IMES DISCUSSION PAPER SERIES

暗号と特許

相澤英孝・宇根正志・楠田浩二

Discussion Paper No. 98-J-9

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-9
1998 年 6 月

暗号と特許

相澤英孝*・宇根正志**・楠田浩二***

要　旨

インターネットが急速に普及するに伴い、オープンなネットワーク上での情報通信において、情報の秘匿や改ざん防止のための技術が必要とされている。特に、電子商取引、電子マネーにおいては、情報セキュリティ確保のための技術が欠くべからざるものとなっている。そのための技術として、暗号の重要性が認識されている。代表的な暗号として、共通鍵暗号、公開鍵暗号、ディジタル署名、ブラインド署名、ハッシュ関数や鍵配送・共有法が挙げられる。

こうした暗号の中には、特許が取得されているものも存在する。このため、オープンなネットワークにおける情報通信セキュリティ確保のために暗号に対する要請が高まる一方、暗号を利用するにあたっては、特許に配慮する必要がある。この問題を考えるためには、現在、暗号が特許によってどのように保護されているのかについて分析し、暗号に関する特許の問題点について検討することが有効であろう。

本稿では、まず、技術が特許による保護を受けるための要件や特許の効力の範囲等について説明するとともに、暗号の特許法による保護について説明する。続いて、ディジタル署名、ハッシュ関数、共通鍵暗号、鍵配送・共有法やブラインド署名といった暗号の技術的内容について概説し、これらの暗号の開発・研究がどのように進められてきたのか、また個々の暗号がどのように関連しているのかを整理する。その上で、既に特許が成立している主要な暗号の特許について、クレーム（請求の範囲）を中心に分析し、実際にどのようなクレームが特許として認められているのかを明らかにするとともに、具体的な特許を例にとり、特許の保護の要件や効力の範囲について検討する。

キーワード：暗号、特許、クレーム、コンピュータ、ソフトウェア
JEL classification: K11、L86、L96、O34

*　早稲田大学アジア太平洋研究センター
**　日本銀行金融研究所研究第 2 課（E-mail: masashi.une@boj.or.jp)
***日本銀行人事局

本稿は、相澤英孝が日本銀行金融研究所における客員研究員として、宇根正志、楠田浩二と共同で行った研究の一部をまとめたものである。本稿作成にあたっては、横浜国立大学の松本勉助教授、NTT 情報通信研究所の岡本龍明特別研究員および太田和夫特別研究員、東芝知的財産部の光主清範参事から有益なコメントを頂戴した。

目　次

Ⅰ はじめに

インターネットの急速な普及に伴い、オープンなネットワーク上での情報通信における情報の秘匿や改ざん防止のための技術として、暗号が注目されている。

代表的な暗号として、共通鍵暗号、公開鍵暗号、ディジタル署名、ブラインド署名、ハッシュ関数や鍵配送・共有法が挙げられる。共通鍵暗号は、暗号化と復号化に同一の鍵が用いられる暗号であり、公開鍵暗号は、暗号化と復号化に異なる鍵が用いられる暗号である。共通鍵暗号は、一般的に公開鍵暗号よりも高速処理が可能である反面、送受信者間で秘密に鍵を共有する必要があり、そうした鍵共有の技術として鍵配送・共有法がある。公開鍵暗号は、暗号化に用いられる鍵と復号化に用いられる鍵が異なる暗号方式であり、暗号化に用いられる鍵を「公開鍵」として公開する一方、復号化に用いられる鍵は「秘密鍵」として秘密にしておくことによって、予め通信相手と秘密に鍵を共有することなく暗号通信を行うことが可能となる。また、ディジタル署名は、通信データに固有の署名データを作成することにより、データ送信者の認証やデータの改ざんの検出等を可能にする技術であり、主に公開鍵暗号によって実現される。このディジタル署名の応用技術の１つがブラインド署名であり、署名者にデータの内容を知られないように署名を作成することが可能となる。ハッシュ関数は、あるデータを圧縮して一定の長さのデータを生成する技術であり、圧縮したデータから元のデータを推測することを困難にする。

特許は、特許権者にその技術についての排他的権利を与えることにより、技術の研究・開発を促進するインセンティブを付与する制度である。現在、日本の特許法上発明として認められるのは、「自然法則を利用した技術的思想のうち高度のもの」である。これまでの判例、学説によると、暗号が「自然法則を利用した技術的思想」に含まれるか否かについては議論の余地がある。実際に多くの暗号に関連する特許が成立しているという事実も踏まえ、暗号をどのように特許の保護の対象として位置付けるかを考える必要がある。

こうした状況下、暗号を使用する際には、先行する暗号の特許について配慮する必要がある。特許の保護の範囲はそのクレームの記載内容が基準となるが、先端技術である暗号については、特許庁によって認められるクレーム及びクレームの解釈に関する考え方が定着していない。今後、暗号に対する要請が一層高まると予想される中、こうした暗号の特許を巡る問題が重要となると考えられる。このような問題について検討するためには、まず、実際にどのような暗号が特許として認められているのかを整理し、それらの特許の効力やその範囲等について分析することが有効であろう。

そこで、本稿では、特許制度や暗号について概観した上で、いくつかの暗号の特許を取り上げてその内容を整理し、それらの特許を例に暗号を巡る問題点を検討する。まず第 2 章において、日本とアメリカにおける特許制度について概説する。具体的には、技術が特許による保護を受けるための要件や、特許の効力の範囲等について説明するとともに、暗号を特許の対象とする際に生じる問題点について整理する。第 3 章では、ディジタル署名、ハッシュ関数、共通鍵暗号、鍵配送・共有法やブラインド署名について、これらの暗号の開発・研究がどのように進められてきたのかをフォローしつつ、個々の暗号の技術的な内容を概説する。第 4 章におい

ては、数多くの暗号に関連する特許の中から 16 の特許を取り上げ、実際にどのようなクレームが発明として認められているのか、また技術の開示はどのように行われているか等を明らかにする。最後に、第 5 章において、第 4 章で整理した特許を例に、新規性、進歩性、開示、特許の効力やその範囲等に関して分析を行い、どのような問題点が存在するのかを検討する。

Ⅱ 特許法と暗号

1．保護の対象

特許制度は、新たに開発された技術の開発者に対して財産権を与えることにより、新技術の開発に対してインセンティヴを与えることを主たる目的とする制度である[1]。特許制度が産業革命とともに発展してきたという背景もあり、保護の対象となる技術を発明（invention）とし、発明は自然科学を応用した技術であるとする考え方が、19 世紀に発達した特許法の基礎的な考え方となっている。この考え方は、機械技術の背景となる物理学などを自然科学上のものとして特許法の保護の対象外とし、自然科学を応用した技術として機械技術を特許の保護の対象としようとするものである。

日本の特許法は、この考え方を基礎として、特許の保護の対象である発明を、「...自然法則を利用した技術的思想のうち高度のものをいう」と規定した（特許法第 2 条）[2]。この規定の「自然法則を利用した技術的思想」について、中山教授は、「自然法則の利用」は「単なる精神活動、純然たる学問上の法則、人為的な取極等」を除外する意味であるとしている。そして、「純然たる学問上の法則」の例として「ピタゴラスの定理のような数学上の法則」を挙げ、「人為的な取極」の例として「暗号表」を挙げている[3]。豊崎教授は、「自然法則を利用しないもの、ことに単なる頭脳の産物は発明とされない。判決に見えたものとしては暗号や隠語の作成方法などがある。...数式、計算方法、コンピュータ・プログラムなども一般に同様に取り扱われている」としている[4]。

1953 年の最高裁判所の判決は、「欧文字単一電報隠語作成方法」を対象とした特許出願を「何等装置等を用いず、又、自然力を利用した手段を施していないから、特許に値する工業的発明であるとはいえない...」とする原判決[5]を維持し、特許の対象である発明にはあたらないとしている[6]。この「欧文字単一電報隠語作成方法」については、最高裁判決と東京高裁判決によると、「同一の代表番号を有する通信語句と隠語との置き換えにより各通信語句より隠語化（即ち通信語句より隠語への翻訳）又はその逆たる各隠語の通信語句化（即ち隠語より通信語句への翻訳）を完成する」ために利用されるもので、「欧文字数字記号等を適当に組合わせて電報用の暗号を作成する方法」であると記載されている。また、実施形態については、「一定の通信語句集と、暗号変種表と、その実施方法の説明書との外には唯だ筆、墨、紙等」を利用する、と説明されている。これらの記述から、「欧文字単一電報隠語作成方法」は、特段装置を用いることなく、平文と暗号文の対応関係を表す「暗号表」によって平文を暗号文に置き換える方

---

1 歴史的には、外国からの技術の導入等の理由があり、最近では、通商政策的理由もある。

2 この規定が、特許法に規定されたのは戦後のことであり、この定義規定を置いたことについては、現在では、批判も多い。

3 中山信弘『工業所有権法（上）』、弘文堂、1993 年の 102 頁を参照。

4 豊崎光衛『工業所有権法［新版・増補］』、有斐閣、1980 年の 151 頁を参照。

5 東京高判昭和 25 年 2 月 28 日行政事件裁判例集 1 巻 7 号 1066 頁

6 最判昭和 28 年 4 月 30 日民事判例集 7 巻 4 号 461 頁

法のことであるとみられる。特許庁は、1997 年 3 月に発表した「産業上利用することができる発明」の審査の運用指針で、「自然法則以外の法則など及びこれらのみを利用しているもの」は「自然法則を利用した技術的思想の創作」ではないから、「発明」に該当しないとしている。「自然法則以外の法則など及びこれらのみを利用しているもの」の例として、『...発明が...数学上の公式...あるいはこれらのみを利用しているときは、その発明は、自然法則を利用したものとは言えず、「発明」に該当しない』としている。さらに、同時に発表された「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」では、「解決手段が数学的解法...である場合は、自然法則を利用した手段とはいえない」としている[7]。

これらの学説、判例、運用指針の一般的な議論からすると、①暗号[8]は「自然法則を利用した技術的思想」ではないとする議論、②暗号は数値変換の方法であり、数値変換の方法は数学的な方法であるから「自然法則を利用した技術的思想」ではないとする議論、③暗号は通信に関連する技術であるから他の通信装置あるいは通信方法と同様に「自然法則を利用した技術的思想」であるとする議論、④現代の暗号はコンピュータの利用を不可欠としているから、コンピュータを利用することで「自然法則を利用した技術的思想」であるとする議論が成り立ちそうである。実務では、特許庁は、暗号を対象とする多くの特許を認めているが、現代の暗号に関する裁判所の判決例はない。

現代の暗号はコンピュータの利用を不可欠としている。したがって、暗号はコンピュータ・ソフトウェアという側面も持っている。コンピュータ・ソフトウェアが特許の対象になるとすれば、暗号も特許の対象となることになる。コンピュータ・ソフトウェアがコンピュータを利用することで「自然法則を利用した」ものとして特許の対象になる[9]と考えれば、現代の暗号は特許の保護対象になると考えられる。ただし、特許庁は、「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」において、『解決手段が...「コンピュータを用いて処理することのみ」である場合...には、「発明」とはしない』としており、この運用指針では、暗号はコンピュータを用いた技術であるから暗号が特許の対象になるという考え方はとっていない[10]。

---

[7] この運用指針は、1993 年の「審査基準（改訂版）」を踏襲している。

[8] コンピュータが発達する以前は、暗号は専用の暗号装置によって実現されることが多かった。しかし、現在、暗号は汎用コンピュータやマイクロプロセッサーにプログラムを実行させることによって実現させることが多いため、暗号を「装置」としてよりもむしろ「数値変換の方法」として保護することが必要となっている。そこで、本稿では、「装置から独立した数値変換の方法」という意味で「暗号」という言葉を用いることとする。

[9] このような考え方に立つ学説として、Chisum, D. S., “The Patentability of Algorithms,” Vol. 47, Pittsburgh Law Review, pp.959（1986）がある。

[10] もっとも、「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」の本文を実例 3*と整合的に解釈すると、コンピュータを用いて処理することのみであるとされる場合は極めて限定されることになる。実例 3 に示されているクレーム 2 の記載内容は、コンピュータを用いて処理することのみであるとされる範囲が極めて限定的であることを示している。

*実例３は、発明の名称が「コンピュータにより自然数 n から n+k までの和を求める装置」とい

コンピュータ・ソフトウェアが「発明」か否かという議論はほとんど不毛の議論であると言ってもよいであろう。「自然法則を利用した技術的思想」という定義は 19 世紀のドイツの考え方によるもの[11]であり、19 世紀の機械技術時代の概念をそのまま 21 世紀を目の前にしたコンピュータ時代に持ち込むことには大きな疑問がある。悪戯に 19 世紀の概念に拘泥することが法の妥当な解釈とは思われない。現実問題として、コンピュータ・ソフトウェアの技術的な性格は否定しがたく、コンピュータ・ソフトウェアに関連する数多くの特許が付与されている。これを無視して、特許法の形式的な解釈論を根拠に「自然法則を利用した」ものでないから無効であるとするのは、コンピュータ・ソフトウェアの技術的性格を無視し、特許を巡る無用な混乱を招き、コンピュータ・ソフトウェアの研究開発のインセンティヴを削ぐことになろう。

運用指針は、暗号を対象とする特許のクレームが数値変換の方法そのもののみである場合には、特許法の規定の文言を尊重して保護の対象とは認められないが、暗号を利用した装置などのクレームを有する場合には、それは「装置」であるとして特許の保護の対象となるという考え方を取り、実質的に数値変換の方法を特許の保護の対象とすることを認めていると理解することができる。これは、特許法の保護の対象がクレームによって特定されるところに着目し、特許法の保護の対象となるか否かという問題を、実質的にクレームの記載方法の問題としていると言うことができよう[12]。

う特許出願の審査に関する例である。この実例 3 では、発明と認められないクレーム 1 と、発明と認められるクレーム 2 が例示されている。クレーム 1 と 2 は次の通りである。

クレーム 1：

「自然数 n と n+k を入力する手段と、自然数 n から n+k までの和 s を、s=(k+1)(2n+k)/2 により求める演算手段と、演算結果を出力する手段とを備えたことを特徴とする、コンピュータにより自然数 n から n+k までの和を求める装置」

クレーム 2：

「自然数 n と n+k を入力する手段と、入力された n を記憶する n 記憶手段と、入力された n+k を記憶する n+k 記憶手段と、n 記憶手段から n を、n+k 記憶手段から n+k を取得し k を演算する手段と、該 k を記憶する k 記憶手段と、自然数 n から n+k までの和 s を上記 n 記憶手段、k 記憶手段に記憶された n、k を用いて s = (k+1)(2n+k)/2 により求める演算手段と、演算結果を出力する手段とを備えたことを特徴とする、コンピュータにより自然数 n から n+k までの和を求める装置」

特許庁は、クレーム 1 を発明と認めない理由として、『「自然数 n から n+k までの和 s を、s=(k+1)(2n+k)/2 により求める」ことは、数学上の公式のみを利用したことであるから、自然法則を利用した解決手段は、「コンピュータのハードウェア資源を用いた演算処理」のみである。そして、コンピュータのハードウェア資源がどのように（how to）用いられて演算処理されるかを直接的又は間接的に示す具体的な事項が記載されていないから、...「発明」に該当しない』と説明している。一方、クレーム 2 に対しては、『「n 記憶手段から n を、n+k 記憶手段から n+k を取得し k を演算する手段と、該 k を記憶する k 記憶手段と、自然数 n から n+k までの和 s を上記 n 記憶手段、k 記憶手段に記憶された n、k を用いて、s = (k+1)(2n+k)/2 により求める演算手段」は、コンピュータのハードウェア資源がどのように（how to）用いられて上記処理がされるかを直接的に示す具体的な事項であるから、...「発明」に該当する』と説明している。

[11] 注 1 の中山（1993）の 102 頁を参照。

[12] 運用指針実例 3 は、特許庁は、特許の保護対象か否かを決定する場合、コンピュータのハードウェア資源の利用形態に関する記載方法を判断基準としていることを示している。この特許庁の取扱は、特許法の改正が無い限り、特許法の「発明」の定義に配慮せざるを得ない一方、コンピュー

20 世紀も終わろうとしている現在、19 世紀の発明概念を 20 世紀後半にも当てはめるという時代錯誤的な考え方を基礎として[13]、現代技術を律しようとしているところに矛盾があるとみるべきであろう。現代技術を前提とし、過去の呪縛から逃れて「発明」の規定の解釈を考える必要があるのではなかろうか。暗号の技術的性格を正面から認めて、暗号は特許の保護の対象である「発明」に該当すると考えることが明快でよいと思われる。現代の暗号は、コンピュータという装置を利用しないかぎり実現できないため、「自然法則を利用した」コンピュータを用いる技術として特許の対象とするという解釈をとればよいと思われる[14,15,16]。

アメリカ合衆国における状況は複雑になっている。アメリカ合衆国の特許法は、特許の保護の対象が「方法（process）、機械（machine）、製品（manufacture）、もしくは、組成物（composition of matter）」であると規定している（特許法第 101 条）。特許法の保護の対象は、この規定の解釈として判例によって決定されている[17]。コンピュータ・ソフトウェアが特許法の保護の対象となるか否かについても、判例によって決定されている。特許商標庁は、1960 年代にコンピュータ・ソフトウェアの特許法による保護について否定的な見解を示したが、関税特許控訴裁判所は、1968 年に In re Prater 事件の判決[18]においてコンピュータ・ソフトウェアを特許法の保護対象とすることに積極的な姿勢を示し、その後も、1969 年の In re Bernhart 事件の判決[19]、1970 年の In re Musgrave 事件の判決[20]、1971 年の In re Benson 事件の判決などで、その考え方を維持した。一方、最高裁判所は、1972 年の Gottschalk v. Benson 事件の判決[21]、1978 年の Parker v. Flook 事件の判決[22]において、コンピュータ・ソフトウェアを特許法で保

タ・ソフトウェア技術はどんどん進んでいくという状況の中で、やむを得ない選択であったかもしれない。

13 特許法の改正で「発明」の定義を規定したことが問題であるとする指摘もある。

14 このことは、あらゆるものが「発明」になるとしているのではない。「自然法則の利用」によって特許の対象を限定することをコンピュータ・ソフトウェアに当てはめることが不適当であるということを意味するに過ぎない。特許の対象になるか否かは特許法によってインセンティヴを与えられるべき「技術的思想」であるか否かという判断によってなされるべきものである。

15 特許庁の「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」の『解決手段が...「コンピュータを用いて処理することのみ」である場合...には、「発明」とはしない』というガイドラインは変更されるべきであり、「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」の実例 3 に記載されている 2 つのクレームの無意味な区別はやめるべきであると思われる。

16 暗号は通信に関連する技術であるから、他の通信装置あるいは通信方法と同様に「自然法則を利用した技術」であるとしても、暗号が特許によって保護されることになる。

17 アメリカでは、特許の保護の対象が判例によって決定される一方、日本では、特許の保護の対象は実質的に特許庁によって決定されている。

18 415 F.2d 1378（1968）＜米国高等裁判所判例集第 2 版第 415 巻 1378 頁、1968 年＞

19 417 F.2d 1395（1969）＜米国高等裁判所判例集第 2 版第 417 巻 1395 頁、1969 年＞

20 431 F.2d 882（1970）＜米国高等裁判所判例集第 2 版第 431 巻 882 頁、1970 年＞

21 409 U.S. 64（1972）＜米国最高裁判所判例集第 409 巻 64 頁、1972 年＞

22 435 U.S. 584（1978）＜米国最高裁判所判例集第 435 巻 584 頁、1978 年＞

護することに消極的姿勢を示した。しかし、最高裁判所は、1980 年に Diamond v. Chacrabarty 事件の判決[23]で、「太陽の下、人によって創られたすべてのもの（anything under the sun made by man）」が特許の対象となるとして、特許に対する消極的な態度を変更し、1982 年の Diamond v. Diehr 事件の判決[24]で、コンピュータ・ソフトウェアを特許法で保護する方向へとその姿勢を変更した。1990 年代になって、特許商標庁は、再びコンピュータ・ソフトウェアの特許法による保護にやや消極的な姿勢へと変化した[25]が、連邦巡回控訴裁判所は、1994 年の In re Donaldson 事件の判決[26]、In re Alappat 事件の判決[27]、In re Lowry 事件の判決[28]などで、コンピュータ・ソフトウェアの特許法による保護について積極的な姿勢を示した。

特許商標庁は、1996 年 2 月 28 日にコンピュータ関連発明の審査ガイドライン（Examination Guidelines for Computer-Related Inventions Final Version）を発表し、連邦巡回控訴裁判所の裁判例を踏まえて、コンピュータ・ソフトウェアの特許法による保護について積極的な姿勢を示した。具体的には、抽象的な概念（abstract idea）、自然法則（law of nature）、自然現象（natural phenomenon）が、判例によって特許法の保護の対象（subject matter）とならないとされているとした上で、特許の対象は技術（technological art）であり、実用的な応用（practical application）を有しなければならないとしている。そして、特許の対象として認められるか否かは、特許の対象を特定するクレームの記載内容によって決定されるべきものであるとしている[29]。アメリカ合衆国でも、クレームが特許法の保護の対象である発明を特定する側面を有するところから、特許法の保護の対象となるか否かという問題を、実質的にクレームの記載方法の問題としている。

現代の暗号はコンピュータ・ソフトウェアによって実現されることから、コンピュータ・ソフトウェアと同様の論争に巻き込まれても不思議はない。しかし、コンピュータ・ソフトウェアと特許の保護の対象に関する裁判例には暗号に関する事例は見られないほか、実際に多くの暗号に関連する特許が認められている。暗号が数値変換の方法であることに着目すれば、数値変換の方法の特許を否定した Gottschalk v. Benson 事件の最高裁判所の判決との関係が問題となるとも思われるが、Diamond v. Chacrabarty 事件、Diamond v. Diehr 事件で判例

---

[23] 206 U.S.P.Q. 193（1980）＜U. S. Patent Quarterly 第 209 巻 193 頁、1980 年＞

[24] 450 U.S. 175（1981）＜米国最高裁判所判例集第 450 巻 175 頁、1981 年＞

[25] USPTO, Patentable Subject Matter Mathematical Algorithms and Computer Programs（1989）

[26] 29 U.S.P.Q 2d 1845（1994）＜U. S. Patent Quarterly 第 2 版第 29 巻 1845 頁、1994 年＞

[27] 31 U.S.P.Q 2d 1545（1994）＜U. S. Patent Quarterly 第 2 版第 31 巻 1545 頁、1994 年＞

[28] 32 U.S.P.Q 2d 1031（1994）＜U. S. Patent Quarterly 第 2 版第 32 巻 1031 頁、1994 年＞

[29] アメリカ合衆国特許商標庁のガイドラインが、数値変換の方法を特許として認めるか否かの問題をクレームの問題とした背景には、数値変換の方法を巡るいくつかの判例が特許の対象に関する時代錯誤的な考え方から影響を受けているという事情がある。特許商標庁は、こうした判例も考慮しつつガイドラインを作成せざるを得ない。このような事情を考えると、このガイドラインの内容はやむを得ないと言うべきであろう。

が変更されたと考えると、抵触しないとも考えられる[30]。また、通信方法あるいは通信機器は特許の保護の対象として認められてきたことから、暗号も通信方法として認められているとみることもできる[31]。なお、特許商標庁の 1996 年のガイドラインによれば、暗号もクレームの記載の方法によって特許の対象となるか否かが決定されることになろう。

2．クレーム

特許法の保護の対象は抽象的な技術思想であり、これを特定することは困難であるため、この技術思想を具現化する「物」あるいは「方法」によって保護の対象を特定している。この「物」あるいは「方法」という記載によって保護の対象を特定するのがクレームである[32]。そして、このクレームを基準として、特許の保護の範囲の判断や審査が行われることになる。

日本の特許庁がコンピュータ・ソフトウェアの発明が保護の対象になるかという問題をクレームの問題としているところから、いくつかの問題が生じている。日本の特許庁は、コンピュータ・ソフトウェアがハードウェアと関連付けられている場合には、特許法の保護の対象として従前から認められている機械と同様に捉えることができるとして、装置のクレームとする特許は認めてきた[33,34]。

暗号も装置クレームの場合には、特許の対象になるとすることに大きな議論はなされなかった[35]。もっとも、このようなクレームには問題がないわけではない。装置クレームでは、その暗号の本質である数値変換の方法を特許の対象として特定するものではなく、数値変換の方法を特定しなくても認められる余地がある。装置構成を実現する数値変換の方法がなけれ

---

[30] Gottschalk v. Benson 事件の最高裁判所の判決の位置付けによるものと思われる。この最高裁判所の判決がその後の最高裁判所の判決で変更されたと考えると、最高裁判所の判決と特許商標庁の実務は矛盾しないが、変更されていないと考えると特許商標庁のガイドラインと整合しているとは言い難い。なお、その後の判例でも、数値変換方法が特許の保護の対象となることを認めた判例はないので、暗号が保護の対象となることが判例上明確となっているわけではない。

[31] 通信方法が特許の対象となるとした最高裁判所の判決例として、Dolbear v. American Bell Telephone Company, 126 U. S. 1（1888）がある。

[32] 「物」あるいは「方法」として特定する論理的な必然性はないが、日本の特許法は「物」あるいは「方法」として特定すべきものとしている。

[33] このような特許庁の実務は、1975 年の「コンピュータ・プログラムに関する発明についての審査基準（その 1）」、1982 年の「マイクロコンピュータ応用技術に関する発明についての運用指針」、1988 年の「コンピュータ・ソフトウェア関連発明の審査上の取扱（案）」、1993 年の「審査基準（改訂版）」などにより築き上げられてきた。

[34]現在の特許庁のガイドラインでは、暗号は「ハードウェア資源を用いて処理すること」によって特許の保護の対象となると判断できるものの、「特定技術分野の審査の運用指針　第 1 章　コンピュータ・ソフトウェア関連発明」の実例 3 では、無意味なハードウェアとの関連付けがなされているかどうかによって保護対象となるか否かを決定する内容となっており、このことは矛盾を示していると思われる。

[35] そもそも、現代の暗号と特許についての議論はほとんどなされていないが、方法のクレームの場合には、コンピュータ・ソフトウェアに関する方法のクレームと同様の議論があり得た。

ば、装置構成は特許の保護の対象とされる技術思想を指すものではないとも考えられるため、そうしたクレームを認めることには、問題がないとは言えない。暗号の本質である数値変換の方法をクレームすることの方が明確であると思われる。ただ、このような装置構成のクレームは、数値変換の方法ではなく装置構成によって発明を特定させようとする特許庁の行政指導に基づくものであり、コンピュータ・ソフトウェアの特許法による保護のこれまでの経緯からすれば、これを無効とすべきではないであろう[36]。

アメリカ合衆国では、Gottschalk v. Benson 事件の最高裁判所の判決などが、数値変換の方法をクレームすることを特許の保護の対象として認めなかったために、一般的に、コンピュータ・ソフトウェアについては、数値変換の方法を回避する書き方となっているクレームが多くなっていると思われる[37]。もっとも、暗号については、実際の運用では数値変換方法に近いものが認められている。これも、通信の方法として認められていると考えると、これまでの判例と矛盾しないとも考えられる[38]。

3．保護を受ける要件

（1）新規性

発明が特許による保護を受けるためには、新規性を有していなければならない。新規性は出願時の技術水準によって決定される。新規でない技術は社会に対して技術的貢献をもたらさないので、特許法によって保護する理由がないからである。

日本の特許法では、出願時の技術水準は先行する国内特許出願[39]、出願時に発行されている特許明細書[40]、雑誌などの刊行物[41]に記載されている技術情報その他の情報[42]で構成される。刊行物については、国の内外を問わず先行技術となるとしているが、刊行物以外の情報については、国内における情報についてのみ先行技術となるとしている。既に公開されている外国の特許出願、外国の論文なども先行技術を構成する。そこで、インターネット上の情報が先行技術を構成するか否かという問題がある。インターネット上の情報は「刊行物」に当たるか、インターネット上の情報は「公知」といえるか、という問題である。インターネット

---

36 もっとも、日本の特許庁における実際の運用では、数値変換の方法そのものをクレームした特許も認められているので、装置との関連付けがどの程度必要であったかは必ずしも明確ではない。

37 ただし、特許商標庁の審査は日本の特許庁の審査よりもバラツキがあり、必ずしも一般論ではすべての特許例を説明できるものではないと思われる。

38 もっとも、Gottschalk v. Benson 事件の最高裁判所の判決については、判例が変更されたと見るべきであるとする指摘もある。例えば、Chisum（1986）がある。

39 特許協力条約に基づいてなされた国際出願で指定国として含まれている場合も国内出願と同様に含まれる。

40 外国の特許の明細書も含まれる。閲覧謄写が可能であれば、文書で発行されていなくてもよい。

41 特許法では「頒布された刊行物」とされている（特許法第 29 条第 1 項第 3 号）。

42 その他の情報としては、製品として販売されている場合、使用されている場合などがある。特許法では「公然知られた」（公知）、「公然実施をされた」（公用）とされている（特許法第 29 条第 1 項第 1 号、第 2 号）。

上の情報であることを理由に公知であると判断しても、インターネット上の情報が刊行物にあたると判断しても、新規性を阻害することには違いがなく、先行技術の判断の上では異ならない[43]。問題は、「公知」であるとされると、特許法第 30 条の新規性の例外が適用されないことにある[44]。

暗号については、暗号に関する特許出願、暗号に関する論文発表などが先行技術となる。暗号が実際に利用されている場合は「公用」として先行技術を構成する。暗号を含んだ製品が販売されたり、暗号がネットワーク上で利用されたりしている場合に「公用」として先行技術を構成すると考えられる。外国で販売されている製品でも国内で入手可能であるかぎりは、国内「公用」であるとして先行技術に含まれるものと考えられる[45]。もっとも、コンピュータ・ソフトウェアが「刊行物」であると考えられるとすれば、国内における入手可能性は問われないこととなる[46]。

先行技術に同じ技術が含まれていれば、出願は新規性を欠くとして特許が認められないことになる。特許出願された技術が先行技術に含まれているか否かは、その特許出願のクレームを基準として判断されることになる。例えば、公開鍵暗号の具体的な実現方法ではなく原理に関する論文が先行技術として存在する場合、新しい技術によって公開鍵暗号の実現方法を開発した者による特許出願は、公開鍵暗号の原理そのものをクレームとすることには新規性がないが、公開鍵暗号の実現方法のクレームには新規性があることになる。

先行技術には、発明者（あるいは出願人）自身が開示した情報も含まれる。日本の特許法は、発明者（あるいは出願人）自身が開示した情報については、「刊行物に発表」した場合、「特許庁長官が指定する学術団体が開催する研究集会において文書をもって発表」した場合などは、6 ヶ月以内に出願すれば、その時から先行技術を構成しない扱いとしている（特許法第 30 条第 1 項）。そこで、インターネット上の論文が「刊行物」にあたるかということが問題となる。「刊行物」にあたる場合、インターネット上の発表も新規性の喪失の例外となるからである[47]。

---

43 日本の特許法上、「公知」は国内に限られているが、インターネット上の情報は世界中で利用可能であるから、国の内外は問題とならないであろう。

44 特許法の解釈上「公知」であるとみなすためには、「誰かが実際にその情報を入手していることが必要か」、それとも「その情報を入手しうる可能性があればよいか」について議論がある。「刊行物」は実際に誰かが入手したことは必要でないとされているので、この解釈上の議論を避けるためには「刊行物」にあたるとしたほうが解釈上よいかもしれない。

45 ネットワークにおける暗号の利用は国の内外についての違いがないので、ネットワーク上での利用について、国の内外は問われるべきものではないであろう。

46 コンピュータ・ソフトウェアを「刊行物」と考えるには、コンピュータ・ソフトウェアの可読可能性が問題となるであろう。

47 なお、最初の発表が新規性喪失の例外の適用を受ける場合、同一発明についての発表があった場合には、新規性喪失の例外の適用を受けないとする考え方もある。しかし、一旦公表された情報は発明者（あるいは出願人）からすればどのような取り扱いをされるかは分からないほか、発明者（あるいは出願人）が複数の方法で公表することは現在の学会では通常行われていることから、最初の発表が新規性喪失の例外の適用を受ける場合には、6 ヶ月以内の出願であれば新規性を喪失しない

アメリカ合衆国でも、発明が特許による保護を受けるためには、その発明が新規性を有していなければならない。新規性は、特許出願から 1 年以内の発明時の技術水準によって決定される。発明時の技術水準は、先行する国内特許[48]、発明時に発行されている特許明細書[49]、雑誌などの刊行物に記載されている技術情報やその他の情報[50]で構成される。アメリカ合衆国でも、刊行物については国の内外を問わず先行技術となるとしているが、刊行物以外の情報については国内における情報のみ先行技術となるとしている。なお、発明の新規性は出願から 1 年以内の発明日を基準とすることから、発明者による公表から 1 年以内に出願すれば新規性を喪失することはない。暗号に関する特許でも、論文発表後 1 年以内に出願すれば新規性を欠くものとはされない。この判断は、コンピュータ・ソフトウェアについてなされ、非機能的で記述的なもの（non-functional descriptive material）については、判断の資料とされない。

特許法第 112 条第 6 項には、機能によって特定されている手段（means）によって記載されたクレーム（means plus functional claim）の解釈に関する規定があり、その手段はクレームに記載されている機能を有するあらゆる手段を含むものではなく、明細書に記載された手段とその均等物に限るとされている。means plus function claim の新規性を判断する場合については、当該手段（means）を明細書に記載されている手段とその均等物に限定して判断することになる[51]。

（2）進歩性（非自明性）

日本の特許法では、発明が特許による保護を受けるためには、その発明が進歩性を有していなければならない。先行技術と同じ技術は新規性を欠くものとして特許を受けることができないが、先行技術には含まれていないが先行技術から容易に発明できる技術は、進歩性を欠くものとして特許が与えられない。進歩性は出願時の技術水準を基準として判断される。進歩性の判断の基準とされるのは、出願時に発行されている特許明細書、雑誌などの刊行物に記載されている技術情報その他の技術を含む先行技術である[52]。

進歩性は技術的な革新の度合いを問うものであり、その判断は難しい。コンピュータ・ソフトウェアに関しては、それまでに行われていた処理をコンピュータによって処理すること

ものとすべきである。暗号についても、特許出願時に、その暗号に関する記述が論文として発表されている場合などは新規性を阻害することになる。自分が発表した論文に記載されている場合でも、新規性喪失の例外として認められない限りは、新規性を欠くとして特許は認められないことになる。

[48] アメリカ合衆国の場合には、出願公開制度が無いので、先行技術となるのは特許に限られる。

[49] 外国の特許の明細書も含まれる。閲覧謄写が可能であれば、文書で発行されていなくてもよい。

[50] その他の情報としては、製品として販売されている場合、使用されている場合などがある。

[51]注 19 の In re Donaldson を参照。

[52] 進歩性を判断するにあたっての先行技術には、先行する特許出願でも公開されていない出願は含まれない。新規性と進歩性の判断の基準となる先行技術は異なることになる。したがって、先行する特許出願から極めて容易に発明される技術については、これを新規性の問題とするか進歩性の問題とするかによって、結論を異にすることになる。

とした場合に進歩性があるかという問題がある。暗号については、先行技術として存在する暗号からの進歩の度合いというものが進歩性の判断の材料となる。安全性の向上、処理時間の短縮などが進歩性を示すことになろう。また、暗号は、先行技術として存在する数学的問題の応用という性格を持つことから、そのような応用が進歩性を有するか否かについても判断しなければならない。

アメリカ合衆国では、日本の進歩性にあたるものとして、非自明性（non-obviousness）の要件が課されている。非自明性は、出願時から 1 年以内の発明時の技術水準によって決定される。その技術水準は発明時に発行されている特許明細書、雑誌などの刊行物に記載されている先行技術で構成される。アメリカ合衆国における非自明性についての基本的な考え方は日本の進歩性についての基本的な考え方と異なるところはないが、具体的な判断については、日本の特許庁の判断とアメリカ合衆国の裁判所あるいは特許商標庁の判断とが食い違うことが少なくない。

（3）開示

特許法は、その発明が公開されることを求めている。この発明の公開の制度は、発明が秘密にされることを防ぎ、広く社会でその技術を利用できるようにしたものである。技術が単純でその模倣が容易な場合は、技術を公開する制度はあまり大きな意味を持たないかもしれないが、技術が複雑になり模倣が容易でない場合は、技術を公開する制度を確立しないと技術が十分に利用できないようになる。この制度を確立するために、明細書による発明の開示ということが行なわれており、重複した技術開発投資を避け、改良技術の開発を容易にすることができる。そのため、特許の明細書は、出願時の技術状況の下でその特許を実施することができるように記載されていなければならない。出願時の技術常識に含まれるものについては明細書で開示される必要はない。開示が十分になされていない場合には特許は与えられないし、たとえ与えられたとしても特許は無効となる。特許明細書中の開示と出願時の技術常識によって実現できないような特許出願も、開示が不十分なものとして特許出願は拒絶されるし、たとえ特許になったとしても無効原因を有することになる。

暗号の場合には、暗号に関連する製品が販売され、あるいはネットワーク上で利用されているとしても、その数値変換の方法が簡単に解るものではないから、暗号の技術的内容に関する開示は重要となる。暗号の特許のクレームが具体的な数値変換の方法を含んでいる場合でも、その数値変換の方法を実装する方法が出願時の技術常識に含まれていないときには、明細書中に実装方法が記載されていなければならない。装置のクレームが具体的な数値変換の方法を含んでいない場合には、数値変換の方法に関する記述が明細書に含まれていなければならない。

アメリカ合衆国の特許法においても、開示についての基本的な考え方は日本と異ならない。特許の明細書は、出願時の技術状況の下で、その特許を実施することができるように記載されていなければならない。ただし、アメリカ合衆国では出願人には信義誠実の義務があり、明細書における開示には、最良実施態様（best mode）の記載が求められている。

4．特許の効力と保護の範囲

（1）特許の効力

特許の効力は、特許の対象とされた発明を排他的に実施することであり、第三者による発明の実施を排除することにある。第三者が特許権者の許諾を得ることなく発明を実施することは特許権の侵害とされ、権利者に侵害行為の差止、損害賠償などの救済が与えられる。

日本の特許法では、発明の実施は、（1）特許の対象となっている物の生産、使用、譲渡、貸渡、譲渡または貸渡のための展示、輸入、（2）特許の対象となっている方法の使用、（3）その方法により生産された物の使用、譲渡、貸渡、譲渡または貸渡のための展示、輸入に及ぶとされている（直接侵害、第 68 条、第 2 条第 3 項）。さらに、（4）第三者が、特許の保護の対象となっている物の生産にのみ使用される物、あるいは、特許の保護の対象となっている方法にのみ使用される物を製造、販売することにも特許の効力が及ぶとされている（間接侵害、第 101 条）。特許の侵害は、特許のクレームに記載されている物あるいは方法を実施することによって成立するので、クレームの記載事項の一部を欠く実施は、原則として特許の侵害（直接侵害）とはならない。クレームの記載事項の一部を欠く実施でも、侵害行為にのみ使用されるものを製造、販売する場合には、特許の侵害（間接侵害）とされる[53]。日本の特許法の間接侵害に関する規定は、侵害行為の一部を実行する者について、その主観的意図に関わらず侵害行為にのみ使用される物を生産する場合に限って侵害となると規定している。

特許発明の実施が共同でなされた場合が問題となる。日本の特許法の間接侵害に関する規定は、第三者が特許の侵害行為のために利用することを知って部品などを供給した場合でも、その部品が汎用品であれば特許侵害にはならないと規定している。それにもかかわらず、複数人がクレームの記載事項を分担することによって特許の発明が実施されている場合に、これを特許の共同侵害とすることができるであろうか。そもそも、侵害行為が複数人の共同行為によって実施されるシステムのクレームも念頭において、特許法が実施を規定していると考えることは難しいであろう。そこで、複数人によって実現されることを前提として実現されるシステムのクレームについては、複数人による共同の実施を認めるという考え方があるかもしれない。ただ、この考え方によれば、クレームの記載事項の一部を欠く実施で間接侵害に当たらない場合でも特許の侵害となることを認めるものであり、解釈上可能かという疑問がある。あるいはまた、複数人による行為を中心的な者の行為とみなして特許の侵害とする考え方があるかもしれない。しかし、この考え方も、第三者の行為を自己の行為とすることが解釈上可能かという疑問がある。

アメリカ合衆国の特許法は、特許の効力は、特許の対象とされた物を生産すること、使用すること、販売することに及ぶものとされている（第 271 条（a））。方法の特許の効力は、その方法を使用すること、その方法により生産された物を使用すること、販売すること、輸入す

---

[53] クレームの記載事項の一部を欠く実施が、クレームの解釈によって特許の侵害とされる場合については後述。

ることに及ぶとされている（第 271 条（a）（g））ほか、特許の効力は、特許の侵害を誘導することにも及ぶとされている（誘導侵害（Inducing infringement）、第 271 条（b））。さらに、特許の対象である発明の本質的な部分であり、特許された物あるいは特許された方法を実施するために使用される物を、特許を侵害しないで使用されることが適当でないことを知ってする販売に対しても、特許の効力が及ぶとされている（寄与侵害（Contributory Infringement）、第 271 条（c））。そして、アメリカ合衆国の特許法では、複数人の共同行為による特許の侵害を認めている[54]。暗号の特許のクレームが通信の送信者と受信者の両者によって実現されるという内容であり、その特許の対象となっている暗号を用いて通信がなされた場合には、同一の社内における暗号通信がなされた場合でなくても、共同行為による特許の侵害あるいは誘導侵害にあたると思われる。

