次世代認証方式に関する
研究・開発動向の調査報告書

平成13年3月16日

目 次

はじめに

インターネットは、世界中の情報資源の共有に大きく貢献している。一方、オープンなネットワークであるインターネットは、それ自身では、情報へのアクセス制御や情報の改ざん防止機能に乏しく、そこを流れるデータは潜在的に盗聴・改ざんの危険にさらされている。また、不特定多数の人が同時にネットワークを利用するため、通信相手を厳密に特定することが難しく、成り済ましの被害を受ける危険がある。情報の秘匿性、一貫性ならびに通信相手やデータ生成者の認証を行うための技術としては暗号技術があり、それらを誰もが効率よく利用するためには暗号・認証基盤技術の整備が必要不可欠である。

現在最も普及している暗号・認証基盤は、PKI (Public-Key Infrastructure) と呼ばれるもので、公開鍵暗号技術を基に構成されている。とりわけ、素因数分解あるいは離散対数問題の難しさに基づいた公開鍵暗号技術が広く用いられており、現在の暗号・認証基盤の安全性はそのほぼ全てが素因数分解あるいは離散対数問題に頼っていると言っても過言ではない。現在、素因数分解および離散対数問題を解くには、問題のサイズに対して準指数オーダーの計算量が掛かることが知られており、これらの問題が数年中に破られる可能性は小さいと考えられる。しかしながら、数十年先のことを考えた場合、これらの問題が難しい問題であり続けている保証は必ずしも無い。例えば、新たな計算機アーキテクチャが発見されたり、新たな解読アルゴリズムが発見されたりする可能性は無いとは言えない。少なくとも、量子計算機が将来実現された場合には、Shor により発見されたアルゴリズムにより、素因数分解および離散対数問題は（確率的）多項式時間で解かれることが知られている。

一旦、素因数分解あるいは離散対数問題が実時間で解けるようになったとすると、現在利用されている暗号・認証基盤はそのほとんどが機能しなくなり、その基盤上で動作している電子決済などの各種サービスも利用できなくなってしまう。広く普及した暗号・認証基盤が機能しなくなることによる社会的な影響は非常に大きく、場合によっては大きな混乱に陥る危険性もある。このような状況を避けるためには、素因数分解あるいは離散対数問題が仮に破られたとしても利用し続けることができる暗号・認証基盤の整備が必要である。

本調査の目的は、素因数分解問題あるいは離散対数問題と独立な問題を探し、それらに関する研究動向を調査することにある。これにより、仮に素因数分解あるいは離散対数問題が破られたとしても利用し続けることができる暗号・認証基盤の実現可能性を探る。第 1 章では、まず、現行の暗号・認証基盤について調査を行う。具体的には現行の暗号・認証基盤で採用されているセキュリティ技術およびそれらの安全性の根拠についてまとめる。第 2 章では情報量的安全性に基づく方式の研究動向を調査し、第 3 章では、計算量的安全

性に基づく方式の中で素因数分解あるいは離散対数問題の難しさと独立な問題に基づく方式の研究動向を調査する。具体的には、それぞれラティス上の問題、ナップサック問題、線形符号の復号問題に基づく方式について調査する。第 4 章では時間的安全性に基づく方式の研究動向を調査し、第 5 章では耐タンパー性に基づく方式の研究動向を調査する。

第1章　暗号・認証基盤における動向調査

現在、インターネットの普及に伴い、ネットワークを利用した電子商取引が活発化してきている。インターネットのようなオープンネットワークで電子商取引を行う場合、情報を「秘匿」するための暗号化技術が重要であることは既に広く知られているが、それと同様、あるいはそれ以上に重要なのが「認証」のための技術である。例えば、取引先・支払い先の確認や契約内容の改竄を防止することなど、ネットワークを介した電子商取引では、通信相手の確認や通信内容が改竄されていないかどうかの確認が重要となってくる。

「認証」のための技術として公開鍵暗号方式に基づいた「ディジタル署名」技術が知られている。しかし、ディジタル署名だけでは、本人でない別の者が本人に成りすまして鍵を生成し、勝手に検証鍵を公開することで、本人名義で通信や取引を行ってしまうなどの不正は排除することが出来ない。よって、生成されたディジタル署名が本人のものであることを確認する仕組みが必要となる。そこで、当事者とは別の第三者が、公開鍵が本人のものであることを証明する仕組みが考えられる。この第三者を通常「認証機関」と呼んでいる。認証機関が電子証明書を発行するなどして、検証 (公開) 鍵に対応する署名 (秘密) 鍵が本人のものであることを証明することにより、通信相手の本人性を確認することが出来る。

近年、この認証機関としてのサービスを行う会社や、認証機関と同様な機能をもつサーバアプリケーションが相次いで商品化されるなど、いわゆる「電子認証サービス」が注目されており、電子商取引等の普及に従いその市場規模は将来的に大きな成長が見込まれている。また、これら認証機関の信頼性、および認証自体の信頼性を確保するため、認証技術の標準化や認証サービスの法整備が急ピッチで進んでいる。

本章では、これら暗号化・認証基盤の現状を述べると共に、そこでどのような暗号・認証方式が用いられているかについての調査結果を報告する。また、またそれら暗号・認証方式の安全性と将来の展望について考察を行う。

1.1　暗号・認証基盤の現状と利用可能な方式

近年の電子商取引や EC の活発化は、インターネットの普及が背景となっている。これらオープンネットワーク上で安全性を確保するため、通信者同士が認証を行った上で秘匿通信を行うプロトコルが開発され広く用いられている。

まず、インターネットプロトコルレベルにおいて安全性を確保するのが IETF 標準の **IPSec(Internet Protocol Security)** である。IPSec では、インターネットのパケットに認証を行うヘッダをつけたりパケット自体を暗号化することで「秘匿」と「認証」を実現

している。最近では、IPSec の機能を組み込んだルータやゲートウェイを用いて、インターネットを介した 2 点間があたかも安全な専用線で結んだかのようにつなぐ VPN(Virtual Private Network) が構築され始めている。

また TCP/IP の上位で動作し、http などのネットワークアプリケーションを保護するプロトコルとして **SSL(Secure Sockets Layer)**[1-15] がある。SSL は Netscape 社により提唱されたプロトコルであり、NetscapeNavigator、Internet Explorer 等の主要な WWW ブラウザに組み込まれるなど、インターネット上のデファクトスタンダードとなっている。さらに標準化の動きもあり、1999 年 IETF により、SSL をベースとした **TLS(Transport Layer Security)** Ver.1.0 が RFC として公開されている [1-14]。

これらのプロトコルでは、使用する暗号方式が規定または推奨されている (表 1.1)。これらのプロトコルで新たな方式を使用することは可能であるが、サーバ・クライアントともにライブラリを実装している必要があり、ローカルドメインでの使用以外は規定された暗号方式を使用するのが一般的である。また暗号方式の鍵長は、共通鍵暗号方式の場合 64, 128 ビット、公開鍵暗号方式の場合 512, 1024, 2048 ビットが標準的に用いられている。

表 1.1: 各プロトコルで規定されている暗号方式

プロトコル	鍵共有	署名	暗号化	メッセージ認証
IPSec	–	–	DES	SHA1 MD5
SSL v3.0	RSA DH Fortezza	RSA DSS	RC4 RC2 DES/3DES	SHA1 MD5
TLS v1.0	RSA DH	RSA DSS	RC4 RC2 IDEA DES/3DES	SHA1 MD5 SHA1 MD5
Kerberos v5	–	–	DES	MD4 MD5

一方、これらプロトコルやアプリケーションを用いる際、本人でない別の者が本人に成りすまして鍵を生成し、勝手に公開鍵を公開することで、本人に成りすますことが出来てしまう。この回避策として、公開鍵が本人のものであることを証明する電子証明書を第三者が発行し、通信時にこの証明書を通信相手に提示することにより、本人性を証明する仕組みが考えられている。この第三者を「認証機関 (CA: Certification Authority)」と呼ぶ。現在、申請者の公開鍵を含むデータに対して、認証機関の秘密鍵で生成したディジタル署名を付加したものを電子証明書とするものが一般的である。このように、公開鍵暗号方式をベースとし、電子証明書の発行や公開鍵の管理するインフラを PKI(Public Key Infrastructure) と呼び、現在整備が進められている。

これを受けて、近年、発行申請者の本人確認や電子証明書の発行などの電子認証サービスを施行する企業が増え、また電子認証サービスを実現するサーバアプリケーションなども数多く商品化されている。前者としては VeriSign、Entrust、日本では富士通、日立、NEC が共同で設立した日本認証サービス (JCSI)[1-18] が有名である。また、後者としては BALTIMORE、Netscape、Microsoft、日本では三菱、富士通が商品を提供している。

