができるが、署名検証は RSA 暗号よりも時間を要する。

[9] NIST（National Institute of Standards and Technology）：米国商務省の下部組織で、科学技術全般に関する標準を策定する役割を担っているほか、情報通信の分野では、1987 年に成立した Computer Security Act により、米国政府内部における情報通信規格である FIPS（Federal Information Processing Standard）を制定する権限を有している。
[10] DSA 署名は、ISO/IEC FDIS 14888-3 において利用可能なディジタル署名方式として記載されているほか、米国の金融分野に利用されるディジタル署名方式の標準規格 ANSI X9.30 として規定されている。

＜DSA 署名＞
【鍵生成】大きな素数 p と、p-1 の素因数 q を選ぶ。続いて、p を法とする乗法群の要素 x を選び、$y = g^x \bmod p$（g は q を法とする乗法群の原始根）を計算する。
【秘密鍵】 x
【公開鍵】（y, g, p, q）
【署名生成】送信者は乱数 k を生成し、以下の計算によって署名（r, t）を生成する。ただし、M は平文、h はハッシュ関数である。

$r = (g^k \bmod p) \bmod q$

$t = (h(M)+xr)/k \bmod q$

【署名検証】受信者は、公開鍵 y を利用して、以下の等式が成立するか否かを検証する。

$r = (g^{h(M)/t} y^{r/t} \bmod p) \bmod q$

DSA 署名を利用する場合には、ElGamal 署名と同様に、毎回異なる乱数 k を利用する必要がある。

(3) ESIGN 署名

ESIGN 署名は、Okamoto[1990]によって提案された高速処理を特徴とするディジタル署名方式である。 ESIGN 署名は、素因数分解問題の困難性と合同多項不等式求解問題の困難性に基づいている。

＜ESIGN 署名＞
【鍵生成】まず、大きな素数 p と q を、p＞q となるように選び、$n = p^2q$ を計算する。次に、k＞3 となる自然数 k を選ぶ。
【秘密鍵】（p, q）
【公開鍵】（k, n）（n は b bit とする）
【署名生成】送信者は、乱数 x（ただし、0＜x＜pq）を生成し、以下の計算によって署名 s を生成する。ただし、平文を M、ハッシュ関数を h とする。

$Q = (h(M)-(x^k \bmod n))/pq$（Q 以上の最小の整数を w とする）、

$y = w/(kx^{k-1}) \bmod p$、

$s = x + ypq$

【署名検証】受信者は、送信者の公開鍵 k を利用して、以下の不等式が成立するか否かを確認する。ただし、N は、(2b)/3 以上の最小の整数とする。

$h(M) \leqq s^k \bmod n < h(M)+2^N$

ESIGN 署名の検証式に利用されるべき乗 k について、k = 2,3 に対しては、Brickell and deLaurentis[1986]によって署名の偽造方法が発表されているものの、k≧4 の場合に対しては、現時点では効率的な署名の偽造方法は示されていない。現在では、安全性の観点から、ESIGN 署名のパラメーターとして、k と n は 1,024 bit 程度、p と q は 340 bit 程度

が推奨されている。

（4）EC-ElGamal 署名

EC-ElGamal 署名は、ElGamal 署名のアルゴリズムを、楕円曲線によって定義された有限可換群上で実現したディジタル署名方式である[11]。EC-ElGamal 暗号と同様に、有限体上の乗法群における離散対数問題に対する高速解法が適用できないことから、安全性を維持しつつ、署名を短縮することが可能となる。この結果、ElGamal 署名と比較して、高速処理が可能になるという利点を有している。EC-ElGamal 署名の署名生成・検証手順は、以下の通り。

＜EC-ElGamal 署名＞

【鍵生成】まず、素数 p に対して有限素体 F_p 上に定義された楕円曲線を $E(F_p)$とし、$E(F_p)$上の点 G（ベースポイントと呼ばれる）を選ぶ。ただし、G は、その位数 l が大きな素数となるように選ばれる。整数 t を選び、Y=tG を計算する。

【秘密鍵】 t

【公開鍵】（$E(F_p)$, G, Y）

【署名生成】送信者は、法 l の乗法群から乱数 k を選び、$E(F_p)$上で以下の計算を行い、平文 M に対応するディジタル署名(R, S)を生成する。ただし、h はハッシュ関数であり、x(R)は R の x 座標の値を指す。

$R = kG$

$S = (h(M) - t \cdot x(R))/k \bmod l$

【署名検証】受信者は、公開鍵 Y を利用して $E(F_p)$上で以下の計算を行い、平文 M に対するディジタル署名の正当性を検証する。

$h(M)G = SR + Y \cdot x(R)$

（5）EC-DSA 署名

EC-DSA 署名は、DSA 署名のアルゴリズムを、楕円曲線によって定義された有限可換群上にで実現したディジタル署名方式である[12]。EC-DSA 署名の署名生成・検証方法は以下の通り。

---

[11] EC-ElGamal 署名は、楕円曲線によって定義された離散対数問題に基づくディジタル署名方式の標準規格案 ISO/IEC WD 15946-2（Information technology ― Security techniques ― Cryptographic techniques based on elliptic curves ― Part 2: Digital signatures）に記載されている方式の 1 つである。本標準規格案には、EC-ElGamal 署名のほか、EC-DSA 署名（後述）や ECKCDSA 署名（EC-DSA 署名のアルゴリズムのうち、ディジタル署名 S の生成方法を一部を変更した署名方式）が記載されている。

[12] EC-DSA 署名は、ISO/IEC FDIS 14888-2 および ISO/IEC WD 15946-2 においてディジタル署名方式の 1 つとして記載されているほか、現在米国の金融分野におけるディジタル署名の標準案 ANSI X9.62 として、標準化が進められている。

＜EC-DSA 署名＞

【鍵生成】まず、素数 p に対して有限素体 F_p 上に定義された楕円曲線を $E(F_p)$とし、$E(F_p)$ 上の点 G（ベースポイントと呼ばれる）を選ぶ。ただし、G の位数 l は大きな素数となる。整数 t を選び、Y=tG を計算する。

【秘密鍵】 t

【公開鍵】（$E(F_p)$, G, Y）

【署名生成】送信者は、法 l の乗法群から乱数 k を選び、$E(F_p)$上で以下の計算を行い、平文 M に対応するディジタル署名(r, S)を生成する。ただし、h はハッシュ関数、x(R)は R の x 座標の値を指す。

$R = kG$

$r = x(R) \bmod l$

$S = (h(M) + rt)/k \bmod l$

【署名検証】受信者は、公開鍵 Y を利用して $E(F_p)$上で以下の計算を行い、平文 M に対するディジタル署名の正当性を検証する。

$r = x((h(M)/S)G + (r/S)Y) \bmod l$

4. 既存の方式に対する改良方式 ― データ守秘専用方式 ―

(1) OAEP

OAEP（Optimal Asymmetric Encryption Padding）は、Bellare and Rogaway[1995]によって提案されたデータ守秘専用の公開鍵暗号方式である。OAEP は、①「入力データに対してランダムなデータを出力する」理想的なランダム関数が利用可能であり（この仮定はランダム・オラクルモデルと呼ばれている）、さらに②RSA 暗号関数が一方向性を有しているならば、能動的攻撃に対して強秘匿であることが証明されている。RSA 暗号関数の一方向性については、以下の通り。

【RSA 暗号関数の一方向性】

RSA 暗号の公開鍵（e, n）および n-1 以下の自然数 Y が与えられた場合、以下の等式を満足する x を求めることが計算量的に困難であるとき、「RSA 暗号関数は一方向性を有する」と呼ばれる。

$$Y = x^e \bmod n$$

OAEP の暗号化・復号の方法は、以下の通り。ハッシュ関数を利用することによって暗号文の冗長性を増し、不正な暗号文ではないか否かを検証することによって、能動的攻撃に対する安全性を確保している点が特徴である。このため、平文よりも暗号文のサイズの方が大きくなる。OAEP は、SET[13]の公開鍵暗号アルゴリズムに採用されているほか、RSA

[13] SET（Secure Electronic Transactions）：VISA と Mastercard によって提案されたインターネット上でのクレジットカード決済を実現する技術仕様。現在、世界各国において SET を利用した実証実験が行われており、日本においても Smart Collar Club、メディアポート名古屋等の電子商取引実証実験において利用されている。

社が開発した公開鍵暗号による暗号通信に関する規格 PKCS#1 Version 2 にも採用されている。

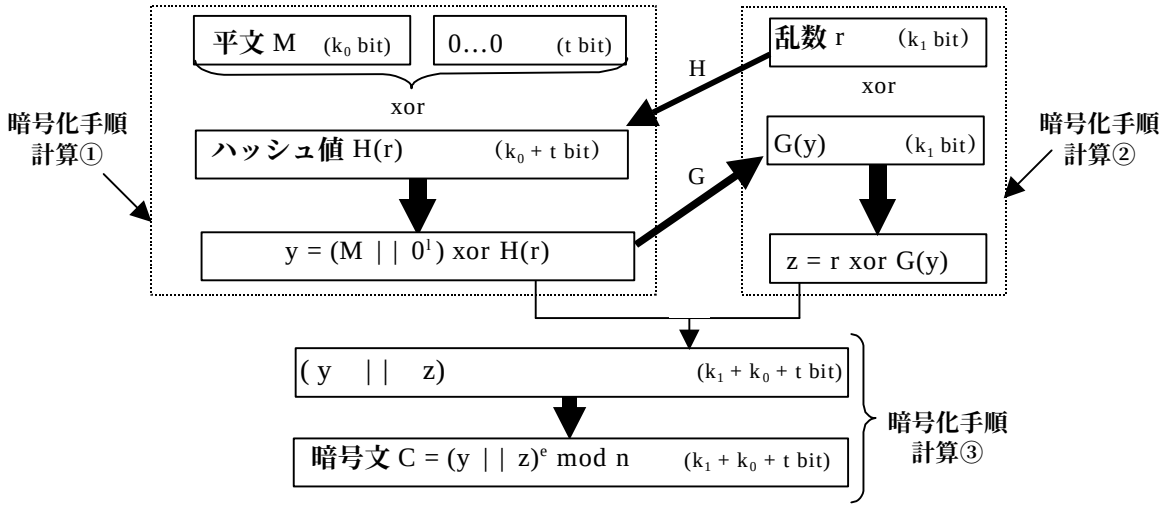

図 4　OAEP の暗号化手順の概念図

＜OAEP＞

【鍵生成】最初に 2 つの大きな素数 p と q を選び、これらの積 n=pq を計算する。次に、p-1 と q-1 の最小公倍数 L = LCM(p-1,q-1)を計算する。GCD(e, L) = 1 を満足する自然数 e を選び、ed = 1 (mod L)を満たす d を求める。

なお、H を、k_1 bit データの入力に対して、k_0+t bit の乱数を生成するランダム関数とし、G を、k_0+t bit データの入力に対して、k_l bit の乱数を生成するランダム関数とする。さらに、$[x]^a$ を x の左から a bit のデータとし、$[x]_a$ を x の右から a bit のデータとする。‖はデータの結合を表す。

【秘密鍵】 (d, p, q)

【公開鍵】（e,n）

【暗号化】平文を M とすると、M は k_0 bit のデータとする（M が k_0 bit 以下の場合にはパディングを行う）。送信者は、k_1 bit の乱数 r を生成し、以下の計算①～③によって暗号文 C を生成する（図 4 参照）。なお、「xor」は排他的論理和を表す。

① $y = (M \| 0^t)$ xor H(r)、②z = r xor G(y)、③$C = (y \| z)^e \bmod n$

【復号】受信者は、まず以下の計算を行う。

$X = C^d \bmod n$、$Y = [X]^{k0+t}$、$R = [X]_{k1}$ xor G(y)

ここで、以下の等式が成立するか否かを確認する。

$[Y \text{ xor } H(R)]_t = 0^t$

上記等式が成立する場合には、以下の計算によって平文 M を復号する。等式が成立しない場合には、不正な暗号文とみなして受け付けない。

$M = [Y \text{ xor } H(R)]^{k0}$

（2）EPOC

EPOC は、Okamoto-Uchiyama 暗号に改良を加えた暗号方式である。EPOC では、暗号文に平文のハッシュ値が含まれており、このハッシュ値を利用して不当な暗号文か否かを検証することができる。こうした改良により、EPOC は、①ランダム・オラクルモデルと、②P 部分群問題の困難性を仮定することによって、能動的攻撃に対して強秘匿であることが証明されている（Fujisaki and Okamoto[1998a]）。EPOC の暗号化・復号の手順は、以下の通り。