（2）保護の範囲

日本の特許法では、特許の範囲はクレーム（請求の範囲）の記載に基づいて定めなければならないと規定されている（第 70 条）。特許の保護の範囲はクレームを基準として判断されるため、特許庁がどのような範囲のクレームを認めるのかが重要となる[55]。したがって、保護の範囲を考えるに際に、どのような範囲のクレームが認められているかということを切り離して考えることは、現実的な発明の十分な保護という観点からは問題がある（もちろん、クレームをあまりに広く解釈することは第三者の利益を害するおそれがあり、この点も考慮にいれなければならない）。

先行する特許に広いクレームがある場合、その技術を利用するためにはその特許の特許権者の許諾が必要となる。先行する特許の権利の範囲がどこまでかということは、クレームを基準として判断されることになる[56]。もっとも、クレームには解釈の余地があり、その解釈によって特許の権利の範囲が異なる。暗号に関連する特許のクレームでも、その文言に解釈の余地がある場合もある。

特許の保護の範囲はクレームの文言の範囲を越えないことが原則であるが、**クレームの文言には含まれていないけれども、実質的には同一の作用効果を生じる技術を保護の範囲に含めるための理論（均等論、**Doctrine of Equivalent**）**もある。均等論は、日本の学説あるいは判例でも認められている理論ではある[57]が、特許の権利の範囲を狭く解釈すべきであるという傾向が強かったため、アメリカ合衆国やドイツに比べると均等論の適用は限定的になっている。均

---

54 イギリスでも、共同不法行為として、共同行為による侵害が認められている（TERRELL ON THE LAW OF PATENTS, 14ed. pp. 198 （1994））。

55 特許庁の決定に対しては、裁判所で争うことは可能であるが、係争によって失われる時間と費用を勘案すると、現実の特許庁の実務は無視しえない。

56 日本の特許法では、クレームが手段の組み合わせで記載されている場合にも、その手段の解釈に関する規定はないが、アメリカ合衆国特許法第 112 条第 6 項における means plus functional claim に関する解釈（後述）を利用すべきとの議論もある。

57 最高裁判所平成 10 年 2 月 24 日判決（平成 6 年（オ）第 1083 号）

等論が適用される要件として、日本の通説では、①均等とされる物あるいは方法（以下、均等物）がクレームによって特定された発明の技術思想に含まれること（置換可能性）、②その均等物が第三者にとって予測可能であること（容易想到性）、が必要であるとされている。また、実際に均等論が適用されるか否かをが判断される場合には、上記の 2 つの要件に加え、③均等論が認められることによって利益を受ける特許権者が均等物を含むクレームを記載することができなかった事情や、④均等論の適用によって不利益を受ける者が均等を予測することができた事情も考慮される[58]。暗号の特許のクレームに記載されている数値変換の方法とは異なる方法を用いた場合に特許の効力が及ぶか否かという問題を検討する際に、この理論が適用される余地があると考えられる。

アメリカ合衆国でも、特許の範囲はクレームの記載に基づいて定めなければならないとされている。ただし、機能によって特定されている手段（means）によって記載されたクレーム（means plus functional claim）については、その手段はクレームに記載されている機能を有するあらゆる手段を含むものではなく、明細書に記載された手段とその均等物に限るとされている。しかし、最近では、この means plus functional claim の解釈に関して次のような判例が出ている。まず、1996 年に米国巡回控訴裁判所は、Cole v. Kimberly-Clark Corp.事件において、「means plus functional claim 形式の記載であっても、構造上の限定が記載されておりその手段の構造が明確になっている場合には、第 112 条第 6 項の適用を受けない場合もある」との見解を示している。このように、means plus functional claim の形式で書かれたクレームの内容が必ずしも明細書に記載されている手段とその均等物に限定されるとは限らないとの見方も存在する。ただし、これらはいずれもコンピュータ・ソフトウェア以外の特許のクレームに関する判例であり、暗号の特許に関しても同様に解釈することが妥当かどうかについては明確になっていない。

均等論は、アメリカ合衆国では日本に比べて広く認められている。米国の場合には、均等論の適用にあたって、均等物が機能（function）、方法（way）、結果（result）において特許の対象と実質的に同一であるかどうかを基に決定される。means plus functional claim についても、均等論が適用されてー旦制限されたクレームについても均等論を認める余地がある。

（3）特許の効力の制限－強制実施権

日本の特許法は、特許権者が特許の対象となっている発明を独占的に利用することを認めつつ、その独占の弊害に対処するために、公共の利益[59]、不実施、利用関係を理由とする裁定実施権（強制実施権）の制度を置いている（第 93 条、第 83 条、第 92 条）。裁定実施権は、特

---

58 相澤英孝「特許制度の在り方－バイオテクノロジーに関する発明への均等論の適用」『星野英一先生古稀祝賀論　日本民法学の形成と課題（下）』（1996）1351 頁を参照。また、最高裁の判例（平成 6 年第 1083 号平成 10 年 2 月 24 日第三小法廷判決）においても、均等論の適用要件としてこの解釈が採用されている。

59 公共の利益を理由とする強制実施権としては、「国民の生命、財産の保全...」などが挙げられている（裁定制度の運用要領）。

許権者の意思に反しても、第三者に特許の対象となっている発明を実施させることを、通産大臣あるいは特許庁長官の裁量による実施権の設定という形式で認める制度である[60]。強制実施権については、強制実施権の濫用による弊害が指摘され[61]、パリ条約、TRIPS 協定などで強制実施権の設定についての条件が規定されている。パリ条約第 5 条 A は強制実施権の設定に関する条件を規定し[62]、TRIPS 協定第 31 条は強制実施権を設定できるより厳しい条件を課している[63]。また、日本は、利用関係を理由とする強制実施権については、アメリカ合衆国との合意によって反競争的行為の是正を目的とする場合以外は設定しないとしている。

特定の通信事業者が特許によって特定の暗号に関する技術を独占したとしても、そのことによって直ちに強制実施権の設定が可能となるものではない。暗号について、軍事目的などに利用される場合、その利用が TRIPS 協定第 31 条の国家的緊急事態（national emergency）への対処にあたるときは、公共の利益を理由として裁定実施権を設定することができるであろう[64]。通信網の建設を理由として、暗号の特許に裁定実施権を設定することが認められるかについては、特許権者から許諾を得る合理的な努力を払った後でなければならず（TRIPS 協定第 31 条（b））、当該特許を利用することが単に経済的に好ましいというだけでは十分ではないと思われる[65,66]。ただし、暗号特許によってネットワークの独占などが生じた場合には、反競争的行為を排除する手段として強制実施権を設定することができると解釈するべきである。

アメリカ合衆国の特許法には強制実施権の制度はないが、反競争的行為を排除する目的で、反トラスト法上の救済として強制実施権を設定することができるとされている。

---

60 ただし、これらの強制実施権が設定された例はない。

61 19 世紀末の特許法の国際的調和に関する国際的な議論でも、強制実施権についての議論がなされている。

62 パリ条約の強制実施権の設定の制限に関する規定については、ボーデンハウゼン『注解　パリ条約』。

63 （1）個別的利益への配慮、（2）特許権者に対して合理的な条件による実施許諾を得る努力がなされ、そのような努力が合理的な期間内に奏効しなかった後に強制実施権が設定されること、（3）強制実施権の範囲と期間はその目的によって限定されること、（4）強制実施権は非排他的であること、（5）強制実施権は、営業と共にする場合以外は、譲度できないこと、（6）強制実施権はその実施権を認めた締約国の国内市場への供給に対するものであること、（7）強制実施権を認めた事情が消滅した場合には、終了さるべきこと、（8）特許権者に十分な実施料が支払われるべきこと、（9）強制実施権の設定及び実施料に関する決定は、司法的再審査に服すべきこと、（10）他の特許（第 2 の特許）が実施可能とされるために強制実施権が設定される場合には、（a）第 2 の特許が強制実施権が設定される特許（第 1 の特許）に対して経済的重要性のある重要な技術的発展を含むこと、（b）第 1 の特許の特許権者が第 2 の特許を実施するための合理的な条件によるクロスライセンスが与えられること、（c）第 1 の特許の強制実施権は第 2 の特許と共にするのでなければ譲度できないこと、を付加すること、が守られなければならないとしている（第 31 条）。

64 特許されている暗号が軍事的に使用できるものであるかどうかは別の問題である。

65 注 1 の中山（1993）の 402 頁を参照。

66 また、アメリカ合衆国国民が特許権者である場合には、反競争的行為に当たらない限り、強制実施権を設定することはできないであろう。

Ⅲ 暗号とその展開

1．公開鍵暗号の原理

公開鍵暗号は、暗号化に用いられる鍵と復号化に用いられる鍵が異なる暗号方式であり、暗号化に用いられる鍵を「公開鍵」として公開する一方、復号化に用いられる鍵は「秘密鍵」として秘密にしておくことによって、予め通信相手と秘密に鍵を共有することなく、暗号通信を行うことを可能にする技術である。

公開鍵暗号は、パラメーターk を持つ特殊な関数（変換）$f_k(x)$を利用した暗号と解釈できる。ここで、特殊であるとは、①パラメーターE に対して逆関数（逆変換）$f_E^{-1}(x) = f_D(x)$を満たすパラメーターD が一意に存在するが、E から D を求めることが計算量的に困難[67]である、②パラメーターE と関数 $f_E(x)$がわかっていても、パラメーターD を知らなければ、$f_E(x)$の逆関数である $f_E^{-1}(x)$（$= f_D(x)$）を求めることが計算量的に困難である、という 2 つの性質を満たすことを意味する。このような関数 $f_k(x)$は、「落し戸付き一方向性関数（trapdoor one way function）」と呼ばれている。落し戸とは、ある問題を解くために必要な情報のことであり、その情報を知っていればその問題が容易に解けるものの、逆にその情報を知らなければその問題を解くのが計算量的に困難となる。例えば、$f_E(x)$の逆関数 $f_E^{-1}(x)$を計算するという問題の落し戸は、パラメーターD となる。

この落し戸付き一方向性関数を利用することによって、公開鍵暗号による暗号通信が可能となる。まず、E を公開鍵として公開すると同時に、D を秘密鍵として秘密に保管する。そして、公開鍵 E による変換 $f_E(x)$を暗号化の変換とし、秘密鍵 D による変換 $f_D(x)(= f_E^{-1}(x))$を復号化の変換とする。これによって、公開鍵を利用した暗号化は誰にでもできるが、復号化は秘密鍵の保持者にしかできないことになり、データの守秘が確保される。さらに、暗号化のための鍵を事前に通信相手との間で秘密に共有する必要がなくなる（図 1 参照）。

Step 1： A は、B の公開鍵 Eb でメッセージ M を暗号化し、暗号文 C を B に送付する。

Step 2： B は、自分の秘密鍵 Db で C を復号化し、元のメッセージ M を得

---

67 計算量的に困難であるとは、その計算が理論的には可能であるが、実際に行うには計算量が莫大であり実行に膨大な費用と時間を要することから、事実上不可能であることを示す。計算量は暗号方式と鍵長に依存する。例えば、現在標準的と考えられている鍵長が 512 bit の RSA 暗号において、公開鍵から秘密鍵を計算するために必要とされる計算量は約 5.4×10^5 MIPS YEAR*、1,024 bit では約 4.0×10^{12} MIPS YEAR と試算されている。この計算を 5 年間で行うために要する費用は、1997 年時点でそれぞれ約 8,700 万円、約 65 兆円と推計されている。一方、楕円曲線上の離散対数問題に基づく方式（後述）では、鍵長が 180 bit のときの計算量が 1,024 bit の RSA 暗号の計算量に相当するとされている（楠田・櫻井、「公開鍵暗号方式の安全性評価に関する現状と課題」、IMES Discussion Paper Series、DPS 97-J-11、1997 年）。

* 1 MIPS は、1 秒間に百万回の命令を実行する計算能力を示しており、1 MIPS YEAR とは、1 MIPS の計算能力をもつ計算機が 1 年間演算処理を実行した場合の計算量を指す。

る。

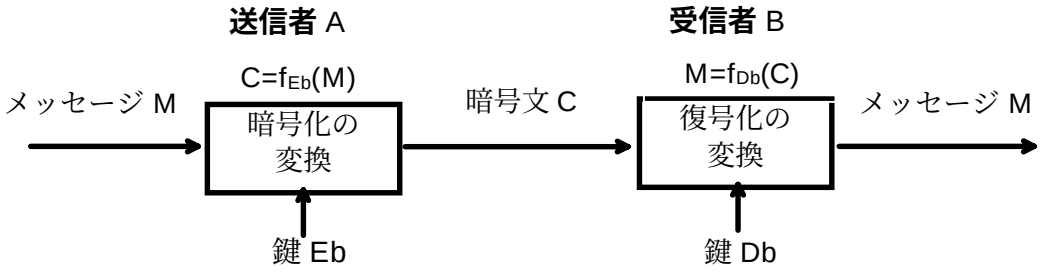


図 1 公開鍵暗号による暗号通信の典型的手順

この公開鍵暗号の原理は、1976 年に Stanford 大学の Diffie と Hellman によって最初に発表された[68]。もっとも、Diffie と Hellman は、このとき発表した論文の中で、上記の特殊な関数 $f_k(x)$の構成方法等、具体的な公開鍵暗号の数値変換の方法も併せて発表した訳ではない[69]。

2．ディジタル署名

署名は、あるデータの署名者を特定するために行われるものであり、署名者が署名対象のデータを自分固有の方法で変換する署名生成のステップと、不特定多数の署名検証者が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とによって構成されると考えられる。ディジタル署名は、署名対象のデータがディジタル・データに置き換わるだけであり、上記の意味の署名と機能的に何ら変わりはない。

ディジタル署名は、公開鍵暗号におけるデータ守秘のための変換をちょうど逆に行うことにより実現される。すなわち、署名者が署名対象データを自分の秘密鍵で変換すれば、その変換は署名者固有の変換となり、署名検証者は対応する公開鍵によって署名の正当性を検証できる。実際のディジタル署名の典型的方法は次の通りである（図 2 参照）。

Step 1： 署名作成者 A は署名対象データ *M* を「ハッシュ関数」と呼ばれる特殊な関数 *h* により一定長に圧縮し[70]、これをディジタル署名方式

---

68 このときに発表された論文の中で、公開鍵暗号の原理のほかに、離散対数問題を利用した鍵共有方式（後述の Diffie-Hellman 鍵共有方式）についても紹介されている（Diffie, W. and M. E. Hellman, “New directions in cryptography,” IEEE Transactions on Information Theory, Vol.IT-22, pp.644-654, November 1976.）。

69 後述する Diffie-Hellman 特許（U. S. Patent Number 4,200,770）は、この同じ論文の中で示された鍵共有方式（Diffie-Hellman 鍵共有法）のクレームを有しているが、ここで説明した公開鍵暗号の基本的な原理に関するクレームは有していない。基本的な公開鍵暗号の原理に関するクレームは、Merkle-Hellman 特許（U. S. Patent Number 4,218,582）に存在する。詳細については、第 4 章を参照。

70 ディジタル署名を作成する前に署名対象データをハッシュ関数によって圧縮しないで、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、この方法を採用した場合、圧縮されたデータのディジタル署名を作成するときに比べてよ

により自分の秘密鍵 d_A で変換することで署名 S を生成する。

Step 2： A は署名付きデータ（M, S ）を署名検証者 B に送信する。

Step 3： B は送信された署名 S を A の公開鍵 e_A で変換した結果と、送信された署名対象データ 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。

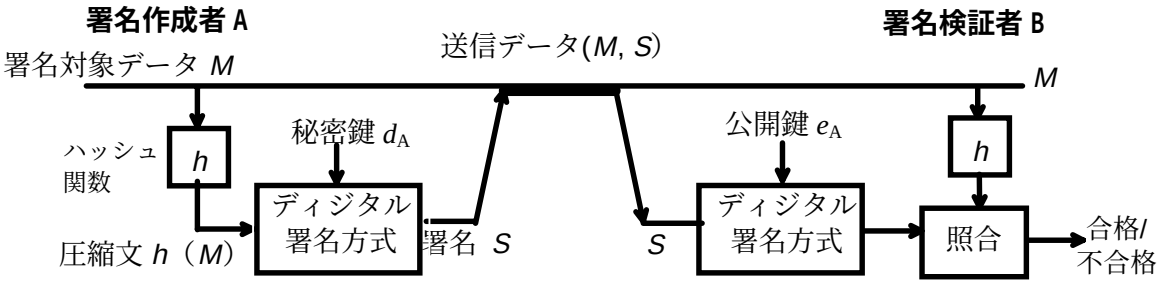


図 2　公開鍵暗号によるディジタル署名の典型的手順

Diffie と Hellman によって公開鍵暗号の原理が初めて発表された後、これまで数多くのディジタル署名の方式が提案されている。代表的なディジタル署名の方式をその数学的原理により分類すれば、「素因数分解問題に基づく方式」、「ナップザック問題に基づく方式」、「離散対数問題に基づく方式」、「楕円曲線により定義された離散対数問題に基づく方式」に分類できる（図 3 参照）。これらの方式が安全性の根拠とする数学の問題は、何れも解くことが計算量的に困難であると多くの数学者に信じられている。従って、あるディジタル署名方式の安全性がこうした数学の問題と厳密な意味で同等の難しさであることが証明されれば、その方式は安全であると確信できる。しかし、残念ながら、上記のいずれの方式もこれらの数学の問題と厳密な意味で同等の難しさであることが証明されていない。また、こうした数学の問題と厳密な意味で同等の難しさであることを証明できる他の方式も提案されてはいるが、それらの方式は未だに実用の域に達していない[71]。以下では、実用的なディジタル署名の方式を上記の４分

り多くのデータを処理する必要があるため、計算効率が低下するという問題が生じる。また、あるデータ x に対して公開鍵で暗号化した場合、データ x が、暗号化されたデータ $f_E(x)$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、暗号化されたデータ $f_E(x)$を秘密鍵 D で変換すると、$f_D(f_E(x)) = f_{E^{-1}}(f_E(x)) = x$ 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、すなわち $f_D(xy) = f_D(x)\, f_D(y)$が成立する場合には、x、y に対する署名 $f_D(x)$、$f_D(y)$を入手した者は xy に対する署名 $f_D(xy)$を偽造できるという問題も生じる。このような問題点が存在するため、ディジタル署名を作成する場合には、ハッシュ関数を用いてデータを圧縮する方法が一般的である。

[71] 素因数分解問題と等価という意味で安全な最初の方式が、Goldwasser, Micali, and Rivest により提案された。しかし、同方式は非常に計算効率が悪く非実用的である。これに対し、Bellare and Goldwasser は、Goldwasser, Micali, and Rivest の方式よりは幾分効率的な方式を提案したが、依然として実用には程遠い方式であった。Bellare and Goldwasser の方式における処理速度の問題に

類に沿って説明する。

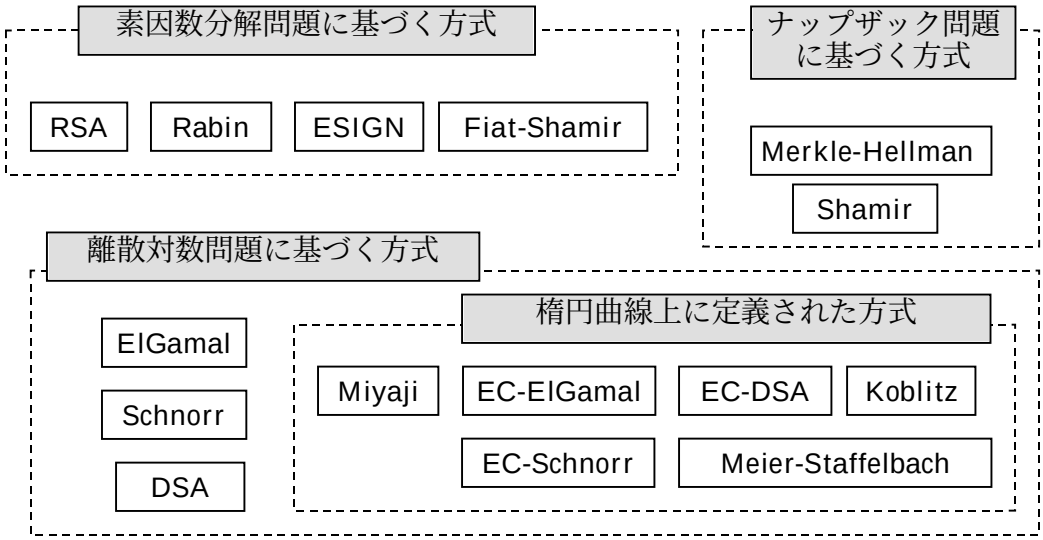


図 3 主要なディジタル署名方式の技術的関係

（1）素因数分解問題に基づく方式

最初に提案された本格的な公開鍵暗号が RSA 暗号[72]である。RSA 暗号は、Rivest、Shamir と Adleman によって提案された方式であり、大きな未知の素数の合成数を素因数分解する問題の困難性に依拠している。厳密に言えば、RSA 暗号に対して素因数分解を上回る解読法が存在しないことは証明されていないが、公表後 20 年近くの間、多くの研究者により安全性が検証されてきたにも拘らず、未だに素因数分解を上回る解読法が示されていないという事実が RSA 暗号の安全性に対する信頼を高めている。

RSA 暗号の問題点としては、①素因数分解問題と厳密な意味で同等の難しさであることが証明されていない点のほか、②非常に高次のべき乗剰余算[73]を行うため計算量が多く、処理速度があまり速くない点、③コンピュータのコスト・パフォーマンスの向上と素因数分解法の進歩に伴い、これまでよりも長い鍵を用いなければ安全性を維持できなくなりつつある点、④利用者は秘密鍵を生成する際に、比較的簡単に解読される弱鍵を排除するように配慮しなければならない点、等が挙げられる。

ついては、Dwork and Naor の方式によって計算量を RSA 暗号の 2～5 倍に抑えることに成功したものの、そのために相当の記憶容量を要する上、署名のサイズは RSA 暗号よりも大きくなるという問題が残されている。

[72] Rivest, R., 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.

[73] べき乗剰余算とは、対象となる自然数をべき乗して余りをとるという演算であり、公開鍵暗号方式において最も良く利用される演算である。例えば、法 p において自然数 N に対する x のべき乗剰余算は、$N^x \bmod p$ と表わされる。

Rabin 暗号[74]は、RSA 暗号を一部改良した方式であり、「受動的攻撃」[75]による解読の困難さが素因数分解問題と等価であることが証明されている暗号方式である。その意味で、Rabin 暗号は RSA 暗号の信頼性の問題を部分的に解決した方式であるといえる。しかし、Rabin 暗号は「能動的攻撃」[76]に対しては容易に解読されることが示されている。

ESIGN 署名[77]は、時間を要するべき乗剰余算の次数を小さくすること等により高速処理を実現したディジタル署名方式である。ESIGN 署名の安全性は、素因数分解問題の困難性に加え、高次合同不等式問題を解くことの困難性にも依存している[78]。高次合同不等式問題は、現時点では一般的な解法は示されていない。

Fiat-Shamir 署名[79]は、零知識対話証明（Zero Knowledge Interactive Proof；ZKIP）[80]を利用した Fiat-Shamir 認証[81]と呼ばれる本人確認の方式をディジタル署名用に改良した方式である。この方式では、ESIGN 署名と同様にべき乗剰余算の次数を小さくすること等により高速処理が実現されているが、反面、⑤署名の都度、乱数生成が必要となる、⑥署名長が長い、という問題が新たに生じている。

---

[74] Rabin, M. O., “Digitalized signatures and public-key functions as intractable as factorization,” M.I.T. Laboratory for Computer Science, Technical Report MIT/LCS/TR-212, 1979.

[75] 受動的攻撃は、攻撃者が選択できない文書に対する署名や署名者の公開鍵しか入手することができない場合に、別の文書の署名を偽造することである。

[76] 能動的攻撃は、攻撃者が予め選んだ幾つかの文書に対する署名を入手可能な場合に、別の文書の署名を偽造することである。

[77] Okamoto, T., “A fast signature scheme based on congruential polynomial operations,” IEEE Transactions on Information Theory, Vol. 36, No. 1, pp. 47-53, 1990.

[78] 高次合同不等式問題は、高次多項式を含む合同不等式の解を求める問題であり、合同不等式とは剰余算に関する不等式のことである。ESIGN 署名においては、署名検証式に $0 \leqq s^k \pmod n - h(m) < 2^{2\lfloor n/3 \rfloor}$ という高次合同不等式が利用されている（ただし、h はハッシュ関数、m は署名対象データ、s は署名であり、k は k≧4）。

[79] Fiat, A. and A. Shamir, “How to prove yourself: practical solutions to identification and signature problems,” Advances in Cryptology — Proceedings of CRYPTO ’86, Lecture Notes in Computer Science, Vol. 263, pp. 186-194, Springer-Verlag, 1986.

[80] 零知識証明は、ある者（証明者）がある知識を有することを当該知識に関する情報を一切漏らすことなく他者（検証者）に証明する方法のことであり、この証明を、証明者・検証者間の双方向通信により実現する方法が零知識対話証明である。

[81] Fiat-Shamir 認証は、零知識対話証明を利用した相手認証プロトコルであり、被認証者と検証者の間で４回以上情報をやり取りして認証を行う方式である。プロトコルを紹介すると、まず①認証センターが合成数 n=pq（p と q は素数）を計算、n を公開し、p と q は秘密とする、②被認証者は、1≦s≦n-1 を満たし n と互いに素な秘密鍵 s を生成し、公開鍵 v（$= s^2 \bmod n$）を生成する、③被認証者は乱数 r を生成し、$x=r^2 \bmod n$ を計算して x を認証者に送付する、④検証者は、1 bit の e（e は 0 もしくは 1）を生成して被認証者に送付する、⑤被認証者は、e=0 のときには $y=r \bmod n$ を、e=1 のときには $y=rs \bmod n$ を計算して y を検証者に送付する、⑥認証者は $y^2 \equiv xv^e \pmod n$が成立することを確認する、⑦②～⑥のステップを予め決められた回数繰り返し、いずれも⑥の合同式が成立した場合に、検証者は被認証者を認証することになる。

(2) ナップザック問題に基づく方式

ナップザック問題（Knapsack Problem）とは、例えば「長さが知られている n 本の棒があるとする。これらのうちの何本かを用いて、ある定められた長さの細長い箱に隙間が生じないように詰めよ」という問題である。この問題は次のように定式化される。

> 正整数 c、正整数を要素に持つ n 次元ベクトル（ナップザック・ベクトル）$\boldsymbol{a} = (a_1, a_2, \cdots, a_n)$ が与えられたとき、
>
> $$c = a\,m^t = \sum_i a_i m_i$$
>
> となるような 2 進数ベクトル $m = (m_1, m_2, \cdots, m_n)$ を求めよ。

このナップザック問題では、n が十分に大きい場合には、解となる 2 進数ベクトルを求めることが計算量的に困難となる。

Merkle と Hellman は、ナップザック問題の困難性に基づく方式である Merkle-Hellman 暗号[82]を開発した。その後、ナップザック問題に基づく多くの公開鍵暗号方式が提案され、Shamir 署名[83]等のディジタル署名方式も提案された。

しかし、これらの方式は、落し戸を仕掛けるためにナップザック問題に何らかの制約が付加された形に再構築されており[84]、その結果、それらの制約を手掛かりにラティス基底縮小アル

---

[82] Merkle-Hellman 暗号を簡単に説明する。まず、ナップザック行ベクトル $A = (a_1,...,a_n)$ を選び、$u > \sum_{i=1}^{n} a_i$ を満足する u を選ぶ。u と互いに素な整数 w を選び、$w^{-1} \cdot w \equiv 1 \pmod u$ を満たす w^{-1} を計算するとともに、$B = wA \bmod u$ を計算する。このとき、公開鍵は B、秘密鍵は A、u と w^{-1} である。送信者は、メッセージを n bit ごとのブロック $M = (m_1,...,m_n)$ に分割し、公開鍵 B を用いて暗号文 $C = BM^t$ を計算する。受信者は、秘密鍵を利用して $C' = w^{-1} \cdot C \bmod u$ を計算し、$C' = AM^t \bmod u$ と A から M を復号化する（Merkle, R. C. and M. E. Hellman, “Hiding information and signatures in trapdoor knapsacks,” IEEE Transactions on Information Theory, Vol.IT-24, No.5, pp.525-530, September 1978.）。

[83] Shamir 署名の署名生成・検証方法について簡単に説明する。この署名方式における秘密鍵 K は k 行 k 列の行列によって表わされ、公開鍵は乱数 n と、$KA^t \equiv B^t \pmod n$ を満足する k 次行ベクトル A である（ただし、$B = (2^0, 2^1, 2^2, ..., 2^{k-1})$）。署名生成では、まず署名を作成したい文章を k bit の 2 進数表現に変換し、それを k 次の行ベクトル M で表わす。そして、署名 S は $S = MK$ によって生成される。署名作成者は通信相手に（M,S）を送信する。署名検証では、署名受信者は $M = SA^t \bmod n$ が成立するかどうかを検証し、成立すれば真正な署名であることが判明する（Shamir, A. and R. E. Zippel, “On the security of the Merkle-Hellman cryptographic scheme,” IEEE Transactions on Information Theory, Vol. IT-26, No.3, pp.339-340, May 1980.）。

[84] 最初に提案された Merkle-Hellman 暗号は、ナップザック・ベクトルに超増加数列（$a_i > a_1 + \cdots + a_{i-1}$ （i=2,…,n）という条件を満たす数列のこと）という制約条件が付けられた「やさしい」ナップザック問題に基づく方式である。また、その後提案されたナップザック暗号の多くも、ナップザック・ベクトルの次元数と最大要素の比が小さくなる低密度と呼ばれる特徴があった。

ゴリズム[85]と呼ばれる強力な解法が開発され、これまで殆ど全てのナップザック暗号に解読法が示されている。

（3）離散対数問題に基づく方式

離散対数問題（Discrete Log Problem）に基づくディジタル署名の方式は、Diffie-Hellman 鍵共有法[86]に端を発している。Diffie-Hellman 鍵共有法の中で、共有する鍵の安全性を確保する方法として初めて離散対数問題が利用された。離散対数問題とは、所与の Y, α と素数 p の下で $Y = \alpha^{X} \bmod p$ を満たす X を導出する問題であり、現時点では、p と α が十分に大きければこの問題を解くことは素因数分解問題を解くことと同程度の難しさになると考えられている[87]。

この Diffie-Hellman 鍵共有法を参考にして開発されたとみられるディジタル署名方式が、ElGamal 署名[88]である。ElGamal 署名にも、RSA 暗号と同様に処理速度と鍵長の問題が残されているほか、Fiat-Shamir 署名と同じく署名生成の度に乱数生成が必要となる。さらに、署名長が長いという問題も生じている。

しかし、ElGamal 署名の署名長の問題は、Schnorr 署名[89]において「離散対数問題における法 p に対し p-1 の約数 q を法とする有限体上で署名を構成する」という署名圧縮技法によって

---

[85] ラティス基底縮小アルゴリズムとは、ラティス L と呼ばれる任意のベクトルによって張られる整数の線形結合の集合から最短ベクトルを求める方法のことである。ナップザック問題は、所与のナップザック・ベクトルに対応するラティス L から最短ベクトルを求める問題に帰着される（ラティス基底縮小アルゴリズムの詳細については、Lenstra, H. W. Jr., “Integer programming with a fixed number of variables,” Univ. of Amsterdam Tech. Report, Vol. 81-03, April 1981.）。

[86] Diffie-Hellman 鍵共有法の具体的な方法については、第 3 節「鍵配送・共有方式」を参照（Diffie, W. and M. E. Hellman, “New directions in cryptography,” IEEE Transactions on Information Theory, Vol.IT-22, pp.644-654, November 1976.）。

[87] 素因数分解問題と離散対数問題がどの程度困難な問題なのかは、数学的に明らかになっていない。ただし、現在知られている解法によってこれらの問題を解くために必要な計算量が見積られており、いずれも準指数関数的な時間が必要であることがわかっている。

[88] ElGamal 署名の署名生成・検証方法を説明する。署名者は、大きな素数 p と秘密鍵 x を選び、$y=\alpha^{x} \bmod p$（但し、α は法 p の原始根）を計算する。公開鍵が y、p と α であり、秘密鍵が x である。署名者は乱数 k を生成して $r=\alpha^{k} \bmod p$ を計算し、メッセージ M に対して $t=(M-xr)/k \bmod (p-1)$を計算する。M の署名が（r,t）である。受信者は、検証式 $\alpha^{M}=y^{r}r^{t} \bmod p$ によって署名検証を行う（ElGamal, T. E., “A public key cryptosystem 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, 1985.）。

[89] Schnorr 署名の署名生成・検証方法を簡単に説明する。署名者は、素数 p と p-1 の素因数 q を選び、$g^{q} \equiv 1 \pmod p$を満たす g を選ぶとともに、秘密鍵 a を用いて $v=g^{-a} \bmod p$ を計算する。次に乱数 k を選び、$r=g^{k} \bmod p$、$e=h(m,r)$と $y=ae+k \bmod q$ （m はメッセージ、h はハッシュ関数）を計算して、署名（e,y）を送付する。受信者は、公開鍵（p,q,g,v）を用いて $e'=h(m,g^{y}v^{e} \bmod p)$を計算し、e=e'が成立することを確認する（Schnorr, C. P., “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.）。

解決が図られた。また、Kravitz によって開発され、後に米国連邦政府ディジタル署名標準となった DSA 署名[90]は、ElGamal 署名の署名生成・検証式に上記の署名圧縮技法を導入している。

（4）楕円曲線により定義された離散対数問題に基づく方式

楕円曲線（elliptic curve）とは、3 次曲線（2 変数 x,y の関数 f(x,y)の次数が 3 であるような代数方程式 f(x,y)=0 の解の集合）のうち、特異点（関数 f の x および y に関する偏微分係数がともに 0 となる(x,y)）を含まない曲線のことである。

1985 年に、この楕円曲線によって定義された有限可換群上の離散対数問題を公開鍵暗号に利用するというアイデアが、Koblitz と Miller によって独立に考案された[91,92]。楕円曲線によって定義された離散対数問題には、一部の楕円曲線を除き、離散対数問題に存在する高速解法が現時点では存在しないため、従来と同程度の安全性を確保しつつ鍵長を大幅に短縮することが可能となるほか、鍵長の短縮に伴い処理速度も向上する。具体的な方式は、ElGamal 署名、Schnorr 署名や DSA 署名等の離散対数問題に基づく方式を、楕円曲線により定義された離散対数問題に基づく方式に翻訳することによって得られる。

楕円曲線により定義された離散対数問題に基づく方式の安全性と処理速度は、定義体や楕円曲線の選択等の具体的な構成法に大きく依存する。初期に提案された方式では、素因数分解問題や離散対数問題に基づく方式に比べて署名生成・検証のための基本演算が複雑なため、鍵長を短くした割には処理の高速化を達成できないという問題があった。こうした中で提案された Menezes-Vanstone による超特異楕円曲線[93]を利用した構成法は、高速処理が可能なため非常に有力な方式とされていたが、Weil 対という写像を応用することによって楕円曲線上で定義された離散対数問題を通常の乗法群上の離散対数問題に帰着させる方法が Menezes-Okamoto-

---

[90] DSA 署名の署名生成・検証方法を簡単に説明する。署名者は、素数 p と p-1 の素因数 q を選び、乱数 x を生成して $y=g^x \bmod p$ を計算する（g は法 q の原始根）。公開鍵は（p,q,g,y）であり、秘密鍵は x である。署名者は、乱数 k を選び、$r=(g^k \bmod p) \bmod q$、$t=(h(m)+xr)/k \bmod q$（h はハッシュ関数）を計算して、署名（r,t）を送付する。受信者は、$r'=(g^{h(m)/t}y^{r/t} \bmod p) \bmod q$ を計算し、r=r'が成立することを確認する（National Institute for Standards and Technology, “Digital Signature Standard (DSS),” Federal Information Processing Standards Publication(FIPS PUB) 186, May 19, 1994.）。

[91] Koblitz, N., “Elliptic curve cryptosystems,” Mathematics of Computation, Vol.48, pp.203-209, 1987.

[92] Miller, V. S., “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.

[93] 超特異楕円曲線（supersingular elliptic curve）は、要素の個数（位数）が q の有限体 F_q 上に定義された楕円曲線 E（F_q）のうち、その位数#E（F_q）を q+1-t と定める場合に、t^2=0,q,2q,3q または 4q となる楕円曲線 E（F_q）のことを言う。

Vanstone[94]によって示された（この方法は MOV 帰着と呼ばれている）。この結果、超特異楕円曲線により定義された離散対数問題を乗法群の離散対数問題に変換することで、従来の高速解法が適用可能となった。また、最近になって、Anomalous 曲線[95]と呼ばれる特殊なクラスの楕円曲線によって定義された離散対数問題についても、同じ要素の個数をもつ加法群に埋め込むことによって容易に解を求めることが可能であることが、佐藤・荒木[96]や、Smart[97]によって示されている。

こうした中、MOV 帰着を避けるとともに処理速度の向上を実現するための研究が進められており、その一つの方向として、Schoof のアルゴリズムと呼ばれる楕円曲線上で定義された加群の要素数を求める方法を実用化する方法が提案されている。他方では、楕円曲線に制約を設けることによって、MOV 帰着を回避するとともに処理速度向上を達成する方法が提案されており、要素の個数が 2^l（l は自然数）となる有限体上の超特異でない楕円曲線を利用した構成法が Koblitz[98]と Meier and Staffelbach[99]によって提案されている。また、楕円曲線上の点（x,y）の x、y 座標の絶対値がともに小さくなるような楕円曲線を利用した構成法が Miyaji[100]によって提案されている。

3．ハッシュ関数

ハッシュ関数は、任意のビット長のデータをある一定長のデータに圧縮変換する関数である。一般には、ハッシュ値を計算する場合、①対象となるデータ M をある一定のビット長のデータブロック M_i(i=1,...,n)に分割し、② M_i を、M_1 から M_n まで順番に圧縮変換関数 $f(M_i,H_{i-1})$（ただし、H_{i-1} は M_{i-1} を圧縮変換関数 $f(M_{i-1},H_{i-2})$によって変換したデータ、また、H_0 は初期値として予め設定）に入力することによって H_n を生成する。最終的に生成される H_n が、データ M のハッシュ値となる。

ハッシュ関数の安全性のうちで最も重要なものは、非衝突一致性（Collision Resistance）

---

94 Menezes, A. J., T. Okamoto, and A. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field,” Proceedings of STOC, pp. 80-89, 1991.

95 Anomalous 曲線は、楕円曲線の位数#E（F_q）が q となる楕円曲線 E（F_q）のことである（但し q は素数）。

96 Satoh, T. and K. Araki, "Fermat Quotients and the Polynominal Time Discrete Log Algorithm for Anomalous Elliptic Curves," preprint, 1997.

97 Smart, N., "Announcement of an attack on the ECDLP for anomalous elliptic curves," 1997.

98 Koblitz, N., “Cm-curves with good cryptographic properties,” Advances in Cryptology — Proceedings of CRYPTO ’91, Lecture Notes in Computer Science, Vol. 576, pp. 279-287, Springer-Verlag, 1992.

99 Meier, W. and O. Staffelbach, “Efficient multiplication on certain nonsupersingular elliptic curves,” Advances in Cryptology — Proceedings of CRYPTO ’92, Lecture Notes in Computer Science, Vol. 740, pp. 333-344, Springer-Verlag, 1993.

100 Miyaji, A., “Elliptic curves suitable for cryptosystems,” IEICE Transactions on Fundamentals, Vol.E77-A, No. 1, pp. 98-105, 1994.