これらの認証局が発行する電子証明書は、基本的に前述のプロトコルで用いられるため、各アプリケーションで標準的に実装されている公開鍵暗号を用いるのが一般的である。また、鍵長についても、クライアント認証のための電子証明書としては 1024 ビット、サーバ認証には 2048 ビット長の署名 (秘密) 鍵が広く用いられている。認証機関によっては、より鍵長の長い電子証明書の発行が可能だが、各アプリケーションでの実装が対応していない、また鍵長が増すにつれて処理速度が遅くなるため、現在はほとんど用いられていない。

また、電子証明書の代わりに秘匿通信用の鍵をセンタ (KDS:Key Distribution Center) から配布することにより認証・通信を行う **Kerberos**[1-16] プロトコルが知られている。Kerberos は MIT で開発された共通鍵暗号方式ベースの認証システムである。しかし、共通鍵暗号方式を用いるため、KDS の鍵管理と相互運用の負担が大きい。最近では、エンティティの初期認証に公開鍵暗号方式を用いることにより鍵管理の負担を軽くする方法も提案され、Windows2000 にも組み込まれている。Kerberos で規定されている暗号方式についても表 1.1 に示す。

1.2 暗号・認証方式の現状

前節で述べたとおり、「秘匿」のための暗号化技術として共通鍵暗号方式を用いることが一般的である。共通鍵暗号方式としては、米国標準暗号方式 DES が 1977 年の制定以来広く世界で用いられてきた。しかし、1999 年米国 RSA Data Security 社により web 上で開催された解読コンテスト (DES Challenge III[1-8]) で、DES がわずか 22 時間で解読されたという事例が示すように、コンピュータの計算能力の向上により DES、および DES に代表される 64 ビットブロック/64 ビット鍵長 (DES の鍵長は 56 ビット) の共通鍵暗号方式の解読の危険性がかなり増大してきている。それゆえ、1997 年 1 月米国連邦標準技術局 (NIST) が DES の後継となる共通鍵暗号方式 AES(Advanced Encryption Standard)[1-19] の公募を開始ししたことにより、次世代でも通用するより強い共通鍵暗号方式の開発が一気に加速した。その結果、2000 年 10 月 Rijndael が選定されたことは記憶に新しい。また米国だけでなく、ISO/IEC が国際標準共通鍵暗号方式の選定をはじめると共に、欧州域内での標準方式を定めるプロジェクト NESSIE[1-20] や、日本国内でも情報処理事業振興協会 (IPA) が事務局を務め日本電子政府での利用を目的とした暗号方式を選定する CRYPTREC[1-21] が進行している。このように、次世代共通鍵暗号方式の開発と標準化の動きが活発化しており、それに合わせてこれら次世代共通鍵暗号方式の使用可能な基盤の構築が進んでいる。

一方「認証」に用いるディジタル署名などの認証方式、鍵共有を実現するため公開鍵暗号方式など、公開鍵暗号ベースの方式の現状はどうであろうか。公開鍵暗号方式としては、1978 年に発明された RSA 暗号、1982 年に発明された ElGamel 暗号、また 1985 年に提案された楕円曲線上の ElGamal 暗号などが有名であり、ディジタル署名方式としては、RSA 署名の他、ElGamel 署名、1991 年 NIST により標準米国標準方式と規定された DSA 署名 (FIPS186 では DSS とよんでいる) などが広く知られている。また、NTT が開発したディ

ジタル署名方式 ESIGN や公開鍵暗号方式 EPOC なども有名である。これら方式の安全性は、全て素因数分解問題 (またはその近似問題) か離散対数問題の困難性に基づいている。つまり、それらの問題が解けないならば安全であろうと広く認識されている。その意味で、新たな数学的な問題を安全性の根拠とし、かつ広く用いられている方式は近年開発されていない。近年の傾向としては、既存方式の安全性の証明および安全性の証明が可能な方式の研究が盛んである。安全性が証明可能とは、方式の安全性が素因数分解や離散対数問題などの困難性に帰着可能なことをいう。有名なものとして Bellare,Rogaway[1-1] により提案された OAEP が挙げられる。これは、既存の公開鍵暗号方式にある種のフォーマットを適用することにより、(用いるハッシュ関数の出力がランダムという仮定の基で) 素因数分解問題などに安全性を帰着可能とする手法である。この手法は現在用いられている公開鍵基盤をそのまま適用できるという点で優れた方式である。また Cramer,Shoup[1-2] により、安全性がある種の問題に帰着可能な離散対数問題に基づいた公開鍵暗号方式も提案されている。また、公開鍵ベースの方式も標準化が進められており、前述の ISO, NESSIE や CRYPTREC などで選定が進められている。

1.3　パラメータサイズと安全性

暗号方式の安全性は、プロトコルの設計やパスワードの選択方法などにより影響を受ける。中でも特に鍵長は安全性を決める重要なファクターといえる。本章では、暗号方式の鍵長と安全性の関係について考察する。

共通鍵暗号方式 DES は現在の暗号・認証基盤で広く用いられており、その鍵長は 56 ビットである。しかし、1999 年 RSA カンファレンスに合わせて Web 上で行われた DES の解読コンテスト (DES Challenge III[1-8]) において、わずか 22 時間 15 分で解読されている。解読手法は、インターネットに接続された 10 万台近いパソコンでの分散処理によるものであるが、普通のパソコンでも潜在的な脅威となるということが証明されたと言える。現在、鍵長 64 ビット程度の共通鍵暗号方式の時代は終焉を迎え、鍵長 128 ビット以上の共通鍵暗号方式が必要だといわれている。それに対応して AES では 128, 196, 256 ビット対応に、またその他の標準化でも 128 ビット以上の鍵長が規定されている。

一方、現在の暗号・認証基盤で広く用いられている公開鍵暗号は RSA であろう。こちらも米国 RSA Data Security 社により行われた解読コンテストにより、1999 年 2 月に 140 桁 (RSA140[1-9]) の素因数分解が、1999 年 8 月に 155 桁 (512 ビット、RSA155[1-10]) の素因数分解が成功している。こちらは 292 台のコンピュータによる分散処理で、前段階の計算に 9 週間、解読計算に 5.2ヶ月を費やしている。これを受け、RSA 社では 768 ビット長の鍵を使うよう推奨している。

このように、コンピュータの計算能力の向上により、方式の安全性を確保するためには鍵長を長くしなければならない。この傾向は、素因数分解や離散対数問題が容易に解けるアルゴリズムが発明されるなどのパラダイムシフトが起こらない限り、将来的に続くであろう。

では、現状と同じ安全性を確保するためには、将来何ビットの鍵長が必要なのであろうか。特に電子証明書など長期間に渡って有効でなくてはならないデータは、あらかじめ将来的に有効な鍵長によってディジタル署名されなくてはならないため、必要な鍵長の推定は重要な問題である。この問題に対して、共通鍵暗号方式については [1-3]、RSA については [1-4]、楕円暗号については [1-5] など将来必要とされる鍵長を導出する研究が知られている。さらに、Lenstra,Verheul[1-6] が PKC2000 において興味深い研究結果を発表している。この研究では、共通鍵暗号方式、RSA、離散対数問題に基づく公開鍵暗号方式、ハッシュ関数について、攻撃にかかる計算量という同じ尺度で、将来商用に用いるために必要と思われる鍵長の下限および計算コストを導出している (表 1.2∼1.4)。

例えば、2020 年まで安全性を保つことが出来るアプリケーションを開発する場合、表 1.2 の 2020 年の行を見ると、1982 年時点での DES の安全性と同程度の安全性を確保するためには、少なくとも $2.9 * 10^{14}$Mips Years の計算量でも攻撃が成功しないようパラメータを決めなければならないことがわかる。よって、表 1.2 より各方式の鍵長は少なくとも以下の値が必要だと言える。

- 共通鍵暗号の鍵長は少なくとも 86 ビット、ハッシュ長は少なくとも 172 ビットが必要。
- RSA の鍵長 (n のビット数) は少なくとも 1881 ビットが必要。
- サブグループにおける離散対数問題 (SDL: Subgroup Discrete Logarithm) は、少なくとも 1881 ビットの有限体で部分乗法群の位数が 151 ビットである必要がある。
- 素体上の楕円暗号 (EC) は、暗号研究による進展がない場合は少なくとも 161 ビットが、進展がある場合は少なくとも 188 ビットが必要。