＜EPOC＞

【鍵生成】大きな素数 p と q（いずれもサイズは k bit）を生成し、$n = p^2q$ を計算する。法 n の乗法群の要素 g を選ぶ。ただし、$g_p (= g^{p-1} \bmod p^2)$の位数が p となるように g を選ぶ。また、$h = g^n \bmod n$ とする。H を、k bit データ（$k=k_0 + k_1$、k_0 は平文 m のサイズであり、k_1 は乱数 r のサイズ）から 3k bit のサイズの乱数を発生させるランダム関数とする。

【秘密鍵】（p, q）

【公開鍵】（n, g, h）

【暗号化】送信者は乱数 r（サイズは k_1）を生成し、平文 m に対応する暗号文 C を以下の計算によって生成する。

$$C = (g^{m \| r})(h^{H(m \| r)}) \bmod n$$

【復号】受信者は、関数 $L(x) = (x-1)/p$ を利用して以下の計算を行う。

$$C_p = C^{p-1} \bmod p^2、X = L(C_p)/L(g_p) \bmod p$$

これらの計算結果を利用して、以下の等式が成立するか否かを確認する。

$$C = g^X h^{H(X)} \bmod n$$

等式が成立した場合には、以下の計算によって平文 m を復号する。成立しない場合には、不正な暗号文とみなして受け付けない。

$$m = [X]^{k0}$$

（3）Cramer-Shoup 暗号

Cramer-Shoup 暗号は、Cramer and Shoup[1998]によって提案されたデータ守秘専用の公開鍵暗号方式である。Cramer-Shoup 暗号は、ElGamal 暗号に一部改良を加えた方式であり、①Diffie-Hellman 判定問題の困難性と、②汎用一方向性ハッシュ関数（任意の x に対して H(x)=H(y)を満足する y ($x \neq y$)を見つけることが困難なハッシュ関数 H）を仮定することによって、能動的攻撃に対して強秘匿であることが証明されている。なお、Cramer-Shoup 暗号は、有限体上の乗法群における離散対数問題に基づいているが、楕円曲線によって定義された離散対数問題に基づく方式としても利用可能である。

【Diffie-Hellman 判定問題】

素数 p を法とする乗法群 G の原始根 g と 3 つの要素 x、y、z を任意に選ぶ。これらを利用して、$g^x \bmod p$、$g^y \bmod p$、$g^z \bmod p$ を計算する（以下、mod p を省略）。このとき、（g^x、g^y、g^z）のみを所与として、$xy = z \bmod p$ が成立するか否かを判定せよ。

Cramer-Shoup 暗号の暗号化・復号の方法は、以下の通り。

＜Cramer-Shoup 暗号＞

【鍵生成】大きな素数 p、q を生成する（ただし、q は p-1 の素因数とする）。次に、p を法とする乗法群の要素 g_1 と g_2 を選ぶ（ただし、$1 = g_1^{q-1} \bmod p$、$1 = g_2^{q-1} \bmod p$ を満足する）。p を法とする加法群の要素 x1、x2、y1、y2、z を選ぶ。これらを用いて、$c = g_1^{x1} g_2^{x2} \bmod p$、$d = g_1^{y1} g_2^{y2} \bmod p$、$h = g_1^{z} \bmod p$ を計算する。H をハッシュ関数とする。

【秘密鍵】（x1, x2, y1, y2, z）

【公開鍵】（p, q, g_1, g_2, c, d, h）

【暗号化】送信者は乱数 r を生成し、受信者の公開鍵を用いて以下の計算を行う。ただし、平文を M とする。このとき、暗号文は（u_1,u_2,e,v）となる。

$u_1 = g_1^r \bmod p$、$u_2 = g_2^r \bmod p$、$e = h^r M \bmod p$、$\alpha = H(u_1,u_2,e)$、

$v = (c\, d^{\alpha})^r \bmod p$

【復号】受信者は、受け取った暗号文に対して以下の等式が成立することを確認する。

$(u_1^{x1+\alpha y1})(u_2^{x2+\alpha y2}) = v \pmod p$

上記等式が成立する場合には、以下の計算から平文 M を得る。成立しない場合には、不正な暗号文とみなして受け付けない。

$M = e/u_1^{z} \bmod p$

5. 既存の方式に対する改良方式 ― ディジタル署名専用方式 ―

（1）PSS 署名

PSS（Probabilistic Signature Scheme）署名は、Bellare and Rogaway[1996]によって提案されたディジタル署名専用の方式である。PSS 署名は、RSA 暗号を署名専用に改良した方式であり、①ランダム・オラクルモデルと、②RSA 暗号関数の一方向性を仮定することにより、能動的攻撃における存在的偽造（後述）が不可能となることが証明されている。PSS 署名では、署名生成の際に乱数が利用されており、同一の文書に同一の鍵で署名を生成しても、毎回異なる乱数を利用すればそれぞれ異なる署名が生成される。

なお、Bellare and Rogaway[1996]は、Rabin 暗号を同様の方法によって改良することで、能動的攻撃に対する安全性が素因数分解問題と等価になることが証明可能な PRab（Probabilistic Rabin Scheme）署名を同時に発表している。もっとも、PRab 署名の場合、署名生成の際に乱数を利用して合成数 n の平方剰余を計算する必要があり、計算量が比較的多くなるという短所がある。PSS 署名の署名生成・検証方法は、以下の通り（図 5 参照）。

＜PSS 署名＞

【鍵生成】最初に 2 つの大きな素数 p と q を選び、これらの積 n=pq を計算する。次に、p-1 と q-1 の最小公倍数 L = LCM(p-1,q-1)を計算する。最後に、GCD(e, L) = 1 を満足する自然数 e を選び、ed = 1 (mod L)を満たす d を求める。

【署名生成】乱数を出力するハッシュ関数 h（ハッシュ値は k_1 bit）とデータ拡張関数 g（k_1 bit → (k - k_1 - 1) bit）を用意する。なお、g 関数で利用するデータサイズ k（$k > k_1$）を予め設定する。

さらに、g 関数の拡張変換に加えて、出力データの左 k_0 bit を出力する関数 g_1 と、左(k-k_0-k_1-1) bit を出力する関数 g_2 を用意する。平文を M とし、k_0 bit の乱数 r を生成して以下の計算①～③を行い、ディジタル署名 S を生成する。

①$w = h(M \| r)$、②$R = g_1(w)$ xor r、

③$S = X^d \bmod n$ （ただし、$X = (0 \| w \| R \| g_2(w))$）

【署名検証】平文とディジタル署名(M, S)を受け取った受信者は、$X = S^e \bmod n$ を計算し、X の最初の 1 bit を b、次の k_1 bit を y、次の k_0 bit を z、残りの bit をαとする。すなわち、$X = (b \| y \| z \| \alpha)$とする。

次に、$R' = g_1(y)$ xor z を計算する。

このとき、以下の 3 つの等式が成立するか否かを検証する。

$h(M \| R') = y$、$g_2(y) = \alpha$、$b = 0$

3 つの等式がすべて成立すれば、S が真正であることが確認される。1 つでも等式が成立しない場合には、S は不正な署名であるとみなして受け付けない。

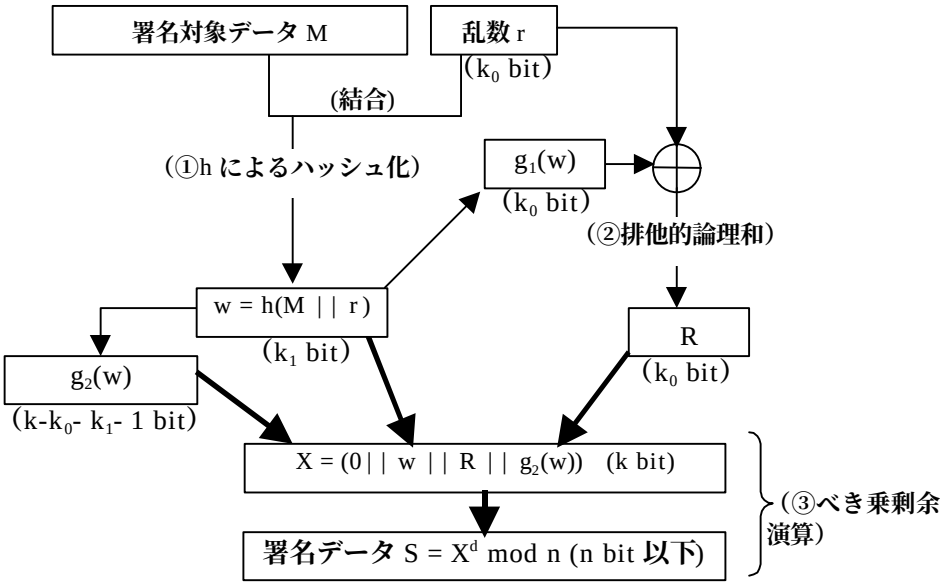

図 5　PSS 署名の署名生成手順の概念図

PSS 署名の署名生成・検証に必要な計算量は、署名生成・検証のいずれの場合も、RSA 暗号の計算量に、1 回のハッシュ関数演算と 2 回のデータ拡張関数演算が追加されるのみであり、実用性が高いとみられている。しかし、ランダム関数である h や g に通常のハッ

シュ関数を利用すると、ランダム・オラクルモデルの仮定が満足されず、安全性の証明が成立しなくなる。

(2) 改良 ElGamal 署名

改良 ElGamal 署名は、Pointcheval and Stern[1996]によって提案された方式であり、ElGamal 署名のアルゴリズムをベースにして開発された。改良 ElGamal 署名は、ElGamal 署名で利用されているハッシュ関数に一部変更を加えた方式である。安全性については、①ランダム・オラクルモデルと、②離散対数問題の困難性を仮定することによって、能動的攻撃における存在的偽造が不可能となることが証明されている。改良 ElGamal 署名は、有限体上の乗法群における離散対数問題に基づいているが、楕円曲線によって定義された離散対数問題に基づく方式としても利用可能である。

なお、Schnorr 署名に対しても同様の改良を施すことによって、能動的攻撃に対する安全性を証明可能とすることもできることが示されている。改良 ElGamal 署名の署名生成・検証手順は、以下の通り。

＜改良 ElGamal 署名＞

【鍵生成】大きな素数 p と、p を法とする乗法群の要素 x を選び、$y = \alpha^x \bmod p$（α は p を法とする乗法群の原始根）を計算する。

【署名生成】まず、ハッシュ関数 h を用意するとともに、乱数 k を生成する。その上で、以下の計算によってディジタル署名（r, t）を生成する。ただし、M は平文である。

$r = \alpha^k \bmod p$

$t = (h(M,r)-xr)/k \bmod (p-1)$

【署名検証】受信者は、公開鍵 y を利用して、以下の等式が成立するか否かを検証する。

$\alpha^{h(M,r)} = y^r r^t \bmod p$

改良 ElGamal 署名は、ElGamal 署名で利用されているハッシュ関数 h(M)を h(M,r)に変更するという比較的マイナーな改良によって証明付きの安全性を実現していることから、改良 ElGamal 署名の処理速度は ElGamal 署名とほぼ同程度とみられている。しかし、ハッシュ関数 h がランダム関数であるという仮定を外すと、安全性の証明が成立しなくなる。このため、既存のハッシュ関数を利用して実用化した場合には、安全性の証明という長所は失われることとなる。

(3) TDH-ESIGN 署名

TDH-ESIGN（Trisection-Domain-Hash ESIGN）署名は、ESIGN 署名を改良したディジタル署名方式である（Fujisaki and Okamoto[1998b]）。TDH-ESIGN 署名では、通常のハッシュ関数の代わりに Trisection-Domain-Hash 関数が利用されており、①ランダ

ム・オラクルモデル（Trisection-Domain-Hash 関数の仮定に含まれる）と、②ESIGN 署名関数の一方向性を仮定すると、能動的攻撃における存在的偽造が不可能であることが証明されている。Trisection-Domain-Hash 関数と ESIGN 署名関数の一方向性については、以下の通り。

【Trisection-Domain-Hash 関数】

ハッシュ関数 h のハッシュ値のサイズが、法 $n = p^2q$ のサイズ $|n|$ の 3 分の 1 となり、h が理想的なランダム関数であるとき、ハッシュ関数 h は Trisection-Domain-Hash 関数と呼ばれる。

【ESIGN 署名関数の一方向性】

ESIGN 署名の公開鍵（k, n）および n と同等のサイズの自然数 Y が与えられた場合、以下の不等式を満足する x を求めることが計算量的に困難であるとき、「ESIGN 署名関数は一方向性を有する」と表現される。

$$0 \leqq (x^k \bmod n) - Y < 2^N$$

ただし、N は、$2|n|/3$ 以上の最小の整数とする。

TDH-ESIGN 署名の署名生成・検証手順は、以下の通り。

＜TDH-ESIGN 署名＞

【鍵生成】まず、大きな素数 p と q を選び、$n = p^2q$ を計算する。ただし、p と q のサイズはいずれも k とする。次に、 $e>3$ となる自然数 e を選ぶ。