である。非衝突一致性とは、ハッシュ値が一致するような二つの異なるデータを見つけることが計算量的に困難であることを意味し、真の署名付きデータから偽造データを作り出すことが困難であることを示す。非衝突一致性に関してハッシュ関数の強度を決定する一般的な要因は、ハッシュ値の長さである。ハッシュ値の長さが n bit のときに、$2^{n/2}$ 個の文書を集めた場合、同じハッシュ値を有する少なくとも一組のデータが発見される確率が 0.5 に近くなることが知られている[101]。従って、ハッシュ関数の安全性を確保するためには、ハッシュ値をある程度長く設定しなければならない。従来は 128 bit 程度で安全とされていたが、現在では 160 bit 以上が望ましいとされている。

ハッシュ関数の主な構成法は現在 2 つに大別される。一つはブロック暗号（後述）を利用する方式であり、もう一つは MD4 にはじまる MD 方式と呼ばれる方式である（図 4 参照）。

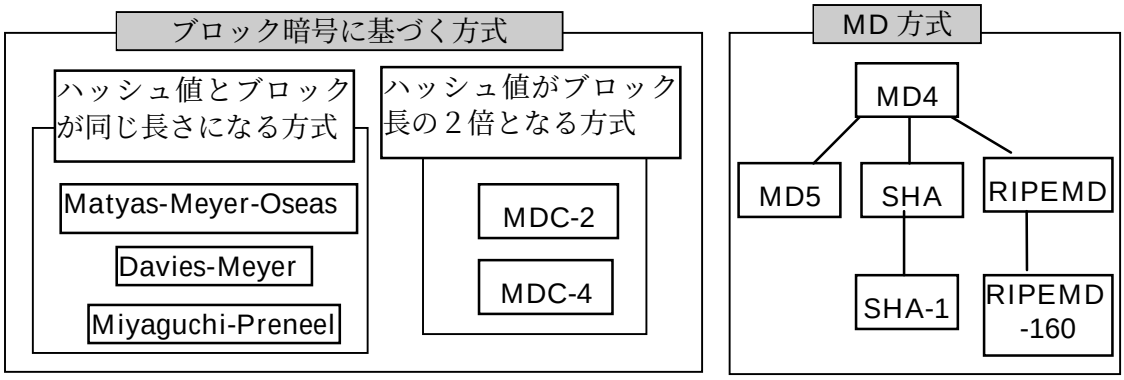


図 4　主要なハッシュ関数の技術的関係

（1）ブロック暗号に基づく方式

暗号鍵 *k* を利用したブロック暗号 *E* によってデータ *M* を暗号化する場合を E_k（*M*）で示し、関数 *H* を *H*（*M* ）=E_k（*M* ）⊕*M* と定義するとき（但し、*k* は固定されており、⊕はビット毎の排他的論理和演算とする）、ブロック暗号 *E* が安全であるならば関数 *H* は非衝突一致性を持つと考えられている。この考え方に基づいて構成されているのがブロック暗号に基づくハッシュ関数であり、その安全性は利用されるブロック暗号の安全性の問題に帰着される。一般的には、変換されたデータブロックを順々にフィードバックすることによって、ハッシュ値を計算する方法が利用されている[102]。

ブロック暗号に基づく代表的な方式としては、ハッシュ値がブロック長と一致する方式と、ブロック暗号を並列に接続しハッシュ値をブロック長の二倍とする方式に大別される。前者としては、Davies-Meyer 方式、Matyas-Meyer-Oseas 方式[103]、Miyaguchi-Preneel 方式等が挙

[101]この攻撃方法は birthday attack と呼ばれている。

[102]例えば、Davis-Meyer 方式では、文書 M のデータブロックを M_i(i=1,...,n)、ハッシュ関数 H、鍵 k によるブロック暗号を E_k とするとき、ハッシュ関数は $H(M_{i+1})=E_k(M_{i+1}) \oplus H(M_i)$ (i=1,...,n-1)となる。

[103] Matyas, S. M., C. H. W. Meyer, and J. Oseas, “Generating strong one-way functions with

げられる。ブロック長が 160 bit 以上で、安全性が高いと評価されているブロック暗号を利用できるのであれば、これらの方式も安全と考えられる。後者の代表的な方式としては、MDC-2 と MDC-4 が挙げられる。両者の違いは、並列処理の一系列において、MDC-2 がブロック暗号によって 1 回だけ変換を行う仕組みとなっているのに対し、MDC-4 はブロック暗号によって連続して 2 回実行する仕組みとなっている点である。ブロック長が 80 bit 以上で、安全性が高いと評価されているブロック暗号を利用できれば、これらの方式も安全と考えられる。

（2）MD 方式

MD 方式は、ブロック暗号の拡大鍵に相当する部分にデータを入力するという方式であり、ブロック暗号に基づく方式よりも高速処理が可能な方式として、現在ハッシュ関数の主流となっている。通常、拡大鍵はブロック・サイズに比べ相当大きいため、ハッシュ関数内での演算に利用されるデータブロックのサイズを大きくすることができることから、高速処理を実現できる。

MD 方式は、Rivest によって開発された MD4[104]と呼ばれる方式に端を発しているが、MD4 自体には、既に同じハッシュ値をもつ一組のデータを効率的に発見する方法が示されている[105]。その後、MD4 方式の改良版として、Rivest によって MD5[106]が開発されたほか、SHA-1[107]や、RIPEMD-128、RIPEMD-160[108]といった方式が提案されている。MD5 の主要な改良点は、MD4 において 3 つのステップによって構成されていたデータ変換プロセスを、新たにステップを１つ追加して 4 つのステップを有するデータ変換プロセスに変更した点である。MD5 は、暗号化電子メール用のフリーソフトである PGP（Pretty Good Privacy）や PEM（Privacy Enhanced Mail）[109]等で採用され、広範に利用されている。MD5 の安全性に関しては、MD5 の圧縮変換関数に対して、ある一定条件の下で同じ変換結果をもつデータを効率的に発見する

---

cryptographic algorithm," IBM Technical Disclosure Bulletin, Vol. 27, pp. 5658-5659, 1985. この方式の特許を IBM 社が取得しているという情報もある（特許番号等は不明）。

[104] Rivest, R. L., "The MD4 message digest algorithm" Advances in Cryptology — Proceedings of CRYPTO '90, Lecture Notes in Computer Science, Vol. 537, pp. 303-311, Springer-Verlag, 1991.

[105] Dobbertin, H., "A cryptanalysis of MD4," The Third Workshop of Fast Software Encryption, Lecture Notes in Computer Science, Vol. 1039, pp. 53-69, Springer-Verlag, 1996.

[106] Rivest, R. L., "The MD5 message digest algorithm" Request for comments（RFC）, 1321, Internet Activities Board, Internet Privacy Task Force, 1992.

[107] National Institute of Standards and Technology, "Secure hash standard," Federal Information Processing Standards Publication（FIPS PUB）180-1, April 17, 1995.

[108] Dobbertin, H., A. Bosselaers, and B. Preneel, "RIPEMD-160: A strengthened version of RIPEMD," The Third Workshop of Fast Software Encryption, Lecture Notes in Computer Science, Vol. 1039, pp. 71-82, Springer-Verlag, 1996.

[109] RFC 1421, "Privacy Enhancement for Internet Electronic Mail: Part III Algorithms, Modes, and Identifiers," IAB IRTF PSRG, IETF PEM WG, February 1993.

方法が示されているものの[110]、その圧縮変換関数を繰り返し利用してハッシュ値を計算する MD5 全体に対しては、効率的に同じハッシュ値をもつデータを見つける方法はこれまでのところ示されていない。

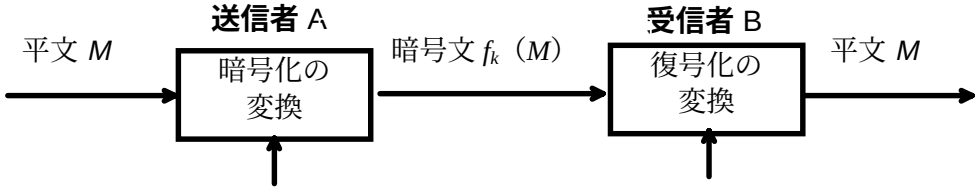
SHA-1 は、最初に米国政府のハッシュ関数標準となった SHA の改良方式である。SHA-1 の改良点は、SHA のデータブロック拡張変換に拡張後の各ブロックを 1 bit だけシフトさせるという変換を追加した点である。この改良は、SHA を開発したアメリカ国家安全保障局（National Security Agency、以下 NSA）が SHA の標準化直後に発見した安全性上の問題点を解決するために行われた[111]。

RIPEMD-160 は初期に提案された RIPEMD[112]の改良方式である。RIPEMD は、ハッシュ値の bit 長が 128 bit であったため birthday attack に対する安全性が問題となったことに加え、非衝突一致性に関する潜在的な脆弱性が示唆された。このため、ハッシュ値のビット長を 160 bit に拡大し、ハッシュ化を行う過程で利用する変換の回数を３回から５回に増やすという改良が行われた。この改良後の方式が RIPEMD-160 である[113]。

SHA-1 と RIPEMD-160 に関しては、現時点では安全性上の問題点は示されてない。また、両方式ともハッシュ値が 160 bit であり、birthday attack に対して当面十分な強度を持つと言われている。

4．共通鍵暗号

共通鍵暗号は、暗号化と復号化に同一の鍵を利用する方式である。共通鍵暗号は、ネットワーク上でやり取りされるデータや記録媒体に保管されるデータを秘匿するために用いられると同時に、無権限者によるデータの改ざんを防止・検出するための技術としても利用される。これらの機能は公開鍵暗号を利用しても実現されるが、公開鍵暗号による暗号化・復号化の処理速度が一般的に共通鍵暗号方式に比べて遅いという欠点が存在する。ただし、共通鍵暗号を利用するためには通信当事者間で鍵を共有しなければならないため、別途鍵を共有するための「鍵共有・配送法」が必要となる。共通鍵暗号によるデータの守秘・認証のための典型的な方法は次の通りである（図 5 参照）。



[110] Boer, B. D. and A. Bosselaers, "Collisions for the compression function of MD5," Advances in Cryptology, EUROCRYPT '93 (LNCS 765), pp. 293-304, 1994.

[111] NSA による SHA の改良については、FIPS PUB 180-1 に記載されている。

[112] Bosselaers, A. and B. Preneel, editors, Integrity Primitives for Secure Information Systems: Final Report of RACE Integrity Primitives Evaluation RIPE-RACE 1040, Lecture Notes in Computer Science Vol. 1007, Springer-Verlag, 1995.

[113] Dobbertin, H., “RIPEMD with two-round compress function is not collision-free,” Journal of Cryptology, Vol. 10 No. 1, pp. 51-70, 1997.

鍵 *K*　　鍵 *K*

図 5 共通鍵暗号方式による守秘・認証の典型的手順

共通鍵暗号は、大別すると「ストリーム暗号（stream cipher）」と「ブロック暗号（block cipher）」に分類される。

ストリーム暗号は、暗号化するデータの各ブロック（1～数 bit）に対応する鍵の系列（鍵ストリーム）を生成し、データの各ブロックとそれに対応する鍵ストリームの排他的論理和を計算することによって暗号化する方式である。ストリーム暗号には、暗号通信プロトコル SSL（Secure Socket Layer）で利用されている RC4 暗号等商用で利用されている方式も存在するが、軍事用に利用されているものが多く、具体的な数値変換の方法も非公開となる傾向にあり、特許が出願されることも少ないことから以下では検討の対象としない。

これに対し、現代の商用暗号の主流となっているのはブロック暗号である（図 6 参照）。ブロック暗号は、暗号化するデータをある一定長のブロックに分割し、すべてのブロックを同じ鍵で暗号化する方式である。現存するブロック暗号の多くが、Shannon が提唱した「合成暗号」[114]と呼ばれる暗号に分類される。IBM 社の Feistel は、Shannon の合成暗号のアイデアを基に、「換字変換」と「転置変換」[115]によって構成される「段関数（round function）」と呼ばれる基本変換を繰返し利用する Lucifer 暗号[116]を開発した。Lucifer 暗号は、２種類の「S-box」と呼ばれる換字変換手段を有しており、変換に利用される暗号鍵によってどちらの S-box を利用するかが決定される。暗号化されるデータのブロック長は 128 bit であり、S-box を含む段関数が 16 回繰り返される構造となっている。Lucifer 暗号の安全性については、差分解読法[117]による攻撃方法がいくつか報告されており、こうした解読法に対して十分な強度をもつとは必ずしも言えない。

---

[114] 合成暗号は、2 つ以上の異なる種類の変換を繰り返し利用する暗号である。Shannon は、個々の変換は単純でも、それらを組み合わせて繰り返し利用することによって強力な暗号を構成することが可能であることを情報理論的に示した（Shannon, C. E., “Communication theory of secrecy systems,” Bell System Technical Journal, Vol. 28, pp. 656-715, 1949.）。

[115] 換字変換とはある数字をある数字に規則的に置き換える変換であり、転置変換とは bit 位置を規則的に入れ替える変換である。

[116] Feistel, H., “Cryptography and computer privacy,” Scientific American, Vol. 228, No. 5, pp. 15-23, 1973.

[117] 差分解読法は、特殊な差分を持つ一組の平文のペアに対して特殊な差分を持つ暗号文のペアが生じる確率が高くなる場合に、それらの特定の暗号文と平文のペアに基づいて暗号鍵を推測する方法である。具体的には、平文における特定のいくつかの bit を 0 から 1、あるいは 1 から 0 に反転させた場合、暗号鍵の値に依存せず、暗号化の途中あるいは暗号化が完了したデータの特定のいくつかの bit が反転する確率が高くなる場合、その情報に基づいて暗号鍵を推測することによって全数探索よりも少ない計算量によって暗号鍵を見つけることが可能となる。

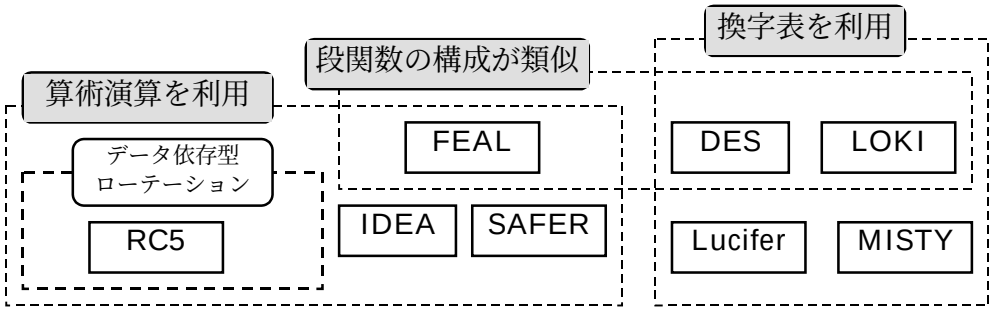


図６　主要なブロック暗号の技術的関係

DES 暗号[118]は、Lucifer 暗号を参考にして開発され、ブロック長 64 bit、鍵長 56 bit、段数 16 段のブロック暗号であり、米国政府標準暗号となっている。換字変換と転置変換によって構成される段関数を繰り返すという点では、DES 暗号は Lucifer 暗号と変わる所はないものの、段関数や、換字変換、転置変換等の設定において、高い安全性と高速処理を同時に達成するために周到な規準が用いられている。合成暗号の強度は、主に「*f* 関数」と呼ばれる暗号変換の非線形性によって決定されるが、DES 暗号の *f* 関数は、段関数を巧みに構成することによって逆関数を持たなければならないという制約から解放されており、効率性と安全性の観点のみから *f* 関数を設定することが可能な仕組みになっている。こうした利点から、DES 暗号における段関数の構成は、その後開発された FEAL 暗号[119]、LOKI 暗号[120]でも採用されている。また、DES 暗号は、差分解読法に対する安全性も考慮に入れて設計されていることが明らかになっている。しかし、線形解読法[121]に対しては、十分な強度を持つとは必ずしも言えないことが示されている。より深刻な問題は、鍵長が 56 bit と短いことから、コンピュータの処理速度の向上に伴って鍵の全数探索による解読の脅威が現実のものとなりつつあることである。

FEAL 暗号[122]は、鍵長 64 bit ないしは 128 bit のブロック暗号である。DES 暗号では *f* 関

---

[118] National Institute of Standards and Technology, “Data Encryption Standard（DES）,” Federal Information Processing Standards Publication（FIPS PUB）46-2, December 13,1993.

[119] Miyaguchi, S., A. Shiraishi, and A. Shimizu, “Fast data encipherment algorithm FEAL-8,” Review of Electrical Communication Laboratories, Vol. 36, No. 4, pp. 321-327, NTT, 1988.

[120] Brown, L., J. Pieprzyk, and J. Seberry, “LOKI — A cryptographic primitive for authentication and secrecy applications” Advances in Cryptology — Proceedings of ASIACRYPT ’90, Lecture Notes in Computer Science, Vol. 739, pp.36-50, Springer-Verlag, 1993.

[121] 線形解読法は、暗号文・平文のいくつかの bit の排他的論理和が暗号鍵のいくつかの bit の排他的論理和と等しくなる確率が 0.5（すなわち、無相関の場合）から乖離する場合に、この乖離をもっとも大きくする線形の近似式を構成し、その近似式を利用して暗号鍵を推測する方法である。この線形解読法は、差分解読法に比べて解読に必要とされる既知の平文量が少ないため、差分解読法よりも実現可能性が高いとされている。

[122] Miyaguchi, S., A. Shiraishi, and A. Shimizu, “Fast data encipherment algorithm FEAL-8,” Review of Electrical Communication Laboratories, Vol. 36, No. 4, pp. 321-327, NTT, 1988.

数における換字変換に換字表を用いているが、FEAL 暗号では算術演算（左 2 bit 循環シフト、256 を法とする加算）を用いている点が特徴である。算術演算は換字よりも処理が容易なため、暗号処理の高速化を実現できる。しかし、最初に提案された 8 段の FEAL 暗号（FEAL-8 暗号）は、差分解読法に対して簡単に解読されることが示された。そこで、差分解読法に対して十分な強度を確保するために、現在では 32 段の FEAL 暗号（FEAL-32 暗号）が推奨されている。FEAL-32 暗号の安全性については、差分解読法と線形解読法に対して近似的な評価から十分な安全性を有しているとみられている[123]。処理速度に関しては、算術演算が用いられているため、各段の処理速度は DES 暗号の処理速度を上回っているが、段数が DES 暗号の 2 倍の 32 段であるため DES 暗号と同程度となっている。

LOKI 暗号[124]は、鍵長とブロック長が 64 bit のブロック暗号である。LOKI 暗号は、段関数のほか *f* 関数の構成も DES 暗号とかなり類似しており、両者の違いは、LOKI 暗号が DES 暗号よりも大きな換字表を利用している点である。大きな換字表を利用することによって、暗号強度を高めるために換字表に要求される諸々の規準を達成し易くなることが示されており、LOKI 暗号はこうした性質を利用したものと考えられている。もっとも、初期に提案された LOKI'89 暗号は、差分解読法に対する潜在的な脆弱性が示されたこと等[125,126]を背景に改良がなされ、LOKI'91 暗号[127]が開発された。LOKI'91 暗号では、LOKI'89 暗号に対して、暗号鍵生成方法に bit シフトや bit 入れ替えが加えられ、最初と最後のデータブロックの排他的論理和変換が削除されたほか、換字変換の方法も一部変更された。LOKI'91 暗号の安全性については、差分解読法と線形解読法に対して近似的な評価から十分な強度を持つとみられている[128]。

---

[123] 差分解読法に対しては、FEAL-32 暗号は DES 暗号と同程度の安全性を有することが示されているほか、線形解読法に対しては、16 段の FEAL 暗号（FEAL-16 暗号）が DES 暗号と同程度の安全性を有することが示されている（Ohta, K. and K. Aoki, "Linear cryptanalysis of the Fast Data Encipherment Algorithm," Advances in Cryptology — Proceedings of CRYPTO '94, Lecture Notes in Computer Science, Vol. 839, pp. 12-16, Springer-Verlag, 1994.）。

[124] Brown, L., J. Pieprzyk, and J. Seberry, "LOKI — A cryptographic primitive for authentication and secrecy applications" Advances in Cryptology — Proceedings of AUSCRYPT '90, Lecture Notes in Computer Science, Vol. 453, pp. 229-236, Springer-Verlag, 1990.

[125] Knudsen, L. R., "Cryptanalysis of LOKI," Advances in Cryptology — Proceedings of AUSCRYPT '92, Lecture Notes in Computer Science, Vol. 718, pp. 196-208, Springer-Verlag, 1993.

[126] Biham, E. and A. Shamir, "Differential cryptanalysis of Snefru, Khafre, REDOC-II, LOKI, and Lucifer," Advances in Cryptology — Proceedings of CRYPTO '91, Lecture Notes in Computer Science, Vol. 576, pp. 156-171, Springer-Verlag, 1992.

[127] Brown, L., M. Kwan, and J. Pieprzyk, "Improving resistance to differential cryptanalysis and the redesign of LOKI," Advances in Cryptology — Proceedings of ASIACRYPT '91, Lecture Notes in Computer Science, Vol. 739, pp. 36-50, Springer-Verlag, 1993.

[128] Tokita, T., T. Sorimachi, and M. Matsui, "Linear cryptanalysis of LOKI and s^{2}DES," Advances in Cryptology — Proceedings of ASIACRYPT '94, Lecture Notes in Computer Science,

なお、LOKI'91 暗号は「関連鍵攻撃（Related-Key Attack）」[129]に弱いことが示されているが、この攻撃は実際の脅威とはならないといわれている。

IDEA 暗号[130]は、データブロック長 64 bit、鍵長 128 bit、段数 8 段の合成暗号である。IDEA 暗号では、入力データブロックを４つの 16 bit のブロックに分割して変換処理を行うなど段関数の構成が DES 暗号と異なるほか、非線形変換には算術演算（2^{16} を法とする加算、2^{16} を法とする乗法）が利用されている。IDEA 暗号の安全性については、差分解読法と線形解読法に対して近似的な評価から十分な強度を持つとみられている[131,132]。

SAFER 暗号は、鍵長とブロック長が 64 bit（SAFER K-64[133]）あるいは 128 bit（SAFER K-128[134]）のブロック暗号である。段数は 6 段、8 段、10 段、12 段のいずれも利用可能であり、非線形変換には全単射[135]のべき乗剰余算が用いられている。しかし、6 段の SAFER K-64 は、差分解読法に対し潜在的な脆弱性を持つことが明らかにされたため[136]、２種類の 64 bit の暗号鍵を入力して別々に鍵パラメーターを生成し、別々に生成された鍵パラメーターからデータブロックの暗号変換装置に交互に入力する仕組みに改良した。この改良版が SAFER K-128 であり、暗号鍵は 128 bit となる。また、SAFER SK-64 は、SAFER K-64 の鍵パラメーター生成部を改良した方式である。改良点は、具体的には、64 bit の暗号鍵を 8 bit の 8 つの

Vol. 917, pp. 293-303, Springer-Verlag, 1995.

129 関連鍵攻撃は、鍵が変更された場合、変更前と変更後の二つの鍵が特殊な既知の関係にあるときのみ実行できる解読法である（Biham, E., "New types of cryptanalytic attacks using related keys," Advances in Cryptology ― Proceedings of EUROCRYPT '93, Lecture Notes in Computer Science, Vol. 765, pp. 398-409, Springer-Verlag, 1994.）。

130 Lai, X., J. L. Massey, and S. Murphy, "Markov ciphers and differential cryptanalysis," Advances in Cryptology ― Proceedings of EUROCRYPT '91, Lecture Notes in Computer Science, Vol. 547, pp. 17-38, Springer-Verlag, 1991.

131 Lai, X., J. L. Massey, and S. Murphy, "Markov ciphers and differential cryptanalysis," Advances in Cryptology ― Proceedings of EUROCRYPT '91, Lecture Notes in Computer Science, Vol. 547, pp. 17-38, Springer-Verlag, 1991.

132 Harpes, C., G. G. Cramer, and J. L. Massey, "A generalization of linear cryptanalysis and the applicability of Matsui's piling-up lemma," Advances in Cryptology ― Proceedings of EUROCRYPT '95, Lecture Notes in Computer Science, Vol. 921, pp. 24-38, Springer-Verlag, 1995.

133 Massey, J. L., "SAFER K-64: A byte-oriented block-ciphering algorithm," Proceedings of Fast Software Encryption, Cambridge Security Workshop, Lecture Notes in Computer Science, pp. 1-17, Springer-Verlag, 1994.

134 Massey, J. L., "SAFER K-64: One year later," Proceedings of Fast Software Encryption, Second International Workshop, Lecture Notes in Computer Science, Vol. 1008, pp. 212-241, Springer-Verlag, 1995.

135 全単射とは、関数において逆関数が存在するということである。

136 Knudsen, L. R. and T. Berson, "Truncated differentials of SAFER," Proceedings of Fast Software Encryption, Third International Workshop, Lecture Notes in Computer Science, Vol. 1039, pp. 15-26, Springer-Verlag, 1996.

鍵ブロックに分割した後、鍵ブロックの排他的論理和を計算して新たに 9 番目の鍵ブロックを計算する点であり、鍵パラメーター生成のための演算を行う際に、従来利用されていた 8 つの鍵ブロックに加えて、その 9 番目の鍵ブロックも利用して鍵パラメーターが生成される。SAFER SK-128 は、SAFER K-128 に上記のように改良された鍵パラメーター生成部が組み込まれた方式である。SAFER SK-64（段数は 8 段以上を推奨）の安全性については、これまでのところ全数探索法よりも効率的な解読法は報告されていないようである。

RC5 暗号[137]はブロック長、鍵長、段数とも可変のブロック暗号である。但し、ブロック長 64 bit、鍵長 128 bit、段数 12 段または 16 段が一般的とされている。RC5 暗号の段関数の構成は *f* 関数に相当する関数の位置を除けば DES 暗号と同様であり、*f* 関数が算術演算（データ依存型循環シフト、加算）を用いているという点は FEAL 暗号等と同様である。ただし、算術演算の一つとしてデータ依存型循環シフトを用いている点が特徴である。RC5 暗号の安全性はブロック長と段数に依存するとされており、ブロック長 64 bit、段数 12 段の場合、差分解読法や線形解読法に対して近似的な評価から十分な強度を持つとみられている[138,139]。

このように、DES 暗号、FEAL 暗号、IDEA 暗号、LOKI 暗号や RC5 暗号については、いずれも差分解読法や線形解読法に対する厳密な評価が得られていない。Nyberg and Knudsen は、DES 暗号と同様の構造をもつ段関数と *f* 関数を有するブロック暗号の場合、段数が４段以上のときの平均差分確率[140]が、１段のときの確率を平方した値の 2 倍以下になることを示した[141]。また、Nyberg は線形解読法に対しても同様に評価できることを示した[142]。松井は、これらの結果から、*f* 関数が逆関数を持つ合成暗号で段数が 3 段以上のときに、平均差分確率と平均線形確率が段関数における同確率の平方以下になることを示し、この原理に基づいて MISTY 暗号を設計した[143]。

---

137 Rivest, R. L., “The MD5 message digest algorithm” Request for comments （RFC）, 1321, Internet Activities Board, Internet Privacy Task Force, 1992.

138 Kaliski, B. S. Jr. and Y. L. Yin, “On differential and linear cryptanalysis of the RC5 encryption algorithm,” Advances in Cryptology — Proceedings of CRYPTO ’95, Lecture Notes in Computer Science, Vol. 963, pp. 171-184, Springer-Verlag, 1995.

139 後にこの評価は必ずしも適切ではないことが指摘されているが（盛合・青木・太田[1996]）、同評価の結論を覆すようなものではない。

140 平均差分確率は、一定の入力データの差分値に対して、出力データの差分値がある一定の値をとる確率の最大値のことで、この確率が小さいほど、差分解読法に対して安全であるといわれている。また、平均線形確率は、入力データと出力データのいくつかの bit の排他的論理和が暗号鍵のいくつかの bit の排他的論理和に等しくなる確率が、0.5 から最も乖離するときの、その乖離の度合いを示す量である。平均線形確率が小さいほど、線形解読法に対して安全であるといわれている。

141 Nyberg, K. and L. R. Knudsen “Provable security against a differential attack,” Journal of Cryptology, Vol. 8, pp. 27-37, Springer International, 1995b.

142 Nyberg, K., “Linear approximation of block ciphers,” Advances in Cryptology — Proceedings of EUROCRYPT ’94, Lecture Notes in Computer Science, Vol. 950, pp. 439-444, Springer-Verlag, 1995a.

143 松井充、「ブロック暗号アルゴリズム MISTY」、電子情報通信学会技術研究報告、ISEC96-11、

MISTY 暗号は、ブロック長 64 bit、鍵長 128 bit のブロック暗号であり、差分解読法と線形解読法に対する安全性が数値的に保証されるように設計されている。また、鍵長を 128 bit とすることで、全数探索法に対しても十分な安全性が確保されている。MISTY 暗号は、サイズの異なる 2 種類の換字表を利用した非線形変換と、暗号鍵によってその形が変化する線形変換関数によって構成されており、MISTY 暗号全体の平均差分確率と平均線形確率を、非線形変換単体に対する同確率の 4 乗以下となるように設計されている。非線形変換には、差分・線形解読の成功確率を最小にするために、ガロア体上のべき乗演算が利用されている。処理速度については、非線形変換に換字表が利用されていると同時に、その非線形変換が並列処理されていることから、高速処理が可能となっている。MISTY 暗号には、数値変換の方法が異なる MISTY1、MISTY2 という 2 つのタイプが存在するが、安全性と処理速度のバランスから、MISTY1 は 8 段、MISTY2 は 12 段で利用することが推奨されている。

5．鍵配送・共有方式

鍵配送・共有法は、必ずしも秘密でない通信経路上において二者間（或は複数）で特定の情報を交換することにより、通信当事者間で秘密鍵を共有する方法である。主要な鍵配送・共有法を利用されている数学的問題によって分類すると、素因数分解問題に基づく方式、離散対数問題に基づく方式や、これら両方に基づく方式等に分類することができる（図 7 参照）。

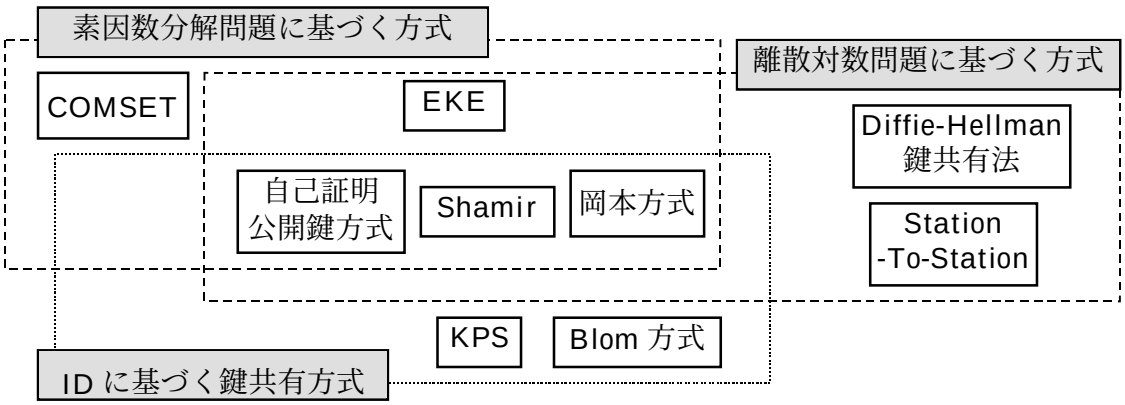


図 7　主要な鍵配送・共有方式の技術的関係

初期の鍵共有法としては、Diffie と Hellman の論文によって具体的方法が示された Diffie-Hellman 鍵共有法がある[144]。利用者 A と B の間で、Diffie-Hellman 鍵共有法によって鍵を共有する場合の具体的な方法は下記の通りである。

1996 年 a.

[144] Diffie, W. and M. E. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol. IT-22, pp.644-654, November 1976.

Step 1： A は[0, p-1]の整数 X_A をランダムに選び、公開されている素数 p とその原始根 α によって、

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

を計算し、Y_A を B に送信する。

Step 2： 同様に B は[0, p-1]の整数 X_B をランダムに選び、

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

を計算し、Y_B を A に送信する。

Step 3： A は鍵を次のように計算して、鍵 K を得る。

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

Step 4： B も次の計算によって鍵 K を得る。

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

このように、Diffie-Hellman 鍵共有法は離散対数問題の困難性に安全性の根拠を置いている。ただし、Diffie-Hellman 鍵共有法には、「中間侵入攻撃（intruder-in-the-middle-attack）」と呼ばれる攻撃が成立することが指摘されている。この攻撃は、攻撃者が当事者 AB 間の通信文を奪取することにより、A と自分、B と自分の間で鍵を共有し、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。この攻撃は通信当事者が相手を相互に確認していないために生じる。Diffie-Hellman 鍵共有法に相手を相互確認する仕組みを導入した「Station-To-Station プロトコル」[145]と呼ばれる方法が、Diffie-Hellman 鍵共有法の上記の問題点を解決した方法として提案されている。このプロトコルでは、通信相手との間で交換する情報にディジタル署名を作成・付加することによって、送付された情報が通信途中で改ざんされていないかどうかをチェックすることが可能になる。このほか、Rabin 暗号と零知識対話証明（ZKIP）が利用されている「COMSET」[146]や、共通鍵暗号方式と公開鍵暗号方式を組み合わせた「EKE（Encrypted Key Exchange）」[147]等が提案されている。

---

[145] Diffie, W., P. C. van Oorschot, and M. J. Wiener, “Authentication and authenticated key exchanges,” Designes, Codes and Cryptography, Vol.2, pp.107-125, 1992.

[146] Brandt, J., I. B. Damgard, P. Landrock, and T. Pederson, “Zero-knowledge authentication scheme with secret key exchange,” Advances in Cryptology ― Proceedings of CRYPTO ’88, Lecture Notes in Computer Science, Vol. 403, pp. 583-588, Springer-Verlag, 1990.

[147] EKE のプロトコルは、①暗号通信利用者 A、B は秘密情報を何らかの方法で共有し、まず A がその情報から暗号鍵を生成し、共通鍵暗号を使って自分の公開鍵を B に送付する、② B は共有する鍵 K を A の公開鍵で暗号化するとともに、秘密情報を利用して共通鍵暗号によって暗号化する、③ A は K を入手し、乱数 Ra を生成して K で暗号化して B へ送付する、④ B は K で Ra を復号化するとともに新たに乱数 Rb を生成し、Ra と Rb を K で暗号化して A に送付する、⑤ A は B から送付されたデータに Ra が含まれていることを確認し、残りのデータ Rb を K で暗号化して B に送付する、⑥ B は A から送られてきたデータを K で復号化し、Rb が含まれていることを確認する、

また、各利用者の ID（名前、電話番号等の情報が利用可能）を利用して、公開鍵を生成・共有する方式が Shamir によって提案されている[148]。この方式では、信頼できる鍵生成センターが各利用者の ID から秘密鍵を生成して、各利用者に秘密に配送する。公開鍵は、各利用者が通信相手の ID から生成することが可能である。その後提案された ID に基づく鍵共有方式としては、Diffie-Hellman 鍵共有法と RSA 暗号を利用した「岡本方式」[149]や、Blom 方式[150]を改良し、線形スキームを利用した「KPS（Key Predistribution System）」[151]、RSA 暗号と離散対数問題を利用した「自己証明鍵方式」[152]等が挙げられる。

という手順から成る。このように、①～②で鍵を共有した後、③～⑥で鍵が正しく共有されていることを乱数を利用して確認する点が特徴である（Bellovin, S. M. and M. Merritt, “Encrypted key exchange: Password-based protocols secure against dictionary attacks,” Proceedings of the 1992 IEEE Computer Society Conference on Research in Security and Privacy, pp. 72-84, 1992.）。

[148] Shamir, A., “Identity-based cryptosystems and signature schemes,” Advances in Cryptology — Proceedings of CRYPTO ’84, Lecture Notes in Computer Science, Vol. 196, pp. 47-53, Springer-Verlag, 1985.

[149] 「岡本方式」では、最初に①鍵生成センターが、各利用者の ID 情報と自分の秘密鍵から各利用者固有の情報 S を生成して各利用者に送付し、自分の公開鍵を公表する、②利用者はそれぞれ乱数を生成して、乱数と S から通信相手に送付する情報 X を生成、送付する、③通信相手から X を受け取った後、X、相手の ID、自分の乱数と鍵生成センターの公開鍵から鍵 K を生成、共有する、という手順によって構成されている（Okamoto, E. and K. Tanaka, “Key distribution based in identification information,” Electronics Letters, Vol. 7, No. 4, pp. 481-485, May 1989.）。

[150] Blom 方式では、まず、信頼できる第三者が鍵を共有するための各利用者固有のデータを生成・公開するとともに、共通鍵を生成するための関数（通信利用者２名の固有データが変数）を生成・公開する。暗号通信利用者は、鍵を生成・共有する際に自分の固有データと通信相手の固有データを関数に入力し、その計算結果が共有鍵となる（”Blom, R., “Non-public key distribution,” Proceedings of Crypto ’82, pp. 231-236, Plenum Press, 1983.”）。

[151] 線形スキームの KPS では、①鍵生成センターは、k 個の n 次対称行列 D_i（i=1,...,k）を生成するとともに（この対称行列は秘密にしておく）、各利用者 i に s_i=（$s_{i,1}$,...,$s_{i,k}$）=（$ID_i \cdot D_1$,...,$ID_i \cdot D_k$）を計算して配布する（ID_i は i の ID 情報であり、n 次行ベクトル）、②利用者 i が利用者 j と鍵を共有する場合、共有鍵 K_{ij} は K_{ij} =h($s_{i,1} \cdot ID_j^t$,...,$s_{i,k} \cdot ID_j^t$)という計算によって生成し、利用者 j は K_{ji} =h($s_{j,1} \cdot ID_i^t$,...,$s_{j,k} \cdot ID_i^t$)によって生成する、という仕組みによって鍵が共有される。KPS は、インターネット上での電子メールソフト Eudora や MS exchange 等にプラグイン形式で機能追加できる形で既に製品化されている（Matsumoto, T. and H. Imai, “On the key distribution system: A practical solution to the key distribution problem,” Advances in cryptology — Proceedings of CRYPTO ’87, Lecture Notes in Computer Science, Vol. 293, pp. 185-193, Springer-Verlag, 1988.）。

[152] 自己証明鍵方式は、共有する鍵を生成する際に、鍵生成センターが各利用者の ID を用いて生成した各利用者固有の公開情報 P を利用する点に特徴がある。鍵生成センターは、各利用者固有の公開情報を生成する際に、各利用者から秘密情報 S とそれをべき乗剰余算によって変換したデータ X を入手し、X が正当に S から生成されたものであることを確認して、X、利用者の ID と自分の秘密鍵から RSA 方式によって P を生成する。したがって、公開情報 P は鍵生成センターによってある特定の利用者の ID と結び付けられており、P 自体がその利用者の ID 証明書の役割を果たしている。なお、第三者がこの P を偽造しようした場合、X から S を計算するために離散対数問題を解く

6．ブラインド署名

ブラインド署名はディジタル署名の応用技術であり、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ある。例えると、「署名を受ける文書とカーボン紙を封筒に入れ、文書の内容を知られることなく封筒の上から署名をして貰う」方法と言えよう。主要なブラインド署名も、利用されている数学的問題によって、素因数分解問題に基づく方式、離散対数問題に基づく方式やこれら両方に基づく方式に分類することができる（図 8 参照）。

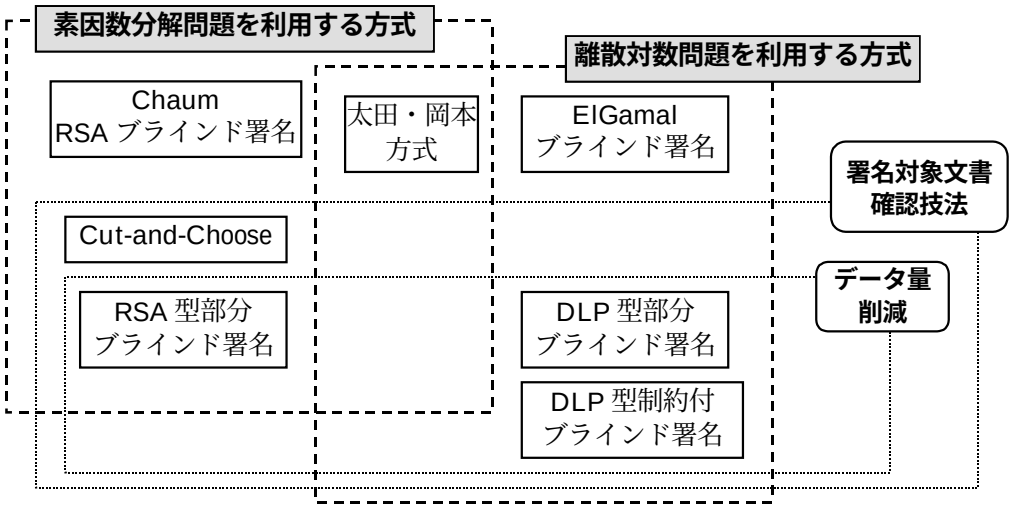


図 8　主要なブラインド署名方式の技術的関係

ブラインド署名の概念を最初に提示したのは Chaum であるが、彼は RSA 暗号を用いた具体的な方式も併せて示した[153]。以下では、RSA 暗号によるブラインド署名の基本手順を、署名依頼者 A が文書 M に署名者 B のブラインド署名を受ける場合で説明する。なお、署名者 B の RSA 暗号の公開鍵を（*e,n*）、秘密鍵を *d* とする。

Step 1：　署名依頼者 A は秘密の乱数 *r* を生成した後、署名者 B の公開鍵（*e,n*）を用いて

$$X = Mr^{e} \mod n$$

を計算し、結果 *X* を B に送信する。

Step 2：　B は自分の秘密鍵（*d,n*）を用いて

$$Y = X^{d} \mod n$$

必要がある（Girault, M., “Self-certified public keys,” Advances in Cryptology — Proceedings of EUROCRYPT ’91, Lecture Notes in Computer Science, Vol. 547, pp. 490-497, Springer-Verlag, 1991.）。

[153] Chaum, D., “Blind signatures for untraceable payments,” Advances in Cryptology — Proceedings of CRYPTO ’82, pp. 192-203, Prenum Press, 1983.