また Lenstra らは、表 1.2 と現状を比較することにより、以下の点を指摘している。

1. DSS では、160 ビットの部分乗法群をもつ 512∼1024 ビットの体と 160 ビットのハッシュ関数を使用するよう規定している。しかし、この体のサイズでは 2002 年までしか安全とはいえない。また、ハッシュ関数も 2013 年まで、部分群サイズも 2026 年までである。
2. RSA を 2040 年まで安全に用いるためには、現在の 1024 ビットの 3 倍程のビット数が必要である。これを実現した場合、署名検証・暗号化で現在の 9 倍、署名生成・復号で現在の 27 倍処理速度が遅くなるであろう。ElGamel 暗号など有限体上の離散対数問題に基づいた方式で 27 倍、DSS などの部分群上の離散対数に基づいた方式でも 11 倍程度処理速度が落ちると思われる。一方、楕円暗号は高々4 倍程度しか遅くならない。
3. 512 ビットの RSA は、現在も SSL の鍵共有などで用いられている。しかし 1986 年時点で既に安全ではなくなっている。また 768 ビットの RSA も既に安全とはいえない。
4. 1024 ビットの RSA と 160 ビットの楕円暗号は同レベルの安全性と言われているが、それは正しくない。 (1375 ビットの RSA と 160 ビットの楕円暗号が同レベルだと思われる。)

この研究結果により、現状の暗号・認証基盤で行われている電子商取引などのデータの将来に渡る安全性、つまりデータのライフタイムは非常に短いと思われる。例えば、電子債権・電子手形など長期にわたって保存されるデータについては、十分注意が必要であろう。だからといって鍵長を長くすると、現在のコンピュータの処理能力では実用に耐えない。今後は、過去のデータを効率よく保存し、将来的にも有効でありつづけるための方式および基盤の整備が必要であろう。

一方、1994 年 Shor[1-7] により量子計算機によって素因数分解問題・離散対数問題を多項式時間で解くアルゴリズムが発明されている。よって、将来量子計算機が実用化された場合、Lenstra らの研究結果は意味をもたない、つまり、いくら鍵長を長くしても素因数分解問題・離散対数問題に基づいた公開鍵暗号は破られてしまうことに注意されたい。

1.4　標準化と法整備

暗号方式の標準化は、前節でも述べた通り、現在多くの機関が選定作業に取り掛かっている状態である。ISO/IEC による国際標準暗号方式、NESSIE プロジェクトによる欧州標準暗号、日本での標準暗号を選定する CRYPTREC などがそれにあたる。また、認証および PKI についての標準化も現在急ピッチで進行している。それについては [1-11] に詳しい。

一方、電子認証サービスを行う認証機関の信頼性・安全性、また利用者の安全性を保証するための法整備が必要とされており、我が国でも「電子署名及び認証業務に関する法律」(平成 12 年法律第 102 号 [1-12]) が 2000 年 5 月に交付され、2001 年 4 月 1 日から施行される。認証機関の信頼性・安全性の最大の基準は、どのようなセキュリティシステムを用いているかという点であり、相当程度のセキュリティ技術に基づいていなければならない。特に認証業務に用いるディジタル署名方式の安全性のレベルについては、本法律に基づく関係政省令において「1,024 ビットの素因数分解等の困難性相当以上の安全性を有する電子署名」という基準を示そうとしている (表 1.5[1-13])。

しかし、前節の Lenstra らの予想によれば、1024 ビットの素因数分解、1024 ビットの有限体上での離散対数問題が十分に安全であるのは数年程度とされている。もし十分に安全とは言えない状態が起こった場合どのように対処すべきなのか、といった観点から認証サービスの検討と法整備も必要であろう。また、これらは全て素因数分解と離散対数のみに基づいた方式であり、量子計算機の出現によるパラダイムシフトが起こった場合の対応についても検討が望まれる。

1.5　まとめと将来展望

以上述べたように、現在の暗号・認証基盤として、公開鍵暗号技術をベースとしたいわゆる PKI とよばれるインフラが構築されつつある。また、それら基盤で用いられている公開鍵暗号方式の安全性は、ほぼ全て素因数分解か離散対数問題の困難性に頼っていると

言っても過言ではない。確かにこれらの問題が数年内に破られる可能性は小さいと考えられる。しかし中長期的には、必ずしもそれらの問題が破られないとは言い切れない。例えば、前述のようにコンピュータの計算能力が大幅に向上し、実用的なパラメータサイズでは安全な通信を行うことが出来なくなるかもしれない。また、将来量子計算機が実現すると素因数分解や離散対数問題が多項式時間で解けてしまう。ただし、それまでに素因数分解や離散対数問題に代わる、量子計算機でも解くことが出来ない新たな問題が発見され、それに基づいた公開鍵暗号系が発明されている可能性もないわけではない。現在の暗号・認証基盤は新しい公開鍵暗号系への変更が比較的容易に設計されており、プロトコルやデータフォーマットなどは現行のまま使用することが出来るであろう。しかし、将来全ての公開鍵暗号系が危険な状態に陥るというパラダイムシフトが起こったとき、はたして現在の暗号・認証基盤は生き残れるだろうか。

以上を考慮すると、第一に素因数分解・離散対数問題以外の問題に基づく公開鍵暗号系の研究・開発が必要だと考えられる。さらに、長期的な研究として、公開鍵暗号系に依存しない暗号・認証基盤を模索することが必要であろう。また、忘れてならないのは、パラダイムシフト起こったときに、それまでの基盤および暗号技術のもとでやりとりされたデータの安全性を保護する必要があるという点である。たとえば、電子現金や電子債権・手形などが無に帰してしまわぬよう保護される基盤を整備していかなければならない。

表 1.2: 計算量的に安全な鍵長の下限とその予想 ([1-6],Table1)