【秘密鍵】（p, q）

【公開鍵】（e, n）（n は 3k bit）

【署名生成】送信者は、乱数 x（ただし、$0<x<pq$）を生成し、以下の計算によって署名 s を生成する。ただし、M を平文、h を Trisection-Domain-Hash 関数とし、h のハッシュ値のサイズは k とする。また、‖ は、データの結合を表す。

$Z = 0 \| h(M) \| 0^{2k-1}$、 $\alpha = (Z - x^e) \bmod n$

$Q = \alpha/pq$（Q 以上の最小の整数を W_0 とする）、

$W_1 = pqW_0 - \alpha$（$W_1 \geqq 2^{2k-1}$ の場合には、乱数 x を選び直す）、

$t = W_0/(ex^{e-1}) \bmod p$、

$s = (x + tpq) \bmod n$

【署名検証】受信者は、送信者の公開鍵 e を利用して、以下の等式が成立するか否かを確認する。ただし、$[Y]^k$ は Y の左 k bit 分のデータを表す。

$[s^e \bmod n]^k = 0 \| [h(M)]^{k-1}$

Ⅳ．公開鍵暗号の安全性評価に関する研究

公開鍵暗号の安全性評価は、攻撃や解読のタイプを分類した上で、①公開鍵暗号方式の安全性と、その暗号方式において利用されている数学の問題の困難性との関連を評価する（相対評価）、②利用されている数学の問題における困難性を評価する（絶対評価）、という 2 つのステップによって構成される。この結果、各暗号方式に対して、「どのようなタイプの攻撃に対して、どの程度の解読がどれだけ困難なのか」を評価することが可能となる。

1. 安全性評価のための概念整理

公開鍵暗号の安全性評価を行う場合、まず、攻撃および解読のタイプを分類しておく必要がある。本節では、公開鍵暗号をデータ守秘およびディジタル署名にそれぞれ利用する場合において、攻撃のタイプと解読・偽造のタイプを整理する。

（1）公開鍵暗号方式をデータ守秘に利用する場合

①攻撃のタイプ

公開鍵暗号方式をデータ守秘に利用する場合に、攻撃のタイプは 4 つに分類される（表 2 参照）。

表 2　データ守秘に利用する暗号方式への攻撃のタイプ

攻撃方法		内容
受動的攻撃	直接攻撃	公開鍵のみを利用して行う攻撃
	選択平文攻撃（CPA）	攻撃者が予め指定したいくつかの平文に対応する暗号文を入手できる場合の攻撃
能動的攻撃	選択暗号文攻撃 (CCA1)	攻撃者が予め指定したいくつかの暗号文に対応する平文を入手できる場合の攻撃（ただし、攻撃者は、攻撃前に予め入手する暗号文と平文のペアをすべて選択しなければならない）
	適応的選択暗号文攻撃 (CCA2)	選択暗号文攻撃における暗号文の選択を、それまでに入手した暗号文・平文ペアに関する情報を参考にしながら決定できる場合の攻撃

公開鍵暗号の場合、攻撃者は公開鍵を用いて任意の平文を暗号化できるため、選択平文攻撃が容易に実現可能となる。したがって、少なくとも、選択平文攻撃に対して十分な安全性を有することが必要である。また、選択暗号文攻撃、適応的選択暗号文攻撃は一般的には成立し難い攻撃であるとみられているものの、実際に適応的選択暗号文攻撃が可能となるケースも存在することが示されている（PKCS#1 Version 1 における攻撃、詳細は後述）ことから、これらの攻撃に対しても安全であることが望ましいとされている。適応的選択暗号文攻撃が最も強力な攻撃方法である。

直接攻撃や選択平文攻撃（Chosen Plaintext Attack、以下 CPA）は受動的攻撃と呼ば

れており、選択暗号文攻撃（Chosen Ciphertext Attack、以下 CCA1）や適応的選択暗号文攻撃（Adaptive Chosen Ciphertext Attack、以下 CCA2）は能動的攻撃と呼ばれている。

②解読のタイプ － 強秘匿と頑健性

解読のタイプとして、2 つのタイプに分類できる（表 3 参照）。

表 3 データ守秘に利用する暗号方式の解読のタイプ

解読のタイプ	内容
完全解読	暗号文に対応する平文全体を求める
部分解読	暗号文に対応する平文の一部や、平文に関連する情報を求める

完全解読が困難である場合、その暗号方式は「一方向性（one-way）」を有していると呼ばれており、いかなる部分解読も困難である場合、その暗号方式は「強秘匿性（indistinguishability of encryption、以下 IND とする）」を有していると呼ばれている（Goldwasser and Micali[1984]）。

また、任意の平文 M に対する暗号文を C=E(M)とする場合、どのような関数 F に対しても、C を用いて M' = F(M)を満足するような C' = E(M')を求めることができないとき、その公開鍵暗号方式は「頑健性（non-malleability、以下 NM とする）」を有していると呼ばれている（Dolev, Dwork and Naor[1991]）。

③各安全性のレベルに関する分類と関係

3 つの攻撃のタイプ（CPA、CCA1、CCA2）と 2 つの安全性のタイプ（IND、NM）から、6 つの安全性のレベルが定義される。各安全性レベルの相互関係に関する研究成果が発表されており（Bellare et al.[1998]）、図示すると以下の通り（図 3 参照）。

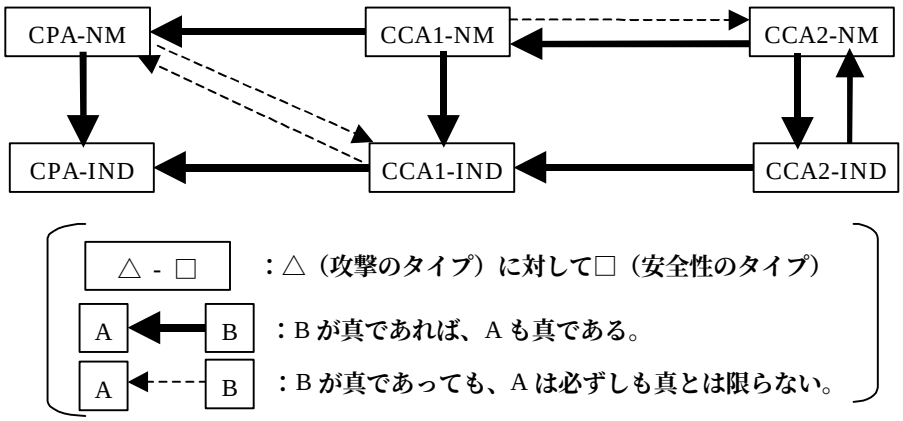

図 3　安全性レベルの相互関係

このように、「適応的選択暗号文攻撃に対して強秘匿（CCA2-IND）」となる方式は、

「適応的選択暗号文攻撃に対して頑健（CCA2-NM）」であることが証明されており、さらに選択暗号文攻撃等他の攻撃に対しても強秘匿および頑健となることから、最も安全性の高い暗号方式であるといえる。

(2) 公開鍵暗号方式をディジタル署名に利用する場合

①攻撃のタイプ

公開鍵暗号方式をディジタル署名に利用する場合、攻撃のタイプは以下の 4 つに分類できる（表 4 参照）。

表 4　ディジタル署名に利用する暗号方式への攻撃のタイプ

攻撃方法		内容
受動的攻撃	直接攻撃	公開鍵のみを利用して行う攻撃
	既知文書攻撃	いくつかのランダムなデータに対応するディジタル署名を入手できる場合の攻撃
能動的攻撃	選択文書攻撃	攻撃者が予め指定したいくつかの署名に対応する署名対象データを入手できる場合の攻撃（ただし、署名者に署名させるデータを攻撃に先立って全て選択しなければならない）
	適応的選択文書攻撃	選択文書攻撃における署名の選択を、それまでに入手した署名とそれに対応する署名対象データに関する情報を参考にしながら決定することができる場合の攻撃

ディジタル署名は、署名付きデータを収集することで既知文書攻撃が可能となるため、少なくとも既知文書攻撃に対して十分な安全性を有する必要がある。また、選択文書攻撃、適応的選択文書攻撃は、一般的には成立し難い攻撃ではあるが、これらの攻撃に対しても安全であることが望まれる。

②偽造のタイプ

ディジタル署名の偽造は、以下の 3 つのタイプに分類することができる（表 5 参照）。

このように、最も安全なディジタル署名方式は、「適応的選択文書攻撃に対して、存在的偽造が不可能な方式」といえる。

表 5 ディジタル署名に利用する暗号方式の偽造のタイプ

偽造のタイプ	内容
一般的偽造	任意のデータに対してディジタル署名を偽造できる
選択的偽造	攻撃者が予め選んだいくつかのデータに対し、ディジタル署名を偽造できる
存在的偽造	少なくともある特定のデータに対して、ディジタル署名を偽造できる

2. 安全性の証明に関する研究

公開鍵暗号の安全性評価の第 1 ステップは、評価の対象である公開鍵暗号方式の安全性と、その暗号方式に利用されている数学の問題の困難性との関連を分析・評価することで

ある。最近では、非現実的と考えられてきた適応的選択暗号文攻撃が、実際に利用可能な場合も存在することを示す研究成果も発表されており、能動的攻撃に対して安全性の証明が付いている公開鍵暗号方式が注目されている。

(1) これまでの研究の経緯

①データ守秘専用の方式

安全性の証明に関する研究は、まず受動的攻撃に対する安全性から始められた。受動的攻撃に対して安全性が証明されている公開鍵暗号方式として最初に提案されたのが、Rabin 暗号である。Rabin 暗号は、受動的攻撃に対する一方向性が素因数分解問題の困難性と等価であることが証明されている。しかし、受動的攻撃に対する強秘匿性や頑健性については証明されていないほか、能動的攻撃に対して容易に完全解読されてしまうことが示されている。

その後、Goldwasser and Micali[1984]が強秘匿の概念を初めて提案し、データ守秘専用の方式として、Goldwasser-Micali 暗号を発表した。Goldwasser-Micali 暗号は、平方剰余判定問題[14]の困難性を仮定すれば、受動的攻撃に対して強秘匿性が証明されるものの、能動的攻撃に対しては完全解読が可能となることが示されている。

能動的攻撃に対しては、Naor and Yung[1990]によって、安全性が証明可能な暗号方式の構成法が初めて提案された。Naor と Yung は、受動的攻撃に対して強秘匿である暗号方式（例えば、Goldwasser-Micali 暗号）と非対話型ゼロ知識証明[15]を組み合わせることによって、適応的選択暗号文攻撃に対しても強秘匿となる暗号方式を構成できることを示した。しかし、Naor と Yung が提案した構成方法は、計算量が非常に多くなること等から、実用的とはいえない。また、Dolev, Dwork, and Naor[1991]は頑健性の概念を初めて提案し、適応的選択暗号文攻撃に対して頑健性を有する暗号方式を提案したが、この方式も計算量が多いことから実用的な方式とはいえない。

こうした中、適応的選択暗号文攻撃に対して強秘匿性と頑健性を満足し、さらに暗号化・復号に必要な計算量の面から実用的といえる暗号方式が、Bellare and Rogaway[1993]によって最初に提案された。その後改良が加えられ、1995 年に Bellare and Rogaway によって再び提案された方式が、RSA 暗号をベースにした OAEP である。OAEP は、ランダム・オラクルモデルと RSA 暗号関数の一方向性の仮定を置くことにより、適応的選択暗号文攻撃に対して強秘匿性かつ頑健性を有することが証明されている。もっとも、実際に

---

[14] 自然数 n と互いに素な自然数 a（a < n）に対して、$x^2 = a \pmod n$を満足する x が存在するとき、a を n に関する平方剰余という。平方剰余判定問題は、「任意の自然数 a が、n に関して平方剰余となるかどうかを判定せよ」という問題である。

[15] ゼロ知識証明は、自分がある情報を知っているという事実を、その情報の内容を一切漏らすことなく他者に対して証明する方法のことである。非対話型ゼロ知識証明は、信頼できる第三者を介して、一回だけ相手に必要な情報を送るだけでゼロ知識証明を実現する方式である。

利用する場合には、ランダム関数の代わりにハッシュ関数が利用されていることから、厳密には安全性の証明に必要な条件が満されていない。したがって、実用化されているOAEPを「厳密に安全性が証明された暗号」とみなすことはできないが、「実用性と安全性の高い暗号」の実現に向けての重要な研究成果と位置付けられている。