$= M^d r \mod n$

を計算し、結果 Y を A に送信する。

Step 3： A は受け取った Y に秘密の乱数の逆数 r^{-1} をかけることにより B の署名 $S = M^d \mod n$ を得る。

Step 4： 最後に A は署名 S の正当性を B の公開鍵により検証する。

$M = S^e \mod n$

このブラインド署名を利用した ElGamal 署名方式[154]も提案されているほか、他のブラインド署名・認証技術として「太田・岡本方式」[155]が挙げられる。太田・岡本方式は、Fiat-Shamir 認証等素因数分解問題に基づく認証方式や離散対数問題に基づく認証方式をベースに、交換される情報が署名者に知られないように零知識対話証明を利用したブラインド認証方式である。そのため、太田・岡本方式の安全性はベースとなる方式によって異なる。

ブラインド署名は署名者依頼が署名者に署名対象データの内容を知られることなく署名を受ける方法であるが、署名依頼者は署名者にとって望ましくないデータに署名させることが可能となり、応用次第では問題が生じる可能性がある。さらに、ブラインド署名を電子現金に応用する場合、利用者のプライバシーを確保するとともに、その電子現金の属性（額面や有効期限等）等については電子現金の受領者も確認できるようにする必要がある。これに対して、Chaum は cut-and-choose methodology と呼ばれる、抜取検査手順を導入する方式を提案している。この方式では、①署名依頼者は署名を希望するブラインド化されたデータを複数用意しておき、②署名者は、それらのデータに署名する前に、幾つかのデータを指定して（抜き取って）署名依頼者にデータの内容を明らかにさせる、③抜き取られたデータの内容や属性が署名者にとって望ましくないものでなければ、署名者はデータすべてに署名をする、という手順に

---

154 Camenisch, J., U. Maurer, and M. Stadler, “Digital payment systems with passive anonymity-revoking trustees,” Computer Security – Proceedings of ESORICS ’96, Lecture Notes in Computer Science, Vol. 1146, pp. 33-43, Springer-Verlag, 1996.

155 太田・岡本方式のうち、ID に基づく Fiat-Shamir 署名をベースにしたブラインド署名方式を簡単に説明する。署名の正当性を証明する証明者を A、署名依頼者を B、署名を確認する確認者を C とする。まず、① A は B の ID 情報 x を公表すると同時に、乱数 r_i を生成して $X_i'=r_i \times x \mod N$ を計算し、X_i'を B に送付する(i=1,...,t)、② B は乱数 u_i と e_i（ただし、e_i は 0 もしくは 1）を生成し、$X_i''=u_i^2 \times x^{-ei} \times X_i' \mod N$ を計算するとともに、ハッシュ関数 f、メッセージ m の下で $\beta_i=f(m,X_i'')$ を計算し、$\beta_i'=\beta_i+e_i \mod 2$ を計算して $\beta'=(\beta_1',...,\beta_t')$を A に送付する、③ A は秘密鍵 S によって $Z_i=r_i \times S^{\wedge} \beta_i' \mod N$ を計算し、$Z=(Z_1,...,Z_t)$を B に送付する、④ B は、$e_i=1$ かつ $\beta_i=0$ の場合には $Z_i'=u_i \times Z_i \times X^{-1} \mod N$ を計算し、それ以外の場合には $Z_i'=u_i \times Z_i \mod N$ を計算して、m、β と Z_i'を C に送付する、⑤ C は $X_i^*=Z_i'^2 \times X^{\wedge} \beta \mod N$ と $\beta^*=f(m,X^*)$を計算し、β^*が受け取ったβに等しいかどうかを確認する（Okamoto, T. and K. Ohta, “Divertible zero-knowledge interactive proofs and communicative random self-reducibility,” Advances in Cryptology ― Proceedings of EUROCRYPT ’89, Lecture Notes in Computer Science, Vol. 434, pp. 134-149, Springer-Verlag, 1990.）。

なる。しかし、このような抜取検査手順を組込むと、ブラインド化や署名の演算対象となるデータ量が増加するという欠点が生じる。

この cut-and-choose methodology を利用せず、署名の一部を確認する方法として、「RSA 暗号による部分ブラインド署名（partially blind signature）」[156]、「離散対数問題に基づく制約付きブラインド署名（restrictive blind signature）」[157]、「離散対数問題に基づく部分ブラインド署名」[158]が提案されている。「部分ブラインド署名」は、署名依頼者と署名者の間で特定の情報（clear part、以下、共有情報と呼ぶ）を共有し、この共有情報を利用して署名を作成する方式である。この方法では、①署名依頼者は、ブラインド化されたメッセージとともにブラインド化されていない共有情報を署名者に送付し、②署名者はブラインド化されていない共有情報を確認した後、ブラインド化されたメッセージと共有情報の両方に対して RSA 方式によってディジタル署名を作成する、という手順となる。このように、署名者に送付されるデータの一部がブラインド化されていることから、「部分ブラインド署名」と呼ばれている。この方式では、利用される公開鍵と秘密鍵が共有情報に基づいて生成されるため、署名されたデータと署名依頼者を関連付けることが可能なほか、署名検証の際に正しい共有情報が利用されているかどうかをチェックすることができる。上記の方式を RSA 暗号に基づいて構成する方式が「RSA 暗号による部分ブラインド署名」、離散対数問題に基づいて構成する方式が「離散対数問題に基づく部分ブラインド署名」である。また、「離散対数問題に基づく制約付きブラインド署名」は、署名者が署名生成時にある特定の情報を署名に混入することができるという方式である。署名の安全性は離散対数問題に基づいている。

なお、「部分ブラインド署名」に関しては、複数の共通情報とその署名を利用して別の署名を作成できる可能性もあり、これを排除するために公開鍵を大きくすると計算量効率性が低下するいう短所が指摘されている。また、「制限付きブラインド署名」についても、署名後に署名依頼者によって一度非ブラインド化されると、署名者は署名に混入した情報を再び確認することができなくなるという短所が存在する。

---

156 Abe, M. and E. Fujisaki, “How to date blind signatures,” Advances in Cryptology — Proceedings of ASIACRYPT ’96, Lecture Notes in Computer Science, Vol. 1163, pp. 244-251, Springer-Verlag, 1996.

157 Brands, S., “An efficient off-line electronic cash system based on the representation problem,” Technical Report, CS-R9323, CWI, April 1993.

158 Abe M. and J. Camenisch, “Partially blind signature schemes,” Proceedings of The 1997 Symposiums on Cryptography and Information Security, SCIS97-33D, January 1997.

Ⅳ 暗号関連特許

1．公開鍵暗号の原理に関する特許

（1）Merkle-Hellman 特許

Merkle-Hellman 特許[159]は、ナップザック問題を利用することによって通信文書の機密性と真正性を確保する方法と装置の特許である。この特許は、公開鍵暗号を利用した暗号通信とディジタル署名の基本的な方法のクレームを有しており、公開鍵暗号の原理に関する特許であると考えることができる。なお、この特許はアメリカと日本で成立しているが、アメリカにおける Merkle-Hellman 特許は 1997 年 8 月 19 日に失効している。

＜アメリカの Merkle-Hellman **特許＞**

アメリカの Merkle-Hellman 特許は 17 のクレームを有している。まず、公開鍵暗号を利用した暗号通信に関するクレームを整理した後、ディジタル署名に関連するクレーム、そしてナップザック問題を利用した方式に関するクレームという順番で説明する。

最初に公開鍵暗号方式に関するクレームをみると、まずクレーム 1 には、暗号通信の典型的な方法（method）が記載されている。

クレーム 1：

「メッセージを送信者から受信者に送付するというタイプの通信を、機密保持性のないチャネルにおいて安全に実行するための方法であって、
①受信者は乱数を生成し、
②受信者は、上記乱数から受信者用の公開暗号鍵を生成し、
③受信者は、上記乱数から、公開暗号鍵と直接関係しているが公開暗号鍵からは計算量的に生成することが実現不可能な秘密復号鍵を生成し、
④受信者は、送信者に公開暗号鍵を送付し、
⑤送信者は、公開暗号鍵を利用すると変換が容易であるが秘密復号鍵なしではその逆変換が計算量的に不可能となるような暗号変換によって公開暗号鍵で暗号文を生成し、
⑥送信者は、暗号文を受信者に送付し、
⑦受信者は、暗号文を復号化するために暗号文を秘密復号鍵で変換する方法」

と記載されている[160]。このクレームには、乱数から公開暗号鍵を生成する方法や公開暗号鍵で

---

159 特許明細書における発明者の記載によると、Hellman-Merkle 特許という呼び方が正式であるが、この特許に記載されている暗号が、一般的には Merkle-Hellman 暗号と呼ばれているため、ここでは Merkle-Hellman 特許と呼ぶこととする。なお、以下で説明する特許においても、その特許に記載されている暗号の一般的な呼び方で呼ぶこととする（U. S. Patent Number は 4,218,582<出願日 1977 年 10 月 6 日、発効日 1980 年 8 月 19 日>、日本国特許公告番号は昭 59-50068<出願日 1978 年 10 月 6 日、公告日 1984 年 12 月 6 日>）。

160 クレーム 1 には、"In a method of communicating securely over an insecure communication

暗号文を生成する暗号変換の方法に関して数値変換の方法が特定されていないが、「実施例の説明」にナップザック問題を利用した数値変換の方法と数値例が記載されている。クレーム 2 はクレーム 1 の従属項、クレーム 3 はクレーム 2 の従属項となっており、いずれもクレーム 1 の暗号通信に受信者確認の方法（method）を追加したものである。クレーム 2 の受信者確認の方法は、受信者が暗号文を正確に復号化できるかどうかをチェックするというものであり、クレーム 3 の方法は、受信者が、受け取ったメッセージを表わすデータ（representation of message）を送信者に返信するというものである。一方、クレーム 6 は、上記のクレーム 1 に対応する装置（apparatus）のクレームであり、装置を構成する要素として、①乱数を生成する手段（means）、②乱数から公開暗号鍵を生成する手段、③乱数から秘密復号鍵を生成する手段、④公開暗号鍵を受信者から送信者に送付する手段、⑤平文を公開暗号鍵によって暗号化する手段、⑥暗号文を送信者から受信者に送付する手段と、⑥受信者側で暗号文を秘密復号鍵で平文に復号化する手段、が記載されている。

ディジタル署名については、クレーム 4 と 5 に、ディジタル署名を利用して通信メッセージの真正性を確保するための方法（method）が記載されている。ディジタル署名に関連するクレームはこの 2 つだけである。

クレーム 4：

「通信メッセージのディジタル署名を生成する方法であって、
①送信者は乱数を生成し、
②送信者は上記乱数から公開鍵を生成し、
③送信者は、公開鍵と直接関係しているが公開鍵から生成することが計算量的に実現不可能な方法によって、上記乱数から秘密鍵を生成し、
④送信者は、公開鍵から生成することが計算量的に実現不可能な方法によって、メッセージを表わすデータを秘密鍵で変換してディジタル署名を生成し、
⑤送信者は公開鍵を受信者に送付し、
⑥送信者はメッセージとそのディジタル署名を受信者に送付し、
⑦受信者は、メッセージとディジタル署名を受け取り、ディジタル署名を公開鍵で復号化

---

channel of the type which communicates a message from a transmitter to a receiver, the improvement characterized by: providing random numbers at the receiver; generating from said random numbers a public enciphering key at the receiver; generating from said random numbers a secret deciphering key at the receiver such that the secret deciphering key is directly related to and computationally infeasible to generate from the public enciphering key; communicating the public enciphering key from the receiver to the transmitter; processing the message and the public enciphering key at the transmitter and generating an enciphered message by an enciphering transformation, such that the enciphering transformation is easy to effect by computationally infeasible to invert without the secret deciphering key; transmitting the enciphered message from the transmitter to the receiver; and proceeding the enciphered message and the secret deciphering key at the receiver to transform the enciphered message with the secret deciphering key to generate the message." と記載されている。

してメッセージを表わすデータを得て、
⑧受信者は、ディジタル署名から生成したメッセージを表わすデータとメッセージを比較し、ディジタル署名の正当性を確認する、
というステップによって構成される方法」[161]

一方、クレーム 5 には、送信者がメッセージ全体を秘密鍵で変換してディジタル署名を作成し、受信者はディジタル署名を公開鍵で変換して、変換後のディジタル署名の冗長性[162]によってその正当性を検証する方式が記載されている。このように、ディジタル署名作成方法として、①メッセージを表わすデータを秘密鍵で変換してディジタル署名を作成する方法と、②メッセージ自体を秘密鍵で変換してディジタル署名とする方法の 2 通りが記載されている。ディジタル署名の具体的な数値変換の方法はクレーム 4 と 5 には記載されていないが、「実施例の説明」の中に、ナップザック問題を利用した Merkle-Hellman 暗号のディジタル署名生成・検証の数値変換の方法が記載されている。

クレーム 7〜17 には、ナップザック問題を利用した暗号通信の方法と装置が特定されている。クレーム 7 には、加法型のナップザック問題に基づく Merkle-Hellman 暗号を利用した暗号通信の方法（method）が記載されており、その数値変換の方法が特定されている。また、クレーム 9 には、乗法型のナップザック問題に基づく Merkle-Hellman 暗号を利用した暗号通信方法（method）が記載されており、その数値変換の方法が特定されている。クレーム 10 はクレーム 7 に対応する装置（apparatus）のクレームとなっており、クレーム 12 はクレーム 9 に対応する装置（apparatus）のクレームとなっている。また、クレーム 13 には、公開鍵暗号を利用した暗号化装置（apparatus）が記載されている。

---

161 クレーム 4 は、"In a method of providing a digital signature for a communicated message comprising the steps of providing random numbers at the transmitter; generating from said random numbers a public key at the transmitter; generating from said random numbers a secret key at the transmitter such that the secret key is directly related to and computationally infeasible to generate from the public key; processing the message to be transmitted and the secret key at the transmitter to generate a digital signature at said transmitter by transforming a representation of the message with the secret key, such that the digital signature is computationally infeasible to generate from the public key; communicating the public key to the receiver; transmitting the message and the digital signature from the transmitter to the receiver; receiving the message and the digital signature at the receiver and transforming said digital signature with the public key to generate a representation of the message; and validating the digital signature by comparing the similarity of the message to the representation of the message generated from the digital signature."と記載されている。

162 本特許の「実施例の説明」に、ディジタル署名の検証に冗長性を利用する方法として、ディジタル署名にメッセージの作成日付や時間を記載するフィールドを設けてそれをチェックする方法や、メッセージが文法的に正しい英文かどうかをチェックする方法が記載されている。

クレーム 13：

「機密保持性のない通信チャネル上でやり取りされるメッセージを暗号化するための装置であって、秘密に保たれているメッセージを受け取るために接続された入力と、公開暗号鍵を受け取るために接続された別の入力と、暗号化されたメッセージを生成して出力する出力を有し、

①メッセージを受け取り、メッセージを表わすベクトルに変換する手段（means）と、

②公開暗号鍵を受け取り、公開暗号鍵を表わすベクトルに変換する手段と、

③メッセージを表わすベクトルと公開暗号鍵を表わすベクトルのドット積を計算することによって暗号化されたメッセージを生成する手段であって、メッセージを表わすベクトルを受け取るために接続された入力と、公開暗号鍵を表わすベクトルを受け取るために接続された入力と、暗号化されたメッセージを生成するための出力を有する手段、

によって特徴づけられる装置」

クレーム 15 には、公開鍵暗号方式を利用した暗号化・復号化装置（apparatus）が記載されている。クレーム 16，17 には、それぞれ加法型と乗法型ナップザック問題に基づく復号化装置（apparatus）が記載されている。

このように、Merkle-Hellman 特許は、公開鍵暗号を利用した暗号通信およびディジタル署名の基本的な方法と装置のクレームを有している。ただし、暗号通信の数値変換の方法については、ナップザック問題を利用した数値変換の方法がクレーム 7 や 9 で明らかとなっている一方、ディジタル署名方式については、ナップザック問題を利用した数値変換の方法がクレームに記載されていない。もっとも、「実施例の説明」の中には、ナップザック問題を利用した暗号通信やディジタル署名の具体的な数値変換の方法が記載されている。

＜日本の Merkle-Hellman **特許＞**

日本で成立している Merkle-Hellman 特許には 2 つのクレームが記載されている。クレーム 1 は、公開鍵暗号を利用した復号化装置のクレームである。

クレーム１：

「機密保持性のない通信チャンネルを経て受け取った暗号化されたメッセージを解読するための装置であって、

①機密保持さるべきメッセージを公開の暗号作成キーで変換する様な暗号作成変換によって暗号化された暗号メッセージを受け取る様に接続された入力と、機密の暗号解読キーを受け取る様に接続された別の入力と、メッセージを出力する出力とを備えた装置に於いて、

②上記暗号作成変換を逆変換する手段であって、上記暗号メッセージを受け取る様に接続された入力と、上記機密の暗号解読キーを受け取る様に接続された別の入力と、上記暗号メッセージを逆変換したものを出力する出力とを備えた様な手段と、

③メッセージを作り出す手段であって、上記暗号メッセージを逆変換したものを受け取る

様に接続された入力と、メッセージを出力する出力とを有するような手段とを備え、
④上記機密の暗号解読キーは上記公開暗号作成キーから計算によって作り出すことはできず、そして上記暗号作成変換は上記機密の暗号解読キーなしに計算によって逆転することはできないことを特徴とする装置」

このように、クレーム 1 には公開鍵暗号を利用した復号化装置が記載されているが、復号化の数値変換の方法は記載されておらず、「発明の詳細な説明」の中に、ナップザック問題に基づく暗号化と復号化の具体的な数値変換の方法が記載されている。クレーム 2 には、公開鍵暗号を利用した暗号化装置が記載されている。

クレーム 2：

「機密保持性のない通信チャンネルを経て送信さるべきメッセージを暗号化する装置であって、
①機密性を保持すべきメッセージを受け取る様に接続された入力と、公開の暗号作成キーを受け取る様に接続された別の入力と、暗号化されたメッセージを出力する出力とを備えた様な装置に於いて、
②メッセージを受け取り、そしてこのメッセージをそのベクトル表示に変換する手段と、上記公開の暗号作成キーを受け取り、そしてこの公開の暗号作成キーをそのベクトル表示に変換する手段と、上記メッセージのベクトル表示と上記公開の暗号作成キーのベクトル表示とのドット積を計算することによって暗号化メッセージを作り出す手段であって、上記メッセージのベクトル表示を受け取る様に接続された入力と、上記公開の暗号作成キーのベクトル表示を受け取る様に接続された別の入力と、暗号化されたメッセージを出力するための出力とを有する様な手段とを備えたことを特徴とする装置」

このクレーム 2 は、アメリカの Merkle-Hellman 特許のクレーム 13 に対応しており、暗号化における数値変換の方法に関しては「メッセージと暗号作成キーのベクトルのドット積を計算する」とクレームに記載されているだけである。ナップザック問題に基づく Merkle-Hellman 暗号では、暗号鍵がナップザックベクトルに対応するが、このクレームの記載内容からは、暗号作成キーがナップザックベクトルに対応するかどうかは必ずしも明らかではない。

このように、日本の Merkle-Hellman 特許には、公開鍵暗号を利用した暗号化装置と復号化装置のクレームが存在する。

2．ディジタル署名特許

（1）RSA 暗号特許

RSA 暗号は、大きな未知の素数の合成数を素因数分解する問題の困難性に依拠した公開鍵暗号方式の 1 つである。RSA 暗号特許[163]は、この素因数分解問題を応用した初めての暗号特許であり、素因数分解問題を利用した暗号化・復号化の数値変換の方法がクレームや「実施例の

[163] U. S. Patent Number は 4,405,829（出願日 1977 年 12 月 14 日、発効日 1983 年 9 月 20 日）。

説明」[164]に記載されている。

RSA 暗号特許は 40 のクレームから構成されており、クレーム 1～22 が素因数分解問題に基づく基本的な暗号化・復号化処理とディジタル署名のシステム（system）のクレームであり、クレーム 23～32 がクレーム 1～22 のシステムに対応する方法（method）のクレームとなっている。さらに、クレーム 33～36 が、暗号化と復号化の演算にメッセージデータの多項式を利用したシステム（system）のクレームであり、クレーム 37～40 がクレーム 33～36 で記載されたシステムの方法（method）のクレームとなっている。

まず、RSA 暗号方式による暗号化・復号化の方法とディジタル署名の生成・検証の方法が記載されているクレーム 23～32 から説明する。クレーム 23, 24 には RSA 暗号方式における暗号化と復号化の方法が記載されている。

クレーム 23：

「暗号通信を行うための方法であって、素数 p と q に対して n = pq となる合成数 n を生成し、この n に対して 0≦M≦n-1 を満足するディジタルメッセージ M を暗号文 C に変換するというステップによって構成される方法で、(p-1)(q-1)と互いに素な関係にある整数 e に対して、M を C = M^e （mod n）を満たす C に変換することによって表される方法」[165]

このクレーム 23 には、法 n、公開鍵 e や、暗号文 C の作成方法（C=M^e（mod n））等、素因数分解問題に基づく公開鍵暗号の暗号化の具体的な数値変換の方法が記載されているが、復号化については記載されていない。復号化の数値変換の方法は、クレーム 23 の従属項のクレーム 24 に記載されている。

クレーム 24：

「クレーム 23 で表される方法であって、上記暗号文 C を上記メッセージ M に復号化するというもう 1 つのステップによって構成され、その復号化ステップは、(p-1)と(q-1)の最小

---

[164] 本特許の「実施例の説明」には、「このシステムのセキュリティは n の素因数である p と q を求める能力に依存する。p と q を大きな素数として選ぶことによって合成数 n も大きくなり、素因数分解もより困難になる。例えば、コンピュータを利用してよく知られた素因数分解法によって素因数分解を行う場合、例えば 200 bit の合成数を素因数分解するためには、10^9 年のオーダーの時間が必要となる。したがって、実際に利用者が暗号化鍵を公開しても、n の素因数分解には膨大な時間が必要となるため、n の素因数 p と q を他人に知られることはない」と記載されている。

[165] クレーム 23 の原文は、"A method for establishing cryptographic communication comprising the step of: encoding a digital message word signal M to a ciphertext word signal C, where M corresponds to a number representative of a message and 0</= M </= n-1 where n is a composite number of the form n = p･q where p and q are prime numbers, and where C is a number representative of an encoded form of message word M, wherein said encoding step comprises the step of: transforming said message word signal M to said ciphertext word signal C whereby C identical = M<e> (mod n) where e is a number relatively prime to (p-1), (q-1)."である。

公倍数を法としたときに e の逆数となる整数 d に対して、M = C^d (mod n) という変換によって表される方法」[166]

なお、「実施例の説明」の中でも、素因数分解問題を利用した暗号化や復号化の具体的な数値変換の方法が記載されており、前述のナップザック問題を利用した Merkle-Hellman 特許の方式とは異なる数値変換の方法を利用していることが明らかになっている。

以上のクレーム 23、24 には 1 対 1 での暗号通信の方法が記載されているが、クレーム 25 には、k 個のターミナルが存在し、それぞれのターミナル間でメッセージ通信を可能とする方法（method）が記載されている。特に、メッセージの変換ステップの中に、送付するメッセージを自分の暗号鍵で変換することによってメッセージのディジタル署名を作成するサブステップが含まれており、素因数分解問題に基づくディジタル署名の作成方法が記載されている。

クレーム 25：

「各々の暗号鍵 E[i]=（e[i],n[i]）と復号鍵 D[i]=（d[i],n[i]）によって特徴づけられる k 個のターミナルが存在する通信システムにおいてメッセージ M[i]（i = 1,2,...,k）を送受信する方法であり、ターミナル A からターミナル B に送付されるメッセージ M[A]を変換するステップから構成され、上記変換ステップは、M[A]から署名付きメッセージ M[As]を作成する M[As] = M[A]^d[A] (mod n[A]) という変換を行うサブステップによって構成される。ただし、M[i]は 0≦M[i]≦n[i]-1 を満たし、n[i]は n[i]=p[i]q[i]となる合成数であり、p[i]と q[i]は素数であり、e[i]は（p[i]-1）と（q[i]-1）の最小公倍数と互いに素な整数であり、d[i]は（p[i]-1）と（q[i]-1）の最小公倍数を法とした場合に e[i]の逆数となる」[167]

クレーム 25 の従属項のクレーム 26 では、クレーム 25 の方法に署名付きメッセージの送付と検証のステップが加えられた方法（method）が記載されている。

---

[166]クレーム 24 の原文を紹介すると、"The method according to claim 23 comprising the further step of: decoding said ciphertext word signal C to said message word signal M, wherein said decoding step comprises the step: transforming said ciphertext word signal C, whereby: M identical = C<d> (mod n) where d is a multicative inverse of e (mod (lcm((p-1), (q-1))))."である。

[167] クレーム 25 には、"A method for transferring a message signal M[i] in a communications system having k terminals, wherein each terminal is characterized by an encoding key E[i] = (e[i],n[i]) and decoding key D[i]=(e[i],n[i]), where i = 1,2,...,k, and wherein M[i] corresponds to a number representative of message-to-be-transmitted from the i<th> terminal, and 0 </= M[i] </= n[i] - 1, n[i] is a composite number of the form n[i] = p[i]･q[i] p[i] and q[i] are prime numbers, e[i] is relatively prime to lcm(p[i]-1,q[i]-1), d[i] is a multiplicative inverse of e[i] (mod (lcm((p[i]-1),(q[i]-1)))) comprising the step of: encoding a digital message word signal M[A] for transmission from a first terminal (i=A) to a second terminal (i=B), said encoding step including the sub-step of: transforming said message word signal M[A] to a signed message word signal M[As], M[A] corresponding to a number representative of an encoding from of said message word signal M[A], whereby: M[As] identical = M[A]<d> S A (mod n[A])."と記載されている。

クレーム 26：

「クレーム 25 の方法に、署名済みメッセージ M[As]を第二のターミナルに送付し、M[As]を変換してメッセージ M[A]を生成するステップを加えた方法であって、M[As]を変換するステップでは、M[A]=M[As]^e[A]（mod n[A]）という変換が行われる方法」

さらに、クレーム 26 の従属項のクレーム 28 には、クレーム 26 の方法に、①送信者が暗号化された署名付きメッセージを受信者に送付し、②受信者はそれを復号化する、というステップが加えられた方法が記載されており、メッセージの暗号化によって情報の秘匿を実現すると同時に、ディジタル署名を添付することによってその真正性を確保する方法が記載されている。このように、素因数分解問題を利用した暗号化・復号化方法、ディジタル署名の生成・検証方法、そしてこれらを組み合わせて情報の機密性と真正性を同時に確保する方法がクレームに記載されている。これらの方法も「実施例の説明」の中で詳細に説明されており、Merkle-Hellman 特許とは異なる数値変換の方法を利用していることが明らかとなっている。

次に、システムのクレームを説明する。クレーム 1 には暗号通信システムが記載されており、クレーム 24 に対応するシステムのクレームとなっている。

クレーム 1：

「暗号通信システムであって、

A. 通信チャネル、

B. 上記通信チャネルに連結され、送信メッセージ M を暗号化メッセージ C に変換し、上記通信チャネルを使って送信する機能をもつ暗号化手段（means）であって、メッセージ M は 0≦M≦n-1 を満足し、合成数 n は素数 p と q に対して n=pq となり、（p-1）と（q-1）の最小公倍数と互いに素な関係にある e の下で、C=M^e（mod n）を満足する暗号化メッセージ C を生成する手段、

C. 上記通信チャネルに連結され、通信チャネルから受信した暗号化メッセージ C をメッセージ M'に変換する機能をもつ復号化手段であり、C を復号化したメッセージ M'は、（p-1）と（q-1）の最小公倍数を法として e の逆数となる d の下で、M'=C^d（mod n）を満足するように復号化する手段、

により構成されるシステム」

このように、通信チャネル、暗号化手段、復号化手段を有する暗号通信システムが記載され、暗号化と復号化のそれぞれの具体的な数値変換の方法も明らかになっている。クレーム 2 はクレーム 1 の従属項であり、クレーム 1 のシステムに、

①送受信されるデータを受信・保管する第一登録手段、べき乗演算に必要な指数を受信・保管する第二登録手段と、剰余演算に必要な法を受信・保管する第三登録手段によって構成される変換手段、

②メッセージを受信し別の手段に送信する出力手段、

③指数を生成するセレクター手段、

④ 2 通りの指数セレクター機能のうちどちらを選択するかを決定する手段、
⑤剰余演算を行う手段、
という 5 つの手段（means）を加えた暗号通信システムが記載されている。また、クレーム 3 はクレーム 25 に対応するシステムのクレームとなっており、k 個のターミナルによって構成される暗号通信システムが記載されている。

クレーム 37～39 には、クレーム 23～26 に記載されている素因数分解問題を利用した数値変換の方法とは別の数値変換の方法を利用した暗号通信方法とディジタル署名方法が記載されている。まず、クレーム 37 には暗号化の方法が記載されている。

クレーム 37：

「暗号通信を行うための方法であって、合成数 n が存在し、この n に対して 0≦M≦n-1 を満足するディジタルメッセージ M を数字 C によって表される暗号文に変換するステップによって構成され、このステップは、数字 e、a[e]、a[e-1]、…、a0 に対して、M を C = a[e]M^e + a[e-1]M^(e-1) +⋯ + a0 (mod n)となる C に変換するステップで表される方法」[168]

前述のクレーム 23 では暗号文 C は C=M^e mod n という演算によって生成されているが、クレーム 37 では、C = a[e]M^e + a[e-1]M^(e-1) +⋯ + a0 (mod n)という変換が利用されていることがわかる。クレーム 38 はクレーム 37 の従属項となっており、クレーム 37 の暗号化に復号化が加えられた方法が記載されている。この復号化の方法には、①整数 e が λ(n)と互いに素であり[169]、② a[e]=1 かつ a[e-1],a[e-2],…,a0 = 0 である、という 2 つの条件が加えられており、このとき λ(n)を法とする剰余算において e の逆数となる整数 d を計算し、その d を用いて M = C^d (mod n)を計算するという復号化方法が記載されている。これらのクレームで特定されている数値変換の方法は、「実施例の説明」においても詳細に記載されている。

このように、RSA 暗号特許には、素因数分解問題を利用した暗号化・復号化の方法が記載されているクレームが存在するほか、ディジタル署名生成・検証の方法を記載したクレームも存在する。また、RSA 暗号の暗号化・復号化のシステムのクレームや、ディジタル署名生成・検証のシステムのクレームも存在する。これらのクレームでは、暗号化や復号化に利用される数

---

[168] クレーム 37 の原文は、"A method for establishing cryptographic communication comprising the step of: encoding a digital message word signal M to a ciphertext word signal C, where M corresponds to a number representative of a message and 0</= M </= n-1 where n is a composite number and where C corresponds to a number representative of an encoded form of message word M, wherein said encoding step comprises the step of: transforming said message word signal M to said ciphertext word signal C whereby C identical = a[e]M<e> + a[e-1]M<e-1> + ⋯ + a0 (mod n) where e and a[e], a[e-1],…, a0 are numbers."である。

[169] λ(n)はオイラー関数である。オイラー関数とは、自然数 n に対して、1,2,3,...,n の中で n と互いに素な数の個数を表す関数である。このオイラー関数は、n と互いに素な任意の整数 S に対して S^λ(n) = 1 (mod n)を満足する。この関係はオイラーの定理と呼ばれている。

値変換の方法が特定されている。

（2）ESIGN 署名特許

ESIGN 署名方式は、RSA 暗号方式における低処理速度の問題を解決した高速ディジタル署名方式であり[170]、素因数分解問題の困難性と合同多項不等式問題の困難性を利用した方式である。ESIGN 署名特許[171]は、アメリカのほかに日本でも成立している。

＜アメリカの ESIGN 署名特許＞

アメリカの ESIGN 署名特許は 25 のクレームを有しており、まず ESIGN 署名方式における署名生成・検証方法（method）がクレーム 1, 2 と 3 に記載されている。最も基本的な形の署名生成・検証方法がクレーム 3 に記載されており、ESIGN 署名方式が素因数分解問題に基づく方式であることが明らかであるが、その署名を生成するために用いられる関数は任意関数となっており具体的な関数形はクレーム 3 には記載されていない。

クレーム 3：

「署名付きデータの送信方法であって、

①送信者が、乱数 x、送信データ m、秘密鍵である素数 p と q、$n = p^2 \cdot q$ となる公開鍵 n、2 以上の整数 l、任意関数 g(m)、任意関数 $f_i(m)$に対して $f(x,m) = \sum_{i=0}^{l} f_i(m)x^i$ となる関数 f(x,m)の下で、g(m)と f(x,m)を基に整数 W を計算し、

②送信者が、f(x,m)の x に関する偏導関数 f '(x,m)に対して y = W/f '(x,m)となる y によって S=x+ypq となる署名 S を生成し、

③送信者は、送信データ m と署名 S を受信者に送信し、

④受信者は、署名 S を用いて f(S,m)を計算し、f(S,m) (mod n)と g(m)の計算結果によって m と S の正当性をチェックする、

というステップによって構成される方法」[172]

---

[170] 「発明の効果」には、「本発明によって、署名生成・検証速度は RSA 署名方式に比べて約 100 倍以上高速になる」と記載されている。

[171] U. S. Patent Number は 4,625,076（出願日 1985 年 3 月 11 日、発効日 1986 年 11 月 25 日）、日本国特許公告番号は平 5-86699（出願日 1985 年 3 月 4 日、公告日 1993 年 12 月 14 日）。

[172] クレーム 3 には、"A signed document transmission method comprising the steps of determining, on the transmitting side, an integer value W based on g(m) and a congruent polynomial f(x,m) (mod n), where x is an integer random number, m is a document to be transmitted, n is a public key given by $n=p^2q$, p and q are secret keys of prime numbers, g(m) is an arbitrary function with respect to m, f(x,m) is a polynimial given to $f(x,m)=\sum_{i=0}^{l} f_i(m)x^i$, $f_i(m)$ is an arbitrary function with respect to m, and l is an integer equal to or greater than 2; generating a signature S as given by S=x+ypq, where y is a congruent division of W and a differentiation f '(x,m) of f(x,m) with respect to x; transmitting the document m and the signature S; obtaining, on the receiving side, a congruent polynomial f(S,m) (mod n) using the signature S in place of x in the polynomial f(x,m), and the document m and the public key n; and

このように、このクレーム 3 にはデータ通信方法が記載されているほか、剰余算の法 n が $n = p^2q$ という 2 種類の素数の積となっており、素因数分解問題の困難性を利用していることが明らかになっている。なお、「発明の実施例」の中で、任意関数の具体的な数式、署名検証の方法のほか、高速処理が可能な理由等が記載されており、4 つの署名方式が実施例として開示されている。クレーム 1 と 2 には、クレーム 3 で具体的に示されていない署名検証式が数式でそれぞれ特定されており、クレーム 1 と 2 の署名検証式はそれぞれ異なっている[173]。また、クレーム 1 の従属項のクレーム 4 には、クレーム 1 の署名方法に加えて、公開鍵（δ, ε）のうち δ の範囲が特定されている署名付きデータの通信システム（system）が記載されているほか、クレーム 4 の従属項のクレーム 5 には、ε の範囲が特定されている通信システム（system）が記載されている。クレーム 8～12 には、署名検証式が具体的な数式で特定されている署名付きデータ通信方法（method）が記載されている。

また、クレーム 23 には、署名データの特定のビットを調べることにより署名の検証を行う方法（method）が記載されており、「クレーム 1 と 2 の署名付きデータの送受信方法であって、公開鍵 δ が 2 のべき乗の形となり、受信者は、合同多項式によって得られた数値のバイナリーデータにおいて予め決められたビット位置が 0 であるかどうかを確認するステップを含む方法」と記載されている。クレーム 24 にはクレーム 2 に対応するシステム（system）が記載され、クレーム 25 にはクレーム 1 に対応するシステム（system）が記載されている。

このように、アメリカの ESIGN 署名特許には、ESIGN 署名方式の署名生成・検証方法のクレームのほか、署名生成・検証を行うシステムのクレームも存在する。なお、署名生成・検証の数値変換の方法がクレームに記載されている。

＜日本の ESIGN 署名特許＞

日本の ESIGN 署名特許は 4 つの署名文書通信方式のクレームを有している。クレーム 1 には、署名の作成方法と検証方法が記載されている。

クレーム 1：

「送信側で素数 p、q（p＞q）を秘密情報として用意し、

①これら素数 p、q を用いて、公開情報 $n=p^2q$、δ（$1 \leqq \delta \leqq \theta(n^{2/3})$、θ(B)は B のオーダーを意味する）、$\gamma = \theta(n/T)$（T は 10^{10}～10^{30} 程度の値）を作り、

②これら公開情報 n、δ、γ を送信者 D の識別番号（ID）と共に公開簿に登録しておき、

③乱数 X（$1 \leqq X \leqq pq-1$、X は n と互いに素な整数）を生成し、

---

verifying the validity of the document m and the signature S based on the calculation results of F(S,m) (mod n) and g(m)"と記載されている。

[173] クレーム 1 の署名検証式は、$\llcorner (g(m)+\{f(S,m) \pmod n\}) / \delta \lrcorner = 0 \pmod \varepsilon$ と記載されている（なお、$\llcorner X \lrcorner$ は X の中で最小の整数を表わす）。一方、クレーム 2 の署名検証式は、$-g(m) \leqq f(S,m) \pmod n \leqq -g(m)+\delta$ と記載されている。

④その乱数 X に対して f(X）(mod n)（ｆ(X)=Σ $^{I}_{i=0}$ g_i・X^i（i=1,2,...,I）、0＜g_i≦n-1、g_i は整数、I≧3）を演算し、
⑤その演算結果と送信すべき文書 m との差 Z=m- (f(X) (mod n))を求め、
⑥その Z を pq で割算し、
⑦その割算結果を切り上げて W=「Z/(pq)」を求め、
⑧上記乱数 X に対し、f ‘(X) mod p (f ‘(X)= Σ $^{I}_{i=0}$ i・g_i・X^{i-1})を演算し、
⑨その演算結果で上記 W を割算し、
⑩この演算結果 y と上記 pq とを乗算し、
⑪これに乱数 X を加算して署名 S とし、
⑫この署名 S と上記文書 m と、上記識別番号 ID とを送信し、
⑬受信側で受信した上記識別番号 ID を利用してその公開情報 n、δ、γを上記公開簿より求め、
⑭受信した上記署名 S に対し、f(S) (mod n)を演算し、
⑮その演算結果と、上記文書 m と上記δとを用いて m≦f(S) (mod n)＜m+δが成立するかどうかを検証し、
⑯上記 n と上記γと、上記署名 S とを用いてγ≦S≦n-γが成立するかどうかを検証し、
⑰上記両検証が共に成立した場合に、上記受領したm、S は上記公開簿に（n、δ、γ）を登録した者によって確かに作成されたものであるとする署名文書通信方式」

このように、クレーム 1 には 3 次以上の合同多項式（I が 3 以上となっており、f(X)は X の 3 次以上の多項式となっている）に基づく署名の作成・検証の方法が記載されているほか、署名作成・検証の数値変換の方法が明らかになっており、この署名方式が素因数分解問題に基づくことも明らかである。クレーム 1 の署名検証には不等式が利用されているが、クレーム 2 では、署名データの特定のビットを調べることによって署名の正当性を検証する方法が記載されている。また、クレーム 3 と 4 には、2 次以上の合同多項式（I が 2 以上となっており、f(X)は X の 2 次以上の多項式となっている）に基づく署名の作成・検証方法が記載されている。なお、「発明の詳細な説明」には、アメリカの特許同様に具体的な数値変換の方法が記載されている。これらの 4 つのクレームに対応するクレームは、アメリカの特許には存在しない。

（3）Fiat-Shamir 署名特許

Fiat-Shamir 署名方式は、零知識対話証明を利用した Fiat-Shamir 認証と呼ばれる本人認証方式を署名用に改良した方式である。Fiat-Shamir 署名方式の安全性は素因数分解問題に依拠しているが、べき乗剰余算の次数を小さくすることで RSA 暗号を利用したディジタル署名方式よりも高速処理が実現可能である。しかし、署名生成の度に乱数を生成する必要があるほか、署名長が長くなるという欠点をもっている。

Fiat-Shamir 署名特許[174]は 42 のクレームから構成されている。まずクレーム 1 において、ユーザー認証とメッセージ認証に必要な ID 情報の作成方法（method）が記載されている。

[174] U. S. Patent Number は 4,748,668（出願日 1986 年 7 月 9 日、発効日 1988 年 5 月 31 日）。

クレーム 1：

「他人が改ざんすることが不可能な ID 情報を作成する方法であって、

（a）少なくとも 2 つの素数の積である法 n を選択し、

（b）任意の記号をある数値に対応させる疑似乱数生成関数 f を選択し、

（c）各主体に一意な情報を含む記号 I を準備し、

（d）$v_j = f(I,j)$ が法 n に対し平方剰余 s_j をもつように k 個の整数 j を選び、

（e）v_j の平方剰余 s_j を計算し、

（f）I、k、s_j と j を読み出し可能な媒体に記録する[175]、

というステップによって構成される方法」

この個人情報を利用したユーザー認証方法（method）がクレーム 4 に記載されているほか、クレーム 10 にはメッセージ認証方法（method）が記載されている。