Year	Symmetric Key Size	Classical Asymmetric Key Size and SDL Field Size	SDL Key Size	Elliptic Curve Key Size $c = 0$	Elliptic Curve Key Size $c = 18$	Infeasible number of Mips-Years	Lower bound for hardware cost in US$ for a 1 day attack	Corresponding number of years on a 450MHz Pentium II PC
1982	56	417	102	105	85	$5.00 * 10^5$	$3.98 * 10^7$	$1.11 * 10^3$
1984	58	463	105	108	89	$1.45 * 10^6$	$4.57 * 10^7$	$3.22 * 10^3$
1986	60	513	107	111	96	$4.19 * 10^6$	$5.25 * 10^7$	$9.31 * 10^3$
1988	61	566	109	114	101	$1.21 * 10^7$	$6.04 * 10^7$	$2.69 * 10^4$
1990	63	622	112	117	106	$3.51 * 10^7$	$6.93 * 10^7$	$7.80 * 10^4$
1991	63	652	113	119	109	$5.97 * 10^7$	$7.43 * 10^7$	$1.33 * 10^5$
1992	64	682	114	120	112	$1.02 * 10^8$	$7.96 * 10^7$	$2.26 * 10^5$
1993	65	713	116	121	114	$1.73 * 10^8$	$8.54 * 10^7$	$3.84 * 10^5$
1994	66	744	117	123	117	$2.94 * 10^8$	$9.15 * 10^7$	$6.53 * 10^5$
1995	66	777	118	124	121	$5.00 * 10^8$	$9.81 * 10^7$	$1.11 * 10^6$
1996	67	810	120	126	122	$8.51 * 10^8$	$1.05 * 10^8$	$1.89 * 10^6$
1997	68	844	121	127	125	$1.45 * 10^9$	$1.13 * 10^8$	$3.22 * 10^6$
1998	69	879	122	129	129	$2.46 * 10^9$	$1.21 * 10^8$	$5.48 * 10^6$
1999	70	915	123	130	130	$4.19 * 10^9$	$1.29 * 10^8$	$9.31 * 10^6$
2000	70	952	125	132	132	$7.13 * 10^9$	$1.39 * 10^8$	$1.58 * 10^7$
2001	71	990	126	133	135	$1.21 * 10^{10}$	$1.49 * 10^8$	$2.70 * 10^7$
2002	72	1028	127	135	139	$2.06 * 10^{10}$	$1.59 * 10^8$	$4.59 * 10^7$
2003	73	1068	129	136	140	$3.51 * 10^{10}$	$1.71 * 10^8$	$7.80 * 10^7$
2004	73	1108	130	138	143	$5.98 * 10^{10}$	$1.83 * 10^8$	$1.33 * 10^8$
2005	74	1149	131	139	147	$1.02 * 10^{11}$	$1.96 * 10^8$	$2.26 * 10^8$
2006	75	1191	133	141	148	$1.73 * 10^{11}$	$2.10 * 10^8$	$3.84 * 10^8$
2007	76	1235	134	142	152	$2.94 * 10^{11}$	$2.25 * 10^8$	$6.54 * 10^8$
2008	76	1279	135	144	155	$5.01 * 10^{11}$	$2.41 * 10^8$	$1.11 * 10^9$
2009	77	1323	137	145	157	$8.52 * 10^{11}$	$2.59 * 10^8$	$1.89 * 10^9$
2010	78	1369	138	146	160	$1.45 * 10^{12}$	$2.77 * 10^8$	$3.22 * 10^9$
2011	79	1416	139	148	163	$2.47 * 10^{12}$	$2.97 * 10^8$	$5.48 * 10^9$
2012	80	1464	141	149	165	$4.19 * 10^{12}$	$3.19 * 10^8$	$9.32 * 10^9$
2013	80	1513	142	151	168	$7.14 * 10^{12}$	$3.41 * 10^8$	$1.59 * 10^{10}$
2014	81	1562	143	152	172	$1.21 * 10^{13}$	$3.66 * 10^8$	$2.70 * 10^{10}$
2015	82	1613	145	154	173	$2.07 * 10^{13}$	$3.92 * 10^8$	$4.59 * 10^{10}$
2016	83	1664	146	155	177	$3.51 * 10^{13}$	$4.20 * 10^8$	$7.81 * 10^{10}$
2017	83	1717	147	157	180	$5.98 * 10^{13}$	$4.51 * 10^8$	$1.33 * 10^{11}$
2018	84	1771	149	158	181	$1.02 * 10^{14}$	$4.83 * 10^8$	$2.26 * 10^{11}$
2019	85	1825	150	160	185	$1.73 * 10^{14}$	$5.18 * 10^8$	$3.85 * 10^{11}$
2020	86	1881	151	161	188	$2.94 * 10^{14}$	$5.55 * 10^8$	$6.54 * 10^{11}$
2021	86	1937	153	163	190	$5.01 * 10^{14}$	$5.94 * 10^8$	$1.11 * 10^{12}$
2022	87	1995	154	164	193	$8.52 * 10^{14}$	$6.37 * 10^8$	$1.89 * 10^{12}$
2023	88	2054	156	166	197	$1.45 * 10^{15}$	$6.83 * 10^8$	$3.22 * 10^{12}$
2024	89	2113	157	167	198	$2.47 * 10^{15}$	$7.32 * 10^8$	$5.48 * 10^{12}$
2025	89	2174	158	169	202	$4.20 * 10^{15}$	$7.84 * 10^8$	$9.33 * 10^{12}$
2026	90	2236	160	170	205	$7.14 * 10^{15}$	$8.41 * 10^8$	$1.59 * 10^{13}$
2027	91	2299	161	172	207	$1.21 * 10^{16}$	$9.01 * 10^8$	$2.70 * 10^{13}$
2028	92	2362	162	173	210	$2.07 * 10^{16}$	$9.66 * 10^8$	$4.59 * 10^{13}$
2029	93	2427	164	175	213	$3.52 * 10^{16}$	$1.04 * 10^9$	$7.81 * 10^{13}$
2030	93	2493	165	176	215	$5.98 * 10^{16}$	$1.11 * 10^9$	$1.33 * 10^{14}$
2032	95	2629	168	179	222	$1.73 * 10^{17}$	$1.27 * 10^9$	$3.85 * 10^{14}$
2034	96	2768	171	182	227	$5.01 * 10^{17}$	$1.46 * 10^9$	$1.11 * 10^{15}$
2036	98	2912	173	185	232	$1.45 * 10^{18}$	$1.68 * 10^9$	$3.22 * 10^{15}$
2038	99	3061	176	188	239	$4.20 * 10^{18}$	$1.93 * 10^9$	$9.33 * 10^{15}$
2040	101	3214	179	191	244	$1.22 * 10^{19}$	$2.22 * 10^9$	$2.70 * 10^{16}$
2042	103	3371	182	194	248	$3.52 * 10^{19}$	$2.55 * 10^9$	$7.82 * 10^{16}$
2044	104	3533	185	197	255	$1.02 * 10^{20}$	$2.93 * 10^9$	$2.26 * 10^{17}$
2046	106	3700	187	200	260	$2.95 * 10^{20}$	$3.36 * 10^9$	$6.55 * 10^{17}$
2048	107	3871	190	203	265	$8.53 * 10^{20}$	$3.86 * 10^9$	$1.90 * 10^{18}$
2050	109	4047	193	206	272	$2.47 * 10^{21}$	$4.44 * 10^9$	$5.49 * 10^{18}$
2050	109	4047	193	206	272	$2.47 * 10^{21}$	$4.44 * 10^9$	$5.49 * 10^{18}$

表 1.3: 表 1.2 の各列の値

Symmetric Key Size :
共通鍵暗号方式の鍵長の下限．(単位:ビット)

Asymmetric Key Size (and SDL Field Size) :
RSA、および有限体 F_p と生成元 g からなる離散対数問題に基づく公開鍵暗号方式 (TDL: Traditional Discrete Logarithm) の鍵長の下限．(単位:ビット)
かつ、有限体 F_p の部分乗法群上での離散対数問題に基づく公開鍵暗号方式 (SDL) の体のサイズの下限．(単位：ビット)

Subgroup Discrete Logarithm Key Size :
SDL の鍵長の下限．(単位:ビット)

Elliptic Curve Key Size :
楕円暗号 (EC) の鍵長の下限．(単位:ビット)
($c = 0$,$c = 18$ の別は表 1.4 を参照).

Infeasible number of Mips Years :
攻撃が成功するための計算量の下限．(単位:Mips Years).

Lower bound for Hardware cost in US$ for a 1 day attak :
1 日で攻撃が成功するために必要なコスト．(単位：米ドル)

Corresponding number of years on 450MHz PentiumII PC :
450MHz PentiumII プロセッサ搭載の PC で攻撃成功にかかる年数．(単位：年)

表 1.4: 表 1.2 作成の前提条件

表 1.2 は、以下の前提条件の基で、共通鍵暗号方式、RSA、TDL、SDL、EC に対する攻撃に必要な計算量が等しくなるような各方式の鍵長の下限を導出している.

(1) 共通鍵暗号に対する攻撃は鍵の全数探索とする.
(2) RSA および TDL に対する攻撃は数体ふるいとする.
(3) SDL、EC に対する攻撃は ρ-法とする.
(4) ハッシュ関数に対する攻撃は中間一致攻撃とする．鍵長 b の共通鍵暗号に対する全数探索攻撃は、ハッシュ長 $2b$ のハッシュ関数に対する中間一致攻撃に匹敵すると仮定する.
(5) 1982 年時点での DES の安全性を限界基準とする．DES の鍵の全数探索にかかる計算量を $5.0 * 10^5$Mips Years とする.
(6) プロセッサのスピードおよびメモリサイズは 18ヶ月で 2 倍になる (ムーアの法則).
(7) 1 ドルで得られる計算能力およびメモリは 18ヶ月で 2 倍になる.
(8) 10 年で国民総生産が 2 倍になる.
(9) 暗号研究の発展により、共通鍵暗号に対する攻撃にかかるコストは 18ヶ月で 1/2 倍になる.
(10) 暗号研究の発展により、楕円暗号に対する攻撃にかかるコストは変わらない (c=0)、または 18ヶ月で 1/2 倍になる (c=18).
(11) 450MHz PentiumII 64Mbyte RAM 搭載の PC の計算量を 450Mips Years とし、値段を 100 米ドルとする.
(12) 共通鍵暗号の 1 ブロック暗号化の段数は 1 段とする.

表 1.5: 「電子署名及び認証業務に関する法律に基づく関係政省令等に盛り込む事項について 別添 1」(平成 12 年 11 月 20 日配布)[1-13] より抜粋

認定の対象となる認証業務〈法第 2 条第 3 項関係、省令事項〉

(1) 認定の対象となる認証業務は、次のいずれかの困難性相当以上の安全性を有する電子署名について行われる認証業務とする.
ア ほぼ同じ大きさの 2 乗数の積である 1,024 ビットの正数の素因数分解
イ 大きさ 1,024 ビットの有限体上の乗法群における離散対数計算
ウ 楕円曲線上の点がなす大きさ 160 ビットの群における離散対数計算
エ その他各号に規定する困難性に相当するものとして、専門家の意見を踏まえ、主務大臣が定めるもの

(中略)

(3) 次に挙げる電子署名方式は、(1) の基準を満たすものとして取り扱う.
ア RSA 方式 (電子署名アルゴリズム識別子 1 2 840 113549 1 1 5) であって、鍵長が 1,024 ビット以上のもの. (当面の間、電子署名アルゴリズム識別子 1 2 840 113549 1 1 4 の RSA 方式も認めるものとする.)
イ ESIGN 方式 (国際標準化機構規格 14888-3) であって、鍵長が 1,024 ビット以上、検証に利用されるべき乗指数が 8 以上であって、ハッシュ関数が SHA1 方式のもの. (当面の間、MD5 方式のハッシュ関数を使用したものも認めるものとする.)
ウ ECDSA 方式 (米国規格協会規格 (ANSI)X9.62) であって、鍵長が 160 ビット以上のもの.
エ DSA 方式 (電子署名アルゴリズム識別子 1 2 840 10040 4 3) であって、鍵長が 1,024 ビットのもの.