また、1998年には、Okamoto and Uchiyama[1998]によって、Okamoto-Uchiyama暗号が提案されている。Okamoto-Uchiyama暗号は、最も実現性が高い受動的攻撃に対する一方向性が素因数分解問題の困難性と等価であることが証明されているほか、暗号化・復号に必要な計算量もRSA暗号と同程度とされており、証明可能な安全性と実用性を兼ね備えている。このOkamoto-Uchiyama暗号の改良版であるEPOCは、ランダム・オラクルモデルとP部分群問題の困難性を仮定することによって、適応的選択暗号文攻撃に対して強秘匿であることが証明されている。さらに、Cramer and Shoup[1998]は、ランダム・オラクルモデルよりも緩い仮定とみられている（i）Diffie-Hellman判定問題の困難性と（ii）汎用一方向性ハッシュ関数の仮定を置くことによって、適応的選択暗号文攻撃に対する強秘匿性が証明可能なCramer-Shoup暗号を提案している。Cramer-Shoup暗号は、ElGamal暗号をベースにした方式であり、実用性も高いといわれている。

②ディジタル署名専用の方式

「最も安全なディジタル署名方式」すなわち「能動的攻撃において存在的解読が不可能なディジタル署名方式」の概念は、1984年にGoldwasser, Micali and Rivestによって定義され、その後、最も安全なディジタル署名方式として、Goldwasser-Micali-Rivest署名が提案された（Goldwasser, Micali and Rivest[1988]）。Goldwasser-Micali-Rivest署名は、素因数分解問題が困難であるとの仮定の下で、能動的攻撃における存在的偽造が不可能であることが証明されている。しかし、アルゴリズムが複雑であり、署名生成・検証に必要な計算量が多くなるため、実用的な方式とはいえない。

その後、Naor and Yung[1989]は、汎用一方向性ハッシュ関数を利用することによって、能動的攻撃における存在的偽造が不可能な署名方式を提案した。また、Rompel[1990]は、Naor and Yung[1989]の考え方を拡張して、より緩い仮定である一方向性関数の利用可能性を仮定すれば、能動的攻撃における存在的偽造が不可能となる署名方式を構築できることを示した。しかし、Naor and YungおよびRompelが提案した署名方式は、莫大な計算量が必要となるため、実用性は低い。

こうした中、証明可能な安全性と比較的高い実用性を有する署名方式が、Bellare and Rogaway[1996]やPointcheval and Stern[1996]によって提案されている。Bellare and Rogawayは、ランダム・オラクルモデルとRSA暗号関数の一方向性を仮定すれば、能動的攻撃における存在的偽造が不可能であることが証明可能なPSS署名を提案した。PSS署名の署名生成・検証の計算量は、RSA暗号の計算量に3回のハッシュ関数計算量を加えたものに等しくなっており、それまでに提案されてきた証明可能な安全性を有している署

名方式に比べて実用性が高い。一方、Pointcheval and Stern は、ElGamal 署名のアルゴリズムを基に、ランダム・オラクルモデルと離散対数問題の困難性を仮定すれば、能動的攻撃における存在的偽造が不可能となる改良 ElGamal 署名を提案している。改良 ElGamal 署名は、ElGamal 署名に利用されているハッシュ関数を一部変更した方式であり、署名生成・検証に必要となる計算量は ElGamal 署名の計算量とほぼ同程度となっている。

このように、一定の仮定の下で最も安全なディジタル署名がいくつか提案されている。しかし、これらの方式を実用化する場合には、ランダム・オラクルモデルにおいて仮定されている「理想的なランダム関数」を利用することが不可能であるため、安全性の証明は成立しなくなるという問題が残されている。

(2) 能動的攻撃の適用可能性 －PKCS#1 への攻撃－

近年、能動的攻撃に対して証明可能な安全性と実用性を兼ね備えた方式がいくつか提案されているが、これまでは能動的攻撃が実際にどれだけ適用可能なのかについて疑問視する見方が少なくなかった。しかし、1998 年 6 月に、RSA 社によって開発された公開鍵暗号を利用した暗号通信データ形式の規格 PKCS#1 Version 1 をある特定の双方向通信環境において実装した場合、適応的選択暗号文攻撃によって任意の暗号文に対する平文を効率的に解読する方法が、Bleinchenbacher[1998]によって発表された。

PKCS#1 Version 1 の概要を簡単に説明すると、以下の通り。

PKCS#1 Version 1 において利用される RSA 暗号の公開鍵を(e,n)(n は k byte とする)、秘密鍵を d、平文を M、パディングデータを PS とすると、暗号化の対象となるデータ ED（ED は k byte）の形式は以下のように規定されている。

$$ED = 00\ 02 \ \| PS \| 00 \| M$$

ED の最初の 2 byte はそれぞれ 00 と 02 であり、PS と M は 1 byte のデータ 00 によって区切られる。M が m byte であった場合、PS は k-m-3 byte となる。送信者は、この ED を受信者の公開鍵を用いて以下の計算によって暗号文 C に変換し、送付する。

$$C = ED^e \bmod n$$

受信者は、自分の秘密鍵 d で暗号化データ C を復号して ED を入手し、ED が規定通りの形式のデータになっていることを確認して M を平文として入手する。

PKCS#1 Version 1 に対する攻撃が有効となるのは、例えば、共通鍵暗号の鍵配送等に利用される双方向通信において、①「受信者が受け取った暗号化データを復号したときに、復号データの最初の 2 byte が"00 02"となっていない等規定の形式とは異なっていた場合に、何らかのエラーメッセージを送信者に送る」等の手続きが組み込まれており、攻撃者が不当な暗号文を送信した場合、そのデータを復号アルゴリズムによって変換した値が上記ファーマットと適合しているか否かを確認することができる環境になっている、さらに、

②「特定の送信者から送信された暗号化データにおいて一定回数エラーが発生した場合、その送信者からのデータ受信をストップする」といった、攻撃者が繰り返し不当な暗号文を送信できないようにする対策が講じられていない、というケースである。こうした実装環境であれば、攻撃者は、自分の選択した暗号化データがエラーとなるか否かを何度でも検証することが可能となり、エラーとならない場合には、その時送信した暗号化データに対応する復号データが"00 02"で始まることがわかる。

上記の実装環境における適応的選択暗号文攻撃の概要は以下の通り。ただし、攻撃対象者（暗号文の受信者）の秘密鍵を(d, p, q)、公開鍵を(e, n)とし、攻撃者がある暗号文 C に対する平文 M を入手しようとする場合を想定する。

（ステップ 1）攻撃者は、次のようなデータ系列 C_i（i = 1,2,…）を作成する。

$$C_i = C \cdot r_i^e \bmod n$$

ただし、r_i（i = 1,2,…）は $0 < r_i < n-1$ を満足する整数であり、攻撃者は、それまでに送付した $C_1,\ldots,C_{i-1}$ に対する攻撃対象者からの反応を確認しながら r_i を決めることができる。

（ステップ 2）攻撃者は、攻撃対象者に C_i（i = 1,2,…）を送付し、攻撃対象者の反応を観察する。復号データが PKCS#1 Version 1 に規定されているデータ形式と異なる場合にはエラーメッセージが送信され、規定のデータ形式（上記の ED の形式）に適合している場合にはエラーメッセージは送信されない。

（ステップ 3）攻撃対象者からエラーメッセージが送信されない場合、攻撃者は、C_i に対する平文 M_i ($= C_i^d \bmod n = Mr_i \bmod n$)の特定の bit に関する情報を入手することができる。

（ステップ 4）攻撃者は、平文 M_i の特定の bit に関する情報を集めて平文 M を求める。

Bleinchenbacher[1998]は、上記の攻撃に必要な選択暗号文数を約 2^{20} 個と試算している。本攻撃法が有効なのは、特定の条件を満たす実装環境においてのみであるが、インターネット上での暗号・認証プロトコルとして広く利用されている SSL Version 3[16]に対して本攻撃が適用できる可能性が指摘されたことから、インターネット技術者の間で大きな注目を集めた。これまでのところ、実際に本攻撃を利用して SSL Version 3 の暗号文を解読したという事例は報告されていないが、本攻撃はインターネットのセキュリティに対する潜在的な脅威と認識されている。

なお、RSA 社は、この発表を受けて PKCS#1 Version 1 を改良し、OAEP を利用した平文の事前処理やパディングを実行する PKCS#1 Version 2 を発表している。

[16] SSL（Secure Socket Layer）Version 3：Netscape 社が提唱する暗号通信、認証等のセキュリティ機能が付加された HTTP プロトコル。SSL Version 3 で利用されるデータ形式等について、PKCS#1 を採用することが規定されている。

（3）証明付きの安全性のレベル比較

①データ守秘専用の方式

これまで提案されているデータ守秘専用の方式の中で、証明付きの安全性と実用性を兼ね備えた主要な方式として、OAEP、EPOC、Cramer-Shoup 暗号が挙げられる。各暗号方式の適応的選択暗号文攻撃に対する安全性を比較すると、以下の表 6 の通り。

表 6　各データ秘匿専用方式の適応的選択暗号文攻撃に対する安全性

	証明されている安全性	安全性の証明に必要となる前提	利用されている数学の問題
OAEP	強秘匿	・ランダムオラクルモデル ・RSA 暗号関数の一方向性	素因数分解問題
EPOC	強秘匿	・ランダムオラクルモデル ・P 部分群問題の困難性	素因数分解問題
Cramer-Shoup 暗号	強秘匿	・汎用一方向性ハッシュ関数 ・Diffie-Hellman 判定問題の困難性	離散対数問題

安全性の証明に利用されている問題の困難性については、長年の研究の中で準指数関数時間の計算量による解法しか示されていない素因数分解問題や離散対数問題の困難性が、最も高い信頼を得ている。したがって、これらの問題の困難性を前提条件として安全性が証明されている暗号方式が望ましい。一方、RSA 暗号関数の一方向性、Diffie-Hellman 判定問題、P 部分群問題の困難性については、素因数分解問題や離散対数問題ほど研究が進んでいるわけではなく、素因数分解問題等との関連性に関する厳密な証明も示されていないことから、素因数分解問題や離散対数問題ほど困難性に対する信頼性は高くない。なお、ランダム・オラクルモデルや汎用一方向性ハッシュ関数の仮定については、後者の方が前者よりも緩い仮定であるとみられている。

3 つの暗号方式の適応的暗号文選択攻撃に対する安全性のレベルを比較すると、いずれも強秘匿であることが証明されているが、Cramer-Shoup 暗号が、他の 2 つの方式（ランダム・オラクルモデル）と比較して緩い仮定（汎用一方向性ハッシュ関数）の下で証明されていることから、Cramer-Shoup 暗号が比較的高い安全性を有していると考えられる。

②ディジタル署名専用の方式

これまで提案されているディジタル署名専用の方式の中で、証明付きの安全性と実用性を兼ね備えた主要な方式として、PSS 署名、改良 ElGamal 署名、TDH-ESIGN 署名が挙げられる。各方式の適応的選択文書攻撃に対する安全性を比較すると、以下の表 7 の通り。

いずれの方式も、適応的選択文書攻撃に対して最も高いレベルの安全性を有している。しかし、どの方式も証明の仮定としてランダム・オラクルモデルが必要となっており、実用化の際にランダム関数をハッシュ関数で置き換える場合には、この仮定が満足されなくなり、安全性の証明は成立しなくなる。なお、ランダム・オラクルモデルを仮定した場合、

3 つの方式の中でも、離散対数問題の困難性を前提としている改良 ElGamal 署名の安全性が比較的高いと考えられる。

表 7　各ディジタル署名専用方式の適応的選択文書攻撃に対する安全性

	証明されている安全性	安全性の証明に必要となる前提	利用されている数学の問題
PSS 署名	存在的偽造不可能	・ランダムオラクルモデル ・RSA 暗号関数の一方向性	素因数分解問題
改良 ElGamal 署名	存在的偽造不可能	・ランダムオラクルモデル ・離散対数問題の困難性	離散対数問題
TDH-ESIGN 署名	存在的偽造不可能	・ランダムオラクルモデル ・ESIGN 署名関数の一方向性	素因数分解問題

3. 主要な数学問題の解法に関する研究

各公開鍵暗号方式の安全性の証明可能性に関する評価に加え、その安全性が依拠している数学の問題がどの程度困難であるかを評価する必要がある。これまで様々な数学の問題の解法に関する研究が進められているが、最も研究が進んでおり、多くの実用的な暗号方式に利用されているものが、①素因数分解問題と②有限体上の乗法群における離散対数問題である。また、最近では、楕円曲線によって定義された離散対数問題を利用した暗号方式がいくつか提案されており、③楕円曲線によって定義された離散対数問題の解法に関する研究も盛んに行われている。以下では、上記 3 つの数学の問題に対する解法について説明する。