クレーム 4：

「クレーム 1 の ID 情報を利用する方法であって、

（a）被認証者はクレーム 1 の ID 情報を、法 n と疑似乱数生成関数 f を有する認証者に送付し、

（b）被認証者は I と j を認証者に送付し、

（c）認証者は $v_j = f(I,j)$を計算し、

（d）被認証者は乱数 $r_i \in (0,n)$を生成し、

（e）被認証者は $x_i = r_i^2$（mod n）を計算して x_i を認証者に送付し、

（f）認証者は乱数ベクトル $e_{i1},\ldots,e_{ik}$ を予め生成しておいたベクトル集合の中から選び、それを被認証者に送付し、

（g）被認証者は $y_i = r_i \prod_{eij=1} s_j$ （mod n）を計算して、y_i を認証者に送付し、

（h）認証者は、$x_i = y_i^2 \prod_{eij=1} v_j$ （mod n）が成立するかどうかをチェックし、

（i）上記（d）から（h）を t 回（t は 1 以上の値）繰り返される、

というステップから構成される方法」

クレーム 10：

「被認証者と認証者との間で交換されるメッセージ m に署名する方法であって、

（a）被認証者は乱数 $r_1,\ldots,r_t \in (0,n)$を生成し、

（b）被認証者は $x_i = r_i^2$ （mod n）を計算して x_i を認証者に送付し、

（c）被認証者は $f(m,x_1,\ldots,x_t)$ を計算して、その計算結果のうちの kt ビットを e_{ij} として設定し、

（d）被認証者は $y_i = r_i \prod_{eij=1} s_j$ （mod n）を計算し、

[175] ID 等の情報を記録するステップにおいて、読み出し可能な媒体（retrievable medium）を利用する（（f） recording on a retrievable medium of and identifier I, k, sj and related indices j）と記載されている。

（e）被認証者は I、j、m、e_{ij} と y_i を認証者に送付し、
（f）認証者は $v_j = f(I,j)$を計算し、
（g）認証者は $z_i = y_i^2 \prod_{eij=1} v_j$ （mod n）を計算し、
（h）認証者は $f(m,z_1,\ldots,z_t)$を計算して、その計算結果のうちの kt ビットを取り出して e_{ij} と一致するかどうかをチェックする、という方法」

これら 2 つのクレームから、これらの認証方法に関する数値変換の方法が明らかとなっている。関数 f の具体的な関数形については「発明の実施例」の中で開示されている。

続いて、上記の認証方法を実現する装置の主なクレームを説明する。クレーム 15 には、クレーム 1 に対応する ID 情報作成装置（apparatus）が記載されているほか、クレーム 18 にはクレーム 4 に対応するユーザー認証装置（apparatus）が、またクレーム 24 にはクレーム 10 に対応するメッセージ認証装置（apparatus）が記載されている。

このように、Fiat-Shamir 署名特許には、素因数分解問題に基づくユーザー認証とメッセージ認証の方法と装置のクレームが存在する。Fiat-Shamir 署名方式の数値変換の方法もクレームに記載されているほか、「発明の実施例」の中にも詳細に記載されている。

（4）Schnorr 署名特許

Schnorr 署名方式は離散対数問題を利用した署名方式であり、署名長が法の 2 倍になる ElGamal 署名方式に対して、「離散対数問題において、法 p に対して p-1 の素因数 q を法とする有限体上で署名を構成する」という署名長圧縮技術によって、署名長を 2p から 2q に短縮することを可能にした方式である。Schnorr 署名特許では、Schnorr 署名方式による本人確認とディジタル署名の方法が記載されているが、Schnorr 署名方式を IC カードによって実装することが想定されている点が特徴である[176]。

Schnorr 署名特許[177]は 11 のクレームを有しており、すべてのクレームが方法（method）のクレームとなっている。まず、クレーム 1 には、IC カードと署名確認装置との間で 4 回の情報交換を行う相互認証方法が記載されている。

クレーム 1：

「IC カードと共に機能し、署名確認装置の公開鍵に対応する秘密鍵によって暗号化され、カード発行センターによって各 IC カードに埋め込まれている ID データを利用するデータ

---

[176] 本特許の「実施例の説明」には、本特許の中で想定されているチップカードについて、「チップカードを利用したデータ交換システムにおいては、各ユーザー固有のチップカードが、政府機関やクレジットカード会社等の発行機関（classification centers）から発行される。（中略）各ユーザーの ID 情報、公開鍵 v、秘密鍵 s や素数 p が、各チップカードが発行される前にカードに保管される。」と記載されている。この内容から、本特許のチップカードには、耐タンパー性のある IC チップを有する IC カードのような装置が想定されていると考えられる。

[177] U. S. Patent Number は 4,995,082（出願日 1990 年 2 月 23 日、発効日 1991 年 2 月 19 日）。

交換システムにおいて、署名確認装置との間で相互認証を行うための方法であり、
① IC カードにおいて、カード発行センターの署名付きの ID データを、IC カードとの間でデータ交換を行うように接続された署名確認装置に送付するステップと、
②署名確認装置において、公開されているリストやセンターの署名を参照することによって、変換された ID データの正当性をチェックするステップと、
③ IC カードにおいて、公開されている素数 p と乱数 $r \in (1,...,p-1)$から、$x=2^r \pmod p$という式によって x を作成するステップと、
④ IC カードにおいて、署名確認装置に x を送付するステップと、
⑤署名確認装置において、ランダムなビット列 $e=(e_{1,1},...,e_{t,k}) \in \{0,1\}^{kt}$ を IC カードに送付するステップと、
⑥ IC カードにおいて、ランダムなビット列 e と IC カードに記録されている秘密鍵 s_j との積を計算し、乱数 r を加え、$y=r+(\Sigma\ ^{k}_{j=1}\ s_j)(\Sigma\ ^{t}_{i=1}\ e_{i,j}2^{i-1}) \pmod{p-1}$という式から y を生成するステップと、
⑦ IC カードにおいて、y を署名確認装置に送付するステップと、
⑧署名確認装置において、y から、$x' = (2^y)(\Sigma\ ^{k}_{j=1}\ v_j)(\Sigma\ ^{t}_{i=1}\ e_{i,j}2^{i-1}) \pmod p$という演算により x'を生成するステップと、
⑨署名確認装置において、x と x'を比較して IC カードの確認を行うステップ、
によって構成される方法」

このように、クレーム 1 から、この認証方式が離散対数問題に基づいていることが明らかになっている。クレーム 1 の従属項のクレーム 2 には、クレーム 1 の相互認証方法に加えて、メッセージの署名生成方法が記載されているほか、クレーム１の従属項のクレーム３には、クレーム 2 とは別の署名生成方法が記載されている。

クレーム 2：
「クレーム 1 に記載されている方法に関してメッセージの署名を生成する方法であり、
① 1 と p-1 の間の乱数 r を生成し、$x=2^r \pmod p$によって x を計算するステップと、
② x とメッセージ m を変数とする公開されたハッシュ関数 h によって、$e=h(x,m) \in \{0,1\}^{kt}$ となるランダムなビット列 e を計算するステップと、
③乱数 r、IC カードに保管されている秘密鍵 s_j とランダムビット列 e を使って、$y=r+(\Sigma\ ^{k}_{j=1}\ s_j)(\Sigma\ ^{t}_{i=1}\ e_{i,j}2^{i-1}) \pmod{p-1}$という式によって y を計算するステップと、
④メッセージ m と署名（x,y）を、IC カードと交信している署名確認装置に送信するステップ、によって構成される方法」

クレーム 3：
「クレーム 1 に記載されているデータ交換システムにおいて交換されるメッセージの短縮化された署名を生成する方法であって、
① IC カードにおいて、1 と p-1 の間の乱数 r を生成し、$x=2^r \pmod p$によって x を計算するステップと、
② IC カードにおいて、x とメッセージ m を変数とする関数によって、$e=h(x,m) \in \{0,1\}^{kt}$

となるランダムなビット列 e を計算するステップと、

③ IC カードにおいて、乱数 r、IC カードに保管されている秘密鍵 s_j とランダムビット列 e を使って、$y=r+(\Sigma\ ^{k}_{j=1} s_j)(\Sigma\ ^{t}_{i=1} e_{i,j}2^{i-1})(\mathrm{mod}\ p-1)$という式によって y を計算するステップと、

④ IC カードから、メッセージ m と署名（e,y）を、IC カードと交信している署名確認装置に送信するステップ、によって構成される方法」

クレーム 2 とクレーム 3 に記載されている署名作成方法は、その数値変換の方法からいずれも離散対数問題に基づく方式であることが明らかであるが、クレーム 2 の署名が（x,y）となる一方、クレーム 3 の署名は（e,y）となっている点が異なっている[178]。クレーム 2 と 3 の署名作成方法に対応する署名検証方法が、それぞれクレーム 10 と 11 に記載されている。

クレーム 10：

「クレーム 2 の方法によって署名されたメッセージ m を受け取る署名確認装置において署名（x,y）の正当性を検証する方法であって、

①メッセージ m と x を用いて $e=h(x,m)\in\{0,1\}^{kt}$ によってランダムなビット列 e を計算するステップと、

②ランダムビット列 e、公開鍵 v と y から、$x' = (2^{y})(\Sigma\ ^{k}_{j=1} v_j)(\Sigma\ ^{t}_{i=1} e_{i,j}2^{i-1})(\mathrm{mod}\ p)$を計算するステップと、

③計算した x'と署名の一部の x を比較するステップ、によって構成される方法」

クレーム 11：

「クレーム 3 の方法に対応する署名の検証方法であって、署名付きメッセージ m を受信する署名確認装置において行われ、

①送信されたメッセージ m と署名（e,y）を用いて、$x' = (2^{y})(\Sigma\ ^{k}_{j=1} v_j)(\Sigma\ ^{t}_{i=1} e_{i,j}2^{i-1})(\mathrm{mod}\ p)$という式によって x'を計算するステップと、

②署名の一部である e と h(x',m)が一致するかどうか確認するステップ、によって構成される方法」

これらのクレームには署名作成・検証の具体的な数値変換の方法が記載されているが、「実施例の説明」においても記載されており、t や k の設定等に関しても詳細に記載されている。

その他、クレーム 3 の従属項のクレーム 4 には、IC カード内での乱数 r や x の生成方法が記載されている。

---

[178]（x,y）という署名の代わりに（e,y）を用いることについて、「実施例の説明」には、「少なくとも 512 bit 程度となる x と y を署名とする方法をベースに、この署名長を短縮する方法がいくつか考えられる。１つの方法は、x の代わりに 72 bit 程度のハッシュ値 e=h(x,m)を利用し、e と y で署名を構成する方法である。」と記載されている。

クレーム 4：

「クレーム 3 に記載されている方法であって、

① IC カード内で乱数 r と x を生成し、その r と x を一組にして IC カードに記録するステップと、

②相互認証を行う場合に保管されている乱数の組（r_v,x_v）を利用し、利用後は、r_v を IC カード内に保管されている他の r によって変換することで別の値に変えるステップと、

③ x についても乱数によって変換し、新しい r とペアで IC カードに再び保管するステップ、によって構成される方法」

なお、クレーム 4 の従属項のクレーム 5 には、クレーム 4 の②と③のステップに記載されている r と x の新しいペアの生成方法が数式によって記載されている。クレーム 6 には、べき乗剰余算に必要な法、公開鍵と秘密鍵の関係や、署名生成の際に（p-1）の素因数 q を位数とする原始根αを利用する方法が記載されている。このαを利用してべき乗剰余算を行うことにより、署名長を短縮することが可能となる[179]。

クレーム 6：

「クレーム 5 に記載されている方法に、素数の法 p を(p-1)が素数 q によって割り切れるように選ぶステップ、α^q = 1 (mod p)かつ α ≠ 1 (mod p)を満足するように原始根αを選ぶステップ、さらに鍵の要素である s_j と v_j が $v_j = \alpha^{(-s_j)} \pmod p$という関係を満足するように、q を法として y、r と s_j を計算するステップを加えた方法」

このように、Schnorr 署名特許は、IC カードを利用した相互認証方法や署名生成・検証方法のクレームを有しており、特に 2 種類の署名生成・検証の方法（クレーム 2 と 10、クレーム 3 と 11）が記載されている。また、相互認証や署名生成・検証の数値変換の方法に離散対数問題を利用している点もクレームから明らかとなっているほか、法 q を利用することで署名長を圧縮する方法についてもクレーム 6 に記載されている。

（5）DSA 署名特許

DSA 署名方式は、ElGamal 署名の改良方式であり、ElGamal 署名と同様の署名生成・検証式に Schnorr 署名の署名圧縮技法を持ち込んだ方式といわれている。

DSA 署名特許[180]は 44 のクレームを有している。まず、クレーム 1 にディジタル署名の生成方法（method）が記載されている。

---

[179] クレーム 6 の原文は、"The method of claim 5, and further defined as: selecting the prime number modulus p such that the number （p-1） is divisible by a prime number q and by such a selection of the base α of a discrete logarithm that α^q = 1 （mod p）, α ≠ 1 （mod p） holds true; and calculating discrete logarithms y, r, s_j modulo q such that α key components s_j and v_j are in the relationship $v_j = \alpha^{(-s_j)} \pmod p$."である。

[180] U. S. Patent Number は 5,231,668（出願日 1991 年 7 月 26 日、発効日 1993 年 7 月 27 日）。

クレーム1：

「情報をやり取りするシステムにおいて、メッセージ m のディジタル署名(r,s)を生成する方法であって、

（a）各メッセージ m に一意に決まる秘密値 k を生成し、

（b）公開値 g を生成し、

（c）素数 p と(p-1)の素因数 q を用いて、r=(g^k mod p) mod q となる r を計算し、

（d）メッセージ m をハッシュ関数 H によって変換してハッシュ値 H(m)を生成し、

（e）上記 s を m の関数として s=f(H(m))によって生成し、

（f）上記の r と s を用いてデジタル署名(r,s)を作成する、

というステップによって構成される方法」[181]

上記の（c）のステップから、この署名方式が離散対数問題を利用する方式であることと、署名生成時に（p-1）の素因数 q を法として利用することによって署名を圧縮する方式であることが明らかとなっている。このクレームには、k や g の数値の決定方法やハッシュ関数 H や関数 f の具体的な形が特定されていないが、「発明の実施例」の中にこれらの数値変換の方法が記載されているほか、署名長の圧縮方法も記載されている[182]。

---

[181] クレーム 1 の原文は、"A method for generating a digital signature （r,s） of a message m in a system wherein information is transmitted and received by users of said system, comprising the steps of: （a） providing a secret value k unique to said message m; （b） providing a public value g; （c） calculating said value r proceeding from a prime modulus p and a value q selected to be a prime divisor of p-1 according to the rule r = （g^k mod p） mod q; （d） applying a hashing transform H only to said message m to generate a transformed message H（m）; （e） calculating said value s according to the rule s = f（H（m）） where said value s is a function of m only by way of said transformed message H（m）; and,（f） generating a signal representative of said digital signature（r,s） in accordance with said value r and said value s and transmitting said generated signal. "である。

[182] 本特許の「発明の実施例」には、「ディジタル署名を生成する方法の中に、署名の一部 r を求める g^k mod p という式がある。（中略）しかし、この署名はかなり長い。したがって、r = (g^k mod p) mod q という mod q の演算によって署名長を 160 bit まで圧縮する。（中略）p は素数の法であり、2^{511} から 2^{512} の間の値となる。また、q は（p-1）の素因数であり、2^{156} から 2^{160} の間の値である」と記載されており、Schnorr 署名方式と同様の方式によって署名の一部 r を 512 bit から 160 bit に圧縮し、処理速度を向上させることが可能であると記載されている。また、利用可能なハッシュ関数の例として、MD4 が例示されている。なお、本特許の「発明の背景」には、本特許の署名方式が ElGamal 署名と Schnorr 署名を利用していることを示唆する内容が記載されており、「Schnorr の特許（U.S. Patent No. 4,995,082）は、ElGamal の方式よりも情報の通信やその内容の真正性確認を高速に処理できるシステムを提供しており、そのシステムは非常に高速でオンラインでの署名を行う機能を有している。しかし、署名の強度等 ElGamal の方式の長所のいくつかは、Schnorr の方式には適用されていない。そのため、ElGamal の方式との互換性を維持した上で、オンライン上での署名、通信、真正性確認を Schnorr のシステムに匹敵する処理速度で実行するシステムを実現することが望ましい」と記載されている。

クレーム 1 の従属項クレーム 2～7 には、秘密鍵の生成、メッセージのハッシュ化や署名データの生成方法（method）が記載されている。クレーム 1 の従属項クレーム 3 には、クレーム 1 の方法に、g を位数 q の原始根となるように選ぶステップが加えられた方法が記載されている。

クレーム 3：

「クレーム 1 に記載されているデジタル署名（r,s）を生成する方法であって、（b）のステップが、h^((p-1)/q)が法 p に関して 1 と合同にならないようなゼロでない整数 h の下で、g=h^((p-1)/q) mod p を満足する g を計算するステップを有する方法」

このように計算される g をべき乗剰余算に利用することにより、署名の一部である r を q 以下にすることが可能となる。クレーム 5 にはディジタル署名の一部である s の生成式 s=k^(-1)*(H(m)+xr) mod q が記載されているほか、クレーム 8 にはディジタル署名の検証方法（method）が記載されている。

クレーム 8：

「クレーム 7 に記載されているディジタル署名(r,s)を生成するための方法に、
(g)受信したディジタル署名(r,s)を含む送信された署名付きメッセージを受信するステップと、
（h）上記受信したディジタル署名(r,s)を検証するステップを付け加えたディジタル署名検証方法」

クレーム 9～14 では、クレーム 8 の（g）と（h）のステップがより詳細に記載されており、例えばクレーム 10 には、署名検証式 r = ((g^(H(m)/s))*(y^(r/s)) mod p) mod q が記載されている（ただし、y=g^x mod p）。なお、クレーム 5 と 10 に記載されている署名生成・検証式は、メッセージ m の代わりにハッシュ値 H（m）が利用されている点と、法（p-1）の代わりに法 q が利用されている点等を除けば、ElGamal 署名方式の署名生成・検証式と一致する[183]。

クレーム 15～20 は、クレーム 1～6 に対応するシステム（system）のクレームとなっているほか、クレーム 21～27 はクレーム 8～14 に対応するシステム（system）のクレームとなっている。クレーム 28 では、署名生成方法（クレーム 1）と署名検証方法（クレーム 8）が一連のステップとして含まれる署名生成・検証方法（method）が記載されており、ディジタル署

[183] ElGamal 署名方式の署名生成･検証式を DSA 署名特許のクレームで利用されている記号を使って書き直すと、署名生成式は① s = k^(-1)*(m-xr) mod (p-1) と② r = g^k mod p となり、署名検証式は③ g^m = (r^s)*(y^r) mod p となる。この ElGamal 署名の署名生成式①において、m を-H(m)に、s を-s に、そして法(p-1)を q に置きかえるとともに、署名生成式②において mod q を加えることで、DSA 署名生成式に一致する。また、署名検証式についても、式③を変形すると r=g^(m/s)*y^(-r/s) mod p となり、m を-H(m)に、s を-s に、そして mod q を加えることによって DSA 署名検証式と一致する。

名の一部である r の生成方法が若干異なっている点を除けば、クレーム 1 とクレーム 8 を合わせた署名生成方法と一致する。

クレーム 28：

「あるシステムにおいてメッセージ m のディジタル署名(r,s)を生成・検証するための方法であって、

（a）上記メッセージ m に対して一意に決定する秘密値 k を生成し、

（b）公開値 g を生成し、

（c）素数の法 p と上記メッセージ m に依存しない圧縮関数 F によって、r = F（g^k （mod p））という演算から上記 r を計算し、

（d）上記メッセージ m と上記ディジタル署名(r,s)によって構成される署名付きメッセージを受信し、

（e）g^k mod p を生成し、

（f）上記 g^k mod p を圧縮関数 F で変換した値が、上記受信した r に等しいかどうかを検証し、

（g）上記署名(r,s)が正当な署名かどうかを検証し、

（h）ステップ（g）の検証結果によって正当性確認済シグナルを生成し、上記シグナルを送信する、というステップから構成される方法」

上記のクレーム 28 をクレーム 1 と比較すると、① r を生成するステップ（c）において必ずしも（p-1）の素因数 q を法として剰余算を行うとは記載されておらず、圧縮関数 F という一般的な形で記載されている、② s の生成ステップが記載されていない、という 2 点が異なっていることがわかる。クレーム 44 は、クレーム 28 に対応するシステム（system）のクレームとなっている。

このように、DSA 署名特許には、DSA 署名方式の署名生成・検証方法が記載されているクレームが存在する。また、署名生成・検証に利用される数値変換の方法が記載されているクレーム 5 と 10 から、DSA 署名特許が ElGamal 署名とほぼ同様の署名生成・検証方法を採用していることが明らかなほか、（p-1）の素因数 q を法とする署名圧縮方法を採用していることも明らかとなっている。

3．ハッシュ関数特許

（1）MDC-2/MDC-4 ハッシュ関数特許

MDC-2 と MDC-4 は、ブロック暗号を利用したハッシュ関数であり、ともに入力データブロックの 2 倍のブロック長となるハッシュ値を生成する方式である。MDC-2 では、1 回の変換が 2 系列の並列処理によって構成されており、それぞれの系列が予め決められた回数だけブロック暗号を利用した変換を実行することでハッシュ値を生成する仕組みとなっている。また、MDC-4 では、1 回の変換が MDC-2 の 2 回の変換に対応しており、この変換を予め決められた回数だけ実行することによってハッシュ値を生成する仕組みになっている。

MDC-2/MDC-4 ハッシュ関数特許[184]は 21 のクレームを有しており、クレーム 1〜11 と 14〜21 が方法（method）のクレーム、クレーム 12 が装置（apparatus）のクレーム、そしてクレーム 13 がネットワーク（network）のクレームとなっている。

まず、方法のクレームから説明する。クレーム 1 には、データを N bit のブロックに分割して 2N bit のデータ変更検出コード（modification detection code、以下、MDC）を生成する方法が記載されている。

クレーム 1：

「複数の N bit のブロックに分割されたデータから生成される 2N bit の MDC を生成する方法であって、

①最初のサイクルにおいて、上記データブロックの最初のブロックから、第一 N bit コピーと第二 N bit コピーを生成し、

②上記第一 N bit コピーを、第一 N bit 暗号鍵を入力する入力を備え、N bit の出力データを生成する第一一方向暗号化手段に入力し、

③上記の第一出力データのうち、第一フィールドを第一出力記録手段に記録し、第二フィールドを第二出力記録手段に記録し、

④上記第二 N bit コピーを、第二 N bit 暗号鍵を入力する入力を備え、第二 N bit 出力データを生成する第二一方向暗号化手段に入力し、

⑤上記第二 N bit 出力データのうち、第一フィールドを第二出力記録手段の第一フィールドに記録し、第二フィールドを第一出力記録手段の第二フィールドに記録し、

⑥ 2 番目のデータブロックをコピーし、第一コピーを次のサイクルにおける第一一方向暗号化手段に入力し、N bit 鍵として第一出力記録手段に保管されている上記数値を第一一方向暗号化手段の第一鍵入力に入力し、第二コピーを次のサイクルにおける第二一方向暗号化手段に入力し、N bit 鍵として第二出力記録手段に保管されている上記数値を第二一方向暗号化手段の第一鍵入力に入力し、次の第一結果データを生成して上記第一出力記録手段に出力し、次の第二結果データを生成して上記第二出力記録手段に出力し、

⑦すべてのデータブロックに対して処理を続け、最後の第一結果データが第一出力記録手段に出力され、最後の第二結果データが第二出力記録手段に出力され、

⑧第一および第二出力記録手段にそれぞれ記録された第一結果データと第二結果データを結合し、2N bit の MDC を生成する、というステップによって構成される方法」

このように、クレーム 1 に記載されている方法が、暗号化処理を 2 つ並列に接続し、データブロックの 2 倍の bit 長を有する MDC を生成する方法であることが明らかになっている。2 つの一方向暗号化手段で利用される数値変換の方法はクレームには記載されていないが、「発明の実施例」の中で利用が想定されるものとして DES 暗号が例示されている。このクレーム 1 の MDC 生成方法は、MDC-2 のハッシュ値生成方法に対応している。このクレーム 1 に対応する装置のクレームが、クレーム 12 である。

[184] U. S. Patent Number は 4,908,861（出願日 1987 年 8 月 28 日、発効日 1990 年 3 月 13 日）。

クレーム 2〜11 はクレーム 1 の従属項となっており、それぞれ出力ブロックの各フィールドの bit 長が異なる場合の MDC 生成方法（クレーム 2）、bit 長が同一の場合の MDC 生成方法（クレーム 4）、また 2 つの暗号鍵の初期値が常に同じ場合の MDC 生成方法（クレーム 7）、暗号鍵の初期値が可変的な場合の MDC 生成方法（クレーム 8）が記載されている。

クレーム 15〜21 には、MDC-4 ハッシュ関数による MDC 生成方法が記載されている。

クレーム 15：

「複数の N bit のデータブロックによって構成されるデータに対応する、2N bit の MDC を生成する方法であって、

①第一暗号鍵を使って第一データブロックを暗号化して第一 N bit 結果を出力し、

②第二暗号鍵を使って上記第一データブロックを暗号化して第二 N bit 結果を出力し、

③第一 N bit 結果を鍵として第二暗号鍵を暗号化して第三 N bit 結果を出力し、

④第二 N bit 結果を鍵として第一暗号鍵を暗号化して第四 N bit 結果を出力し、

⑤第三 N bit 結果を次の第一鍵としてフィードバックするとともに、第四 N bit 結果を次の第二鍵としてフィードバックし、

⑥上記のステップを最後の入力ブロックまでくり返し、最後の第三 N bit 結果と最後の第四 N bit 結果を生成し、

⑦最後の第三 N ビット結果と最後の第四 N ビット結果を結合して、2N ビットの MDC を作成する、というステップから構成される方法」

クレーム 16 には、クレーム 15 の方法に、暗号化された各 N bit 結果の bit を入れ替えるステップが加えられた MDC 生成方法が記載されており、MDC-2 のハッシュ値生成方法に対応している。

ネットワークのクレームであるクレーム 13 には、MDC 生成機能を有し、通信データの真正性を確認することが可能な 2 つのコンピュータを含むネットワークが記載されている。

クレーム 13：

「暗号化機能を有する相互接続された第一および第二コンピュータを含むネットワークであって、

①第一の安全ではない通信チャネルと、第一および第二コンピュータと相互に接続する第二の安全な通信チャネルと、

②情報 I を第一 MDC に変換し、情報 I を第一通信チャネルによって、また第一 MDC を第二通信チャネルによって相手に送信する機能を有しており、第一コンピュータに搭載される第一暗号化手段（means）と、

③情報 I を安全でない第一通信チャネルから、また第一 MDC を第二通信チャネルから受信して、情報 I を第一暗号化手段と同様の方法で変換して第二 MDC を作成する機能を有しており、第二コンピュータに搭載される第二暗号化手段と、

④第一通信チャネルから入手した情報 I を変換して生成した第二 MDC と、第二通信チャネルから入手した第一 MDC を比較して、両者が同一であれば情報 I を正当なものとみなす機能を有しており、第二コンピュータに搭載される比較手段と、

⑤承認した情報 I をメモリーに移転する手段、によって構成されるネットワーク」

「発明の実施例」には、このネットワークの利用例としてコンピュータのプログラム等を安全に送信するシステムが記載されており、そのネットワークシステムが詳細に記載されている。

このように、MDC-2/MDC-4 ハッシュ関数特許には、MDC-2 と MDC-4 による MDC 生成方法のクレームが存在するほか、MDC を利用したメッセージ認証を実現するネットワークのクレームも存在する。これらの方法、装置、ネットワークの詳細は「発明の実施例」に記載されている。なお、暗号化の数値変換の方法については「発明の実施例」の中で DES 暗号が挙げられているものの、クレームには記載されていない。

4．共通鍵暗号特許

（1）Lucifer 暗号特許

Lucifer 暗号は、換字と転置の 2 種類の変換を組み合わせた段関数を繰り返し利用することによって暗号化を行う方式であり、Shannon の「合成暗号」の考え方に基づいて開発された[185]。Lucifer 暗号特許[186]は 1991 年 3 月 19 日に失効している。

Lucifer 暗号特許は 13 のクレームを有しており、クレーム 1～12 がシステム（system）のクレームで、クレーム 13 がプロセス（process）のクレームである。クレーム 13 には、メッセージブロックの暗号化プロセスが記載されている。

クレーム 13：

「バイナリーデータとなっているメッセージブロックを暗号化するプロセスであり、
（a）メッセージブロックを第一登録手段に入力するステップと、
（b）バイナリーデータの鍵ブロックを第二登録手段に入力するステップと、
（c）バイナリーデータのメッセージブロックを n ビット毎のセットに分割し、上記第二登録手段の鍵ブロックの値によって決定される $2^n!$通り存在する変換方法のうちの 1 つによって、上記の n ビットのメッセージブロックを別の n ビットのブロックデータに変換するステップと、
（d）変換されたバイナリーデータを、そのデータの各ビットを並び替えることによって線形変換を行うステップと、
（e）上記（c）と（d）の変換を予め決められた回数だけ繰り替し、その回数の変換が終了するとメッセージブロックの暗号化が完了するステップ、によって構成されるプロセス」[187]

---

[185]「発明の背景」の中で、複数の変換を組み合わせるという Shannon の考え方が先行技術として挙げられている。

[186] U. S. Patent Number は 3,798,359（出願日 1971 年 6 月 30 日、発効日 1974 年 3 月 19 日）。

[187] クレーム 13 の原文は、"A process for enciphering a message block of binary digits comprising the steps of: a. loading said message block of binary digits into a first register means; b. loading a key block of binary digits into a second register means; c. grouping the message binary digits

このクレーム 13 には、非線形変換（ステップ（c））と線形変換（ステップ（d））を繰り返すことによって暗号化を行う、と記載されているだけで数値変換の方法は記載されていない。

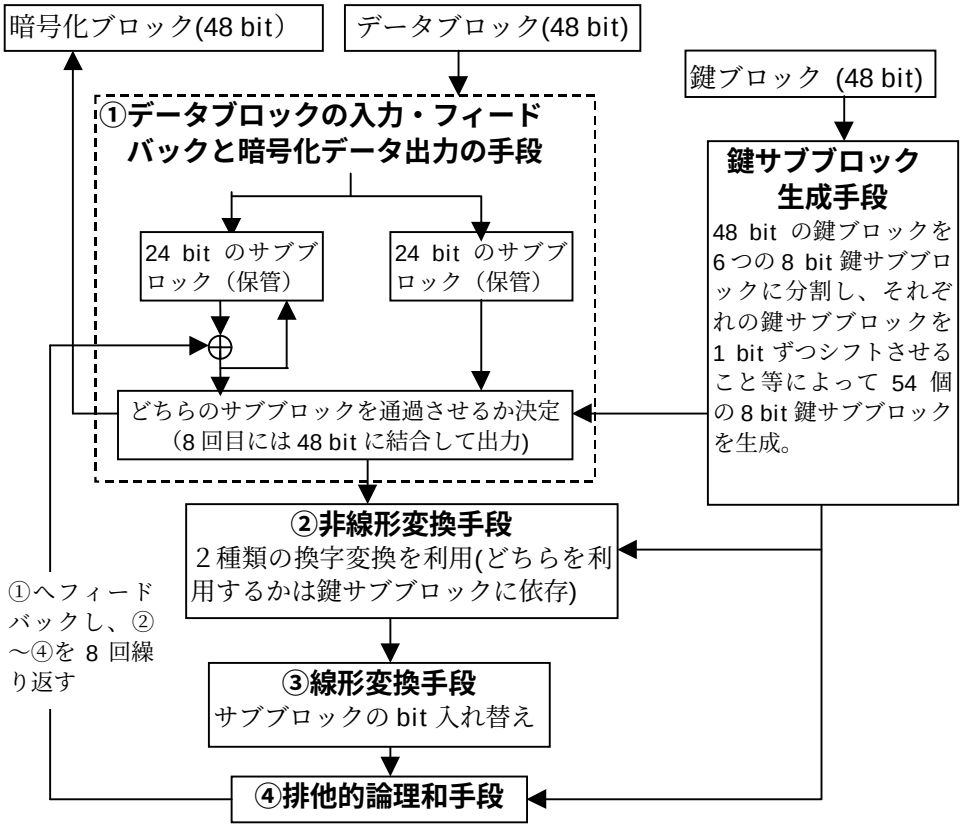


図 9　Lucifer 暗号全体の構成

「発明の詳細な説明」には Lucifer 暗号方式の数値変換の方法が詳細に記載されており、例えば、データブロックの入力・フィードバックと暗号化データ出力のための手段、鍵ブロックの入力・変換手段、非線形変換手段、線形変換手段、排他的論理和手段が記載されている（図 9 参照）[188]。ただし、クレーム 13 に記載されている非線形変換手段は 2 種類の換字変換手段に

---

into a plurality of sets each having n digits; substituting for each said set of n digits one out of $2^n!$ combinations of n binary digits, as determined by the binary condition of selected binary digits in said second register means; d. linearly transforming the substituted binary digits in said second register means; e. repeating steps c and d for a prespecified number of rounds so that upon termination of the prespecified number of rounds the message block is fully enciphered."である。

[188] 「発明の詳細な説明」に記載されている Lucifer 暗号の実施例では、①データブロックの入力・フィードバックと暗号化データ出力のための手段（鍵データに依存）、②非線形変換手段（鍵データに依存、2 種類の換字変換手段により構成される）、③線形変換手段（鍵データに依存しない）、④排他的論理和手段（サブブロックと鍵サブブロックの排他的論理和）の順番で 1 回の変換が行われ、これを 8 回繰り返すことで暗号化を行うと記載されている。入力されるデータブロックと鍵ブ

よって構成され、各換字変換に関しては① 4 bit の入力データを 16 bit に拡張変換し、②その 16 bit の変換結果を転置変換し、③その変換結果である 16 bit のデータを圧縮変換によって 4 bit のデータに変換・出力する、と記載されているだけであり、①の拡張変換、②の転置変換や③圧縮変換の具体的な数値変換の方法は説明されていない。

一方、クレーム 1 には、暗号化システムが記載されている。

クレーム 1：

「データブロックを暗号化ブロックに変換する暗号化システムであって、
①バイナリーデータを入力する入力手段（means）と、
②入力したデータの変換を制御するために利用される、複数のバイナリーデータである鍵を生成する手段と、
③入力手段と連結しており、入力データの換字変換を行う非線形変換手段と、
④上記非線形変換手段に連結し、入力データの bit 位置を入れ替える転置変換を行う線形変換手段と、
⑤鍵の制御によって変換を行う上記非線形変換手段によって構成され、
暗号鍵を変数とするブロック暗号を構成するシステム」

クレーム 1 の従属項のクレーム 2 には、複数の換字変換手段を有する暗号化システムが記載されているほか、クレーム 2 の従属項のクレーム 3 には、暗号鍵ブロックと換字変換手段を結び付けるゲート手段と、換字変換手段からのデータを出力して線形変換手段へデータを入力する出力手段を有する暗号化システムが記載されている。クレーム 3 の従属項のクレーム 4 には、変換後のデータ出力手段と bit 位置の入れ替えを行う線形変換手段をもつ暗号化システムが記載され、そのクレーム 4 の従属項のクレーム 5 には、生成した暗号鍵の保管手段を有する暗号化システムが記載されている。また、クレーム 1 の従属項のクレーム 7 では、クレーム 1 の暗号化システムに、

⑥線形変換されたデータと鍵の排他的論理和を計算する手段（means）と、
⑦排他的論理和を計算する手段によって変換されたデータを入力手段にフィードバックする手段、

が加えられた暗号化システムが記載されている。このクレーム 7 の従属項のクレーム 8 には、

ロックはともに 48 bit に設定されており、48 bit のデータブロックは 24 bit の 2 つのサブブロックに分割され、一方は保管され、もう一方が非線形変換、線形変換によって暗号化が行われると記載されている（ただし、鍵長やブロック長は、48 bit だけでなく 64 bit や 128 bit でも利用可能であると記載されている）。鍵ブロックについては、鍵サブブロック生成手段によって 6 つの 8 bit ブロックに分割され、各サブブロックを 1 bit ずつシフトさせること等によって暗号鍵を順次生成すると記載されている。1 回の変換が終了すると①にフィードバックされ、①に保管されていたもう一方の 24 bit サブブロックによって排他的論理和変換が行われ、その変換結果は保管されるとともに、入力される鍵サブブロックの値によって、その変換結果ともう一方の 24 bit サブブロックのどちらが次の変換②～④に移るのかが決定される。8 回目の変換が終了すると、変換された 2 つの 24 bit サブブロックは結合されて 48 bit となり、暗号化ブロックとして出力される。

フィードバック手段において暗号鍵によってデータブロックの各 bit を操作する bit シフト手段を有する暗号化システムが記載されており、クレーム 8 の従属項のクレーム 9 には非線形変換に利用される暗号鍵の保管手段を有する暗号化システムが記載されている。

このように、Lucifer 暗号特許のクレームには、線形変換、非線形変換と排他的論理和という 3 種類の変換を繰り返して暗号化を行う方法と、それに対応する暗号化システムが記載されている。これらの数値変換の方法については「発明の詳細な説明」の中に詳細に記載されているが、非線形変換については「発明の詳細な説明」の中でも不明な部分が残されている。

（2）DES 暗号特許

DES 暗号は、前節の Lucifer 暗号を参考にして開発されたとされる暗号方式で、現在米国政府標準暗号（FIPS 46-2）となっている。DES 暗号特許[189]はアメリカや日本において成立しているが、アメリカの特許については 1993 年 6 月 8 日に失効している。

＜アメリカの DES 暗号特許＞

DES 暗号特許は 10 のクレームを有している。まず、*f* 関数の変換プロセス（process）が記載されているクレーム 9 から説明する。

クレーム 9：

「暗号鍵の制御の下でデータブロックの暗号化を行うプロセスであって、
（a）データブロックを第一保管手段に保管するステップと、
（b）暗号鍵の bit の位置を入れ替えることによって暗号鍵を線形変換するステップと、
（c）ブロックデータの予め決められた位置の bit を複製してブロックデータを拡張することで、転置した暗号鍵の bit 長に等しくするステップと、
（d）拡張されたデータブロックと、データブロック長に等しい長さの暗号鍵によって換字変換関数を実行するステップと、
（e）換字変換後のデータブロックの bit 位置を入れ替えることによって線形変換を行い、暗号化データブロックを生成するステップとによって構成される暗号化プロセス」[190]

---

[189] U. S. Patent Number は 3,962,539（出願日 1975 年 2 月 24 日、発効日 1976 年 6 月 8 日）、日本国特許公告番号は昭 59-45269（出願日 1976 年 2 月 18 日、公告日 1984 年 11 月 5 日）。

[190] クレーム 9 の原文は、"A process for product block ciphering a block of data represented by a combination of binary digits under control of a cipher key represented by a combination of binary digits comprising the steps of: a. storing said block of data bits in a first store means, b. linearly transforming said cipher key by rearranging a portion of said cipher key combination of binary digits, c. duplicating predetermined ones of the data bits of said block of data bits to provide an expanded block of data bits consisting of said duplicated data bits and said block of data bits, said expanded block of data bits being equal in number to the number of said transformed cipher key bits, d. carrying out a substitution transformation function in accordance with said expanded block of data bits and said transformed cipher key bits to produce a substitution set of bits represented by a combination of binary digits equal in number

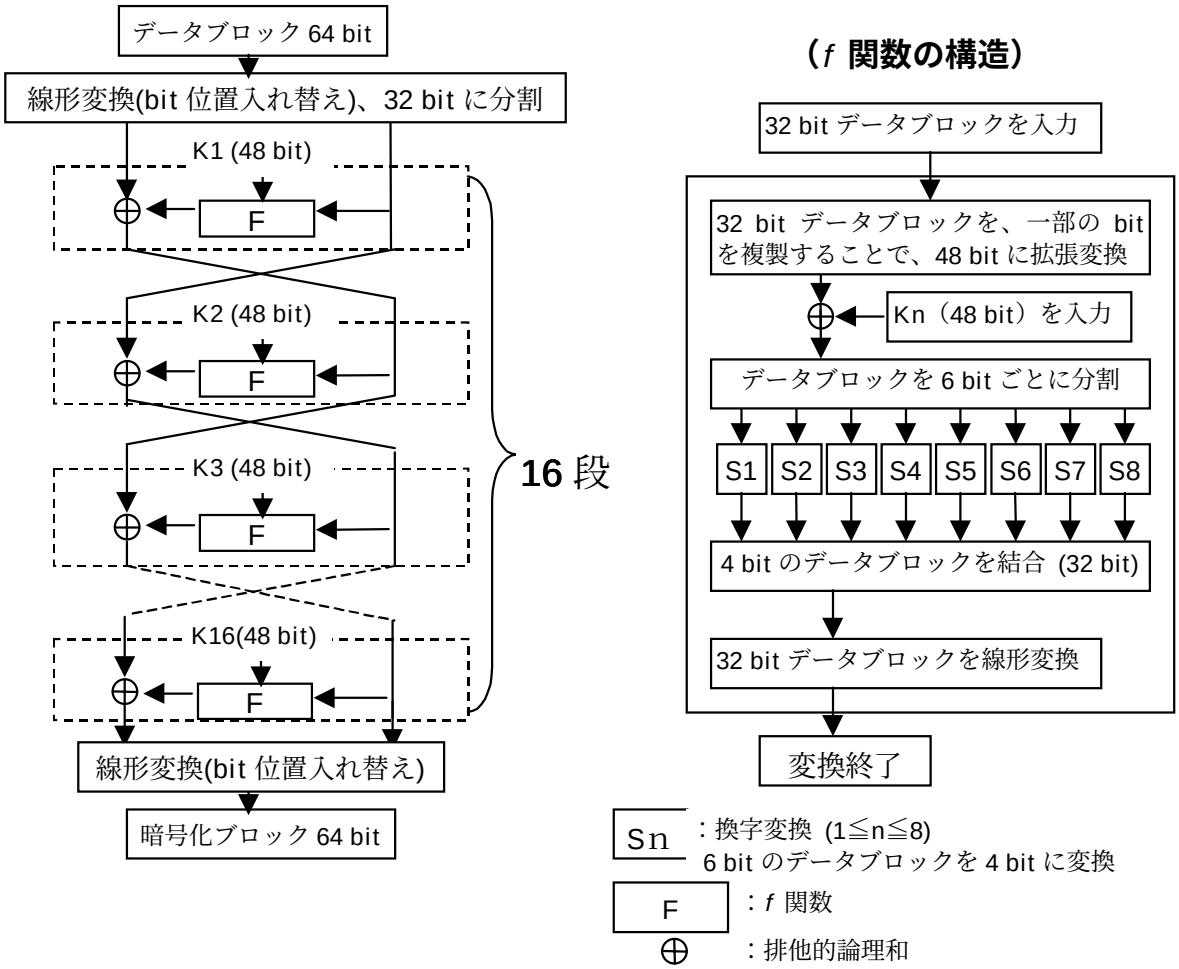


図 10　DES 暗号全体の構成と *f* 関数の構造

このように、クレーム 9 には 5 つのステップから構成される暗号化プロセスが記載されているが、暗号鍵の線形変換、ブロックデータの拡張変換、データブロックの換字変換や線形変換については具体的に記載されていない。ただし、これらの数値変換の方法は「暗号化プロセス」に記載されている[191]。また、*f* 関数を含む DES 暗号方式全体の構成についても「発明の詳細な説明」の中に記載されている（図 10 参照）[192]。