参考文献

[1-1] M. Bellare and P. Rogaway, “Optimal asymmetric encryption.” In Proc. of Eurocrypt’94, LNCS 950, pp.92-111, Springer-Verlag, 1994.

[1-2] R. Cramer and V. Shoup, “A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack.” In Proc. of Crypto’98, LNCS 1462, pp.13-25, Springer-Verlag, 1998.

[1-3] J.R.T.Brazier, “Possible NSA decryption capabilities.” http://jya.com/nsa-study.htm.

[1-4] A.M.Odlyzko, “The future of integer factorization.”, RSA Laboratories’ Cryptobytes, Vol.1, No.2, pp.5-12, 1995, available from http://www.rsa.com/rsalabs/cryptobytes/.

[1-5] D.B.Johnson, “ECC, Future resiliency and high security systems”, March 30, 1999, available from http://www.certicom.com/research.html.

[1-6] A.K.Lenstra and E.R.Verheul, “Selecting cryptographic key sizes.” In Proc. of PKC’2000, LNCS , pp. , Springer-Verlag, 2000.

[1-7] P.W.Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer.” SIAM Journal of Computing, Vol.26, No.5, pp.1484-1509, 1997.

[1-8] RSA Laboratories, DES Challenge III, http://www.rsasecurity.com/rsalabs/des3/.

[1-9] RSA Laboratories, RSA Factoring Challenge, RSA-140, http://www.rsasecurity.com/rsalabs/challenges/factoring/rsa140.html.

[1-10] RSA Laboratories, RSA Factoring Challenge, RSA-155, http://www.rsasecurity.com/rsalabs/challenges/factoring/rsa155.html.

[1-11] 情報処理振興事業協会 (IPA) セキュリティセンター, “情報セキュリティの現状 2000 年版”, Rev.1.1, 2001 年 2 月, available from http://www.ipa.go.jp/security/fy12/sec2000/.

[1-12] 電子署名及び認証業務に関する法律 (平成 12 年 5 月 1 日法律第 102 号), available from http://www.mpt.go.jp/top/ninshou-law/law-index.html.

[1-13] 電子署名及び認証業務に関する法律の施行に関する意見募集, 平成 12 年 11 月 20 日, available from http://www.mpt.go.jp/top/ninshou-law/pub.html.

[1-14] T.Dierks and C.Allen, "The TLS Protocol Version 1.0," RFC2246, Jan. 1999.

[1-15] A.Frier, P.Karlton and P.Kocher, "The SSL 3.0 Protocol," Netscape Communications Corp., Nov 18, 1996.

[1-16] J.Kohl and C.Neuman, "The Kerberos Network Authentication Service (V5)," RFC1510, Sep. 1993.

[1-17] NIST FIPS PUB 186, "Digital Signature Standard," National Institute of Standards and Technology, U.S. Department of Commerce, May 18, 1994.

[1-18] 日本認証サービス株式会社 (JCSI), http://www.jcsinc.co.jp/.

[1-19] National Institute of Standards and Technology (NIST), Advanced Encryption Standard (AES), http://csrc.nist.gov/encryption/aes/.

[1-20] Computer Security and Industrial Cryptography (COSIC), New European Schemes for Signatures, Integrity, and Encryption(NESSIE), https://www.cosic.esat.kuleuven.ac.be/nessie/.

[1-21] 情報処理振興事業協会 (IPA), CRYPTREC, http://www.ipa.go.jp/security/enc/CRYPTREC/.

[1-22] Microsoft Corp., Windows2000 Kerberos Authentication, July 9, 1999, http://www.microsoft.com/windows2000/.

第2章　情報量的安全性に基づく方式の研究動向調査

近年、計算機や計算アルゴリズムの研究および開発は急激な進歩をみせている。これに伴い、現在利用されている暗号・認証方式の将来における安全性は必ずしも保証されないものとなってきている。とくに、将来完成するであろう量子計算機はこれらの暗号・認証方式の安全性を著しく脅かすものであることがわかっている [2-7, 40]。このような問題は、素因数分解や離散対数問題といった計算量的な困難性の仮定を用いる限り、不可避であるといえる。したがって、長期にわたる安全性が必要とされる状況においては、計算量的な仮定を必要とする方式の利用は適切ではない。将来にわたる高い安全性を実現するためのアプローチとして情報量的安全性に基づいて暗号・認証方式を設計することが考えられる。本章では、情報量的安全性に基づく暗号・認証方式に関する研究の動向について報告する。

2.1　情報量的安全性に基づく暗号・鍵共有方式

計算量的な困難性に関するさまざまな仮定を利用することができないため、情報量的安全性に基づいて暗号・認証方式を設計することは必ずしも容易ではない。そのため、情報量的安全性に基づく方式の利用には、通常なんらかの制限が要求される。具体的には、システム利用者数、結託攻撃者数、利用回数などに上限が求められることになる。これらの値の上限は利用可能な記憶容量に大きく依存するものであるため将来的にはこのような問題は徐々に無視できるものとなっていくものと思われる。本節においては、情報量的安全性に基づく暗号方式および鍵共有方式の実現例およびこれらに必要となる記憶容量に関する研究について述べる。なお、暗号システムを構成する際、実際には鍵共有を行う必要が生じるため、情報量的安全性に基づく鍵共有方式についてもあわせて述べる。

2.1.1　方式の説明

2.1.1.1　情報量的安全性に基づく暗号方式

通常、暗号方式は共通鍵暗号および公開鍵暗号に分類される。共通鍵暗号については、情報量的安全性に基づいて構成する方法は古くから知られており、One-time pad もしくは Vernam 暗号 [2-46] と呼ばれている。One-time pad においては、メッセージの送信者およ

び受信者はあらかじめ鍵 $k \in GF(q)$ を共有しておくものとする。送信者は平文 $m \in GF(q)$ に対して、

$$c = m + k$$

を計算し、暗号文 c を受信者に送信する。受信者は

$$m = c - k$$

により m を復号する。Shannon[2-39] によって示されているように、暗号文より平文に関する情報が一切得られないことを保証するためには暗号化に用いられる鍵のもつ情報量は平文の情報量以上でなくてはならない。上記の暗号方式においては、m, k はいずれも $GF(q)$ の元であり、平文および鍵の情報量は等しい。したがって、この方式は鍵長に関し最適であるといえる。しかしながら、平文と同じサイズの鍵を利用することは効率的であるとはいえず、とくに、そのような鍵を送信者-受信者間で共有するための方法について検討を行う必要がある。現実的には、現在提案が行われている共通鍵暗号方式のうち、AES など充分安全性の検討が行われているものに関しては、量子計算機などの出現後も高い安全性を維持するものと考えられているため、当面は効率性を重視しこれらの計算量的な安全性に基づく共通鍵暗号方式を利用していくことになると思われる。

一方、公開鍵暗号方式については、現在のところ情報量的安全性に基づく方式は知られていない。一般的な公開鍵暗号方式のモデルにおいては送信者と攻撃者の間で持ちうる情報に差異がないため、情報量的安全性のみに基づいてこれを構成することは困難である。情報量的安全性に基づき、公開鍵暗号方式に類似した機能を実現するためには、従来の公開鍵暗号のモデルとは異なる構成を行う必要があると考えられる。

2.1.1.2　情報量的に安全な鍵事前配送方式

すでに述べられたとおり、情報量的安全性に基づき共通鍵暗号方式を構成することは可能であり、また、すでに安全性の検討が充分に行われている（計算量的に安全な）共通鍵暗号方式は量子計算機の出現後も高い安全性を維持するものと思われている。これらの共通鍵暗号方式を利用する際、安全に鍵配送を行う必要が生じる。ここでは、情報量的に安全な鍵配送方式の研究動向を報告する。

情報量的に安全な鍵配送方式は、Blom[2-4] により MDS 符号を用いた方式がはじめて提案され、後に、松本, 今井 [2-30] により一般的な構成方法が示されている。松本, 今井の構成方法に基づく鍵配送方式は、計算量的な安全性に基づくものも含め、鍵事前配送方式 (KPS: Key Predistribution Systems) と総称される。