(1) 主な素因数分解問題の解法

これまでに提案されてきた素因数分解に有効なアルゴリズムの多くは、「まず適当な整数 z を求め、n と z の最大公約数 p = GCD(n, z)を計算することによって n の約数 p を求める」という手法が基本となっている。これは、2 つの整数 m と n の最大公約数を計算する極めて効率的なアルゴリズム（Euclid の互除法[17]）が存在するためである。したがって、問題は「どのようにして初期値 z を効率的に求めるか」であり、これまでに様々な手法が提案されている。これらの手法は、計算量が素因数 p の性質によって決定する「素因数依存型」と、合成数 n の大きさのみによって決定する「合成数依存型」に大別される。

[17] Euclid の互除法：2 つの整数の最大公約数を求めるアルゴリズムの 1 つ。例えば、210 と 18 の最大公約数を求める場合、次のような手順となる。① 210 を 18 で割って余り 12 を得る、②18 を 12 で割って余り 6 を得る、③12 を 6 で割ると余りは 0 となる、④割り切れたときの割る数 6 が最大公約数となる。このように、最初に 2 つの数のうち大きい方の数を小さい方の数で割り、もし余りが 0 でなければ割る数を余りで割る。この計算を余りが 0 となるまで繰り返し、余りが 0 となったときの割る数が求める最大公約数となる。この手順を用いれば、計算の対象となる数が大きくなっても、効率的に最大公約数が計算できることが知られている。

（i）素因数依存型アルゴリズム

素因数依存型アルゴリズムとしては、「モンテカルロ法」（Pollard[1975]）、p-1 が小さな素数の積になっている場合に有効な「p-1 法」（Pollard[1974]）、p+1 が小さな素数の積になっている場合に有効な「p＋1 法」（Guy[1975]）、p-q が小さい場合に有効な「Fermat 法」（Brillhart[1981]）、楕円曲線を利用する「楕円曲線法」（Lenstra[1987]考案、Montgomery[1987]改良）等が挙げられる。

（ii）合成数依存型アルゴリズム

合成数依存型の主要なアルゴリズムは、2 次合同式 $s^2 = t^2 \pmod n$ を満たす s と t を求め、$z = s \pm t$ とおいて最大公約数 GCD(n, z)を計算して素因数を見つけるという手法である。具体的なアルゴリズムとしては、「連分数法」（Morrison and Brillhart[1975]）、「線形ふるい法」（Schroeppel[1977]）、「2 次ふるい法」（Pomerance[1983]考案、Silverman[1987]改良）、そして、現時点で最高速の素因数分解アルゴリズムである「数体ふるい法」（Lenstra et al.[1990]）が挙げられる。

なお、以下では、計算量の評価の際には一般的に次の記法が用いられる。

$L_p[a, b]=\exp((b+o(1))(\log p)^a(\log\log p)^{1-a})$（ただし、$p\to\infty$のときに、$o(1)\to 0$）

$L_p[a,b]$の値が小さいほど計算量が少ないことを意味する。

①p－1 法と p＋1 法

まず、p－1 法は、Pollard[1974]によって提案された解法であり、その計算量は、合成数 n の素因数 p に対して p－1 の最大素因数のサイズに依存する。このため、p－1 が小さな素数の積になっている場合、すなわち、「ある整数 M に対し、$p-1 = \prod p_i^{e_i}$（$p_i \leqq M$）と表される場合」（このような p－1 は、M-smooth と呼ばれる）に p－1 法が有効となる。p－1 法による素因数分解の考え方は以下の通り。

＜p – 1 法＞

p を n の素因数とする。このとき、n と互いに素な自然数 a が存在して、

$$\mathrm{GCD}(a, n) = 1 \text{ ならば、} \mathrm{GCD}(a, p) = 1$$

となる。したがって、Fermat の小定理[18]から、

$$a^{p-1} = 1 \pmod p$$

となり、p－1 の倍数 K に対して、$a^K=1 \bmod p$ が成立する。素因数 p は a^K-1 と n の最大公約数 $\mathrm{GCD}(a^K-1, n)$を計算することによって求めることができるため、試行錯誤によって p－1 の倍数となるような K を探し当てることが必要となる。

ただし、K を効率的に見つけることができるのは、p－1 の最大素因数が比較的小さな自然数 M 以下の場合に限定される。このとき、必要となる計算量は M のサイズに依存する。

[18] Fermat の小定理は、「素数 p と、p と互いに素な任意の整数 m に対して、$m^{p-1} \bmod p = 1$ が成立する」という定理である。

一方、n の素因数 p に関して、p＋1 が比較的小さな素数の積で表される場合には、p－1 法を若干変形した p＋1 法と呼ばれる方法によって効率的に素因数分解を実行することができる。その際の必要な計算量は、p＋1 の最大素因数のサイズに依存することが示されている（Guy[1975]）。

②Fermat 法

基本的な Fermat 法は、Fermat によって提案されているが、この手法が|p－q|が小さい場合に非常に有効であることを評価したのは、Brillhart[1981]である。この Fermat 法による解法を避けるためには、素数 p と q を|p－q|が大きくなるように選ぶことが必要となる。Fermat 法による素因数分解の考え方は以下の通り。

＜Fermat 法＞

n = pq,　p < q の場合、x = (q+p)/2、y = (q-p)/2 とおくと、

$$x^2 - y^2 = n$$

となる。そこで、（x, y）=（K , 0）（K は $n^{1/2}$ 以下で最大の整数）を初期値として、(x, y) を少しずつ変化させ、$r = x^2-y^2-n$ が 0 となるような（x, y）をみつける。この（x, y）を利用して連立方程式を解くことによって、（p, q）を求めることができる。

なお、$(x+1)^2 = x^2+2x+1$ なので、A = 2x+1、B = 2y+1 とおけば、x → x+1 は r → r+A となり、y → y + 1 は r → r - B となる。このように、加減法だけで計算できるのが本手法の長所である。

③2 次ふるい法

2 次ふるい法（quadratic sieve method）は、2 次合同式 $k^2 = l^2 \bmod n$ を満たす k と l（$k \neq \pm l \pmod n$）を求める解法であり、Pomerance[1983]によって考案された。

2 次ふるい法による素因数分解の考え方は以下の通り。

＜2 次ふるい法＞

2 次合同式 $k^2 = l^2 \bmod n$ の k と l を探すために、次の関数を導入する。

$$Q(x) = (x+m)^2-n$$　（ただし、m は $n^{1/2}$ 以下の最大の整数）

Q(x)に x = 0,±1,±2,…を代入し、Q(x)を素因数分解する。Q(x)は比較的小さな値となるので、小さな素因数を持つ可能性が高い。そこで、小さな素因数 p_j のみを持つ Q(x)、言い換えると、全ての素因数が比較的小さな自然数 v 以下となるような Q(x)をたくさん集める（このような Q(x)は、v-smooth と呼ばれる）。このとき、Q(x)は、

$$Q(x) = p_1^{d1} \cdots p_r^{dr} \pmod n$$

と表される（ただし、$p_1,\cdots,p_r$ は v 以下の素因数）。

次に、Q(x)の集合の中から、次式を満たすような部分集合 S を探し出す（ただし a_j (j=1,

…,r)は偶数）。

$$\prod_{Q(x)\in S} Q(x) = \prod_{S} p_j^{a_j} \pmod{n}$$

このような部分集合 S が見つかれば、

$$k = \prod_{S} (x+m) \bmod n, l = \prod_{S} p_j^{(a_j/2)} \bmod n$$

とおくことにより、$k^2 = l^2$（mod n）が成立する。もし、GCD(k+l, n)、またはGCD(k-l, n)が n の素因数であれば、アルゴリズムは終了する。そうでなければ、別の部分集合 S を見つけて同じ手順を繰り返す。

このアルゴリズムの漸近的な平均計算量は、$L_n[1/2, 1.061]$と準指数関数時間の計算量で評価されている。

④楕円曲線法

楕円曲線法は、p－1 法において利用されている法 p の乗法群を、有限体 F_p 上で定義された楕円曲線の点によって構成される有限可換群 $E(F_p)$で置き換えるとともに、整数 a を $E(F_p)$上の点で置き換えた解法である。前述の p－1 法では、合成数 n が小さな素因数のみによって構成される場合に有効となるのに対し、楕円曲線法では、こうした適用可能な合成数 n に関する制約が存在しない。楕円曲線法による素因数分解の考え方は以下の通り。

＜楕円曲線法＞

素因数分解の対象となる合成数を n として、楕円曲線 E_n：$y^2 = x^3 + ax + b$（mod n）を選び、E_n 上の点 M= (x, y)を適当に選ぶ。

次に、p-1 法と同様に、小さい素数の積からなる合成数 k を生成し、楕円曲線 E_n 上で定義された加法によって、kM を計算する（点 M を k 回加える）。

ここで、n の未知な素因数の 1 つを p とすると、法 p による特殊な写像により、E_n 上の点は、有限体 F_p 上での楕円曲線 $E(F_p)$： $y^2 = x^3 + a'x + b'$（mod p）に移される。この $E(F_p)$ の要素の個数が k の約数となっていたとする（つまり、$E(F_p)$の要素の個数が小さな素数の合成数であったと仮定する）。このとき、$E(F_p)$上において kM = O (mod p)（O は無限遠点）が成立し、E_n 上における kM の x、y 座標の分母の値と n は互いに素ではなくなり、1 よりも大きな公約数を有する。

したがって、$E(F_p)$の要素の個数が k の約数となっていたとすれば、E_n 上の点 kM を$(c_1 / d_1 , c_2 / d_2)$とすると、GCD(d_1, n)と GCD(d_2, n)を計算することによって n の素因数を見つけることができる。そこで、有限体 F_p 上での楕円曲線 $E(F_p)$における要素の個数が小さな素数の積となるように試行錯誤を行い、さらに適当な k を見つけることができれば、n の素因数分解が可能となる。

楕円曲線法の場合、利用する楕円曲線 E を適宜変えることによって楕円曲線の要素の個数を変えることが可能であり、要素の個数が小さな素数の積となるような楕円曲線を比較

的早く探し当てることができる点が特徴である。

楕円曲線法の計算量は素因数 p のサイズに依存しており、平均的な計算量は $L_p[1/2, 1.414]$であり、準指数関数時間の計算量で評価されている。

⑤数体ふるい法

数体ふるい法は、2 次ふるい法の概念を代数的整数上で実現した解法である。数体ふるい法は、素因数分解の対象となる合成数を 2 種類の異なる代数体上で表現し、その表現の相違を利用して、$k^2 = l^2 \bmod n$ を満たす k と l（$k \neq l \pmod n$）を求める方法である。

数体ふるい法による素因数分解の考え方は以下の通り。

＜数体ふるい法＞

n を合成数、f を次数 k の既約多項式（多項式の係数は $\mathbf{Z}_n$（n を法とする加法群）の要素）とする。ただし、f における k 次（最高次）の係数は 1 とする。m を $f(m) = 0 \pmod n$ を満足する整数とし、α を $f(\alpha) = 0$ を満足する数とする。さらに、このαを要素とする特殊な集合 O_K（代数体 $K=Q(\alpha)$の整数環）を用意するとともに、特殊な準同型写像φを以下のように定義する。

$$\varphi : a \mapsto m \bmod n$$

以上の前提の下で、整数のペア（c_j, d_j）を適当に選んで（$c_j + d_j m$）と（$c_j + d_j \alpha$）を計算し、それぞれ小さな素因数の積になるような（c_j, d_j）のペアを大量に集める。

$$c + dm = \prod_{p_i \in F} p_i^{e_i} \text{、} c + da = \prod_{q_i \in G} q_i^{e_i'} \prod_{r_i \in U} r_i^{e_i''}$$

ただし、F をある値以下の素数の集合、G を集合 O_Kにおけるある値以下の素数の集合、U を集合 O_Kにおけるある特殊な基底の集合とする。

集めた（c_j, d_j）のペアを適当に組み合わせて、以下の等式が成り立つようにする

$$\prod (c_j + d_j m) = (\prod_{p_i \in F} p_i^{e_i})^2 \cdots (1) \text{、} \prod (c_j + d_j a) = (\prod_{q_i \in G} q_i^{e_i'} \prod_{r_i \in U} r_i^{e_i''})^2 \cdots (2)$$

ここで、上記の等式（2）をφを用いて変換し、以下の等式を導出する。

$$\prod (c_j + d_j m) = f(\prod_{q_i \in G} q_i^{e_i'} \prod_{r_i \in U} r_i^{e_i''})^2 \bmod n = (\prod_{q_i \in G} f(q_i)^{e_i'} \prod_{r_i \in U} f(r_i)^{e_i''})^2 \bmod n \cdots (3)$$

等式（1）と（3）より、

$$(\prod_{p_i \in G} p_i^{e_i})^2 = (\prod_{q_i \in G} f(q_i)^{e_i'} \prod_{r_i \in U} f(r_i)^{e_i''})^2 \bmod n \cdots (4)$$