---

to the number of bits in said block of data, and linearly transforming said substitution set of bits by rearranging said substitution combination of binary digits, whereby the combined transformation results in a product block cipher of said block of data bits."と記載されている。

[191]「暗号化プロセス」では f 関数の変換プロセスが記載されており、① 32 bit のデータブロックを 8 つの 4 bit ブロックに分割した後、各 4 bit の後ろ 2 bit を複製して 6 bit のブロックに拡張変換し、②同様に 8 つの 6 bit のブロックとなっている 48 bit の暗号鍵によって排他的論理和変換を行い、③各 6 bit のデータブロックを 8 種類の非線形変換手段（換字表を利用、データの 1 bit 目と 6 bit 目の値に依存）によって 4 bit のデータブロックに変換し、④ 8 つの 4 bit データブロックを線形変換手段によって変換を行う、と記載されている。拡張変換、非線形変換や線形変換等は表の形で記載されている。

[192] 「発明の詳細な説明」には、DES 暗号方式全体の構成が次のように記載されている。まず、入

クレーム 1〜8 はすべて装置（device）のクレームである。クレーム 1 にはブロックデータの暗号化装置が記載されており、その装置を構成する手段が記載されている。

クレーム 1：

「暗号鍵の制御の下でデータブロックの暗号化を行う装置であって、
①データブロックを保管する第一保管手段（means）、
②暗号鍵の転置変換を行う第一線形変換手段、
③転置変換後の暗号鍵と同じ bit 長のデータブロックを生成するために、データブロックの一部を複製してデータブロックを拡張する手段、
④データブロックの拡張手段と第一線形変換手段に連結され、暗号鍵と拡張されたデータブロックを変数とする換字変換を実行し、上記データブロックと同じ長さのデータブロックを生成する手段、
⑤換字変換手段に連結され、データブロックの bit 位置を入れ替えることによって変換を行う第二線形変換手段、によって構成される装置」

クレーム 1 はクレーム 9 に対応する装置を特定している。クレーム 1 の従属項のクレーム 2 には、クレーム 1 の手段に加えて、データブロックの半分を複製することでデータブロックを拡張する手段を有する暗号化装置が記載されている。また、クレーム 1 の従属項のクレーム 3 には、拡張されたデータブロックと転置変換された暗号鍵を結合・変換する手段と、その変換されたデータブロックを非線形変換関数によって変換する非線形変換手段を有する暗号化装置が記載されている。クレーム 4 では、クレーム 1 の暗号化装置にデータと暗号鍵を一定のブロックに分割する手段が加えられた暗号化装置が記載されており、クレーム 8 には、データブロックと暗号鍵ブロックをそれぞれ 32 bit および 48 bit に設定した暗号化装置が記載されている。なお、クレーム 7 には、データブロックの両端の 2 bit を変数とする換字変換手段を有する装置が記載されている。

このように、DES 暗号特許は、データのブロック化、データの拡張変換、暗号鍵の転置変換、暗号鍵を利用した換字変換によって構成される暗号化のプロセスと装置のクレームを有している。このプロセスは、*f* 関数の変換に対応すると考えられる。ただし、これらのクレームには線形変換や換字変換の数値変換の方法が記載されておらず、「発明の詳細な説明」の中で詳

力データと鍵は 64 bit のデータブロックとして入力されるが、鍵 64 bit のうち、8、16、24、32、40、48、56 と 64 bit 目はパリティビットなので、鍵入力後最初の線形変換において削除されて 56 bit となり、2 つの 28 bit の鍵ブロックに分割される。各 f 関数で利用される鍵ブロックは、bit シフト等によって変換され、16 個（f 関数による変換が 16 回行われるため）の 48 bit の暗号鍵が生成される。入力データ 64 bit は、最初に線形変換によって bit 位置の入れ換えが行われ、2 つの 32 bit のデータブロック（左 32 bit と右 32 bit）に分割される。そのうち右 32 bit のデータブロックは暗号鍵とともに f 関数に入力され、その出力結果は左の 32 bit のデータブロックとの排他的論理和によって変換されて 1 回の変換が終了する。左右の 32 bit のデータブロックは入れ替えられて次の変換が開始し、16 回繰り返される。その後、2 つのデータブロックが結合して 64 bit となり、bit 位置の入れ替えによって変換され、暗号化ブロックとして出力される。

細に記載されている。また、DES 暗号の *f* 関数以外の部分もクレームには記載されていないが、それらについても「発明の詳細な説明」の中で詳細に記載されている。

＜日本の DES **暗号特許＞**

日本の DES 暗号特許のクレームは１つしかなく、アメリカ特許のクレーム 1 がそのまま記載されている。

クレーム 1：

「一組の暗号キー・ビットの制御のもとにデータ・ビットのブロックに対して積ブロック暗号処理操作を実行するための暗号装置にして、上記データ・ビットのブロックを記憶するための記憶手段と、上記一組の暗号キー・ビットを置換して出力するための第一置換手段と、上記記憶手段に接続され、上記ブロック中の選択されたデータ・ビットを二重にすることによって、上記第一置換手段から出力された暗号キー・ビットの数に等しいデータ・ビットのブロックを生成するための手段と、該手段からの拡張されたデータ・ビットのブロック及び上記第一置換手段からの置換された暗号キー・ビットを論理的に組み合わせて代替変換を実行することにより、元のデータ・ビットの数に等しいビット数を有する代替ビット群を生成するための手段と、該手段から出力された上記代替ビット群を置換することにより、上記データ・ビットのブロックの積ブロック暗号を生成するための第二置換手段とを有することを特徴とする暗号装置」

このように、暗号化に利用される数値変換の方法はクレームでは記載されておらず、「発明の詳細な説明」の中に記載されている。

（3）FEAL **暗号特許**

FEAL 暗号は、*f* 関数に 2 種類の算術演算（左 2 bit 循環シフトと 256 を法とする加算）を利用した暗号方式である。算術演算を利用することで、暗号化・復号化を DES 暗号よりも高速で行うことが可能である。

FEAL 暗号特許[193]は 26 のクレームを有しており、すべてのクレームが装置（equipment）のクレームとなっている。クレーム 1 には、データランダム化装置が記載されている。

クレーム 1：

「データランダム化装置であって、
①入力データを同じ長さのブロックに分割する手段（means）と、
②上記データ分割手段から送信されるデータブロックを次の処理手段に伝達する複数のチャネルと、
③上記チャネルから送信されるデータを変換する関数処理手段と、
④チャネルから送信されるデータを関数処理手段に伝達するデータの分岐手段と、

[193] U. S. Patent Number は 4,850,019（出願日 1986 年 11 月 3 日、発効日 1989 年 7 月 18 日）。

⑤ランダム化されたデータを生成するために、すべてのチャネルから送信されるデータを結合する手段とによって構成される装置」

このクレーム 1 のデータランダム化装置は、FEAL 暗号における *f* 関数に対応しているとみられる。関数処理手段の具体的な数値変換の方法はこのクレームには記載されていないが、「発明の詳細な説明」の中に、左 2 bit の循環シフトと 256 を法とする加算やデータブロックの数値変換の方法が詳細に記載されている（図 11 参照）[194]。

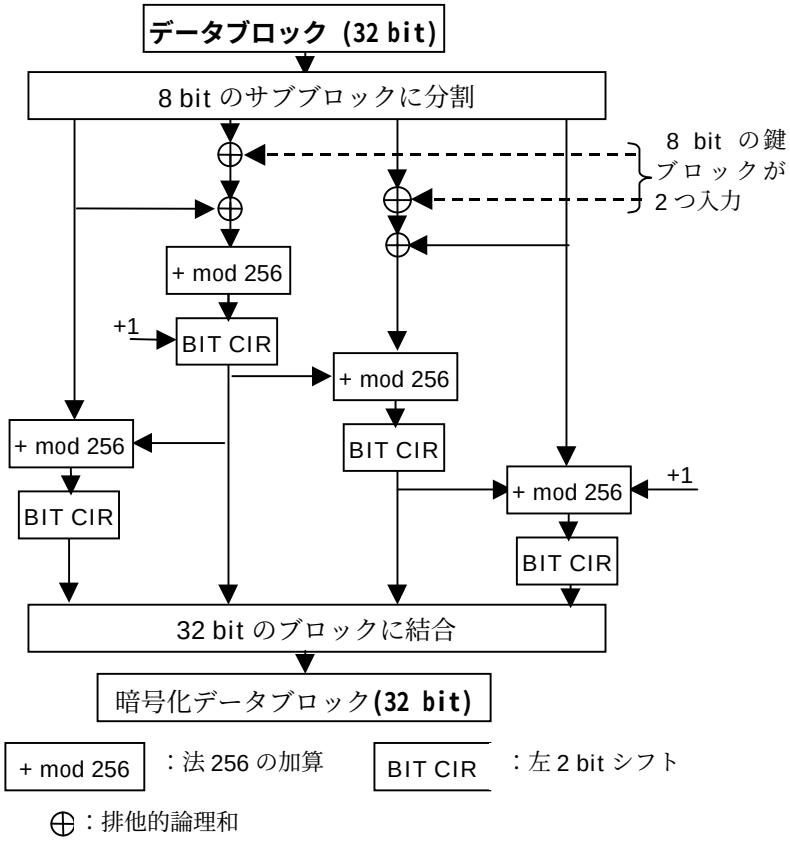


図 11　FEAL 暗号の *f* 関数の構造

クレーム 1 の従属項クレーム 2 には、データランダム化装置の構成要素として、⑥関数処理手段によって変換されたデータをさらに変換するための変換処理手段が加えられた装置が記載されている。また、クレーム 1 の従属項のクレーム 6 には、⑦パラメーターデータ（鍵）を

194 「発明の詳細な説明」における f 関数の説明では、① 32 bit のデータブロックをさらに 4 つの 8 bit のデータブロックに分割するとともに、② 16 bit の暗号鍵も 2 つの 8 bit の鍵ブロックに分割し、③ 8 bit のデータブロックをそれぞれ法 256 の加算（関数処理手段に相当）、左 2 bit の循環シフト（変換処理手段に相当）と、他のデータブロックまたは鍵ブロックとの排他的論理和（第五および第六関数処理手段に相当）によって変換し、④変換した 4 つの 8 bit のデータブロックを結合して 32 bit のデータとして出力する、と記載されている。鍵ブロックについても、64 bit の暗号鍵のもとになるデータから生成する数値変換の方法が詳細に記載されている。

入力し、鍵を関数処理手段に送信するパラメーター処理手段が加えられた装置が記載されている。クレーム 6 の従属項のクレーム 8 には、パラメーター処理手段が関数処理手段よりも上位に設置されるデータランダム化装置が記載されている。クレーム 2 の従属項のクレーム 3 には、⑧変換処理を切り替える処理切替手段を有する装置が記載されている。クレーム 11 には、「関数処理手段は、2 つのデータとある定数を加えてその合計値の剰余演算を行う」と記載されており、加算変換手段を有する暗号化装置が特定されている。また、クレーム 19 には、「変換処理手段は、チャネルデータに対して bit 循環処理を行う」と記載されており、データブロックの bit 循環シフトによる変換手段を有する暗号化装置が特定されている。もっとも、これらのクレームには、加算変換手段等の数値変換の方法が特定されていない。

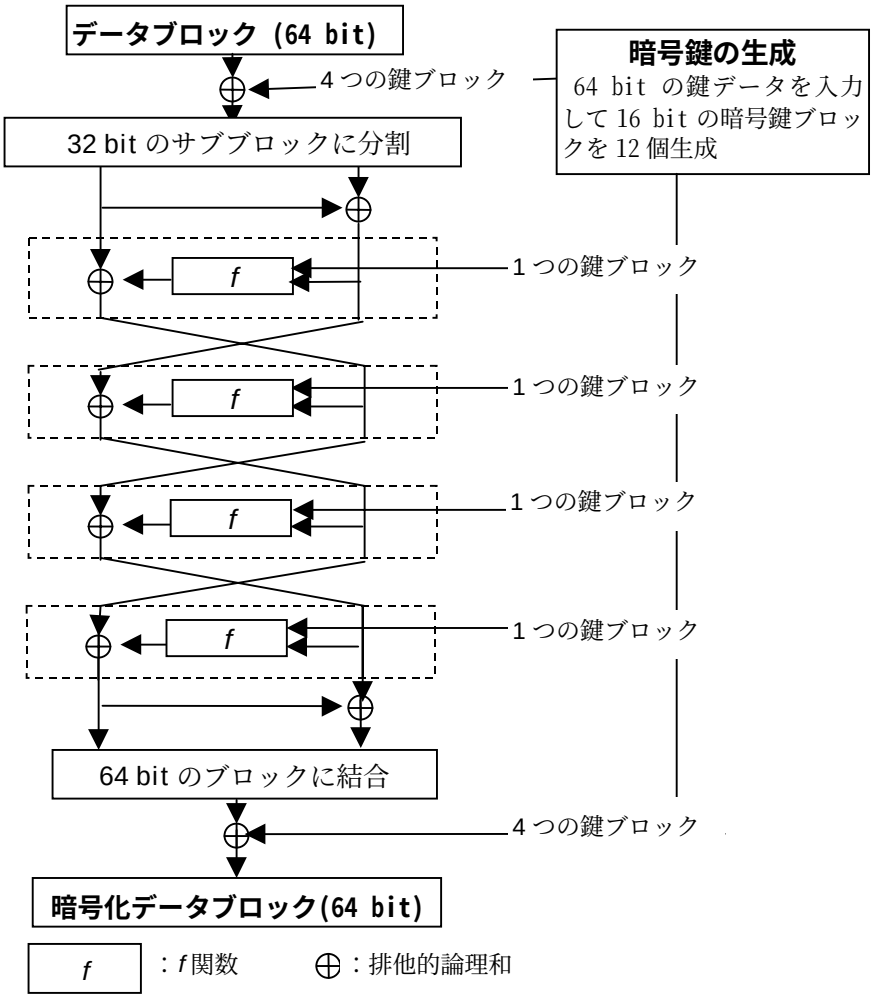


図 12　FEAL 暗号の構成（段数が 4 の場合）

このように、FEAL 暗号特許には、加算と bit 循環シフトという 2 種類の変換手段を有するデータランダム化装置のクレームが存在しており、この装置は FEAL 暗号における *f* 関数に対応しているとみられる。加算や bit 循環シフトの数値変換の手段はクレームには記載されていないが、「実施例の詳細な説明」の中で詳細に記載されているほか、FEAL 暗号の *f* 関数以外

の部分についても記載されている（図 12 参照）[195]。

（4）IDEA 暗号特許

IDEA 暗号は、DES 暗号や FEAL 暗号と異なる構造の段関数を有する暗号方式であり、2 種類の算術演算（2^{16} を法とする加算と、（$2^{16}+1$）を法とする積）を利用する点が特徴である。IDEA 特許[196]は国際特許として出願されており、ここではアメリカと日本の特許の内容をそれぞれ整理する。

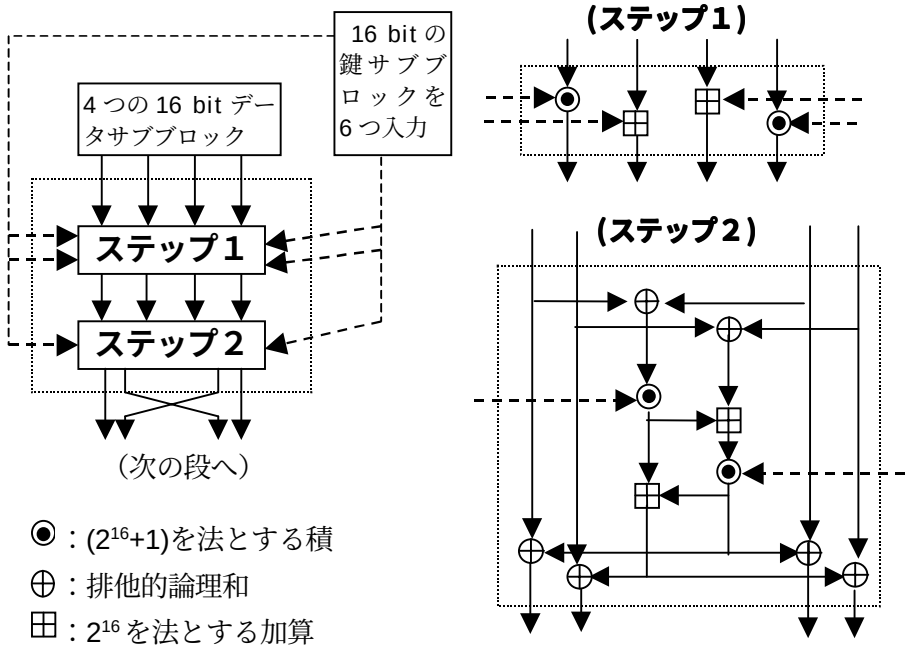


図 13　IDEA 暗号の段関数の構造

---

[195] 「実施例の説明」の中には f 関数以外の FEAL 暗号の方法が記載されており、① 64 bit のデータブロックは 4 つの 16 bit の鍵ブロックとの排他的論理和によって変換され、②その変換結果は 2 つの 32 bit データブロック（左 32 bit と右 32 bit）に分割され、③右 32 bit データブロックは左 32 bit のデータブロックとの排他的論理和によって変換され、④その右 32 bit データブロックは、次の変換の左 32 bit データブロックとなる一方、1 つの 16 bit の鍵ブロックとともに f 関数によって変換され、左 32 bit データとの排他的論理和によって次の変換の右 32 bit データブロックとなる（④の変換が 1 段に相当する）。「発明の詳細な説明」では 4 段の場合と 6 段の場合が記載されている。なお、それぞれ既定の段数の変換が終了すると、⑤右の 32 bit データブロックは左の 32 bit データブロックとの排他的論理和によって変換され、⑥その結果生成される 32 bit データブロックと左 32 bit データブロックが結合されて 64 bit データブロックとなり、⑦再び 4 つの 16 bit 鍵ブロックとの排他的論理和が計算され、その結果が暗号化データブロックとなる、と記載されている。

[196] 国際特許出願番号は PCT/CH91/00117（出願日 1991 年 5 月 16 日）、U. S. Patent Number は 5,214,703（発効日 1993 年 5 月 25 日）、日本国特許公表番号は平 5-500121（公表日 1993 年 1 月 14 日）。

＜アメリカの IDEA 暗号特許＞

アメリカの IDEA 暗号特許は 10 のクレームを有しており、すべて装置（device）のクレームとなっている。まず、クレーム 1 にはデータランダム化装置が記載されている。

クレーム 1：

「自由に選択可能な制御ブロックによって、データブロックを同じブロック長（第一 bit 番号 N）のブロックに変換するための装置であって、

①データブロックを入力し、それを 2 つ以上の同じブロック長（第二 bit 番号 m）のサブブロックに分割する第一入力手段（means）と、

②サブブロックと同じブロック長の 2 つ以上の制御ブロックを受け取る第二入力手段と、

③制御ブロックを利用して、少なくとも 4 つの処理ユニットを含む、少なくとも 2 種類の連続する変換処理によってサブブロックを変換する演算手段であり、それぞれの処理ユニットは、変換処理のためのサブブロックと制御ブロックを受け取るための 2 つの入力機能と、変換後のブロックを出力する出力機能を備えており、連続する 2 つの処理ユニットの変換方法が異なるように配置され、3 種類の処理方法のいずれかの方法が利用されるような演算手段と、

④少なくとも 2 つの最終変換処理後のサブブロックを出力する出力手段、とによって構成される装置」

このクレーム 1 には処理ユニットにおいて実行される数値変換の方法が記載されていないが、「実施例の説明」の中には記載されており、IDEA 暗号で利用される算術演算（2^m を法とする加算（m=4,8,16）、（2^m－1）を法とする積と排他的論理和）が具体的に説明されている。

クレーム 1 の従属項であるクレーム 2〜4 には、クレーム 1 の 4 つの手段を有し、演算手段の中に 4 つの処理ユニットが含まれる装置（クレーム 2 に記載）、演算手段に 6 つの処理ユニットが含まれる装置（クレーム 3 に記載）、さらに演算手段が 2 つのステップ（第一、第二ステップ）の変換によって構成され、第一ステップには 4 つの処理ユニットが、また第二ステップには 6 つの処理ユニットが含まれる装置（クレーム 4）がそれぞれ記載されている。このクレーム 4 には、入力されるデータブロックと処理ユニットの関係、処理ユニットと出力ブロックとの関係や、処理ユニットと入力される制御ブロックとの関係が記載されており、このクレームで特定される装置が IDEA 暗号の段関数に対応しているとみられる。このクレーム 4 の装置の具体的な構造については「実施例の説明」の中で開示されている（図 13 参照）。

さらに、クレーム 1 の従属項のクレーム 8 で特定されている装置は、IDEA 暗号全体に対応する暗号化装置とみられる。このクレームにも処理ユニットにおける数値変換の方法が記載されていないが、「実施例の説明」の中に記載されている（図 14 参照）[197]。

---

[197] 「実施例の説明」には IDEA 暗号全体の構成が次のように記載されている。①入力される 64 bit のデータブロックは 4 つの 16 bit のデータサブブロックに分割され、② 128 bit の鍵ブロックも、鍵サブブロック生成手段において左 25 bit 循環シフトによって 52 の 16 bit 鍵サブブロックに変換され、③第一暗号化ステージでは、4 つのデータサブブロックが 6 つの鍵サブブロックによって変

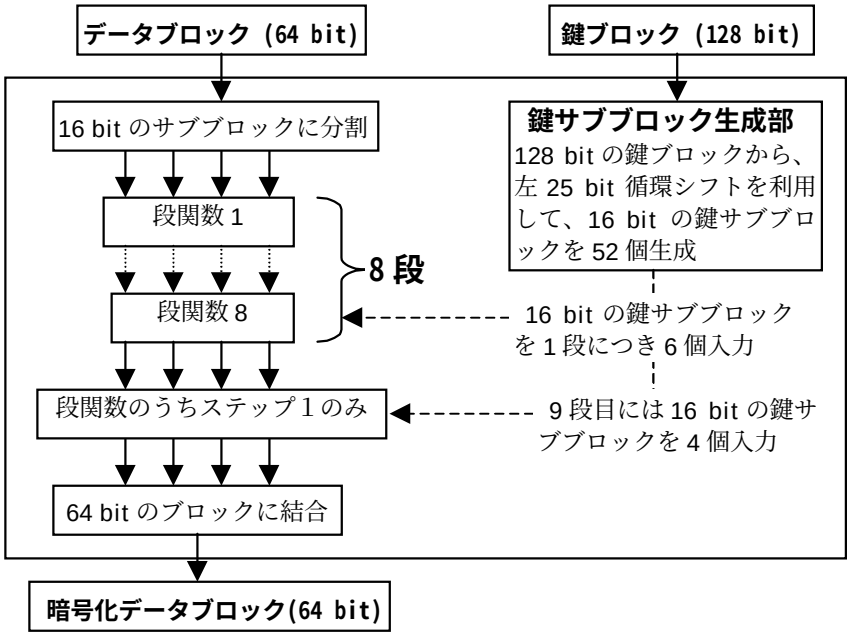


図 14　IDEA 暗号の構成

クレーム 8：

「クレーム 1 の装置であって、

①第一、第二、第三と第四データブロックを入力する第一入力手段と、

② X 個の制御ブロックを入力する第二入力手段と、

③同一の演算を実行する Y 個のステージと最終演算ステージによって構成され、各ステージでは 4 つの入力データを 4 つの出力データに変換し、あるステージの出力データが次のステージの入力データとなり、最終ステージでは 4 つの最終データブロックを生成する演算手段と、

④第一、第二、第三と第四の最終サブブロックを出力する出力手段によって構成され、第二入力手段によって入力される制御ブロックの数 X はステージの数 Y とすると X = Y*6+4 という関係で表わされ[198]、Y 番目のステージでは、第一処理ユニットが第一ブロックを第 X-3 制御ブロックによって変換し、第二処理ユニットが第二ブロックを第 X-2 制御ブロックによって変換し、第三処理ユニットが第三ブロックを第 X-1 制御ブロックによって変換し、第四処理ユニットが第四ブロックを第 X 制御ブロックによって変換する装置」

換・出力され、第二暗号化ステージの入力データとなり、④この暗号化ステージが 8 回繰り返され、第 9 暗号化ステージでは 4 つの鍵サブブロックによって変換が行われ、⑤変換後の 4 つのデータサブブロックが結合して 64 bit の暗号化ブロックとなる。③の変換内容については、2^{16} を法とする加算、（2^{16}＋1）を法とする積と、排他的論理和の 3 種類の演算を有する変換として詳細に説明されている。

[198] クレーム 8 には、ステージの数Ｙと暗号鍵ブロックＸとの間の関係について、「X = Y + 4」と記載されているが、「実施例の説明」には「X = Y*6 + 4」と記載されている。

このように、アメリカの IDEA 暗号特許には、IDEA 暗号の段関数に対応するとみられる装置のクレームが存在するほか、IDEA 暗号全体に対応するとみられる装置のクレームも存在する。各処理ユニットにおいて実行される数値変換の方法はクレームには記載されておらず、「実施例の説明」において記載されている。

＜日本の IDEA 暗号特許＞

日本の IDEA 暗号特許は 17 のクレームを有しており、クレーム 1～13 が暗号変換装置のクレーム、クレーム 14～17 が方法のクレームとなっている。まず、装置のクレームのうち、クレーム 1，2，3，5、6 と 7 が、アメリカの特許のクレーム 1、2、3、4、6 と 8 にそれぞれ対応している。クレーム 9 では、「特許請求の範囲第 7 項または第 8 項において、第 2 の数 T が 52 であって、そして、第 1 の数 S が 8 であることを特徴とする暗号変換装置」と記載されており、暗号鍵ブロックの数（T）と段数（S）が特定されている。さらに、クレーム 10 では、2 種類の算術演算の方法が具体的に記載されている。

クレーム 10：

「特許請求の範囲第 1 項において、

① 第 1 種の演算（加算的係数 2^m）が以下のようにして求められること、

・すべての各入力ブロック（E_1,E_2）が 2 進数の整数として扱われ、かつ、集合｛0、1、2、3、...、（2^m-1）｝の構成要素であり、

・求められた出力ブロック（A）が入力ブロック（E_1,E_2）の和の係数 2^m であり、

② 第 2 種の演算（乗算係数（2^m-1））が以下の様にして求められること、

・入力ブロック（E_1,E_2）、および出力ブロック（A）のすべてのブロックビットがゼロである場合において、このブロックは 2 進法表記において、2^m であり、そうでない場合には、各入力ブロック（E_1,E_2）は、2 進法表記において整数とみなされ、かつ、集合｛0、1、2、3、...、（2^m-1）｝であり、

・対応する出力ブロック（A）を 2 進法表記すると、入力ブロック（E_1,E_2）の乗算係数（2^m-1）であり、

③ そして第 3 種の演算装置（ビットバイビット排他的論理和）において、以下のように演算が実行されること、

・各入力ブロック（E_1,E_2）および出力ブロック（A）のすべてのブロックが、連続するビットのシーケンスであり、

・固定された位置が各ビットに割り当てられており、各出力ブロック（A）のビットシーケンスは各々、対応する入力ブロック（E_1,E_2）によって与えられる位置における 2 つのビットの排他的論理和であることを特徴とする暗号変換装置」

このクレーム 10 の従属項のクレーム 11 には、m が 4、8、16 のいずれかの値となる暗号変換装置が記載されている。アメリカの特許とは異なり、クレームの中に 2 種類の算術演算の方法が記載されている。また、クレーム 14 は暗号変換装置の使用方法のクレームであり、平文・暗号文や暗号鍵のブロック分割方法、（2^m+1）を法とする積と 2^m を法とする加算を利用した

データ変換方法が記載されている。クレーム 14 の従属項のクレーム 16 には、暗号鍵のサブブロック化の方法が記載されているほか、同じくクレーム 14 の従属項のクレーム 17 には、キーサブブロック数（T）が 52、段数（S）が 8、サブブロック長（m）が 16 となる暗号化装置の使用方法が記載されている。これらのクレームも、アメリカの IDEA 暗号特許には存在しないクレームである。

（5）MISTY 暗号特許出願

MISTY 暗号は、換字表を利用した非線形変換と、暗号鍵によって形が変わる線形変換関数を有するブロック暗号であり、差分解読法と線形解読法に対する安全性が理論的に保証されている。また、換字表を利用した非線形変換を並列処理することで、高速処理を可能としている。

現在公開されている MISTY 暗号特許[199]は 31 のクレームを有しており、そのうちクレーム 9 と 10 だけがデータ変換方法のクレームであり、残りはすべてデータ変換装置のクレームである。クレーム 9 のデータ変換装置は、MISTY 暗号の基本的な構成に対応しているとみられ、MISTY 暗号の特徴の 1 つである非線形変換の並列配置が明らかになっている。ただし、非線形変換の数値変換の方法はこのクレームには記載されておらず、「発明の開示」と「発明を実施するための最良の形態」に詳細に記載されている[200]。

クレーム 9：

「任意の 2 つの A 入力データと B 入力データに対し、

① B 入力データをそのまま第 1 の A 中間データとして出力する第 1 ステップ（S1）と、

② 上記 A 入力データを第 1 の鍵パラメータで非線形変換し、該非線形変換後の出力データと上記 B 入力データとの排他的論理和をとり第 1 の B 中間データとして出力する第 2 ステップ（S2）と、

③ 上記第 1 の B 中間データを入力し、そのまま第 2 の A 中間データとして出力する第 3 ステップ（S3）と、

④ 上記第 1 の A 中間データを入力して第 2 の鍵パラメータで非線形変換し、該非線形変換後の出力データと、上記第 1 の B 中間データとの排他的論理和をとり、第 2 の B 中間データとして出力する第 4 ステップ（S4）とを備え、

上記第 1 から第 4（S1～S4）を繰り返し、最後に第 2 又は第 4 ステップで終えるようにし、また、最終 A 中間データと B 中間データを変換データとするデータ変換方法」

また、クレーム 10 には、クレーム 9 と同様に 4 つのステップからなるデータ変換方法が記載

---

199 国際特許出願番号は PCT/JP96/02154（出願日 1996 年 7 月 31 日）。日本における MISTY 暗号特許は 1997 年 3 月 13 日に公開されているが、現時点では成立していない。

200「発明を実施するための最良の形態」には、クレーム 9 の「非線形変換」が、排他的論理和と 2 種類の換字表を利用した非線形変換によって構成され、これらの変換手段がどのように配置されるのかが「実施の形態」として詳細に記載されている。

されている。ただし、クレーム 10 のデータ変換方法は、排他的論理和と非線形変換の配置がクレーム 9 とは異なっている。

次に、データ変換装置のクレームを整理する。まず、クレーム 1 は、クレーム 9 に対応する装置のクレームとなっており、4 つの変換ステップを有するデータ変換装置が記載されている。また、クレーム 1 の従属項のクレーム 2 には、排他的論理和と非線形変換を備えた 2 種類の副変換処理部を交互に必要な個数だけ接続したデータ変換装置が記載されている。クレーム 2 の従属項のクレーム 3 と 4 には、クレーム 2 の副変換処理部の非線形変換部分に、非線形変換と排他的論理和を実行する内部副変換処理部を交互に n 段接続した変換手段を有するデータ変換装置が記載されており、副変換処理部の非線形変換の部分が入れ子の構造になっていることが明らかとなっている（図 15 参照）[201]。

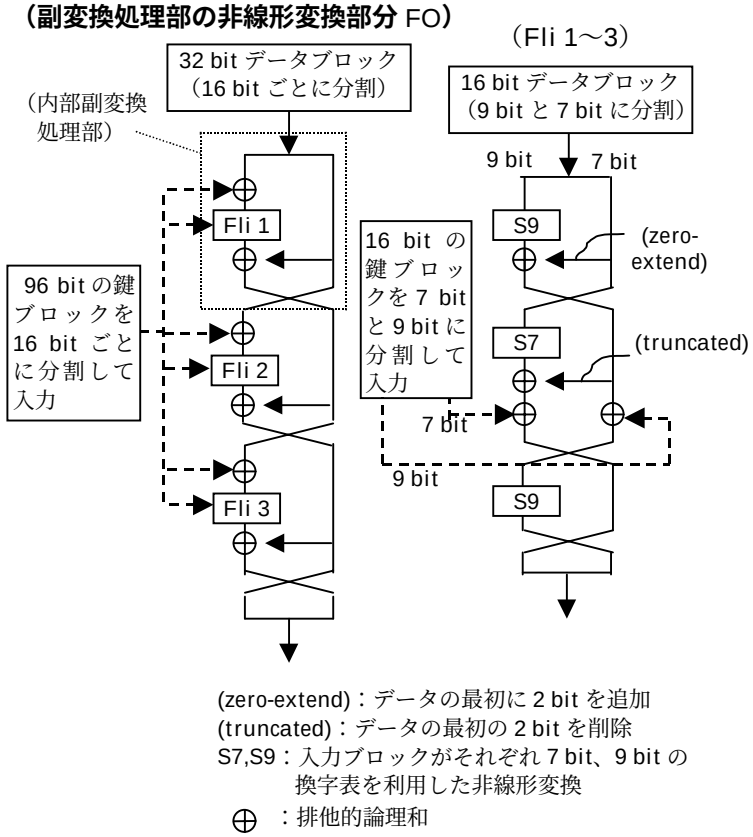


図 15　MISTY 暗号の副変換処理部の非線形変換部分

クレーム 1 の従属項のクレーム 11 には、クレーム 1 の装置のデータ入力部分にデータ選択部を付加するとともに、データ出力部分にデータ保持部を付加し、データ保持部に出力された変換データをデータ入力部にフィードバックして、予め決められた回数だけ変換を繰り返すデ

[201] 副変換処理部の非線形変換部分の構造は「発明を実施するための最良の形態」で明らかになっており、① 16 bit の鍵ブロックをパラメーターとして入力するとともに、②入力された 16 bit のデータブロックを 7 bit と 9 bit のブロックに分割し、②それぞれのデータブロックを換字表を用いた非線形変換によって変換し、④最後に再び 16 bit のデータブロックに結合して出力する、という変換方法が記載されている。

ータ変換装置が記載されている[202]。クレーム 2 の従属項のクレーム 13 には、クレーム 2 の副変換処理部における非線形変換部分に 2 種類の内部副変換処理部を有するデータ変換装置が記載されており、特に最初の内部副変換処理部の入力側に内部データ選択部を付加するとともに、最後の内部副変換処理部の出力側に内部データ保持部を付加して、クレーム 11 と同様に、変換後のデータをフィードバックする形でデータ変換を行う装置が記載されている。

一方、クレーム 19 と 20 には、非線形変換の方法が特定されているデータ変換装置が記載されている。もっとも、いずれのクレームでも具体的な数値変換の方法は記載されておらず、「発明を実施するための最良の形態」の中で説明されている。

クレーム 19：

「非線形変換回路として、少なくともそのどれかにガロア体上の元 X の n 乗回路を用いることを特徴とする請求の範囲第 1 項又は請求の範囲第 5 項いずれか記載のデータ変換装置」

クレーム 20：

「ガロア体上の元 X の n 乗回路を、正規基底で構成することを特徴とする請求の範囲第 19 項記載のデータ変換装置」

クレーム 2 の従属項であるクレーム 24 には、2 つの排他的論理和回路と、暗号鍵の一部をパラメーターとして論理和または論理積を計算する 2 つの論理演算回路を有するデータ変換部が第 1 副変換処理装置に付加されたデータ変換装置が記載されている。このデータ変換部が、鍵に依存して形が変わる線形変換関数に対応しているとみられる。

残りのクレーム 25～31 には、データ変換装置の構成が記載されている。まずクレーム 25 には、非線形変換回路と排他的論理和回路を有するデータ変換装置が記載されており、非線形変換回路を並列に配置してデータの変換を行う装置であることが明らかになっている。

クレーム 25：

「第 1 と第 2 の系統のデータ（A と B）を入力し、鍵パラメータ（111～114）[203]を用いて非線形変換処理し、非線形変換処理された第 1 と第 2 の系統のデータ（A と B）を出力するデータ変換装置において、

① 第 1 の系統のデータ（A）を鍵パラメータによって非線形変換処理する非線形変換回路（131～134）と、第 1 と第 2 の系統のデータ（A と B）の排他的論理和を演算する排他的論理和回路（141～144）とを有する副変換処理部（121～124、又は 161～164）

---

202 この装置構成については、「発明を実施するための最良の形態」の中で、「このように構成することで、（中略）高速のデータ変換ができ、かつ、副変換処理部の数を少なくすることができ、装置規模を小さくすることができる」と、そのメリットについて説明されている。

203 （）内の数字は、明細書の図 1 に記載されているデータ変換装置において、装置の各部分に付されている番号を示している。

を少なくとも 2 つ、第 1 の副変換処理部（121 又は 161）及び第 2 の副変換処理部（122 又は 162）として備え、

② 第 1 の副変換処理部（121 又は 161）から出力される第 1 と第 2 の系統のデータ（A と B）を第 2 の副変換処理部（122 又は 162）の第 2 と第 1 の系統のデータ（B と A）として入力し、

③ 第 1 と第 2 の副変換処理部（121 と 122、又は 161 と 162）の非線形変換回路（131 と 132、又は 132 と 133）の非線形変換処理を同時に実行することを特徴とするデータ変換装置」

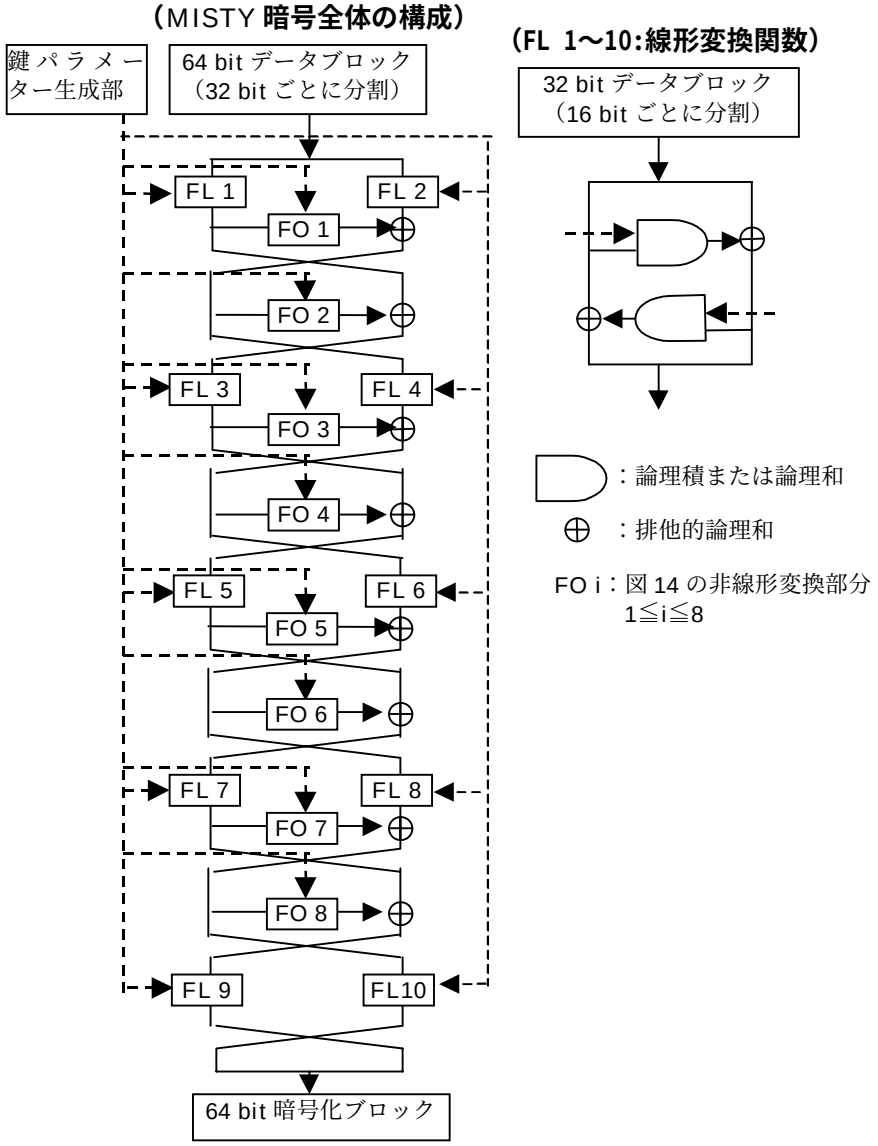


図 16 MISTY 暗号全体の構成

また、クレーム 25 の従属項のクレーム 28 には、クレーム 25 の装置に繰り返し処理部と鍵パラメータ供給部を付加することで、副変換処理部の非線形変換回路を所定の回数だけ繰り返し実行する装置が記載されている。これらのクレームには MISTY 暗号全体の構成が記載されて