KPS における、送信者-受信者間の鍵共有手順の概略を次に示す。まず、信頼できる機関 (TA) が対称関数 $f(x, y)$ を作成し、利用者 i に対し秘密アルゴリズム $s_i(x) := f(x, i)$ を発行する。利用者 i は利用者 j との共有鍵 k_{ij} を $k_{ij} = s_i(j)$ により導出する。厳密な KPS の構成方法については [2-30] を参照されたい。情報量的安全性に基づく KPS の構成例として線形スキーム [2-30] などが挙げられる。線形スキームにおいては、TA は $(\omega + 1) \times (\omega + 1)$

対称行列 A を作成し、利用者 i に対し秘密アルゴリズム

$$\mathbf{s}_i := \mathbf{v}_i A$$

を発行する。ここで、ω は想定される最大結託者数とし、$\mathbf{v}_i$ は i より一意に定まる $\omega+1$ 次の横ベクトルとする。また、$\mathbf{v}_i$ より i が一意に求まるものとする。利用者 i は利用者 j との共有鍵 k_{ij} を

$$k_{ij} = \mathbf{s}_i \cdot {}^t\mathbf{v}_j$$

により導出する。ここでは、二者間の鍵共有方法について述べたが、KPS により三者以上によるグループ鍵配送も容易に実現することができる [2-5, 14, 10]。

次に、情報量的安全性に安全性に基づく KPS に必要となるメモリサイズについて議論を行う。ここまで、情報量エントロピーを用いて KPS に必要となるメモリサイズの下界を導出する研究が行われている [2-5, 28, 6, 29]。これらの結果のひとつとして、想定される最大結託者数を ω とすると、t 人の利用者によるグループ鍵の共有が可能な KPS において、利用者がもつ秘密アルゴリズムに必要となるメモリサイズは

$$H(S_i) \geq \left(\begin{array}{c} \omega+t-1 \\ t-1 \end{array} \right) H(K)$$

となることが導かれている [2-5]。ここで、$H(S_i)$ は利用者の秘密アルゴリズムに必要となるメモリサイズとし、$H(K)$ を共有する鍵のメモリサイズとする。線形スキームや Blom の方式において秘密アルゴリズムのメモリサイズは $(\omega+1)H(K)$ であり、$t=2$ の場合の上記の不等式の等号が成立する。したがって、線形スキームおよび Blom の方式は $t=2$ の場合における最適な KPS の実現方式であるといえる。一般的な t に対する最適な KPS の実現方式は Blundo ら [2-5] によって提案されている。また、各利用者の秘密アルゴリズムの配布後に t の値を決定可能な方式のうち最適な KPS の実現方式は Fiat, Naor[2-14] により提案されている。TA が保持する秘密情報に必要となるメモリサイズの下界については [2-28, 44] などで同様な議論がなされており、TA の保持する秘密情報のメモリサイズについても線形スキーム, Blom の方式, Blundo らの方式が最適であることが示されている。ここに示されているように、現在すでに必要となるメモリサイズを最適とする KPS は存在しており、機能と安全性を維持したまま、必要となるメモリサイズをこれ以上削減することは不可能である。しかし、通常 KPS によって提供される機能のうち必ずしもすべてが必要となるわけではない。花岡, 西岡, Zheng, 今井 [2-18, 20] は利用者間の通信のうち、鍵共有を必要としないものの鍵共有機能を削除することでメモリサイズの削減を行っている。

KPS においては、TA に権力が集中するため問題となりうる。TA の分散化方法については [2-30, 28] などで述べられている。

2.1.1.3 同報通信暗号

KPS の応用技術のひとつとして同報通信における暗号方式が挙げられる。同報通信路上に送信される暗号文を、あらかじめ契約を行った利用者のみが復号可能となるような暗

号方式を同報通信暗号とよぶ [2-14]。同報通信暗号のもっとも簡単な実現方法のひとつとして次のような手法が考えられる [2-44, 29]。まず、情報発信者が TA となって各利用者に KPS の秘密アルゴリズムを配布する。この後、配信される情報の受信を希望する利用者は情報発信者と契約を行う。すべての契約者が決定すると、情報発信者はこれらの契約者のみが KPS により共有する鍵を導出し、この鍵で配信される情報の暗号化を行う。契約者は他のすべての契約者との共有鍵を導出し、これを用いて暗号化された情報を復号する。この際、これまでに述べられた情報量的安全性に基づく KPS を利用することで情報量的安全性に基づく同報通信暗号を実現することができる。ただし、利用者数が大きい場合、この手法を単純に適用したのでは利用者がもつ秘密情報のメモリサイズは非常に大きくなることが知られている。この問題に対する解決法のひとつとして、暗号文に冗長性を持たせる方法がある [2-44, 25]。ただし、この場合暗号文のメモリサイズが増大するため、適切にパラメータの設定を行う必要がある。

2.1.1.4　量子鍵配送

情報量的に安全な鍵共有を行う KPS とは異なる手法のひとつとして量子チャネルを用いて送信者-受信者間でインタラクティブなプロトコルを実行するものがある。このようなプロトコルは Bennet, Brassard[2-3] により初めて提案がなされた。Bennet, Brassard のプロトコルにおける送信者-受信者間の鍵共有の概略を次に示す。まず、二種類の量子状態の測定系を作成する。たとえば、情報のキャリアとして光子を利用する場合、ある任意に選んだ測定系 ($\oplus$ 測定系) およびこれを 45 度傾けた測定系 ($\otimes$ 測定系) をとる。このとき、$\oplus$ 測定系においては、偏光の状態は水平方向に直線偏光した状態 (水平偏光) と垂直方向に偏向した状態 (垂直偏光) の 2 つの直交規格化された状態で表現できる。このとき、$\otimes$ 測定系においては、偏光の状態は 45 度偏光した状態 (45 度偏光) と 135 度偏光した状態 (135 度偏光) という 2 つの直交規格化された状態で表現されることになる。送信者はランダムに測定系を選択し、その測定系における直交規格化された状態のうちのひとつをランダムに選び、選択した状態を光子を用いて受信者に送信する。受信者はランダムに測定系を選択し、受信した光子の状態を選択した測定系を用いて観測する。これを送信者-受信者間で繰り返した後、受信者は選択した測定系を受信者に公開の通信路を用いて伝える。送信者は自分の測定系と一致したものを同じ通信路を用いて回答する。送信者-受信者間で選択した測定系が一致したときに送信された光子の状態に関する情報は、送信者-受信者間で共有される。上記のやりとりを $2N$ 回繰り返した場合、送信者-受信者間では平均して N ビットの鍵を共有することができる。このプロトコルの詳細は [2-3] を参照されたい。上記の Bennet, Brassard のプロトコルの提案以降、量子チャネルを用いたさまざまな鍵共有プロトコルの提案が行われている。たとえば、非直交な状態を用いた構成法 [2-1] や EPR 対と呼ばれるエンタングルした量子対を用いる構成法 [2-13] などがある。量子チャネルを用いた鍵共有方式の実装実験として [2-2] などがある。

2.1.1.5 雑音のある通信路を用いた鍵共有法

最近、検討が行われている情報量的に安全な鍵共有方式として雑音のある通信路を用いる方法がある [2-33, 34]。単一の情報源から発信された情報が雑音のある通信路をとおって各利用者に送信されるものとする。このとき、各利用者のもつ情報にのっている雑音は利用者ごとに異なるものであるため、信号雑音比によらず、任意の送信者-受信者間でのみ共有される情報が存在することになる [2-31]。これを利用して鍵共有を行うことができる。情報源から送信者もしくは受信者までの通信路の信号雑音比が、情報源から攻撃者までの通信路の信号雑音比に劣るものであったとしても理論上鍵共有は可能である。

2.1.2 研究動向

計算量的な安全性に対する危機感から、すでに述べられたように、近年、情報量的安全性に基づく暗号・鍵共有方式に関する研究が活発に行われている。

KPS に関しては、一般的なモデルに対してはすでに最適な方式が存在しており、現在はなんらかの制限が与えられたモデルに対する最適化に関し研究が行われている。たとえば、有料道路料金自動収集システムに対し最適化された KPS の実装方法 [2-19] などが提案されている。

同報通信暗号に関しても、近年、活発に研究が行われており、利用者のもつ秘密情報のサイズと通信量のトレードオフを効率的に設定可能な方式の検討などがなされている。また、暗号化された情報の復号鍵を漏洩した利用者を特定することが可能な方式の研究も活発に行われている。

量子鍵配送の実装の成功例が多数報告されており、国内でも三菱電機の研究グループによる実装実験の報告がなされている [2-24]。現在は、長距離区間における鍵配送の実装実験などが行われている。