が成立することから、$y = \prod_{p_i \in F} p_i^{e_i}$ 、$x = \prod_{q_i \in G} f(q_i)^{e_i'} \prod_{r_i \in U} f(r_i)^{e_i''}$ とおけば、以下の合同式が得られる。

$$x^2 = y^2 \pmod n$$

このような（x, y）のペアをいくつか集めることによって、GCD(x-y, n)を計算して n の素因数を求めることが可能となる。

以上のように、数体ふるい法では、$\mathbf{Z}_n$ と O_K という 2 つの異なる集合を利用することに

よって、合成数 n を 2 通りに分解する点がポイントである。

数体ふるい法による計算量は合成数 n のサイズに依存し、準指数関数時間の計算量 $L_n[1/3, 1.922]$と評価されている（Adleman[1991]、Lenstra et al.[1990]）。現時点では、この数体ふるい法が最高速の素因数分解アルゴリズムとなっている。

（2）有限体上の乗法群における離散対数問題の主な解法

有限体上の乗法群における離散対数問題は、有限体上において対数を計算する問題のことである。実数体上で対数を計算することは大きな数であっても容易であるが、有限体上での対数の計算は数が大きくなるにつれて指数関数的に難しくなることが知られている。素因数分解問題の困難性と理論的に厳密な比較は示されていないものの、最高速の解法における必要計算量によってほぼ同程度の困難度とみられている。

これまでに提案されている離散対数問題の解を求めるアルゴリズムは、次の 2 種類に大別される。

（i）素数 p を法とする乗法群上での離散対数の計算量が、p-1 の最大素因数のサイズに依存するアルゴリズム

代表的なアルゴリズムとして、Shanks[1972]、Pohlig and Hellman[1978]、Pollard[1978]のアルゴリズムが挙げられる。これらのアルゴリズムの計算量は、p-1 における最大素因数のサイズの指数関数時間となる。

（ii）素数 p を法とする乗法群上での離散対数の計算量が、p のサイズに依存するアルゴリズム

このアルゴリズムは「指数計算法」と呼ばれている。代表的な指数計算法としては、Adleman[1979]、Coppersmith[1984]、ElGamal[1985b]、Coppersmith, Odlyzko and Schroeppel[1986]（Gauss の整数法）のアルゴリズムや、数体ふるい法を利用した Gordon[1992]のアルゴリズム、その改良法の Schirokauer[1993]のアルゴリズム、Adleman[1994]の関数体ふるい法等が挙げられる。これらのアルゴリズムの計算量は、p のサイズに対する準指数関数時間となる。

現在、全ての有限体 F_{p^k} に対して最高速となるようなアルゴリズムは存在せず、対象となる有限体によって最高速のアルゴリズムが変わる。k が $k < (\log p)^{1/2-\varepsilon}$（ε：任意の自然数）を満足する場合には数体ふるい法が最高速となり、計算量は $L_{p^k}[1/3, 1.922]$となっている。一方、k が $k \geqq (\log p)^2$ の場合には関数体ふるい法が最高速となり、計算量は $L_{p^k}[1/3, 1.922]$となっている。もっとも、k が $(\log p)^{1/2-\varepsilon} \leqq k < (\log p)^2$ を満足する場合に、これらのアルゴリズムと同程度の計算量となるアルゴリズムは、現時点では提案されていない。

以下では、離散対数問題の主要な解法として、①楕円曲線上で定義された離散対数問題

や、DSA 署名における q を法とする有限体上での離散対数問題に直接適用できるとともに、有限群の要素の個数が特殊な場合には多項式時間の計算量で解くことができる Pohlig-Hellman[1978]のアルゴリズム、②有限体における要素の個数に対する準指数関数時間の計算量によって解くことができる指数計算法の一般的なアルゴリズムとその改良アルゴリズムについて説明する。

①Pohlig-Hellman のアルゴリズム

Pohlig and Hellman[1978]のアルゴリズムは、素数 p を法とする乗法群に対して、要素の個数 p-1 における最大素因数のサイズに対する指数関数時間の計算量を要する。Pohlig-Hellman のアルゴリズムが最も有効となるのは、p-1 の最大素因数がある一定の値（p のサイズ）以下となる場合である。このときの計算量は、多項式時間の計算量 $O((\log p)^2)$ となることが示されている。

Pohlig-Hellman のアルゴリズムの考え方は、以下の通り。

＜Pohlig-Hellman のアルゴリズム＞

p を法とする乗法群 G の原始根を g、$g^x \bmod p = a$ とし、所与の a、g、p の下で離散対数 x を求めるケースを考える。このとき、乗法群 G における要素の個数は p-1 となる。ただし、p-1 は、以下のような合成数となっていることがわかっている。

$$p-1=\prod_{i=1}^{k} q_i^{e_i} \ (q_1 < \cdots < q_k \le \log p)$$

離散対数 x に対して $x \bmod q_i^{e_i}$ (1≦i≦k)をすべて求めることができれば、中国人剰余定理[19]によって、以下のように x を求めることができる。

$$x \bmod \prod_{i=1}^{k} q_i^{e_i} = x \bmod p-1 = x \quad (\because \ x < p\text{-}1)$$

そこで、以下の等式を満たす $b_{i,j}$ (1≦i≦k、0≦j≦e-1)を求める。

$$x \bmod q_i^{e_i} = \prod_{j=1}^{e-1} b_{i,j} q_i^j \ (0 \le b_{i,j} \le q_i - 1)$$

$b_{i,j}$ は、以下のような手順によって見つけることができる。

（**ステップ** 1） $a^{(p-1)/q_i}$ と $g = g^{(p-1)/q_i} \bmod p$ を計算する。

（**ステップ** 2） $g^i = g^{(p-1)/q_i} \pmod p$ を満足する i を、i = 0,1,...と次々に試して見つける。i が見つかったら、$b_{i,0} = i$ とする。

（**ステップ** 3） $a_1 = ag^{-b_{i,0}}$ として、$g^i = a_1^{(p-1)/q_i^2} \pmod p$ となる i を探し、$b_{i,1} = i$ とする。

---

[19] 中国人剰余定理は、「$m_1, m_2, \ldots, m_k$ が互いに素であるとし、$M = m_1 \times \cdots \times m_k$ とする。このとき、連立合同式 $X = d_i \bmod m_i$（i = 1,…,k）の解 X は、法 M の下で一意に存在する」という定理である。

（**ステップ** 4）$a_2 = a_1 g^{-qb_{i,1}}$ **として、**$g^i = a_2^{(p-1)/q_i^3} \pmod p$ **となる** i **を探し、**$b_{i,2} = i$ **とする。**

以上の手順を $b_{k,e-1}$ **まで実行し、**$b_{i,j}$ **を利用して** $x \bmod q_i^{e_i}$ (1≦i≦k)**を求め、**$x \bmod \prod_{i=1}^{k} q_i^{e_i}$ **から** x **を求める。**

②指数計算法

(i) 指数計算法の一般的アルゴリズム

指数計算法の一般的アルゴリズムは、最初に有限体 F_p **上での対数表を作成し、その表を利用して離散対数を計算するという方法である。**

指数計算法の一般的アルゴリズムの考え方は、以下の通り。

＜指数計算法の一般的アルゴリズム＞

まず、最大の素数が v **以下となる素数の集合**$\{p_1,\cdots, p_n \mid 0<p_i<v\}$**に対し、**g **を底とする有限体** F_p **上での対数表を以下の要領で作成する。**

①$g^r \bmod p$ **の最大素因数が** v **以下となるように、有限体** F_p **の要素** r **を選ぶ。このとき、**$0<p_i<v$ **を満足する素数** p_i(i = 1,…,n)**が存在して、以下の等式が成立する。**

$$g^r \bmod p = p_1^{a1}\cdots p_n^{an}$$

②上記等式を満足する$(a_1,\cdots, a_n)$**を求める。**

③①の等式の両辺について、g **を底とする対数をとる。**

$$r \bmod p-1 = a_1 \log_g p_1+\cdots+ a_n \log_g p_n \quad \cdots(1)$$

④適当に選んだ n **個の** r **に対して上記①～③の手順を繰り返すことによって、**n **本の独立な方程式(1)を得ることが可能となり、連立方程式を解いて**$(\log_g p_1 ,\ldots,\log_g p_n)$ **を得る。これらを記録して対数表を作成する。**

$y = g^x \bmod p$ **の離散対数** x **を計算する場合を考える。**$yg^t \bmod p$ **の最大素因数が** v **以下となり、さらに以下の等式が成立するような** $t \in F_p$ **を選ぶ。**

$$yg^t \bmod p = p_1^{b1}\cdots p_n^{bn} \quad \cdots(2)$$

両辺の対数を計算して次式を得る。

$$x = \log_g y = (b_1\log_g p_1+\cdots+ b_n\log_g p_n) - t$$

上記の式を基に、対数表の値（$\log_g p_1 ,\ldots,\log_g p_n$**）を利用することで、離散対数** x **を求めることができる（**b_i (i = 1,…,n)**は、(2)式の計算過程で既知）。**

後述するように、この一般的アルゴリズムに様々な改良を加えることによって、準指数関数時間の計算量による解法がいくつか提案されている。

(ii) 数体ふるい法と関数体ふるい法

指数計算法の一般的アルゴリズムに対して、Gordon[1992]**は、素因数分解のために開発**

された数体ふるい法の概念を応用し、従来の手法よりも高速である準指数関数時間の計算量のアルゴリズムを 2 種類提案した。1 つは一般数体ふるい法に基づくものであり、もう 1 つは特殊数体ふるい法に基づくものである。「一般」と「特殊」については、アルゴリズム自体はほとんど同じであるが、適用可能な乗法群 Z_p における要素の個数 p の性質に差異が存在する。「一般」の場合は、すべてのタイプの p に対して適用可能である一方、「特殊」の場合は、以下のような性質を有する p に対してのみ適用可能である。

性質 1： 乗法群 Z_p の要素を係数とする多項式 F の係数が、ある一定の整数以下となる。
性質 2： $y^k F(x/y) = 0 \pmod p$ を満足するとともに、$p^{1/k}$ 程度の大きさとなる整数 x と y が存在する。
性質 3： 数体ふるい法で利用する特殊な集合 O_K の各要素が一意に分解できる。

要素の個数が素数 p の有限体における特殊数体ふるい法の計算量は L_p [2/5, 1.005]となるほか、一般数体ふるい法の計算量は L_p [1/3, 2.080]となり、いずれも準指数関数時間の計算量になると評価されている。

現在では、上記の Gordon のアルゴリズムにさらに改良が加えられ、一般数体ふるい法よりも高速で解を求めることが可能なアルゴリズムが提案されている。まず、Schirokauer[1993]が、有限体 F_{p^k} に対して k が $k < (\log p)^{1/2-\varepsilon}$（ε：任意の自然数）を満足する場合に最高速となるアルゴリズムを提案している。Schirokauer 版の数体ふるい法における計算量は $L_{p^k}[1/3, 1.922]$となっており、素因数分解問題に対する Adleman-Lenstra 版数体ふるい法と同じ計算量となる。また、Adleman[1994]は、F_{p^k} に対して k が $k \geqq (\log p)^2$ の場合に最高速となる「関数体ふるい法」と呼ばれるアルゴリズムを提案している。関数体ふるい法における計算量は、$L_{p^k}[1/3, 1.922]$となっており、Schirokauer の数体ふるい法における計算量と同じになると評価されている。

表 8　離散対数問題に利用される有限体のタイプと最高速の解法

有限体のタイプ		最高速の解法	計算量
法 p による乗法群の要素の個数 p-1 における最大素因数が log p 以下の場合		Pohlig-Hellman のアルゴリズム	多項式時間 $O((\log p)^2)$
上記以外の場合	有限体 F_{p^k} に対して、$k < (\log p)^{1/2-\varepsilon}$（ε:任意の自然数)となる場合	Schirokauer 版数体ふるい法	準指数関数時間 $L_{p^k}[1/3, 1.922]$
	有限体 F_{p^k} に対して、$k \geqq (\log p)^2$ となる場合	Adleman の関数体ふるい法	準指数関数時間 $L_{p^k}[1/3, 1.922]$
	有限体 F_{p^k} に対して、$(\log p)^{1/2-\varepsilon} \leqq k < (\log p)^2$ となる場合	Schirokauer-Weber-Denny のアルゴリズム	準指数関数時間 $L_{p^k}[1/2, c]$

k が $(\log p)^{1/2-\varepsilon} \leqq k < (\log p)^2$ を満足する場合については、Schirokauer や Adleman によって提案された高速解法と同程度の計算量 $L_{p^k}[1/3, c]$で解を求めるアルゴリズムは、現時

点では提案されていない。計算量が $L_p[1/2, c]$となる Schirokauer-Weber-Denny のアルゴリズムが最高速となっている（Schirokauer et al.[1996]）