いないが、「発明を実施するための最良の形態」に記載されている（図 16 参照）[204]。

以上のように、MISTY 暗号特許は、内部で非線形変換と排他的論理和を実行するデータ変換手段と装置のクレームを有している。また、非線形変換を実行する部分に、排他的論理和回路と非線形変換回路を有する内部副変換処理部が組み込まれているデータ変換装置もクレームに記載されているほか、副変換処理部を繰り返し利用して変換を行う方法として、データ選択部やデータ保持部等を使って変換後のデータをフィードバックする機能をもつデータ変換装置もクレームに記載されている。換字表を用いた非線形変換や、論理和、論理積を利用することもクレームから明らかになっているほか、MISTY 暗号全体の構成についても「発明を実施するための最良の形態」の中に記載されている。

5．鍵配送・共有法特許

（1）Diffie-Hellman 特許

Diffie-Hellman 特許[205]は、1976 年に Diffie と Hellman が発表した論文の中で紹介された鍵共有方式に関する特許である。Diffie and Hellman 特許は 1997 年 4 月 29 日に失効している[206]。

Diffie-Hellman 特許は 8 つのクレームから構成されており、そのうちクレーム 2、3、4 と 6 が方法（method）のクレームであり、クレーム 1、5、7 と 8 が装置（apparatus）のクレームである。まず、方法のクレームから整理する。クレーム 2 には、鍵生成装置を利用した暗号鍵の共有方法と暗号通信の方法（method）が記載されている。

クレーム 2：

「送信者から受信者にメッセージを送信するというタイプのデータ通信を、安全とはいえない通信チャネルにおいて実行するための方法であって、

①送信者が変換された第一データを生成するために、第一データを逆変換が困難な方法で変換し、

②受信者が変換された第二データを生成するために、第二データを逆変換が困難な方法で変換し、

③送信者が上記変換された第一データを受信者に送信し、

④受信者が上記変換された第二データを送信者に送信し、

---

[204] 「発明を実施するための最良の形態」には MISTY 暗号全体の構成が記載されており、64 bit のデータを 32 bit に分割し、排他的論理和回路、論理積および論理和変換の手段を有する線形変換関数と換字表を利用した非線形変換回路を有する複数の副変換処理部の数値変換の方法が詳細に記載されている。なお、MISTY1 のほか、MISTY2 の構造も記載されている。

[205] U. S. Patent Number は 4,200,770（出願日 1977 年 9 月 6 日、発効日 1980 年 4 月 29 日）。

[206] Diffie-Hellman 特許を所有しているスタンフォード大学は、この特許の排他的なライセンス権を Cylink 社（米国 California 州 Sunnyvale を本拠とする情報セキュリティ製品のベンダーで、ネットワーク関連ソフトウェアの開発・販売等を行っている）に委譲していたが、Cylink 社は Diffie-Hellman 特許が既に失効していたことを 1997 年 9 月 16 日に公表している。

⑤送信者が上記変換された第二データと上記第一データを用いて、上記変換された第一データと変換された第二データだけから生成することが不可能な方法によって、暗号鍵となる第三データ（共有する暗号鍵）を生成し、
⑥受信者が上記変換された第一データと上記第二データを用いて、第三データの生成方法と同じ方法によって、暗号鍵となる第四データを生成し、
⑦送信者はメッセージを上記暗号鍵（第三データ）で暗号化し、
⑧送信者は暗号化されたメッセージを受信者に送付し、
⑨受信者は暗号化されたメッセージを暗号鍵（第四データ）で復号化する、
という方法」

このクレーム２には、共有する鍵の生成方法に関して「逆変換が困難な方法で変換する」と記載されているだけであり、具体的な数値変換の方法は記載されていないが、「実施例の説明」の中で説明されている[207]。また、クレーム 2 の従属項のクレーム 3 には、クレーム２の方法に加えて、共有している暗号鍵で暗号文をまともな平文に復号化できるかどうかをチェックすることにより、受信者の本人確認も行うことが可能であると記載されている。クレーム 4 では、クレーム 2 の一部分である暗号鍵の共有方法（①～⑥）が暗号鍵生成方法として記載されている。

また、クレーム 6 には、暗号文送信者と受信者が鍵を共有する方法が記載されており、その中で、暗号文送信者が鍵の生成に必要な第一データの変換と第三データの生成の際にべき乗剰余算を実行するステップと、受信者も第二データの変換と第四データの生成の際にべき乗剰余算を実行するステップが記載されている。

クレーム 6：
「送信者と受信者の間で安全に暗号鍵を生成する方法であって、以下の各ステップによって構成されており、
①送信者は、変換された第一データを生成するために第一データを変換するが、その変換方法は、逆変換が不可能となるように第一の数値を第一データでべき乗するという演算と、そのべき乗演算された数値に対して第二の数値を法とする剰余演算によって構成され、
②受信者は、変換された第二データを生成するために第二データを変換するが、その変換方法は、逆変換が不可能となるように第一の数値を第二データでべき乗するという演算と、そのべき乗演算された数値に対して第二の数値を法とする剰余演算によって構成され、

[207] Diffie-Hellman 鍵共有方式を簡単に説明する。まず、暗号通信者間で予め原始根 a と法 q を定め、各々が秘密の乱数 X_1 と X_2 と生成し、$Y_i = a^{X_i} \pmod q$を生成する（ただし、i=1,2）。暗号通信者は互いに相手に自分の Y_i を送付し、相手の Y_i に X_j を乗じて $Y_1^{X_2} = Y_2^{X_1} = a^{(X_1X_2)} \pmod q$ を計算する。この $a^{(X_1X_2)} \pmod q$が共有される暗号鍵となる。ここで記載されている第一データおよび第二データが X_1 および X_2 に対応し、第三データおよび第四データが、$Y_2^{X_1} \pmod q$および $Y_1^{X_2} \pmod q$に対応するとみられる。

③送信者が受信者に対して、変換された第一データを送信し、
④受信者が送信者に対して、変換された第二データを送信し、
⑤送信者は、変換された第二データを第一データで変換し、変換された第一データや変換された第二データから生成するのが不可能な、暗号鍵に相当する第三データを生成するが、その方法は、変換された第二データを第一データでべき乗するという演算と、そのべき乗演算された数値に対して第二の数値を法とする剰余演算によって構成され、
⑥受信者は、変換された第一データを第二データで変換し、変換された第一データや変換された第二データから生成するのが不可能な、暗号鍵に相当する第四データを生成するが、その方法は、変換された第一データを第二データでべき乗するという演算と、そのべき乗演算された数値に対して第二の数値を法とする剰余演算によって構成される方法」

次に装置のクレームをみると、まずクレーム 1 には、暗号鍵を生成する鍵生成装置（a secure key generator）が記載されている。

クレーム 1：
「鍵生成装置であって、
①第一データを受け取るように接続された第一入力と、
②第二データを受け取るように接続された第二入力と、
③第一出力と、
④第二出力と、
⑤第一出力において逆変換が困難な方法によって上記第一データを変換して第三データを生成し、第二出力において、上記第二および第三データだけから生成することが不可能な方法によって上記第一データで上記第二データを変換し、暗号鍵となる第四データを生成するための手段（means）、
によって構成される装置」

このクレームにおいても、クレーム 2 と同様に、暗号鍵生成に関して「逆変換が困難な方法」としか記載されていない。しかし、「実施例の説明」の中で、離散対数問題を利用した数値変換の方法が記載されている。また、クレーム 5 には、送信者と受信者の鍵生成装置（apparatus）がそれぞれ第一鍵生成装置および第二鍵生成装置として記載されており、それぞれの鍵生成装置の構成要素はクレーム 1 に記載されている内容と同一となっている。クレーム 7 は、クレーム 5 と同様に鍵生成装置のクレームであるが、利用される数値変換の方法が数式を使わずに記載されている。クレーム 8 には、べき乗剰余算が数式によって特定された暗号鍵生成装置（apparatus）が記載されており、鍵共有法の数値変換の方法が明らかになっている。このクレーム 8 では、前述のクレーム 6 とは異なり、べき乗剰余算のなかでも離散対数問題に基づく方式であることが明らかになっている。

クレーム 8：

「安全な暗号鍵を生成するための装置であって、第一鍵生成装置と第二鍵生成装置によって構成され、

①第一鍵生成装置は、第一データ X_i を受信するために接続された第一入力部分と、第二データ Y_j を受信するために接続された第二入力手段と、第一および第二出力手段と、第一出力手段において第三データ Y_i と暗号鍵となる第四データ K_{ij} を生成する手段、ただし、大きな素数 q、$1 \leqq a \leqq q-1$ を満足する乱数 a、$1 \leqq X_i \leqq q-1$ を満足する乱数である第一データ X_i の下で、Y_i を $Y_i = a^{X_i} \bmod q$ という逆変換が不可能な変換により生成し、第二データ Y_j の下で、K_{ij} を $K_{ij} = Y_j^{X_i} \bmod q$ という第二および第三データから生成することが不可能な変換によって生成する装置であり、

②第二鍵生成装置は、第五データ X_j を受信するために接続された第一入力部分と、第三データ Y_i を受信するために接続された第二入力手段と、第一および第二出力手段と、第一出力手段において第二データ Y_j と暗号鍵となる第六データ K_{ji} を生成する手段、ただし、$1 \leqq X_j \leqq q-1$ を満足する乱数である第五データ X_j の下で、Y_j を $Y_j = a^{X_j} \bmod q$ という逆変換が実現不可能な変換により生成し、K_{ji} を $K_{ji} = Y_i^{X_j} \bmod q$ という第二および第三データから生成することが不可能な変換によって生成する装置」

このように、Diffie-Hellman 特許のクレームには、Diffie-Hellman 鍵共有方式における鍵共有方法とその鍵共有方法を利用した暗号通信方法が記載されている。また、べき乗剰余算を利用した鍵共有方法がクレーム 6 に記載されているほか、それが離散対数問題に基づくものであることがクレーム 8 や「実施例の説明」の記載内容から明らかになっている。

6.ブラインド署名特許

（1）RSA ブラインド署名特許

RSA ブラインド署名特許[208]は、ブラインド署名の概念を利用した初めての特許である。

RSA ブラインド署名特許は 41 のクレームを有している。まず、クレーム 1～19 が方法（method）のクレームとなっており、クレーム 20～41 は装置（apparatus）のクレームとなっている。クレーム 1 にはブラインド署名方式の方法が記載されている。

クレーム 1：

「署名者が公開鍵ディジタル署名方式によって変換する前に、複数の署名依頼者がディジタルメッセージを変換し（ブラインド化する）、さらに公開鍵ディジタル署名方式によって変換された後にそのメッセージを変換する（非ブラインド化する）方法であって、公開鍵ディジタル署名は公開鍵によって検証可能であるにもかかわらず、署名者はブラインド化されたディジタルメッセージと、公開鍵ディジタル署名方式によって変換された後に非ブラインド化されたメッセージの対応関係を知ることができないという特徴を有し、

①署名依頼者が、第一鍵によって複数のメッセージを変換してディジタル第一メッセージを生成する（ブラインド化）ステップと、

[208] U. S. Patent Number は、4,759,063（出願日 1983 年 8 月 22 日、発効日 1988 年 7 月 19 日）。

②署名者が、ディジタル第一メッセージを公開鍵ディジタル署名方式で変換して第二メッセージを生成するステップと、
③署名依頼者が、上記第一鍵でディジタル第二メッセージを変換し、もとのディジタルメッセージに公開鍵ディジタル署名が残っているディジタル第三メッセージを生成する（非ブラインド化）ステップによって構成される方法であり、
上記ブラインド化ステップは、署名依頼者が第一鍵を利用して、第一鍵をもっていない署名者にディジタル第三メッセージとディジタル第一メッセージとの間の対応関係を知られないようにするために行われる方法」

このように、クレーム 1 にはメッセージのブラインド化、ブラインド化されたメッセージの署名作成、署名付きメッセージの非ブラインド化の方法が記載されている。もっとも、これらの数値変換の方法は特定されていない。また、クレーム 14 には、このクレーム 1 に記載されているブラインド署名方法を利用した価値移転の方法が記載されている。

クレーム 14：

「公開鍵ディジタル署名方式による変換を受ける前にそのディジタルメッセージを処理し（ブラインド化する）、公開鍵ディジタル署名方式による変換を受けた後に、その結果のディジタルメッセージを処理する（非ブラインド化する）ことによって追跡不可能な価値の移転手段を提供する方法であって、ディジタルメッセージの公開鍵ディジタル署名は公開鍵によって検証可能であるにもかかわらず、公開鍵やそれに対応する秘密署名鍵の所有者でさえも、ブラインド化されたディジタルメッセージと、ディジタル署名によって変換された後に非ブラインド化されたディジタルメッセージとの間の対応関係を実際に知ることが非常に困難となる方法であり、
①署名依頼者が、複数のメッセージを第一鍵で変換してブラインド化し、対応するブラインド化された第一ディジタルメッセージを生成するステップと、
②署名者が、上記第一メッセージを受け取り、それらのうち少なくとも 2 つのメッセージを署名秘密鍵で変換して第二ディジタルメッセージを生成するステップと、
③署名者が、価値の移転と交換で、対応する上記第二メッセージを署名依頼者に送付するステップと、
④署名依頼者が、署名者から対応する上記第二メッセージを受け取って上記第一鍵で変換し、署名者が第一鍵を入手しない限り第一メッセージと第三メッセージを関連付けることができないような方法で、署名者の署名が付されている非ブラインド化された第三メッセージを生成するステップと、
⑤署名検証者が、少なくとも 1 つの上記第三メッセージを受け取り、それに付されている公開鍵ディジタル署名が正当かどうかをチェックするステップと、
⑥署名検証者が、チェックした上記第三メッセージを記録すると同時に、過去の記録からその第三メッセージとディジタル署名が二重使用されていないか否かをチェックし、チェックに合格したディジタル署名と交換で価値を提供するステップにより構成される方法」

さらに、クレーム 15 には、ブラインド化・非ブラインド化にそれぞれ積および逆数の積を利用する方法が記載されている。

クレーム 15：

「クレーム 1、2、3、4、7、8、9、10 と 14 に記載されている方法であって、
①上記ブラインド化のステップには、それぞれのディジタルメッセージに対応する第一鍵を生成して、両者の積を計算するステップが含まれているほか、
②上記非ブラインド化のステップでは、各第二メッセージとそれに対応する第一鍵の逆数の積を計算するステップが含まれている方法」

クレーム 15 のより具体的な数値変換の方法については、クレーム 15 の従属項のクレーム 17 に記載されている。

クレーム 17：

「クレーム 15 に記載されている方法であって、
①上記ブラインド化のステップにおいて、もとのメッセージ m_i は、k_i を第一鍵、e を公開署名鍵、n を公開鍵ディジタル署名方式における法とすると、$t_i = m_i \times (k_i \text{ ^e}) \pmod n$ という演算によって第一メッセージ t_i に変換され、
②上記ディジタル署名の変換のステップにおいて、第一メッセージ t_i に署名を付加した第二メッセージ t_i‘は、秘密署名鍵 d の下で、$t_i' = t_i \text{ ^d} \pmod n$という演算によって生成され、
③上記非ブラインド化のステップにおいて、第三メッセージ m_i‘は、第二メッセージ t_i‘と第一鍵 k_i によって $m_i' = m_i \text{ ^d} \pmod n$を満足する m_i‘として生成される、
という 3 つのステップによって構成される方法」

このクレーム 17 のブラインド署名方式には、署名生成に利用される数値変換の方法にべき乗剰余算が利用されていることが明らかである。なお、「実施例の詳細な説明」の中で、このブラインド署名方式が RSA 方式の署名生成・検証方法を採用している旨が記載されており、その実施例も記載されている。

残りのクレーム 20～41 は装置（apparatus）のクレームである。まず、クレーム 20～29 はクレーム 1～9 に対応する装置のクレームであり、クレーム 31～35 がクレーム 10～14 に対応する装置のクレームである。また、クレーム 37～41 が、クレーム 15～19 に対応する装置のクレームとなっている。

クレーム 20：

「署名者が公開鍵ディジタル署名方式によって変換する前に、複数の署名依頼者がディジタルメッセージを変換し（ブラインド化する）、さらに公開鍵ディジタル署名方式によって変換された後にそのメッセージを変換する（非ブラインド化する）装置であって、公開鍵ディジタル署名は公開鍵によって検証可能であるにもかかわらず、署名者はブラインド

化されたディジタルメッセージと、ディジタル署名によって変換された後に非ブラインド化されたメッセージの対応関係を知ることができないという特徴ともち、
①署名依頼者が、第一鍵によって複数のメッセージを変換してディジタル第一メッセージを生成する（ブラインド化）ための手段（means）と、
②署名者が、ディジタル第一メッセージを公開鍵ディジタル署名方式によって変換して第二メッセージを生成するための手段と、
③署名依頼者が、上記第一鍵でディジタル第二メッセージを変換し、もとのディジタルメッセージに公開鍵ディジタル署名が残っているディジタル第三メッセージを生成する（非ブラインド化）ための手段、
によって構成される装置であり、上記ブラインド化のための手段は、署名依頼者が第一鍵を利用して、第一鍵をもっていない署名者にディジタル第三メッセージとディジタル第一メッセージとの間の対応関係を知られないようにするために利用される装置」

このように、RSA ブラインド署名特許には、ブラインド署名の生成・検証方法と装置のクレームのほか、ブラインド署名方式を利用した価値情報のやり取りの方法のクレームが存在する。署名生成・検証に利用される数値変換の方法はクレーム 17 の中で具体的に記載されており、RSA 方式を利用している点が明らかになっているほか、「実施例の詳細な説明」の中でも RSA 方式を利用した数値変換の方法が記載されている。

（2）太田・岡本型ブラインド認証特許

太田・岡本型ブラインド認証方式[209]は、Fiat-Shamir 署名や Schnorr 署名等の方式をベースにしたブラインド認証方式であり、ユーザー認証やメッセージ認証の際に交換される情報の内容を認証者や確認者に知られないように認証を行うことを可能にする方式である。太田・岡本型ブラインド認証方式の安全性の根拠はベースとなる方式によって異なり、Fiat-Shamir 方式をベースとする場合には素因数分解問題に基づくが、Schnorr 方式をベースとする場合には離散対数問題に基づくことになる。

太田・岡本型ブラインド認証特許は 31 のクレームを有している。まずクレーム 1 には、ユーザー認証システム（system）が記載されている。

クレーム 1：

「証明者 A、認証依頼者 B と確認者 C が関係者として参加するシステムであって、
①ステップ 1：証明者 A は乱数 r を用いて初期情報 x'を作成し、認証依頼者 B に x'を送付する、
②ステップ 2：認証依頼者 B は乱数を生成し、この乱数を用いて x'を変換して x"を生成し、確認者 C に x"を送付する、
③ステップ 3：確認者 C は質問情報 β を生成して認証依頼者 B に送付する、

[209] U. S. Patent Number は 4,969,189（出願日 1989 年 6 月 19 日、発効日 1990 年 11 月 6 日）、ヨーロッパ特許公告番号は 348,812（出願日 1989 年 6 月 21 日、公告日 1990 年 1 月 3 日）。

④ステップ 4：認証依頼者 B は、乱数を用いてβを変換してβ'を生成し、証明者 A にβ'を送付する、
⑤ステップ 5：証明者 A は、秘密鍵 s、乱数 r とβ'から認証情報 z を生成し、認証依頼者 B に z を送付する、
⑥ステップ 6：認証依頼者 B は、z と乱数から z'を生成し、確認者 C に z'を送付する、
⑦ステップ 7：確認者 C は、z'が正当なデータかどうかを x''とβを利用して確認する、
というステップによって構成されるシステムであり、認証依頼者 B が変換に利用する乱数を秘密にしておくことによって、認証依頼者 B と証明者 A との間でやり取りされる情報（x', β', z）と、確認者 C と認証依頼者 B との間でやり取りされる情報（x'', β, z'）との間の関連性を誰にも知られないようにする認証システム」

このクレーム 1 には、ユーザー認証システムの具体的な数値変換の方法が記載されていないが、「実施例の説明」の中で、本認証システムのほか、本ユーザー認証システムのベースとなる Fiat-Shamir 方式、拡張された Fiat-Shamir 方式や離散対数問題に基づく方式が詳細に記載されている。一方、クレーム 2 では、メッセージ認証システム（system）が記載されている。

クレーム 2：

「証明者 A、署名依頼者 B と確認者 C が関係者として参加するシステムであって、
①ステップ 1：証明者 A は、乱数 r を用いて初期情報 x'を生成し、署名依頼者 B に x'を送付する、
②ステップ 2：署名依頼者 B は、乱数を生成し、この乱数を用いて x'から x''を生成する、
③ステップ 3：署名依頼者 B は、x''とメッセージ m からβを生成し、ステップ 2 で生成した乱数を用いてβからβ'を生成して証明者 A に送付する、
④ステップ 4：証明者 A は、秘密鍵 s、乱数 r とβ'から z を生成して、署名依頼者 B に z を送付する、
⑤ステップ 5：署名依頼者 B は、ステップ 2 で生成した乱数と z から z'を生成し、メッセージ m、βと z'を確認者 C に送付する、
⑥ステップ 6：確認者 C は、z'とβがメッセージ m の正当な署名かどうかをチェックする、
というステップによって構成されるシステムであって、署名依頼者 B が乱数を秘密にしておくことによって、署名依頼者 B と証明者 A との間でやり取りされる情報（x', β', z）と、確認者 C と署名依頼者 B との間でやり取りされる情報（m, β, z'）との間の関連性を誰にも知られないようにする認証システム」

このクレーム 2 についても、メッセージ認証システムの数値変換の方法が記載されていないが、前述のユーザー認証システムと同様に「実施例の説明」に記載されている。クレーム 1 の従属項のクレーム 3 には、ステップ 1、2、4 と 5～7 の各ステップでの演算に剰余算を利用し、ステップ 5 においては秘密鍵 s が公開情報 x と少なくとも 2 つの素数の合成数 N の下で $s^2 \bmod N = x$ を満たすように決定されるシステムが記載されている。

クレーム 3：

「クレーム 1 に記載されている認証システムであって、

①上記ステップ 1 において、上記証明者 A は、上記乱数 r と公開情報 x を利用した剰余算によって上記初期情報 x'を計算し、

②上記ステップ 2 において、上記認証依頼者 B は、上記 x'、上記乱数、上記公開情報 x を利用した剰余算によって上記 x''を計算し、

③上記ステップ 4 において、上記認証依頼者 B は、上記 β と上記乱数を利用した剰余算によって上記 β'を計算し、

④上記ステップ 5 において、上記証明者 A は、上記乱数 r、上記 β'、上記秘密鍵 s を利用した剰余算によって上記 z を計算する、ただし、少なくとも 2 つの素数の合成数である N を法として、$s^2 \bmod N = x$ を満足するように秘密鍵 s は決定される、

⑤上記ステップ 6 において、上記認証依頼者 B は、上記 z'を少なくとも上記 z、上記乱数と上記公開情報 x を利用する剰余算によって計算し、

⑥上記ステップ 7 において、上記認証者 C は、認証依頼者 B から受け取った x''が上記 z'、上記 β と上記公開情報 x を利用した剰余算によって計算した値に等しいかどうかを調べることにより認証を行う、という認証システム」

このクレーム 3 からも、ユーザー認証システムの数値変換の方法が明らかではないが、「実施例の説明」には、Fiat-Shamir 方式をベースとしたブラインド認証方式の具体的な数値変換の方法が記載されている。同様に、クレーム 2 の従属項のクレーム 4 にも、クレーム 2 のステップ 2〜6 における演算に剰余算が利用されるシステムが記載されている。クレーム 1 の従属項のクレーム 5 と、クレーム 2 の従属項のクレーム 6 には、それぞれ複数の秘密鍵 s_j を利用する認証システム（system）が記載されており、秘密鍵 s_j は $s^2_j \bmod N = x_j$ を満足するように決定されると記載されている。

クレーム 1 の従属項のクレーム 7 には、クレーム 1 のユーザー認証システムにおいて、ある整数 L の下で、$s^L \bmod N = x$ を満足するように決定された秘密鍵 s を利用するユーザー認証システムが記載されている。このユーザー認証システムは、拡張された Fiat-Shamir 方式をベースにしたブラインド認証方式に対応するとみられるが、「実施例の説明」の中でこの方式の具体的な数値変換の方法が記載されている。また、クレーム 2 の従属項のクレーム 8 には、クレーム 7 と同様に $s^L \bmod N = x$ によって決定された秘密鍵 s を利用するメッセージ認証システムが記載されている。

クレーム 1 の従属項のクレーム 9 には、クレーム 1 のステップ 1 を、公開情報 g の r 乗を剰余算することで x'を計算するというステップに置き換え、ステップ 2 と 7 に g を利用して剰余算を行うステップを付加したユーザー認証システムが記載されている。クレーム 2 の従属項のクレーム 10 でも同様に g を用いた剰余算を利用するステップを有するメッセージ認証システムが記載されている。さらに、クレーム 9 の従属項のクレーム 14 には、離散対数問題に基づく方式をベースとしたユーザー認証システムが記載されている。また、クレーム 10 の従属項のクレーム 16 には、離散対数問題に基づく方式をベースとしたメッセージ認証システムが記載されている。

クレーム 14：

「クレーム 9 に記載されている認証システムであって、上記ステップ 1、2 と 7 の剰余算はそれぞれ公開される素数 P を法とする剰余算とし、上記ステップ 5 と 6 の剰余算はそれぞれ P-1 を法とする剰余算とし、上記秘密鍵 s が $g^s \bmod P=x$ を満足するように決定される認証システム」

クレーム 16：

「クレーム 10 の認証システムであって、上記ステップ 1、2 と 6 の剰余算はそれぞれ公開される素数 P を法とする剰余算とし、上記ステップ 4 と 5 の剰余算はそれぞれ P-1 を法とする剰余算とし、上記秘密鍵 s が $g^s \bmod P=x$ を満足するように決定される認証システム」

クレーム 19～31 は、メッセージ認証のための署名装置（device）のクレームである。まず、クレーム 19 には、この署名装置を構成する 4 つの手段が記載されている。

クレーム 19：

「証明者 A が確認者 C に対して署名依頼者 B のメッセージの正当性を証明するためのメッセージ認証システムにおける署名装置であって、

①乱数を生成する乱数生成手段（means）と、

②証明者 A からの初期情報 x'を上記乱数によってランダム化し、ランダム化された初期情報 x"を生成する初期情報ランダム化手段と、

③メッセージ m と上記のランダム化された初期情報 x"から質問情報βを出力し、βと上記乱数を使ってβ'を生成する質問情報生成手段と、

④証明者 A の生成した情報 z と上記乱数を受け取り、z から上記乱数の影響を除去し、非ランダム化された情報 z'を生成する非ランダム化手段によって構成され、

上記メッセージ m、上記βと z'を確認者 C に送付する装置」

このように、クレーム 19 には署名装置の 4 つの構成要素が記載されているが、これらの各構成要素の詳細な内容は「実施例の説明」に記載されている。クレーム 19 の従属項のクレーム 28～31 には、クレーム 19 の４つの手段の演算内容が数式で表現された装置（device）が記載されている。例えば、クレーム 28 には、Fiat-Shamir 方式をベースとしたブラインド認証方式を利用した署名装置が記載されている。

クレーム 28：

「クレーム 19 の署名装置であって、

①上記乱数生成手段は、上記乱数のほかに、t 個のランダムビット e_i と乱数 u_i（i=1,...,t）を生成する手段であり、

②上記初期情報ランダム化手段は、t 個の e_i と u_i、公開情報 x と上記 x'を証明者 A から受

け取り、t 個の x''_i を 2 つの秘密の素数の合成数 N を利用して、$x''_i=(u_i^2)(x^{-e_i})(x'_i) \pmod N$ という計算によって生成する手段であり、

③質問情報生成手段は、メッセージ m と x''_i を受け取り、一方向関数 f によって $\beta_i=f(m,x''_i)$ を計算し、β_i を出力する一方向関数手段と、β_i と e_i を受け取って $\beta_i+e_i \bmod 2$ を計算し、その結果の β'_i を出力する剰余算手段を有し、

④非ランダム化手段は、上記 u_i、e_i、β_i、x、N と z_i を証明者 A から受け取って、z'_i を、$e_i=1$ かつ $\beta_i=0$ の場合には $z'_i=(u_i)(z_i)(x^{-1}) \bmod N$ によって計算し、上記以外の場合には $z'_i=(u_i)(z_i) \bmod N$ によって計算する手段を有する署名装置」

同様に、クレーム 30 には、拡張された Fiat-Shamir 方式をベースとしたブラインド認証方式に利用される署名装置が記載されており、さらにクレーム 31 には、離散対数問題に基づく方式をベースとしたブラインド認証方式に利用される署名装置が記載されている。これらのクレームには、その数値変換の方法が記載されている。

このように、太田・岡本型ブラインド認証特許には、ユーザーとメッセージのブラインド認証の方法のクレームが存在し、Fiat-Shamir 方式、拡張された Fiat-Shamir 方式や離散対数問題に基づく方式をベースとしたブラインド認証方式のクレームが存在する。また、メッセージ認証における署名装置のクレームも存在する。署名装置のクレームでは、各ブラインド認証方式の具体的な数値変換の方法が明らかになっているほか、「実施例の説明」においても詳細に記載されている。

Ⅴ 特許法による暗号の保護

1．保護の対象とクレーム

（1）日本の特許の場合

日本の特許庁は、特許法に「発明」を定義する規定が存在することから、コンピュータ・ソフトウェアを対象とする特許を正面から認めることに躊躇あるいは戸惑いを示していたが、コンピュータ・ソフトウェアがハードウェアと関連付けられている場合には、特許法の保護の対象として従前から認められている機械と同様に捉えることができるとして、装置をクレームとする特許は認めてきた。暗号についても例外ではなく、装置のクレームとすることによって特許が認められてきた[210]。

DES 暗号特許[211]のクレーム 1 を例に、クレームの記載内容について考える。クレーム 1 は装置クレームである。

DES **暗号特許（日本）のクレーム 1：**

「一組の暗号キー・ビットの制御のもとにデータ・ビットのブロックに対して積ブロック暗号処理操作を実行するための暗号装置にして、上記データ・ビットのブロックを記憶するための記憶手段と、上記一組の暗号キー・ビットを置換して出力するための第一置換手段と、上記記憶手段に接続され、上記ブロック中の選択されたデータ・ビットを二重にすることによって、上記第一置換手段から出力された暗号キー・ビットの数に等しいデータ・ビットのブロックを生成するための手段と、該手段からの拡張されたデータ・ビットのブロック及び上記第一置換手段からの置換された暗号キー・ビットを論理的に組み合わせて代替変換を実行することにより、元のデータ・ビットの数に等しいビット数を有する代替ビット群を生成するための手段と、該手段から出力された上記代替ビット群を置換することにより、上記データ・ビットのブロックの積ブロック暗号を生成するための第二置換手段とを有することを特徴とする暗号装置」

このクレーム 1 は暗号装置のクレームとなっているが､数値変換の方法が記載されていない。数値変換の方法を含まない装置構成のクレームについては、特許の保護の対象とされない抽象的な思想を指すものと解釈することも可能であろう。ただ、このような装置構成のクレームは、数値変換の方法ではなく、装置構成によって発明を特定させようとする特許庁の行政指導に一因を有するものであり、コンピュータ・ソフトウェアの特許法による保護のこれまでの経緯からすれば、これを無効とすべきものではないであろう[212]。もっとも、後述のよう

---

210 実際に、前章で取り上げた日本の暗号に関連する 4 つの特許のクレームを分類してみると、装置および方式クレームは 20、方法クレームが 4 つとなっていることがわかる。

211 特許公告番号は昭 59-45269。

212 このような装置構成による特許が認められていることは、数値変換の方法を特定しないことによって、広い特許の保護の範囲を主張する可能性を求めるという出願人の意図にも添うものとなっている。

に、その権利の範囲については、装置構成に過度に捕らわれることなく、発明の本質である数値変換の方法にも配慮すべきであると思われる。

一方、具体的な数値変換の方法が記載されているクレームも存在する。例えば、日本のESIGN 署名特許[213]のクレーム 1 が挙げられる。このクレームは署名文書通信方式のクレームである。

ESIGN **署名特許（日本）のクレーム 1：**

「送信側で素数 p、q（p＞q）を秘密情報として用意し、

①これら素数 p、q を用いて、公開情報 $n=p^2q$、δ（$1 \leqq \delta \leqq \theta(n^{2/3})$、θ(B)は B のオーダーを意味する）、$\gamma = \theta(n/T)$（T は 10^{10}～10^{30} 程度の値）を作り、

②これら公開情報 n、δ、γ を送信者 D の識別番号（ID）と共に公開簿に登録しておき、

③乱数 X（1≦X≦pq-1、X は n と互いに素な整数）を生成し、

④その乱数 X に対して f(X)（mod n)（$f(X)=\Sigma^{I}_{i=0} g_i \cdot X^i$（i=1,2,...,I）、0＜$g_i$≦n-1、$g_i$ は整数、I≧3）を演算し、

⑤その演算結果と送信すべき文書 m との差 Z=m-（f(X) (mod n)）を求め、

⑥その Z を pq で割算し、

⑦その割算結果を切り上げて W=「Z/(pq)」を求め、

⑧上記乱数 X に対し、f ‘(X) mod p（$f'(X)=\Sigma^{I}_{i=0} i \cdot g_i \cdot X^{i-1}$）を演算し、

⑨その演算結果で上記 W を割算し、

⑩この演算結果 y と上記 pq とを乗算し、

⑪これに乱数 X を加算して署名 S とし、

⑫この署名 S と上記文書 m と、上記識別番号 ID とを送信し、

⑬受信側で受信した上記識別番号 ID を利用してその公開情報 n、δ、γ を上記公開簿より求め、

⑭受信した上記署名 S に対し、f(S) (mod n)を演算し、

⑮その演算結果と、上記文書 m と上記δとを用いて m≦f(S) (mod n)＜m+δが成立するかどうかを検証し、

⑯上記 n と上記γと、上記署名 S とを用いてγ≦S≦n-γが成立するかどうかを検証し、

⑰上記両検証が共に成立した場合に、上記受領したm、S は上記公開簿に（n、δ、γ）を登録した者によって確かに作成されたものであるとする署名文書通信方式」

（2）アメリカの特許の場合

アメリカ合衆国特許商標庁も、特許発明の保護の対象がクレームによって特定されるため、実質的に保護の対象の問題をクレームの記載方法の問題としている。通信方法、通信装置が特許の対象となるとした判決例や、数値変換の方法を特許の対象とならないとした判決例はあるが、暗号についての判決例はない。そのため、暗号に関してどのようなクレームが認められるかという点は、判例上明確であるとはいえない。また、特許商標庁における審査実務は、日本特許庁の審査実務に比べるとガイドラインの実質的な拘束力は強くない。

[213]特許公告番号は平 5-86699。

アメリカの Merkle-Hellman 特許[214]では、以下のクレーム 1 が認められている。

Merkle-Hellman **特許（アメリカ）のクレーム 1：**

「メッセージを送信者から受信者に送付するというタイプの通信を、機密保持性のないチャネルにおいて安全に実行するための方法であって、
①受信者は乱数を生成し、
②受信者は、上記乱数から受信者用の公開暗号鍵を生成し、
③受信者は、上記乱数から、公開暗号鍵と直接関係しているが公開暗号鍵からは計算量的に生成することが実現不可能な秘密復号鍵を生成し、
④受信者は、送信者に公開暗号鍵を送付し、
⑤送信者は、公開暗号鍵を利用すると変換が容易であるが秘密復号鍵なしではその逆変換が計算量的に不可能となるような暗号変換によって公開暗号鍵で暗号文を生成し、
⑥送信者は、暗号文を受信者に送付し、
⑦受信者は、暗号文を復号化するために暗号文を秘密復号鍵で変換する方法」

このクレームは、公開鍵暗号を利用した基本的な暗号通信方法のクレームであり、公開鍵暗号についての広いクレームが認められた例ということができるであろう。

一方、RSA 暗号特許[215]は、素因数分解問題に基づく数値変換の方法を含む以下のクレーム 23 を有している。

RSA **暗号特許（アメリカ）のクレーム 23：**

「暗号通信を行うための方法であって、素数 p と q に対して n = pq となる合成数 n を生成し、この n に対して 0≦M≦n-1 を満足するディジタルメッセージ M を暗号文 C に変換するというステップによって構成される方法で、(p-1)(q-1)と互いに素な関係にある整数 e に対して、M を C = M^e (mod n)を満たす C に変換することによって表される方法」

これは、先行技術として公開鍵暗号の基本的な暗号通信方法のクレームが Merkle-Hellman 特許に存在するところから、数値変換の方法を限定していると考えることができるであろう。

RSA 暗号特許の他にも、具体的な数値変換の方法を特定しているクレームを有する特許として Fiat-Shamir 署名特許[216]が存在する。Fiat-Shamir 署名特許のクレーム 10 には、ディジタル署名生成・検証方法が記載されている。

Fiat-Shamir **署名特許（アメリカ）のクレーム 10：**

「被認証者と認証者との間で交換されるメッセージ m に署名する方法であって、
（a）被認証者は、乱数 r_1 ,...,$r_t \in (0,n)$を生成し、

[214] U. S. Patent Number は 4,218,582。
[215] U. S. Patent Number は 4,405,829。
[216] U. S. Patent Number は 4,748,668。

（b）被認証者は、$x_i = r_i^2 \pmod n$を計算して x_i を認証者に送付し、
（c）被認証者は、$f(m,x_1,\ldots,x_t)$を計算して、その計算結果のうちの kt ビットを e_{ij} として設定し、
（d）被認証者は、$y_i = r_i \prod_{eij=1} s_j \pmod n$を計算し、
（e）被認証者は、I、j、m、e_{ij} と y_i を認証者に送付し、
（f）認証者は、$v_j = f(I,j)$を計算し、
（g）認証者は、$z_i = y_i^2 \prod_{eij=1} v_j \pmod n$を計算し、
（h）認証者は、$f(m,z_1,\ldots,z_t)$を計算して、その計算結果のうちの kt ビットを取り出して e_{ij} と一致するかどうかをチェックする、という方法」

実際にクレームを整理すると、暗号に関する特許のクレームは数値変換の方法をクレームしているとも解釈できるが、仮に、現在でも Gottschalk v. Benson 事件の判決の考え方が有効であるとすると、問題が全くないとは言えない。しかし、通信方法が特許の対象となる以上、RSA 暗号特許のクレーム 23 のように「暗号通信方法」のクレームとすることによって、暗号通信の方法も特許の対象となると考えることもできる。また、このような数値変換の方法を含むクレームは、その方法の技術的核心が数値変換の方法にのみ存在することから、もしも Parker v. Flook 事件の判決の考え方が有効であるとすると、問題が全くないとは言えない。これも、暗号通信の方法というクレームにすることで特許の対象となると考えることも可能である。

2．保護を受ける要件

（1）新規性

先行技術に同じ技術が含まれていれば、特許出願は新規性を欠くとして認められないことになる。特許出願された技術が先行技術に含まれているか否かは、その特許出願のクレームを基準として判断されることになる。

1976 年に Diffie と Hellman が公開鍵暗号の原理を論文で発表した後、日本の Merkle-Hellman 特許が 1978 年に出願されている。そこで、Merkle-Hellman 特許のクレーム 2 を例に、このクレームの新規性について検討する。

Merkle-Hellman **特許（日本）のクレーム 2：**

「機密保持性のない通信チャンネルを経て送信さるべきメッセージを暗号化する装置であって、
①機密性を保持すべきメッセージを受け取る様に接続された入力と、公開の暗号作成キーを受け取る様に接続された別の入力と、暗号化されたメッセージを出力する出力とを備えた様な装置に於いて、
②メッセージを受け取り、そしてこのメッセージをそのベクトル表示に変換する手段と、上記公開の暗号作成キーを受け取り、そしてこの公開の暗号作成キーをそのベクトル表示に変換する手段と、上記メッセージのベクトル表示と上記公開の暗号作成キーのベクトル表示とのドット積を計算することによって暗号化メッセージを作り出す手段であ