雑音のある通信路を用いた鍵配送は、量子鍵配送と同様、信頼される機関を必要としないため今後の進展が注目される。

2.1.3 課題とまとめ

情報量的安全性に基づく暗号・鍵共有方式を実装する上で、記憶容量が膨大となりうることが知られている。これを踏まえ、今後はより少ないメモリサイズで高い安全性を実現するための研究が活発になっていくものと思われる。記憶装置の大容量化・低価格化も現在急激にすすんでおり、情報量的安全性に基づく方式の有用性はより一層高まっていくものと考えられる。情報量的安全性に基づく方式の実用化に向けて、現在利用されている暗号インフラとの効率的な併用方法などについても検討が必要であろう。

2.2　情報量的安全性に基づく認証方式

情報量的に安全な認証方式は [2-15] において初めて提案がなされて以来、幅広く研究が行われている。本節においては、これらの研究における主要な成果についての紹介を行う。

2.2.1　方式の説明

2.2.1.1　認証符号

もっとも基本的な認証方式は送信者-受信者間で共有される鍵を用いて構成される [2-15, 41]。このような認証方式においては、送信者はまず受信者に対し送信するメッセージ m を選択し、受信者との共有鍵 k を用いて m に対する認証子 c を作成し、これを受信者に送信する。受信者は c が k を用いて作成された正しい認証子であることを k を用いて検証する。この認証方式に対する攻撃として、なりすまし攻撃 (impersonation) と置き換え攻撃 (substitution) が考えられる。なりすまし攻撃において、攻撃者は、正しい認証子として受信者が受理するように不正に認証子を作成する。置き換え攻撃においては、攻撃者は送信者が作成した正しい認証子を観測した後、これを不正な認証子に置き換えるものとする。なりすまし攻撃および置き換え攻撃の成功確率をそれぞれ P_I, P_S とし、K, M, C をそれぞれ k, m, c の確率変数とすると、次のような不等式が導き出されることが知られている [2-32]。

$$\begin{aligned} P_I &\geq 2^{-I(C;K)}, \\ P_S &\geq 2^{-H(K|C)}, \\ \max(P_I, P_S) &\geq 2^{-H(K)/2} \end{aligned}$$

上記の不等式の等号を満足する認証方式の実現例として、次のような構成が考えられる。$m \in GF(q)$ であり、$k = \{a, b\}$, $a, b \in GF(q)$ とする。送信者は、

$$c = am + b$$

を計算し、$\{m, c\}$ を受信者に送信する。受信者は、$c = am + b$ となることを検証する。この構成において、$P_I = 1/q$, $P_S = 1/q$ となることがわかる。

2.2.1.2　非対称認証符号

すでに述べられたように、送信者-受信者間の共有鍵を利用する場合、情報量的に安全な認証方式を容易に構成することができる。しかし、送信者-受信者間の共有鍵を利用した構成においては、送信者によって送信される正しい認証子は、受信者によっても作成可能であるため、受信者は送信されたメッセージと認証子が、送信者により作成されたものであることを第三者に対して主張することはできない。このような問題を解決する方式として、A^2-code [2-42, 43, 26, 27, 35], A^3-code [2-8, 11, 16, 45, 17, 47] と呼ばれる送信者-受

信者間のもつ情報に非対称性を有する認証符号の提案が行われている。A^2-code, A^3-code は、送信者と受信者のもつ秘密情報をそれぞれ異なるものとし、認証子の正当性の真偽に関し送信者と受信者で主張が異なった場合、どちらの主張が正しいかを arbiter とよばれる第三者が客観的に判断できる方式である。A^2-code においては arbiter はいかなる攻撃者とも結託を行わないことが要求されるが、A^3-code においてはこの仮定が弱められたものとなっている。A^2-code, A^3-code の利用により、メッセージの送信元に関して客観的な判断を下すことが可能となるが、これらの方式においては受信者が特定されているため電子署名のように複数の受信者が検証を行うことはできない。

2.2.1.3 同報通信のための認証符号

同報通信における認証を行う際、複数の受信者による検証が可能であることが要求される。送信者と個々の受信者間においてそれぞれ鍵を共有し、これ利用して認証を行う方式が考えられるが、これは非効率的である。この問題を解決する方式として、同報通信のための認証符号 (MRA: Multireceiver authentication code)[2-12, 36, 17, 37, 38] の提案が行われている。MRA においては、想定される最大結託者数を適切に設定することで、各利用者もつ秘密情報のメモリサイズを大幅に削減することが可能である。

MRA は複数の受信者による検証が可能な方式であるが、認証子の正当性の真偽に関し、第三者が必ずしも客観的に判断を下すことができるとはかぎらない。さらに、現在までに提案されている MRA の多くにおいては送信者により発行された認証子に対し、任意のエンティティは、特定の受信者のみがこの認証子を正当なものとして受理するように改竄を加えることができる。そのため、受信者間での認証子の転送を許す場合、認証子を受信した受信者がこれに上記のような改竄を行うことで次のような攻撃を行うことができる。攻撃者は、上記の改竄により攻撃の対象となる受信者のみがこの認証子を受理するような認証子の偽造を施し、これを攻撃対象となる受信者に送信する。攻撃対象となった受信者は認証子を正当なものとして受理してしまうが、第三者に対し、この正当性を主張することはできない。したがって、MRA においては一般的に複数の検証者間で認証子を転送することはできない。

2.2.1.4 情報量的安全性に基づく電子署名方式

ここまでにすでに述べられたように、A^2-code, A^3-code や MRA は電子署名に類似した機能を有しているが、これらを電子署名として利用することはできない。情報量的安全性に基づく電子署名方式の実現は、Chaum, Roijakkers [2-9] により初めて試みられたが、この方式は 1bit のメッセージのみに対し署名の生成が可能であり、また、検証者間における電子署名の転送に関し、安全性に問題を残している。また、Chaum らの方式を A^3-code を用いて拡張した方式 [2-17] の提案がなされているが、検証者間における電子署名の転送に関しては検討がなされていない。最近、花岡, 四方, Zheng, 今井により、検証者間における電子署名の転送が可能である、初めての実用的な情報量的安全性に基づく電子署名方式が提案されており [2-21, 23]、この方式がいかなる計算量的な困難性の仮定も用いずに

安全性を証明できることも併せて示されている。同署名方式は、署名発行回数や署名の対象となるメッセージのサイズに制限があるものの、鍵管理を適切に行う限り将来にわたる確実な安全性を保証することができる。長期にわたる安全性の保証を必要とするアプリケーションに対しては同署名方式が特に有効である。花岡, 四方, Zheng, 今井の署名方式の概略を次に述べる。

利用者数、想定される利用者の最大結託数、利用者一人あたりの最大署名発行回数をそれぞれ n, ω, ψ とする。まず、信頼できる機関 (TA) はランダムに n 個の $GF(q)^\omega$ 上の元 $v_1, v_2, \ldots, v_n$ を利用者 $U_1, U_2, \ldots, U_n$ のそれぞれに対して選択する。また、次のように $F(x, y_1, \ldots, y_\omega, z)$ を生成する：

$$F(x, y_1, \ldots, y_\omega, z) = \sum_{i=0}^{n-1} \sum_{k=0}^{\psi} a_{i0k} x^i z^k + \sum_{i=0}^{n-1} \sum_{j=1}^{\omega} \sum_{k=0}^{\psi} a_{ijk} x^i y_j z^k$$

ここで、各係数 a_{ijk} は F_q からランダムに選択されたものとする．また、$U_l, m \in GF(q)$ (U_l: 利用者の識別子, m: メッセージ) であるものとする。次に、各利用者 U_l $(1 \leq l \leq n)$ に対し、TA は署名鍵 $s_l := F(U_l, y_1, \ldots, y_\omega, z)$ と二種の検証鍵 $v_l := (v_{1,l}, \ldots, v_{\omega,l}) \in GF(q)^\omega$ および $\tilde{v_l} := F(x, v_l, z)$ を安全な通信路を用いて送信する。 その後は TA は利用者の秘密情報を保持する必要はない。

$m \in F_q$ に対して、 U_i は署名

$$sig_{i,m} := F(U_i, y_1, \ldots, y_\omega, z)|_{z=m} = F(U_i, y_1, \ldots, y_\omega, m)$$

を s_i を用いて生成する。

m および m に対する U_i の署名 $sig'_{i,m}$ を受信した U_j は v_j と $\tilde{v_j}$ を用いて署名検証を行なう。U_j は次のように k_{ijm}, k'_{ijm} を計算し、これらの比較を行なう。

$$\begin{aligned} k_{ijm} &:= F(x, v_j, z)|_{x=U_i, z=m}, \\ k'_{ijm} &:= sig'_{i,m}|_{(y_1,\ldots,y_\omega)=(v_{1,j},\ldots,v_{\omega,j})} \end{aligned}$$