このように、現時点では、すべての有限体 F_{p^k} に対して最高速となるアルゴリズムは提案されておらず、適用する有限体に依存して最高速となるアルゴリズムが変わる。これまでの離散対数問題の解法に関する研究成果をまとめると、上記の表 8 の通り。

（3）楕円曲線によって定義された離散対数問題の主な解法

楕円曲線によって定義された離散対数問題には、通常の有限体上の離散対数問題における準指数関数時間の計算量となる高速解法（指数計算法）が適用できないと考えられている。しかし、楕円曲線の中でも（i）MOV 帰着や FR 帰着（後述）によって、拡大次数が小さい拡大体への埋め込みが可能となる楕円曲線（例えば、トレース 0 および 2 の楕円曲線）や、（ii）トレース 1 の Anomalous 曲線については、それぞれ準指数関数時間および多項式時間での解法が適用できることが示されている。トレース 0 の楕円曲線は超特異楕円曲線と呼ばれている。超特異楕円曲線や Anomalous 曲線については、以下の通り。

【超特異楕円曲線】

有限素体 F_p の楕円曲線 $E(F_p)$において、楕円曲線上の点（x,y 座標の値がいずれも F_p の要素となる点）の個数を X とする。このとき、$X = p + 1$ となる（トレースが 0 となる）楕円曲線 $E(F_p)$が、超特異楕円曲線と呼ばれている。

【Anomalous 曲線】

有限素体 F_p の楕円曲線 $E(F_p)$において、楕円曲線上の点（x,y 座標の値がいずれも F_p の要素となる点）の個数を X とする。このとき、$X = p$ となる（トレースが 1 となる）楕円曲線 $E(F_p)$が、Anomalous 曲線と呼ばれている。

①拡大次数が小さい拡大体への埋め込みが可能な楕円曲線

楕円曲線のなす有限可換群 $E(F_p)$は、Menezes、Okamoto and Vanstone[1991]によって提案された「MOV 帰着 （MOV-reduction）」、Frey and Rück[1994]によって提案された MOV 帰着の「FR 帰着（FR-reduction）」、内山・斎藤[1998]によって提案された FR 帰着の改良アルゴリズム「US アルゴリズム」と呼ばれる手法により、多項式時間の計算量によって拡大体 F_{p^k} に埋め込むことが可能となる。しかし、F_{p^k} の拡大次数 k が大きい場合には、数体ふるい法や関数体ふるい法等、有限体上の離散対数問題の高速解法を利用しても現実的な時間内に解を求めることは不可能となる。これらの手法が有効となるのは、拡大次数が小さい場合（$k < \log p$ が必要十分条件とされている（内山・斎藤[1998]））に限定される。このような条件を満足する楕円曲線としては、これまでトレース 0 の超特異楕円曲線やトレース 2 の楕円曲線が知られている。

もっとも、要素の個数が素数となるような任意の楕円曲線に対してこれらの手法が有効となる確率は、$O((\log p)^9 (\log\log p)^2 / p)$ 以下となり、非常に小さいことが示されている

（Balasubramanian and Koblitz[1998]）。

（i）超特異楕円曲線

超特異楕円曲線 $E(F_p)$に対しては、MOV 帰着あるいは FR 帰着（US アルゴリズム）を利用することで、多項式時間によって、k が高々6 程度となる拡大体 F_p^k に埋め込むことが可能となることが知られている（Menezes, Okamoto and Vanstone[1991]）。このため、有限体上の離散対数問題に適用可能な解法によって、準指数関数時間の計算量で解を求めることができる。

（ii）トレース 2 の楕円曲線

トレース 2 の楕円曲線（要素の個数が p-1）$E(F_p)$によって定義された離散対数問題に対しては、FR 帰着（US アルゴリズム）が有効となる（内山・斎藤[1998]）。このタイプの楕円曲線に FR 帰着（US アルゴリズム）を適用することで、楕円曲線上の離散対数問題を確率的多項式時間によって F_p（拡大次数 k が 1）に帰着させることが可能となり、準指数関数時間の計算量の解法が適用可能となることが示されている。一方、MOV 帰着を適用した場合には、拡大次数 k が大きくなるため、準指数関数時間の計算量の解法が適用できなくなる。

②Anomalous 曲線

Anomalous 曲線によって定義された離散対数問題は、フェルマー商と呼ばれる関数を利用することによって、楕円曲線上の有限可換群 $E(F_p)$を同じ要素の個数をもつ加法群に埋め込むことにより、多項式時間の計算量（$(\log p)^3$ に比例する計算量）で解を求めることが可能となる。この解法は、Smart[1997]、Semaev[1998]、Satoh and Araki[1998]によって発表されたことから、「SSSA アルゴリズム」と呼ばれている。

ただし、任意に選んだ楕円曲線が Anomalous 曲線となる確率は非常に小さいとみられており、十分大きな素数 p の下での確率は $O((\log p)(\log\log p)/\sqrt{p})$ 程度と見積もられている（Satoh and Araki[1998]）。

③その他の楕円曲線

上記①、②以外の楕円曲線によって定義された離散対数問題を解くためには、現時点では、指数関数時間の計算量が必要になるとみられている。最高速の解法である Pohlig-Hellman のアルゴリズムにおける計算量は、$\exp(0.5\log p)$程度と見積もられている。以上の研究成果をまとめると、以下の表 10 の通り。

表 10　楕円曲線の種類と各楕円曲線に基づく離散対数問題の解法

楕円曲線のタイプ	適用可能な解法	必要計算量	各解法が適用可能となる確率
拡大次数が小さい拡大体への埋め込みが可能な楕円曲線	MOV 帰着、FR 帰着（US アルゴリズム）	いずれも準指数関数時間	$O((\log p)^9(\log\log p)^2/p)$
超特異楕円曲線（トレース 0）	MOV 帰着、FR 帰着（US アルゴリズム）		
トレース 2 の楕円曲線	FR 帰着（US アルゴリズム）		
Anomalous 曲線（トレース 1）	SSSA アルゴリズム	多項式時間	$O((\log p)(\log\log p)/\sqrt{p})$
上記以外の楕円曲線	——	指数関数時間	——

（4）各数学問題の必要計算量の比較

上記 3 種類の数学の問題を解くために必要な計算量を試算・比較する。比較対象とする各数学問題の解法には、現時点で最高速といわれている解法を適用する。素因数分解問題には Adleman-Lenstra 版の数体ふるい法、有限体上の乗法群における離散対数問題には Schirokauer 版数体ふるい法（あるいは Adleman の関数体ふるい法）、楕円曲線によって定義された離散対数問題には Pohlig-Hellman のアルゴリズムを利用する。各アルゴリズムの必要計算量は、以下の表 11 の通り。

表 11　各数学の問題における高速解法と必要計算量

	高速解法	必要計算量
素因数分解問題	Adleman-Lenstra 版数体ふるい法	$\exp((1.922+o(1))(\log p)^{1/3}(\log\log p)^{2/3})$
有限体上の乗法群における離散対数問題	Schirokauer 版数体ふるい法（Adleman の関数体ふるい法）	
楕円曲線によって定義された離散対数問題	Pohlig-Hellman のアルゴリズム	$\exp(0.5(\log p))$

素因数分解問題の困難性と離散対数問題の困難性との関係については、理論的に厳密な研究成果は現時点では示されていないものの、必要となる計算量によって比較するとほぼ同程度になるとみられている。

問題のサイズによってこれらの計算量を比較すると、以下の表 12 の通り。問題のサイズには、素因数分解問題であれば素数 p のサイズ、離散対数問題であれば法 p のサイズ、楕円曲線によって定義された離散対数問題であれば有限体における要素の個数 p のサイズが対応する。

表 12 問題のサイズと解を求めるために必要な計算量の関係

問題のサイズ (bit)	必要計算量	
	素因数分解問題 離散対数問題	楕円曲線によって 定義された離散対数問題
160	—	2^{80}
174	—	2^{87}
200	—	2^{100}
234	—	2^{117}
240	—	2^{120}
280	—	2^{140}
512	1.86×2^{63}	2^{256}
768	1.39×2^{76}	2^{384}
1,024	1.65×2^{86}	2^{512}
1,536	1.25×2^{103}	2^{768}
2,048	1.77×2^{116}	2^{1024}

このように、素因数分解問題および離散対数問題における問題のサイズが 1,024 bit、2,048 bit **の場合の必要計算量は、それぞれ楕円曲線によって定義された離散対数問題における問題のサイズの** 174 bit、234 bit **の必要計算量とほぼ同等となるとみられる。**

Ｖ．おわりに

従来から、公開鍵暗号方式における安全性の証明に関する研究が盛んに行われ、安全性が証明可能な様々な暗号方式が提案されてきたが、いずれの方式も計算量が多い等の理由から実用化には至らなかった。これに対し、最近では、安全性と実用性を両立させる暗号方式が提案され、OAEP 等一部の方式は既に実用化されている。こうした公開鍵暗号における安全性証明に関する研究は、公開鍵暗号技術に対する一層の信頼性向上に繋がるほか、オープンなネットワークにおける情報セキュリティの向上に資するものである。今後とも、公開鍵暗号の研究動向を注視する必要があろう。

また、最近の公開鍵暗号における理論研究によって、暗号システムのセキュリティを確保するためには、利用されている暗号方式単体の安全性を確保するだけでは不十分であるとの認識が一層強まっている。Bleichenbacher によって発表された PKCS#1 Version 1 に対する攻撃法は、利用されている RSA 暗号を直接攻撃するものではなく、攻撃者が送信したデータに対する攻撃対象者の返信データから、対応する平文の情報を入手するというものである。したがって、Bleichenbacher の攻撃を防ぐためには、少なくとも、攻撃者が送信したデータに対して不用意に平文に関する情報を返信しないような実装が必要となる。このように、情報セキュリティ技術を「総合技術」として位置付け、実装方法を含めて「どのようなセキュリティ技術を採用するか」を検討することが重要である。今後も、PKCS#1 の安全性に関する研究のように、実際に利用されている具体的な実装方法の安全性に関する研究についても注視していく必要があろう。

以　上

参考文献

池野信一・小山謙二、『現代暗号理論』、電子情報通信学会、1986 年.

岩下直行・谷田部充子、「金融分野における情報セキュリティ技術の国際標準化動向」、日本銀行金融研究所情報セキュリティ・シンポジウム提出論文、1998 年 11 月.

内山成憲・齋藤泰一、「トレース 2 の楕円曲線上の離散対数問題について」、ISEC98-27、pp. 51-57、1998 年.

宇根正志、「公開鍵暗号方式 EPOC について」、日本銀行金融研究所ディスカッションペーパーシリーズ、No. 97-J-19, 1998 年 7 月.

宇根正志・太田和夫、「共通鍵暗号を取り巻く現状と課題 －DES から AES へ－」、日本銀行金融研究所情報セキュリティ・シンポジウム提出論文、1998 年 11 月.

岡本龍明・太田和夫（編著）、『暗号・ゼロ知識証明・数論』、共立出版、1995 年.

岡本龍明・藤岡淳・岩田雅彦、「高速ディジタル署名方式 ESIGN」、NTT R&D、Vol. 40、No. 5、1991 年.

岡本龍明・山本博資、『現代暗号』、産業図書、1997 年.

勝野裕文、「RSA 暗号解読への一対策」、電子情報通信学会論文誌(D)、Vol.J66-D、No.8、pp. 963-969、1983 年.

楠田浩二・櫻井幸一、「公開鍵暗号方式の安全性評価に関する現状と課題」、日本銀行金融研究所ディスカッションペーパーシリーズ、No. 97-J-11, 1997 年 7 月.

黒澤馨・伊東利哉・竹内正士、「素因数分解の困難さと同等の強さを有する逆数を利用した公開鍵暗号」、電子情報通信学会論文誌、Vol. J70-A、No. 11、pp. 1632-1636、1987 年 11 月.

小山謙二・桑門秀典・鶴岡行雄、「3 次曲線に基づく公開鍵暗号」、NTT R&D Vol. 44 No. 10、1995 年.

辻井重男・笠原正雄（編著）、『暗号と情報セキュリティ』、昭晃堂、1990 年.

中山靖司・太田和夫・松本勉、「電子マネーを構成する情報セキュリティ技術と安全性評価」、日本銀行金融研究所情報セキュリティ・シンポジウム提出論文、1998 年 11 月.

松本勉・岩下直行、「金融分野における情報セキュリティ技術の現状と課題」、日本銀行金融研究所情報セキュリティ・シンポジウム提出論文、1998 年 11 月.

宮地充子、「ElGamal 型楕円曲線暗号の設計と標準化及び実装化動向」、1997 年暗号と情報セキュリティシンポジウム、SCIS'97-7B、1997 年.