って、上記メッセージのベクトル表示を受け取る様に接続された入力と、上記公開の暗号作成キーのベクトル表示を受け取る様に接続された別の入力と、暗号化されたメッセージを出力するための出力とを有する様な手段とを備えたことを特徴とする装置」

このクレーム 2 には、公開鍵暗号を利用した暗号化装置が記載されており、メッセージと暗号作成キーをベクトル表示に変換する手段と、それらのドット積を計算することによって暗号化メッセージを作り出す手段が記載されている。暗号作成キーがベクトルとして暗号化に利用されていることから、この暗号作成キーがナップザック問題におけるナップザックベクトルに対応すると解釈すれば、この暗号化装置はナップザック問題に基づく公開鍵暗号の方式を利用していると考えることが可能であろう。Diffie と Hellman が発表した公開鍵暗号の原理との関係が問題となるが、Diffie と Hellman が示したのは抽象的な公開鍵暗号のアイデアのみであり、実現方法を発表したわけではない。これに対し、本クレームは公開鍵暗号を利用した暗号化装置を特定したものであり、公開鍵暗号の 1 つの実現方法を示している。その意味で、本クレームは新規性を有すると判断することができよう。

次に、アメリカの RSA 暗号特許のクレーム 25 を取り上げ、Merkle-Hellman 特許に対する新規性について検討する。

RSA **暗号特許（アメリカ）のクレーム 25：**

「各々の暗号鍵 E[i]=（e[i],n[i]）と復号鍵 D[i]=（e[i],n[i]）によって特徴づけられる k 個のターミナルが存在する通信システムにおいてメッセージ M[i]（i = 1,2,…,k）を送受信する方法であって、ターミナル A からターミナル B に送付されるメッセージ M[A]を変換するステップによって構成され、上記変換ステップは、M[A]から署名済みメッセージ M[As]を作成する M[As] = M[A]^d[A] mod n[A] という変換を行うサブステップによって構成される方法。ただし、M[i]は 0≦M[i]≦n[i] -1 を満たし、n[i]は n[i]=p[i]q[i]となる合成数であり、p[i]と q[i]は素数であり、e[i]は（p[i]-1）と（q[i]-1）の最小公倍数と互いに素な整数であり、d[i]は（p[i]-1）と（q[i]-1）の最小公倍数を法とした場合に e の逆数となる整数である」

このクレームは、素因数分解問題に基づく公開鍵暗号の方式を利用してディジタル署名を生成する方法のクレームとなっている。このディジタル署名の生成方法については、RSA 暗号特許に先行して Merkle-Hellman 特許が明細書の中で説明しているが、 Merkle-Hellman 特許の明細書にはナップザック問題に基づく公開鍵暗号方式を利用したディジタル署名方式のみが記載されている。したがって、RSA 暗号特許のクレーム 25 は、素因数分解問題に基づく方式をディジタル署名に利用しているという点で、Merkle-Hellman 特許に対して新規性を有していると考えられる。

また、IDEA 暗号特許[217]のクレーム 1 における、Lucifer 暗号特許、DES 暗号特許と FEAL

[217] U. S. Patent Number は 5,214,703。

暗号特許に対する新規性について検討する。

IDEA **暗号特許（アメリカ）のクレーム 1：**

「自由に選択可能な制御ブロックによって、データブロックを同じブロック長（第一 bit 番号 N）のブロックに変換するための装置であって、

①データブロックを入力し、それを 2 つ以上の同じブロック長（第二 bit 番号 m）のサブブロックに分割する第一入力手段（means）と、

②サブブロックと同じブロック長の 2 つ以上の制御ブロックを受け取る第二入力手段と、

③制御ブロックを利用して、少なくとも 4 つの処理ユニットを含む、少なくとも 2 種類の連続する変換処理によってサブブロックを変換する演算手段であり、それぞれの処理ユニットは、変換処理のためのサブブロックと制御ブロックを受け取るための 2 つの入力機能と、変換後のブロックを出力する出力機能を備えており、連続する 2 つの処理ユニットの変換方法が異なるように配置され、3 種類の処理方法のいずれかの方法が利用されるような演算手段と、

④少なくとも 2 つの最終変換処理後のサブブロックを出力する出力手段、とによって構成される装置」

このクレーム 1 には、第一入力手段、第二入力手段、演算手段と出力手段によって構成される装置が記載されている。まず、Lucifer 暗号特許で開示されている暗号化の方法とクレーム 1 を比較すると、第一入力手段、出力手段については Lucifer 暗号特許にも含まれていると考えられる。しかし、第二入力手段については、Lucifer 暗号特許の暗号化方法では、サブブロック（24 bit）と鍵ブロックに相当する制御ブロック（8 bit）が異なるブロック長となる構造になっているため、第二入力手段が Lucifer 暗号特許の技術に含まれるとは考えにくいほか、演算手段についても、Lucifer 暗号特許において変換処理に相当すると考えられる 3 つの手段（非線形変換手段、線形変換手段、排他的論理和手段）のうち、4 つ以上の処理ユニットを含んでいるのは非線形変換手段のみと考えられるため、IDEA 暗号特許のクレーム 1 の演算手段も Lucifer 暗号特許に含まれているとは言い難い。これらの点を考慮すると、IDEA 暗号特許のこのクレームは Lucifer 暗号特許に対して新規性を有すると考えられる。次に DES 暗号特許との関係についても同様にして調べると、DES 暗号特許に開示されている暗号化方法では、データのサブブロック（32 bit）と鍵ブロックに相当する制御ブロック（48 bit）が異なるブロック長となる構造になっているため、Lucifer 暗号特許と同様、第二入力手段が DES 暗号特許に開示されている暗号化方法に含まれるとは言い難い。また、FEAL 暗号特許に開示されている暗号化方法と比較すると、第一入力手段、第二入力手段と出力手段は FEAL 暗号特許に含まれると考えられるが、演算手段については含まれるとは考えにくい。これは、FEAL 暗号特許においては、処理ユニットに相当する法 256 加算手段、左 2 bit シフト手段と排他的論理和手段を含む変換処理（f 関数に相当）は 1 つしか存在せず、少なくとも 2 種類の連続する変換処理を有する演算手段は FEAL 暗号特許には含まれないと考えられるためである。このように、IDEA 暗号特許のクレーム 1 は、Lucifer 暗号特許、DES 暗号特許と FEAL 暗号特許に対して

新規性を有していると考えられる。

（2）進歩性

暗号は、数学上の問題を応用している。そこで、発明に利用されている数学的問題が先行技術と比較したときに進歩性を有するかどうかが重要となる。

アメリカの Merkle-Hellman 特許が先行技術として存在する場合、公開鍵暗号という暗号の方法では共通するが、使われている数学的問題が異なる RSA 暗号特許出願が非自明性を有するか否かが問題となる。公開鍵暗号にはすべての数学的な問題が利用可能である訳ではなく、特定の数学的問題のみが利用可能であるから、素因数分解問題という別の数学的問題に基づく RSA 暗号の発明は非自明性を有すると言うべきであろう。

また、アメリカの ESIGN 署名特許[218]のディジタル署名方式は、素因数分解問題と合同多項不等式問題を利用した署名方式である。ESIGN 署名特許のディジタル署名方式は、素因数分解問題を安全性の根拠としている点で RSA 方式と共通しているものの、RSA 方式とは異なった署名生成・検証方法を利用することで、より高速な処理を実現している点が特徴である。ESIGN 署名特許の明細書には、「合同多項式の次数を RSA 法の次数に比べ十分小さくすれば、RSA 法に比べ高速な署名作成／検証が可能になる。例えば RSA 法の次数は 10^{200} 程度であるのに対し、この発明の合同多項式の次数を 10^2 程度以下にすれば、ほぼ 100 倍程度以上高速な処理が見込まれる」と記載されている。これらの点から、ESIGN 署名特許は、処理速度の向上という点において RSA 暗号に対して非自明性を有しているとみられる。

アメリカの DSA 署名特許[219]のディジタル署名方式には、ElGamal 署名方式の署名生成・検証式と Schnorr 署名方式[220]の署名圧縮技術が利用されている。実際に、ElGamal 署名方式と DSA 署名特許のディジタル署名方式の署名生成・検証式を比較してみると、ElGamal 署名方式の署名生成式は① s =k^(-1)*(m-xr) mod (p-1) と② r = g^k mod p、署名検証式は③ g^m = (r^s)*(y^r) mod p となるが、①式において、m を-H(m)に、s を-s に、そして法(p-1)を q に置きかえるとともに、②式の右辺に mod q の演算を加えることで、DSA 署名生成式に一致することがわかる。また、署名検証式についても、③式を変形すると r=g^(m/s)*y^(-r/s) mod p となり、m を-H(m) に、s を-s に、そして mod q の演算を加えることによって DSA 署名検証式と一致することがわかる。さらに、DSA 署名特許の明細書には、②式と③式の g には p-1 の素因数 q を位数とする原始根を利用すると記載されており、Schnorr 署名方式の署名圧縮技術を利用していることが明らかである。

このように、DSA 署名特許に記載されているディジタル署名方式は、ElGamal 署名方式と Schnorr 署名方式を組み合わせた方式であると考えられるが、これらの技術の存在を前提

---

[218] U. S. Patent Number は 4,625,076。

[219] U. S. Patent Number は 5,231,668。

[220] ElGamal 署名方式は、特許が取得されていないが、Schnorr 署名方式は、Schnorr 署名特許（U. S. Patent Number 4,995,082）によってカバーされている。

とした場合に、DSA 署名方式のような技術の組み合わせが容易に想定できるかどうかが問題となる。DSA 署名方式は、２つの方式を組み合わせることにより、ElGamal 署名方式における高い安全性を維持しつつ、より高速での処理を可能にした方式である点を考慮すると、非自明性を有するとの判断が妥当であると考えられる。

（3）開示

特許の明細書には、出願時の技術状況の下でその特許を実施することができるように記載されていなければならず、技術内容の開示が不十分な場合には特許は与えられないし、たとえ与えられたとしてもその特許は無効となる。暗号の場合には、その暗号において利用される数値変換の方法が開示されなければならない。したがって、どの程度数値変換の方法が開示されているのかが問題となる。

日本の Merkle-Hellman 特許では、クレームには数値変換の方法が含まれていないが、明細書の「発明の詳細な説明」には数値変換の方法が含まれているので、明細書における開示としては十分であろう。明細書において、どの程度の開示がなされなければならないかについては、出願時の技術状況によって決定される。

Merkle-Hellman **特許の明細書における開示の一部：**

「キー発生器にはナップザックベクトル a'が与えられ、これは(1)式（$a_i > \Sigma^{i-1}_{j=1} a_j$）を満足するものであり、したがって S'=a'＊x を解くことができる様にし、そして容易に解かれたナップザックベクトル a'を、$a_i=w＊a_i'$ モジュロ m（3）という関係によってトラップドア・ナップザックベクトル a に変換する。このベクトル a はライン E 上の公開暗号作成キーE として働き、そして公開ファイルに入れられるか又は機密保持性のないチャネルを経て送信者に送られる。従って送信者も盗聴者もこの暗号作成キーを入手することができる。送信者は a に等しいこの暗号作成キーE を用いて、S=a＊x にせしめることにより、ベクトル x で表された暗号でないテキストメッセージ X から暗号テキスト S を作り出す。..（中略）..送信者の暗号解読装置にはその機密の暗号解読キーD として w、m 及び a'が与えられ、次式を容易に解くことができる。S=..（中略）.. $=\Sigma\ x_i＊a_i'$ モジュロ m（7）..（中略）..このナップザックの問題は x に対して容易に解かれ、これは更に難しいトラップドア・ナップザックの問題 S=a＊x に対する解でもある。それ故、受信者は 2 進数ベクトル x として表わされた暗号でないテキストメッセージ X を復元することができるし、盗聴者のトラップドアナップザックの問題は計算して解くことができず、盗聴者は暗号でないテキストメッセージを復元することができない」

アメリカの Lucifer 暗号特許[221]では、Lucifer 暗号特許の「発明の詳細な説明」の中に、換字変換に関する説明が抽象的で、拡張変換の方法や転置変換の方法について十分な説明がない。この場合、発明日当時の技術に拡張変換の方法や転置変換の方法が含まれていると解釈すれば

[221] U. S. Patent Number は 3,798,359。

問題ないが、含まれていないと解釈すれば開示が不十分であるということになる[222]。

Lucifer **暗号特許の明細書における換字変換の説明の部分：**

「図 5A と 5B には、換字変換ユニット S のより詳細な図が示されている。換字変換ユニット S が n 個の入力を有する場合には、ユニット S の内部で $2^n!$の変換方法のうちの１つの方法によって非線形変換が実行される。図 4 に示されている実施例では、各換字変換ユニットは 4 bit のデータの入力を有している。この換字を実行する簡易な方法は、暗号化と復号化の手続きを実行することである。具体的には、入力される可能性のある 16 個のデータのうちの１つの 4 bit のデータが、入力ライン M によって復号化装置に送信され、復号化装置において 16 bit のデータに変換される。..（中略）..同様に、暗号化装置においては、16 bit の入力データが 4 bit のデータに変換され、出力ライン E によって出力される。..（中略）..$2^n!$個の変換は、復号化装置の出力と暗号化装置の入力とを結ぶラインの組み合わせの数に対応する」

このように、換字変換ユニットは、①「復号化装置」によって 4 bit データを 16 bit データに変換し、② 16 bit データを転置した後、③「暗号化装置」によって 16 bit データを 4 bit データに変換する、という機能を有していることがわかる。しかし、「復号化装置」や「暗号化装置」における変換方法が特定されていない。

3．特許の効力と保護の範囲

（1）特許の効力

特許の侵害は、基本的には、第三者がクレームに記載された発明を特許権者の許諾を得ることなく実施することによって成立する。そこで、クレームの記載によって、侵害行為の成立が左右される可能性がある。

日本の Merkle-Hellman 特許のクレーム 1 のように、復号化装置というクレームになっている場合、復号化のみは受信者の単独の行為によってなされるので、受信者の特許の侵害という捉え方をすることができる。

一方、アメリカ合衆国の Merkle-Hellman 特許のクレーム 1 では、送信者と受信者が存在する暗号通信方法が記載されているが、この方法の実現には送信者と受信者の二者を必要としており、アメリカ合衆国では共同侵害や誘導侵害が認められる可能性もある。なお、この特許のクレーム 13 には復号化装置が含まれているので、受信者行為も特許侵害となる可能性もある。

（2）保護の範囲

先行する特許の権利の範囲がどこまでかということは、クレームを基礎として判断されることになる。もっとも、クレームには解釈の余地があり、その解釈によって特許の権利の範囲が異なってくることになる。

---

[222] 開示が不十分であるとすれば、特許は無効ということになる。

アメリカの Merkle-Hellman 特許のクレーム 1 には、公開鍵暗号を利用した暗号通信の方法が記載されている。

Merkle-Hellman 特許（アメリカ）のクレーム 1：

「メッセージを送信者から受信者に送付するというタイプの通信を、機密保持性のないチャネルにおいて安全に実行するための方法であって、
①受信者は乱数を生成し、
②受信者は、上記乱数から受信者用の公開暗号鍵を生成し、
③受信者は、上記乱数から、公開暗号鍵と直接関係しているが公開暗号鍵からは計算量的に生成することが実現不可能な秘密復号鍵を生成し、
④受信者は、送信者に公開暗号鍵を送付し、
⑤送信者は、公開暗号鍵を利用すると変換が容易であるが秘密復号鍵なしではその逆変換が計算量的に不可能となるような暗号変換によって公開暗号鍵で暗号文を生成し、
⑥送信者は、暗号文を受信者に送付し、
⑦受信者は、暗号文を復号化するために暗号文を秘密復号鍵で変換する方法」

このクレームには、公開鍵暗号に用いられる数値変換の方法が特定されていない。クレームの文言からは、公開鍵暗号を用いたすべての暗号の方式が特許の権利の範囲に含まれることになる。したがって、ある数学的問題を利用することによって公開鍵暗号を実現する最初の方法が Merkle-Hellman 特許によって示されている場合、公開鍵暗号についての権利を Merkle-Hellman 特許に認めるべきであるというのが自然な解釈であると思われる。これに対し、この特許において開示されているのはナップザック問題を用いた方式のみであるから、特許の保護の範囲はナップザック問題を用いた公開鍵暗号に限定されるべきであるとする考え方もありうる。なお、Merkle-Hellman 特許のクレーム 4 には、公開鍵暗号を利用したディジタル署名の方法が記載されており、クレーム 1 の保護の範囲と同様の議論が当てはまるであろう。

アメリカの RSA 暗号特許のクレーム 23 には、素因数分解問題を利用した暗号通信における暗号化の方法が記載されている。

RSA 暗号特許（アメリカ）のクレーム 23：

「暗号通信を行うための方法であって、素数 p と q に対して n = pq となる合成数 n を生成し、この n に対して 0≦M≦n-1 を満足するディジタルメッセージ M を暗号文 C に変換するというステップによって構成される方法で、(p-1)(q-1)と互いに素な関係にある整数 e に対して、M を C = M^e (mod n)を満たす C に変換することによって表される方法」

このクレームの文言から判断すると、クレームに特定されている素因数分解問題を用いた方式には特許の権利が及ぶが、素因数分解問題以外の数学的問題を利用する方式には及ばない。したがって、離散対数問題を利用している ElGamal 署名や Schnorr 署名には、特許の効力は及ばないことになる。

Lucifer 暗号特許のクレーム 13 に記載されている暗号化プロセスの保護の範囲について、

FEAL 暗号との関連で検討する。

Lucifer **暗号特許（アメリカ）のクレーム 13：**

「バイナリーデータとなっているメッセージブロックを暗号化するプロセスであり、
（a）メッセージブロックを第一登録手段に入力するステップと、
（b）バイナリーデータの鍵ブロックを第二登録手段に入力するステップと、
（c）バイナリーデータのメッセージブロックを n ビット毎のセットに分割し、上記第二登録手段の鍵ブロックの値によって決定される $2^n!$通り存在する変換方法のうちの 1 つによって、上記の n ビットのメッセージブロックを別の n ビットのブロックデータに変換するステップと、
（d）変換されたバイナリーデータを、そのデータの各ビットを並び替えることによって線形に変換するステップと、
（e）上記（c）と（d）の変換を予め決められた回数だけ繰り替し、その回数の変換が終了するとメッセージブロックの暗号化が完了するステップ、によって構成されるプロセス」

Lucifer 暗号は Shannon の合成関数の考え方に基づいて初めて開発された暗号方式であり、Lucifer 暗号特許は、Shannon の合成関数の理論に基づいた暗号方式の最初の実現方法を示したものである。この Lucifer 暗号特許の中で、記載内容の抽象度が高く保護の範囲が最も広いと考えられるのがクレーム 13 であり、本クレームで特定されている変換の方法はステップ(c)と(d)の 2 種類である。ステップ(c)の変換方法は、「上記第二登録手段の鍵ブロックの値によって決定される $2^n!$通り存在する変換方法のうちの 1 つ」と記載されていることから、鍵ブロックを変数とする換字変換であると推測できる。また、ステップ(d)の変換は、「変換されたバイナリーデータを、そのデータの各ビットを並び替えることによって線形に変換する」というものであり、転置変換であることが明らかである。もっとも、いずれも具体的な数値変換の手順が特定されておらず、本クレームで特定されている暗号化プロセスは、これらの抽象度の高い 2 種類の変換方法を利用した合成関数に基づく暗号化の手順ということができる。

このクレーム 13 の特許の権利の範囲に FEAL 暗号が含まれるか否かは、FEAL 暗号で用いられる算術演算が転置変換と換字変換のいずれかに含まれると解釈されるか否かによって決定されることになる。FEAL 暗号で利用される算術演算のうち、左 2 bit シフトに 1 を加える変換はステップ(d)の転置変換とは異なる種類の変換と考えることも可能であり、その場合には、FEAL 暗号はクレーム 13 の保護の範囲の外と解釈できる[223]。

（a）means plus functional claim

アメリカの means plus functional claim に相当するクレームの保護の範囲について、ア

[223] 仮に、FEAL 暗号が Shannon によって提案された理論の 1 つの実現方法であるとしても、FEAL 暗号特許で特定されている暗号方式が Lucifer 暗号特許とは異なる方法によって実現されたものか否かが問題となるため、上記と同様に具体的なクレームの内容を比較・検討する必要がある。

メリカの FEAL 暗号特許のクレーム 1 を例に検討する。

FEAL **暗号特許（アメリカ）のクレーム 1：**

「データランダム化装置であって、
①入力データを同じ長さのブロックに分割する手段（means）と、
②上記データ分割手段から送信されるデータブロックを次の処理手段に伝達する複数のチャネルと、
③上記チャネルから送信されるデータを変換する関数処理手段と、
④チャネルから送信されるデータを関数処理手段に伝達するデータの分岐手段と、
⑤ランダム化されたデータを生成するために、すべてのチャネルから送信されるデータを結合する手段とによって構成される装置」

このクレーム 1 には、抽象的な表現で記載されている 5 つの手段（means）によって構成される装置（equipment）が記載されている。アメリカ特許法第 112 条（6）によれば、手段（means）の内容は明細書中に記載されている技術とその均等物に限定されるので、「データを変換する関数処理手段」は明細書中に開示されている関数と、その関数と均等とされる関数に限定されることになる[224]。もっとも、最近の米国における判例では、① mean plus functional claim 形式の記載であっても、構造上の限定が記載されており、その手段の構造が明確になっている場合には、第 112 条（6）の適用を受けない場合がある、②明細書にて実施例との対応が明白に説明されていなければ、クレームの範囲が全ての実施例とその均等物に及ばない可能性がある、ことが指摘されている。

こうした観点から、実際に第 112 条（6）が適用されるかどうかを考慮する必要がある。まず、①については、本クレームに記載されている装置と手段の構造が必ずしも明確になっていないことから、構造上の限定が明確になっているとは言い難い。また、②については、データランダム化装置の構造および機能が明細書中に詳細に記載されており、本クレームの各手段との対応関係が明確となっていると考えられる。以上から、本クレームは第 112 条（6）の適用を受けると考えて問題はなさそうである。実際に FEAL 暗号特許の明細書をみると、法 256 の加算、左 2 bit の循環シフトと排他的論理和が変換方法として記載されており、これらが関数処理手段として解釈されると考えられる。このように、クレーム 1 の範囲は明細書によって相当限定されることとなる。

（b）均等論

特許の保護の範囲はクレームの文言の範囲を越えないことが原則である。しかし、日本では、置換可能性と容易想到性等を満足すると判断された場合には均等論が適用され、クレームの文言には含まれていなくても実質的に同一の作用効果を生じる技術が特許の保護の範囲に含まれると言われている。また、アメリカでも、機能（function）、方法（way）、結果（result）

[224] なお、均等論の適用の可能性は否定されない。

において特許の対象と実質的に同一であると認められた物や方法は、均等物としてその特許の保護の範囲に含まれると言われている。このように、一定の要件の下で均等論を適用することが可能な場合には、クレームの記載内容とは異なった手段を用いた物や方法が特許の保護の範囲に含まれる可能性がある。

アメリカにおける均等論の適用が問題となる可能性のある例として、アメリカの Schnorr 署名特許[225]を取り上げる。Schnorr 署名特許のすべてのクレームが IC カードを利用する方法を特定しており、Schnorr 署名特許によって特定されている方法を IC カードを利用しないで実現した場合、これらが Schnorr 署名特許の発明の均等物であると判断されるか否かが問題となる可能性がある。そこで、Schnorr 署名特許の署名圧縮技術を利用している DSA 署名特許を IC カードを利用しないで実現する場合、その実現方法が Schnorr 署名特許の均等物として解釈されるか否かを検討しよう。

米国で均等論が適用されるためには、①機能、②方法、③結果の 3 点が実質的に同一であることが必要である。まず、①機能と③結果については、いずれも「従来方式よりも署名長の短いディジタル署名の生成・検証方法を提供し、ディジタルデータの守秘・認証を実現するとともに、従来方式よりも高速処理を可能にしている」という意味で実質的に同一であると考えることも可能である。一方、②方法については、両者が同一の署名圧縮技術を採用しているものの、それ以外のディジタル署名生成・検証方法が異なる数式によって特定されており[226]、この数値変換の方法が実質的に同一と解釈し得るか否かが問題となる。仮に、両者の数値変換の方法が異なると解釈される場合には、均等論は適用されず、IC カードを利用しない DSA 署名特許は Schnorr 署名特許の均等物ではないと判断されることになる[227]。

---

[225] U. S. Patent Number は 4,995,082。

[226] 例えば、Schnorr 署名特許における署名検証式 $x = (2^{y})(\Sigma\ {}^{k}_{j=1} v_j)(\Sigma\ {}^{t}_{i=1} e_{i,j} 2^{i-1})(\mathrm{mod}\ p)$ （ただし、e=h(x,m))と、DSA 署名特許における署名検証式 x = ((g^(H(m)/y))*(y^(x/y)) mod p) mod q が実質的に同一の方法と解釈されるか否かが問題となる（具体的な検証式は、Schnorr 署名特許については 56～57 ページを、DSA 署名特許については 59～61 ページを参照）。

[227] NIST は、DSA 署名方式の米国政府標準規格（FIPS PUB 186）に、「商務省はこの標準を抵触するであろう特許には一切関知しない」と記載している。

VI おわりに

本稿では、主要な暗号の特許を検討し、実際にどのようなクレームが認められているのか、また、それらのクレームに関して特許の保護の要件、効力の範囲をどのように解釈すればよいのかについて検討を試みた。その結果、まず暗号に関するクレームとして、装置構成のクレームだけではなく、数値変換の方法に近いと考えられるクレームも認められていることがわかった。また、特許の保護の要件については、例えば Merkle-Hellman 特許において、公開鍵暗号を利用した暗号通信やディジタル署名の基本的な方法のクレームが「抽象的な概念」に当たらないか否かとか、Lucifer 暗号特許において、技術内容の開示が十分であると言えるかどうか等、解釈が非常に難しい問題が存在することが明らかとなったほか、特許の効力と保護の範囲に関しても、複数の当事者の関与が必要とされる特許の効力をどのように考えるか、均等論の考え方をどの程度考慮するかといった問題も残されていることがわかった。

こうした暗号の特許を巡る問題について明確な判断を見出すためには、裁判による判決を待つ以外に手段はない。しかし、コンピュータ・ソフトウェアに関する判例は数少なく、暗号に関する判例は見られない。したがって、ある暗号を利用する際には、関連する暗号の特許が存在するのかどうか、存在するとすればその特許の効力はどこまで及ぶのか、そして利用したい暗号がその特許の効力の範囲に含まれていないかどうかについて、事前に調査・分析を行う必要がある。そうした場合に、本稿で検討したような法律的・技術的な観点からの特許の分析が有用であると考えられる。

今後、オープンなネットワーク上における情報通信のセキュリティを確保する技術として、暗号への要請が一層高まることが予想される中、特許が成立している暗号の使用に関して法律的な問題が生じる可能性も高まると思われる。こうした観点から、今後も暗号の研究開発に関する動向に加えて、暗号の特許や特許制度等の動向にも注視していく必要があると言えよう。

以　上

参考文献

（特許関連）

相澤英孝、「特許制度の在り方 －バイオテクノロジーに関する発明への均等論の適用－」、『星野英一先生古稀祝賀論　日本民法学の形成と課題（下）』、1996 年.

青山紘一、『ソフトウェアの特許』、工業調査会、1993 年.

斎藤治・森田泰子・加藤壮太郎、「金融業務における特許権の成否 －特許法の保護対象について」、『金融研究』第 14 巻第 2 号、日本銀行金融研究所、1995 年 7 月.

ソフトウェア委員会、「米国での機能実現クレームの最近の動向」、『パテント』Vol. 51、No. 2、pp. 115-130、1998 年 2 月.

豊崎光衛、『工業所有権法[新版・増補]』、有斐閣、1980 年.

中山信弘、『工業所有権法（上）』、弘文堂、1993 年.

Chisum, D. S., “The Patentability of Algorithms,” Vol. 47, Pittsburgh Law Review, pp. 959, 1986.

Hanneman, H. W. A. M., *The Patentability of Computer Software*, Kluwer Law and Taxation Publishers, 1985.（H. W. A. M. Hanneman 著、佐野稔監修、ソフトウェア技術者協会ソフトウェア法的保護分科会訳、『コンピュータソフトウェアの特許適格性』、日刊工業新聞社、1993 年.）

（暗号関連）

岡本龍明・山本博資、『現代暗号』、産業図書、1997 年.

楠田浩二・櫻井幸一、「公開鍵暗号方式に関する現状と課題」、IMES Discussion Paper Series 97-J-11、日本銀行金融研究所、1997 年.

松井　充、「ブロック暗号アルゴリズム MISTY」、電子情報通信学会技術研究報告、ISEC96-11、1996 年 a.

————、「ハッシュ関数：算術演算型とテーブル参照型の比較検討」、『暗号アルゴリズムの設計と評価』ワークショップ講演論文集、pp. 63-70、1996 年 b.

盛合志帆・青木和麻呂・太田和夫、「RC5 の線形確率について」、1996 年暗号と情報セキュリティシンポジウム講演論文集、SCIS96-12C、1996 年.

Abe, M. and J. Camenisch, “Partially blind signature schemes,” The 1997 Symposium on Cryptography and Information Security, SCIS97-33D, 1997.

———— and E. Fujisaki, “How to date blind signatures,” Advances in Cryptology — Proceedings of ASIACRYPT ’96, Lecture Notes in Computer Science, Vol. 1163, pp. 244-251, Springer-Verlag, 1996.

Bellare, M. 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.

——— and S. Micali, “How to sign given any trapdoor function,” Proceedings of STOC, pp. 32-42, 1989.

Biham, E., “New types of cryptanalytic attacks using related keys,” Advances in Cryptology — Proceedings of EUROCRYPT’93, Lecture Notes in Computer Science, Vol. 765, pp. 398-409, Springer-Verlag, 1994.

——— and A. Shamir, “Differential cryptanalysis of Snefru, Khafre, REDOC-II, LOKI, and Lucifer,” Advances in Cryptology — Proceedings of CRYPTO’91, Lecture Notes in Computer Science, Vol. 576, pp. 156-171, Springer-Verlag, 1992.

Blom, R., “Non-public key distribution,” Proceedings of CRYPTO ’82, pp. 231-236, Plenum Press, 1983.

Boer, B. D. and A. Bosselaers, "Collisions for the compression function of MD5," Advances in Cryptology, EUROCRYPT '93 (LNCS 765), pp. 293-304, 1994.

Bosselaers, A. and B. Preneel, editors, *Integrity Primitives for Secure Information Systems: Final Report of RACE Integrity Primitives Evaluation RIPE-RACE 1040*, Lecture Notes in Computer Science Vol. 1007, Springer-Verlag, 1995.

Brands, S., “An efficient off-line electronic cash system based on the representation problem,” Technical Report CS-R9323, CWI, Apr. 1993.

Brown, L., M. Kwan, and J. Pieprzyk, “Improving resistance to differential cryptanalysis and the redesign of LOKI,” Advances in Cryptology — Proceedings of ASIACRYPT ’91, Lecture Notes in Computer Science, Vol. 739, pp. 36-50, Springer-Verlag, 1993.

———, J. Pieprzyk, and J. Seberry, “LOKI — A cryptographic primitive for authentication and secrecy applications,” Advances in Cryptology — Proceedings of AUSCRYPT ’90, Lecture Notes in Computer Science, Vol. 453, pp. 229-236, Springer-Verlag, 1990.

Camenisch, J., U. Maurer, and M. Stadler, “Digital payment systems with passive anonymity-revoking trustees,” Computer Security – Proceedings of ESORICS ’96, Lecture Notes in Computer Science, Vol. 1146, pp. 33-43, Springer-Verlag, 1996.

Chaum, D., “Blind signatures for untraceable payments,” Advances in Cryptology — Proceedings of CRYPTO ’82, pp. 192-203, Prenum Press, 1983.

Daemen, J., “Weak keys for IDEA,” Advances in Cryptology — Proceedings of CRYPTO ’93, Lecture Notes in Computer Science, Vol. 733, pp. 224-231, Springer-Verlag, 1994.

Diffie, W. and M. E. Hellman, “New directions in cryptography,” *IEEE Transactions on Information Theory*, Vol.IT-22, pp.644-654, November 1976.

———, P. C. van Oorschot, and M. J. Wiener, “Authentication and authenticated key exchanges,” *Designes, Codes and Cryptography*, Vol.2, pp.107-125, 1992.

Dobbertin, H., “RIPEMD with two-round compress function is not collision-free,” *Journal of Cryptology*, Vol. 10 No. 1, pp. 51-70, 1997.

———, “A cryptanalysis of MD4,” The Third Workshop of Fast Software Encryption, Lecture Notes in Computer Science, Vol. 1039, pp. 53-69, Springer-Verlag, 1996.

———, A. Bosselaers, and B. Preneel, “RIPEMD-160: A strengthened version of RIPEMD,” The Third Workshop of Fast Software Encryption, Lecture Notes in Computer Science, Vol. 1039, pp. 71-82, Springer-Verlag, 1996.

Dwork, C. 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.

ElGamal, T. E., “A public key cryptosystem 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, 1985.

Feige, U. and A. Shamir, “Witness indistinguishable and witness hiding protocols, ” Proceedings of STOC, pp. 416-426, 1990.

Feistel, H., “Cryptography and computer privacy,” Scientific American, Vol. 228, No. 5, pp. 15-23, 1973.

Fiat, A. and A. Shamir, “How to prove yourself: practical solutions to identification and signature problems,” Advances in Cryptology — Proceedings of CRYPTO ’86, Lecture Notes in Computer Science, Vol. 263, pp. 186-194, Springer-Verlag, 1986.

Goldwasser, S., S. Micali, and R. Rivest, “A digital signature scheme against adaptive chosen message attack,” *SIAM J. Comput.*, Vol. 17, No. 2, pp. 281-308, 1988.

Harpes, C., G. G. Cramer, and J. L. Massey, “A generalization of linear cryptanalysis and the applicability of Matsui’s piling-up lemma,” Advances in Cryptology — Proceedings of EUROCRYPT ’95, Lecture Notes in Computer Science, Vol. 921, pp. 24-38, Springer-Verlag, 1995.

Kaliski, B. S. Jr. and Y. L. Yin, “On differential and linear cryptanalysis of the RC5 encryption algorithm,” Advances in Cryptology — Proceedings of

CRYPTO '95, Lecture Notes in Computer Science, Vol. 963, pp. 171-184, Springer-Verlag, 1995.

Knudsen, L. R., "Cryptanalysis of LOKI," Advances in Cryptology — Proceedings of AUSCRYPTO '92, Lecture Notes in Computer Science, Vol. 718, pp. 196-208, Springer-Verlag, 1993.

——— and T. Berson, "Truncated differentials of SAFER," Proceedings of Fast Software Encryption, Third International Workshop, Lecture Notes in Computer Science, Vol. 1039, pp. 15-26, Springer-Verlag, 1996.

Koblitz, N., "Elliptic curve cryptosystems," *Mathematics of Computation*, Vol.48, pp.203-209, 1987.

———, "Cm-curves with good cryptographic properties," Advances in Cryptology — Proceedings of CRYPTO '91, Lecture Notes in Computer Science, Vol. 576, pp. 279-287, Springer-Verlag, 1992.

Lai, X., J. L. Massey, and S. Murphy, "Markov ciphers and differential cryptanalysis," Advances in Cryptology — Proceedings of EUROCRYPT '91, Lecture Notes in Computer Science, Vol. 547, pp. 17-38, Springer-Verlag, 1991.

Lenstra, H. W. Jr., "Integer programming with a fixed number of variables," Univ. of Amsterdam Tech. Report, Vol. 81-03, April 1981.

Massey, J. L., "SAFER K-64: A byte-oriented block-ciphering algorithm," Proceedings of Fast Software Encryption, Cambridge Security Workshop, Lecture Notes in Computer Science, pp. 1-17, Springer-Verlag, 1994.

———, "SAFER K-64: One year later," Proceedings of Fast Software Encryption, Second International Workshop, Lecture Notes in Computer Science, Vol. 1008, pp. 212-241, Springer-Verlag, 1995.

Matsumoto, T. and H. Imai, "On the key distribution system: A practical solution to the key distribution problem," Advances in cryptology — Proceedings of CRYPTO '87, Lecture Notes in Computer Science, Vol. 293, pp. 185-193, Springer-Verlag, 1988.

Matyas, S. M., C. H. W. Meyer, and J. Oseas, "Generating strong one-way functions with cryptographic algorithm," *IBM Technical Disclosure Bulletin*, Vol. 27, pp. 5658-5659, 1985.

Meier, W. and O. Staffelbach, "Efficient multiplication on certain nonsupersingular elliptic curves," Advances in Cryptology — Proceedings of CRYPTO '92, Lecture Notes in Computer Science, Vol. 740, pp. 333-344, Springer-Verlag, 1993.

Menezes, A. J., T. Okamoto, and A. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field," Proceedings of STOC, pp. 80-89, 1991.

Merkle, R. C. and M. E. Hellman, "Hiding information and signatures in trapdoor

knapsacks," *IEEE Transactions on Information Theory*, Vol.IT-24, No.5, pp.525-530, September 1978.

Miller, V. S., "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.

Miyaguchi, S., A. Shiraishi, and A. Shimizu, "Fast data encipherment algorithm FEAL-8," Review of Electrical Communication Laboratories, Vol. 36, No. 4, pp. 321-327, NTT, 1988.

Miyaji, A., "Elliptic curves suitable for cryptosystems," *IEICE Transactions on Fundamentals*, Vol.E77-A, No. 1, pp. 98-105, 1994.

Naor, M. and M. Yung, "Universal one-way hash functions and their chosen ciphertext attacks," Proceedings of STOC, pp. 33-43, 1989.

National Institute of Standards and Technology, "Data Encryption Standard（DES）," Federal Information Processing Standards Publication（FIPS PUB）46-2, December 13, 1993.

————, "Digital Signature Standard（DSS）," Federal Information Processing Standards Publication(FIPS PUB) 186, May 19, 1994.

————, "Secure Hash Standard," Federal Information Processing Standards Publication（FIPS PUB）180-1, April 17, 1995.

Nyberg, K., "Linear approximation of block ciphers," Advances in Cryptology — Proceedings of EUROCRYPT '94, Lecture Notes in Computer Science, Vol. 950, pp. 439-444, Springer-Verlag, 1995a.

———— and L. R. Knudsen "Provable security against a differential attack," *Journal of Cryptology*, Vol. 8, pp. 27-37, Springer International, 1995b.

Ohta, K. and K. Aoki, "Linear cryptanalysis of the Fast Data Encipherment Algorithm," Advances in Cryptology — Proceedings of CRYPTO '94, Lecture Notes in Computer Science, Vol. 839, pp. 12-16, Springer-Verlag, 1994.

Okamoto, T., "A fast signature scheme based on congruential polynomial operations," *IEEE Transactions on Information Theory*, Vol. 36, No. 1, pp. 47-53, 1990.

————, "Provably secure and practical identification schemes and corresponding signature schemes," Advances in Cryptology — Proceedings of CRYPTO '92, Lecture Notes in Computer Science, Vol. 547, pp. 31-53, Springer-Verlag, 1993.

———— and K. Ohta, "Divertible zero-knowledge interactive proofs and commutative random self-reducibility," Advances in Cryptology — Proceedings of EUROCRYPT '89, Lecture Notes in Computer Science, Vol. 434, pp. 134-149, Springer-Verlag, 1990.

Pointcheval, D. 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.

Rabin, M. O., “Digitalized signatures and public-key functions as intractable as factorization,” M.I.T. Laboratory for Computer Science, Technical Report MIT/LCS/TR-212, 1979.

Rivest, R. L., “The MD4 message digest algorithm” Advances in Cryptology — Proceedings of CRYPTO ’90, Lecture Notes in Computer Science, Vol. 537, pp. 303-311, Springer-Verlag, 1991.

————, “The MD5 message digest algorithm” Request for comments （RFC）, 1321, Internet Activities Board, Internet Privacy Task Force, 1992.

————, “The RC5 encryption algorithm” Proceedings of Fast Software Encryption, 2nd International Workshop, Springer-Verlag, 1995.

————, 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.

Rompel, J., “One-way functions are sufficient for secure signatures,” Proceedings of STOC, pp.387-394, 1990.

Satoh, T. and K. Araki, “Fermat Quotients and the Polynominal Time Discrete Log Algorithm for Anomalous Elliptic Curves,” preprint, 1997.

Schnorr, C. P., “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.

Shamir, A. and R. E. Zippel, “On the security of the Merkle-Hellman cryptographic scheme,” *IEEE Transactions on Information Theory*, Vol. IT-26, No.3, pp.339-340, May 1980.

Shannon, C. E., “Communication theory of secrecy systems,” *Bell System Technical Journal*, Vol. 28, pp. 656-715, 1949.

Smart, N., "Announcement of an attack on the ECDLP for anomalous elliptic curves," 1997.

Tokita, T., T. Sorimachi, and M. Matsui, “Linear cryptanalysis of LOKI and s^2DES,” Advances in Cryptology — Proceedings of ASIACRYPT ’94, Lecture Notes in Computer Science, Vol. 917, pp. 293-303, Springer-Verlag, 1995.