L. M. Adleman, “A subexponential algorithm for the discrete logarithm problem with applications to cryptography,” Proceedings of FOCS, pp. 55-60, 1979.

L. M. Adleman, “Factoring Numbers Using Singular Integers,” Proceedings of ACM Annual Symposium on Theory of Computing, pp. 64-71, Springer-Verlag, 1991.

L. M. Adleman, “The function field sieve,” Algorithmic Number Theory, Lecture Notes in Computer Science, Vol. 877, pp. 108-121, Springer-Verlag, 1994.

American National Standards Institute, “X9.30 Part 1: Public Key Cryptography Using Irreversible Algorithm (DSA)” 1997.

American National Standards Institute, “Working Draft: X9.31 Digital Signatures Using Reversible Public Key Cryptography For The Financial Service Industry (rDSA),” 1998.

American National Standards Institute, “Working Draft: X9.62 Public Key Cryptography For The Financial Service Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA),” 1998.

K. Araki, T. Satoh and S. Miura, “Extended Abstract of Overview of Elliptic Curve Cryptography,” Proceedings of 1998 International Workshop on Practice and Theory in Public Key Cryptography, 1998.

R. Balasubramanian and N. Koblitz, “The Improbability That an Elliptic Curve Has Subexponential Discrete Log Problem under the Menezes-Okamoto-Vanstone

Algorithm," Journal of Cryptology, Vol. 11, No. 2, pp.141-145, 1998.
M. Bellare, A. Desai, D. Pointcheval, and P. Rogaway, "Relations Among Notions of Security for Public-Key Encryption Schemes," Advances in Cryptology — Proceedings of CRYPTO '98, Lecture Notes in Computer Science, Vol. 1462, pp. 26-35, Springer-Verlag, 1998.
M. Bellare and S. Goldwasser, "New paradigms for digital signatures and message authentication based on non-interactive zero knowledge proofs," Advances in Cryptology — Proceedings of CRYPTO '89, Lecture Notes in Computer Science, Vol. 435, pp.194-211, Springer-Verlag, 1990.
M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols," Proceedings of the First Annual Conference on Computer and Communications Security, ACM, 1993.
M. Bellare and P. Rogaway, "Optimal asymmetric encryption," Advances in Cryptology — Proceedings of EUROCRYPT '94, Lecture Notes in Computer Science, Vol. 950, pp. 92-111, Springer-Verlag, 1995.
D. Bleichenbacher, "Generating ElGamal signatures without knowing the secret key," Advances in Cryptology — Proceedings of EUROCRYPT '96, Lecture Notes in Computer Science, Vol. 1070, pp. 10-18, Springer-Verlag, 1996.
D. Bleichenbacher, "Chosen Ciphertext Attacks Against Protocols Based on the RSA Encryption Standard PKCS #1," Advances in Cryptology — Proceedings of CRYPTO '98, Lecture Notes in Computer Science, Vol. 1462, pp. 1-12, Springer-Verlag, 1998.
D. Boneh and R. Venkatesan, "Breaking RSA May Not Be Equivalent to Factoring," Advances in Cryptology — Proceedings of EUROCRYPT '98, Lecture Notes in Computer Science, Vol. 1403, pp. 59-71, Springer-Verlag, 1998.
E. F. Brickell and J. DeLaurentis, "An attack on a signature scheme proposed by Okamoto and Shiraishi," Advances in Cryptology — Proceedings of CRYPTO '85, Lecture Notes in Computer Science, Vol. 218, pp. 28-32, Springer-Verlag, 1986.
J. Brillhart, "Fermat's factoring method and its variants," Congressus Numerantium, Vol. 32, pp.29-48, 1981.
J. Camenisch, U. Maurer, and M. Stadler, "Digital payment systems with passive anonymity-revoking trustees," Computer Security — Proceedings of ESORICS '86, Lecture Notes in Computer Science, Vol. 1146, pp. 33-43, Springer-Verlag, 1996.
D. Chaum, "Blind signatures for untraceable payments, "Advances in Cryptology — Proceedings of CRYPTO '82, pp. 199-203, Plenum Press, 1983.
D. Coppersmith, "Fast evaluation of logarithms in fields of characteristic two," IEEE Transactions on Information Theory, Vol. IT-30, pp. 587-594, 1984.
D. Coppersmith, A.M. Odlyzko, and R. Schroeppel, "Discrete logarithms in GF(p)," Algorithmica, Vol. 1, pp. 1-15, 1986.
D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter, "Low-exponent RSA with related messages," Advances in Cryptology — Proceedings of EUROCRYPT '96, Lecture Notes in Computer Science, Vol. 1070, pp. 1-9, 1996.
R. Cramer and V. Shoup, "A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack," Advances in Cryptology — Proceedings of CRYPTO '98, Lecture Notes in Computer Science, Vol. 1462, pp. 13-25, Springer-Verlag, 1998.
W. Diffie and M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol.IT-22, pp.644-654, November 1976.
W. Diffie, P. C. van Oorschot, and M. J. Wiener, "Authentication and authenticated key exchanges," Designes, Codes and Cryptography, Vol.2, pp.107-125, 1992.
C. Dwork and M. Naor, "An efficient existentially unforgeable signature scheme and its applications," Advances in Cryptology — Proceedings of CRYPTO '94, Lecture

Notes in Computer Science, Vol. 839, pp. 234-246, Springer-Verlag, 1994.
T. E. ElGamal, “A public key cryptosystems and a signature scheme based on discrete logarithm,” Advances in Cryptology — Proceedings of CRYPTO ’84, Lecture Notes in Computer Science, Vol. 197, pp. 10-18, Springer-Verlag, 1985a.
T. E. ElGamal, “A subexponential-time algorithm for computing discrete logarithms over GF(p^2),” IEEE Transactions in Information Theory, Vol. IT-31, pp. 473-481, 1985b.
G. Frey and H. G. Rück, “A Remark Concerning m-divisibility and The Discrete Logarithm in The Divisor Class Group of Curve,” Math. Comp., Vol. 62, No. 206, pp. 865-874, 1994.
E. Fujisaki and T. Okamoto, “How to Enhance the Security of Public-Key Encryption at Minumum Cost,” manuscript, 1998a.
E. Fujisaki and T. Okamoto, “TDH-ESIGN: ESIGN with Trisection Domain Hash,” manuscript, 1998b.
S. Goldwasser and S. Micali, “Probabilistic encryption,” Journal of Computer and System Sciences, Vol. 28, No. 2, pp. 270-299, 1984.
S. Goldwasser, S. Micali, and R. Rivest, “A digital signature scheme against adaptive chosen message attack,” SIAM J. Comp., Vol. 17, No. 2, pp. 281-308, 1988.
D. M. Gordon, “Discrete logarithm field in GF(p) using the Number Field Sieve,” SIAM Jounal on Discrete Math., Vol. 6, pp.124-138, 1993.
R. K. Guy, “How to factor a number,” Proceedings of 5th Manitoba Conference on Numerical Mathematics, pp. 49-89, 1975.
International Organization for Standardization, “ISO 11166-2 Banking — Key Management by means of asymmetric algorithms — Part 2: Approved algorithms using the RSA cryptosystem,” 1994.
International Organization for Standardization and International Electrotechnical Commission, “ISO/IEC FDIS 14888-3 Information technology — Security techniques — Digital signature with appendix — Part 3: Certificate-based mechanism,” 1998.
N. Koblitz, “Elliptic curve cryptosystems,” Mathematics of Computation, Vol.48, pp.203-209, 1987.
N. Koblitz, A Course in Number Theory and Cryptography, Springer-Verlag, 1997.（N. Koblitz 著、櫻井幸一訳、『数論アルゴリズムと楕円暗号理論入門』、シュプリンガー・フェアラーク、1997 年.）
A. K. Lenstra, H. W. Jr. Lenstra, M.S. Manasse, and J.M. Pollard, “The number field sieve,” Proceedings of ACM Annual Symposium on Theory of Computing, pp. 564-572, 1990.
H. W. Jr. Lenstra, “Factoring Integers with elliptic Curves,” Annals of Mathematics, Vol. 126, pp. 649-673, 1987.
A. Menezes, T. Okamoto, and S. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field,” Proceedings of STOC, pp. 80-89, 1991.
V. S. Miller, “Use of elliptic curves in cryptography,” Advances in Cryptology — Proceedings of CRYPTO ’85, Lecture Notes in Computer Science, Vol. 218, pp. 417-426, Springer-Verlag, 1986.
P. L. Montgomery, “Speeding the Pollard and elliptic curve methods of factorization,” *Math., Comp.*, pp. 243-264, 1987.
M. Naor and M. Yung, “Universal one-way hash functions and their chosen ciphertext attacks,” Proceedings of STOC, pp. 33-43, 1989.
M. Naor and M. Yung, “Public-key cryptosystems provably secure against chosen ciphertext attacks,” Proceedings of STOC, pp. 427-437, 1990.
National Institute for Standards and Technology, “Specifications for a digital signature standard,” Federal Information Processing Standard Publication 186, 1991.
T. Okamoto, “A fast signature scheme based on congruential polynomial operations,” IEEE

Transactions on Information Theory, Vol. 36, No. 1, pp. 47-53, 1990.
T. Okamoto and S. Uchiyama, “A New Public-Key Cryptosystem as Secure as Factoring,” Advances in Cryptology — Proceedings of EUROCRYPT ’98, Lecture Notes in Computer Science, Vol. 1403, pp. 308-318, Springer-Verlag, 1998.
S. Pohlig and M. Hellman, “An improved algorithm for computing logarithms over GF(p) and its cryptographic significance,” IEEE Transactions on Information Theory, Vol. 24, pp. 106-110, 1978.
D. Pointcheval and J. Stern, “Security proofs for signature schemes,” Advances in Cryptology — Proceedings of EUROCRYPT ’96, Lecture Notes in Computer Science, Vol. 1070, pp. 387-398, Springer-Verlag, 1996.
J. M. Pollard, “Theorems on factorization and primality testing,” Proceedings of Camb. Phil. Soc., Vol. 76, pp.521-528, 1974.
J. M. Pollard, “A Monte Carlo method for factorization,” BIT, Vol. 15, pp. 331-334, 1975.
C. Pomerance, “The quadratic sieve algorithm,” Lecture Notes in Computer Science, Vol. 209, pp.169-182, 1985.
M. O. Rabin, “Digitalized signatures and public-key functions as intractable as factorization,” M.I.T. Laboratory for Computer Science, Technical Report MIT/LCS/TR-212, 1979.
R. L. Rivest, A. Shamir, and L. Adleman, “A method of obtaining digital signatures and public key cryptosystems,” Communications of the ACM, Vol. 21, No. 2, pp.120-126, 1978.
J. Rompel, “One-way functions are sufficient for secure signatures,” Proceedings of STOC, pp. 387-394, 1990.
I. A. Semaev, “Evaluation of Discrete Logarithms in a Group of p-torsion Points of an Elliptic Curve in Characteristic p,” Math. Comp., Vol. 67, No. 221, pp. 353-356, 1998.
T. Satoh and K. Araki, “Fermat Quotients and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves,” Commentarii Math, Univ. Sancti Pauli, 1998.
B. Schneier, *APPLIED CRYPTOGRAPHY*, John Wiley & Sons, Inc., 1993.
O. Schirokauer, “Discrete Logarithms and Local Units,” Phil. Trans. R. Soc. Lond., A 345, pp.409-423, 1993.
O. Schirokauer, D. Weber, and T. Denny, “Discrete Logarithms: The Effectiveness of the Index Calculus Method,” Algorithmic Number Theory, Lecture Notes in Computer Science, Vol. 1122, Springer-Verlag, pp. 335-361, 1996.
C. P. Schnorr, “Efficient signature generation for smart cards,” Advances in Cryptology — Proceedings of CRYPTO ’89, Lecture Notes in Computer Science, Vol. 435, pp.239-252, Springer-Verlag, 1990.
D. Shanks, “Class number, a theory of factorization, and Genera,” Proceedings of Syposium Pure Mathematics, AMS, 1972.
R. D. Silverman, “The multiple polynomial quadratic sieve,” Math. Comp., Vol. 48, pp. 243-264, 1987.
G. J. Simmons, “A ‘weak’ privacy protocol using the RSA crypto algorithm,” Cryptologia, Vol.7, No.2, pp.180-182, April 1983.
N. P. Smart, “The discrete logarithm problem on elliptic curves of trace one,” preprint, 1997.
H. C. Williams, “A modification of the RSA public-key encryption procedure,” IEEE Transactions on Information theory, Vol. IT-26, pp. 726-729, 1980.
H. C. Williams and B.K. Schmid, “Some remarks concerning the MIT public-key cryptosystem,” BIT, Vol. 19, pp. 525-538, 1979.