$k_{ijm} = k'_{ijm}$ であれば、U_j は $sig_{i,m}$ を受理する。

この実装方式を用いた場合、無限の計算能力をもつ攻撃者であっても $2/q - 1/q^2$ 以上の確率で一切の攻撃を成功させることができない。この方式についての詳細は [2-21] を参照されたい。

2.2.2　研究動向

共有鍵を利用した情報量的に安全な認証方式の研究は、ほぼ完成しており、今後大きい進歩は特にないものと思われる。その一方で、A^2-code, A^3-code や MRA の研究は近年活発に行われており、特に、必要となる記憶容量に関してさまざまな発表が行われている。情報量的に安全な電子署名方式の研究はその重要性から今後一層幅広い研究が行われていくものと思われる。記憶容量などに関してより高い効率性を実現する方式の検討が行われている [2-22]。

2.2.3 課題とまとめ

長期にわたる署名の信頼性の実現のために、情報量的に安全な電子署名方式は今後特に重要となるものと思われる。情報量的安全性に基づく暗号・鍵共有方式同様、情報量的に安全な電子署名方式は必要となる記憶容量が膨大となるため、必要となる記憶容量の削減技術の開発が要求される。ただし、すでに述べられたとおり、記憶装置の大容量化・低価格化は現在急激にすすんでおり、情報量的安全性に基づく方式の有用性はより一層高まっていくものと考えられる。

参考文献

[2-1] C.H. Bennet, “Quantum cryptography using any two nonorthogonal states,” Physical Review Letters, vol.68, no. 21, pp.3121-3124, 1992.

[2-2] C.H. Bennet, F. Bessette, G. Brassard, L. Salvail and J. Smolin, “Experimental quantum cryptography,” Journal of Cryptology, vol. 5, no. 1, pp.3-28, Springer-Verlag, 1992.

[2-3] C.H. Bennet and G. Brassard, “Quantum cryptography: public key distribution and coin tossing,” Proc. of IEEE Int. Conf. on Comp. Sys. and Signal Proc., pp.175-179, 1984.

[2-4] R. Blom, “Non-public Key Distribution,” Proc. of CRYPTO’82, Plenum Press, pp.231-236, 1983.

[2-5] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro and M. Yung, “Perfectly Secure Key Distribution for Dynamic Conferences,” Proc. of CRYPTO’92, LNCS 740, Springer-Verlag, pp.471-486, 1993.

[2-6] C. Blundo, L.A. Frota Mattos and D.R. Stinson, “Trade-offs between Communication and Strage in Unconditionally Secure Schemes for Broadcast Encryption and Interactive Key Distribution,” Proc. of CRYPTO’96, LNCS 1109, Springer-Verlag, pp.387-400, 1996.

[2-7] D. Boneh and R. J. Lipton, “Quantum cryptanalysis of hidden linear functions,” Proc. of CRYPTO’95, LNCS 963, Springer-Verlag, pp.424-437, 1995.

[2-8] E. F. Brickell and D. R. Stinson, “Authentication codes with multiple arbiters,” Proc. of Eurocrypt’88, LNCS 330, Springer-Verlag, pp.51-55, 1988.

[2-9] D. Chaum and S. Roijakkers, “Unconditionally secure digital signatures,” Proc. of CRYPTO’90, LNCS 537, Springer-Verlag, pp.206-215, 1990.

[2-10] Y. Desmedt and V. Viswanathan, “Unconditionally Secure Dynamic Conference Key Distribution,” Proc. of IEEE, ISIT’98, 1998.

[2-11] Y. Desmedt and M. Yung, “Arbitrated unconditionally secure authentication can be unconditionally protected against arbiter’s attack,” Proc. of CRYPTO’90, LNCS 537, Springer-Verlag, pp.177-188, 1990.

[2-12] Y. Desmedt, Y. Frankel and M. Yung, “Multi-receiver/Multi-sender network security: efficient authenticated multicast/feedback,” Proc. of IEEE Infocom’92, pp.2045-2054, 1992.

[2-13] A.K. Ekert, “Quantum cryptography based on Bell’s theorem,” Physical Review Letters, vol.67 pp.661-663, 1991.

[2-14] A. Fiat and M. Naor, “Broadcast Encryption,” Proc. of CRYPTO’93, LNCS 773, Springer-Verlag, pp.480-491, 1994.

[2-15] E. N. Gilbert, F. J. MacWilliams and N. J. A. Sloane, “Codes which detect deception,” Bell System Technical Journal, 53, pp.405-425, 1974.

[2-16] T. Johansson, “Lower bounds on the probability of deception in authentication with arbitration”, IEEE Trans. Inform. Theory, IT-40, 5, pp.1573-1585, 1994.

[2-17] T. Johansson, “Further results on asymmetric authentication schemes,” Information and Computation, 151, pp.100-133, 1999.

[2-18] G. Hanaoka, T. Nishioka, Y. Zheng and H. Imai, “An Efficient Hierarchical Identity-based Key-Sharing Method Resistant against Collusion-Attacks,” Proc. of Asiacrypt’99, LNCS 1716, Springer-Verlag, pp.348-362, 1999.

[2-19] G. Hanaoka, T. Nishioka, Y. Zheng and H. Imai, “An Optimization of Credit-Based Payment for Electronic Toll Collection Systems,” IEICE Trans., vol. E83-A, no.8, pp.1681-1690, 2000.

[2-20] G. Hanaoka, T. Nishioka, Y. Zheng and H. Imai, “Optimal Unconditionally Secure ID-Based Key Distribution Scheme for Large-Scaled Networks,” IEICE Trans., vol.E84-A, no.1, pp.222-230, 2001.

[2-21] G. Hanaoka, J. Shikata, Y. Zheng and H. Imai, “Unconditionally secure digital signature schemes admitting transferability,” Proc. of Asiacrypt2000, LNCS 1976, Springer-Verlag, pp.130-142, 2000.

[2-22] 花岡，四方，Zheng，今井, “情報量的安全性に基づく転送可能な One-Time 署名方式,” 2001 年暗号と情報セキュリティシンポジウム予稿集, 2B-3, pp.43-48, 2001.

[2-23] G. Hanaoka, Y. Zheng and H. Imai, “Unconditionally secure ID-based digital signature scheme,” Proc. of the 22nd Symposium on Information Theory and Its Applications (SITA’99), pp.283-286, 1999.

[2-24] 長谷川, 西岡, 石塚, 安部, 清水, 松井, 竹内, “量子暗号システム実験,” 2001 年暗号と情報セキュリティシンポジウム予稿集, 2C-2, pp.67-72, 2001.

[2-25] S. Ravi Kumar, Sridhar Rajagopalan, Amit Sahai, “Coding Constructions for Blacklisting Problems without Computational Assumptions,” Proc. of CRYPTO’99, LNCS 1666, Springer-Verlag, pp.609-623, 1999.

[2-26] K. Kurosawa, “New bound on authentication code with arbitration,” Proc. of CRYPTO’94, LNCS 839, Springer-Verlag, pp.140-149, 1994.

[2-27] K. Kurosawa and S. Obana, “Combinatorial bounds for authentication codes with arbitration,” Proc. of Eurocrypt’95, LNCS 921, Springer-Verlag, pp.289-300, 1995.

[2-28] K. Kurosawa, K. Okada and K. Sakano, “Security of the center in key distribution schemes,” Proc. of Asiacrypt’94, LNCS 917, Springer-Verlag, pp.333-341, 1995.

[2-29] K. Kurosawa, T. Yoshida, Y. Desmedt and M. Burmester, “Some Bounds and a Construction for Secure Broadcast Encryption,” Proc. of Asiacrypt’98, LNCS 1514, Springer-Verlag, pp.420-433, 1998.

[2-30] T. Matsumoto and H. Imai, “On the KEY PREDISTRIBUTION SYSTEM: A Practical Solution to the Key Distribution Problem,” Proc. of CRYPTO’87, LNCS 293, Springer-Verlag, pp.185-193, 1987.

[2-31] U.M. Maurer, “Secret key agreement by public discussion from common information,” IEEE Trans. on Information Theory, vol. 39, no. 3, pp.733-742, 1993.

[2-32] U.M. Maurer, “A unified and generalized treatment of authentication theory,” Proc. of 13rd Symp. on Theoretical Aspects of Computer Science - STACS’96, LNCS 1046, Springer-Verlag, pp.387-398, 1996.

[2-33] U.M. Maurer, “Information-theoretically secure secret-key agreement by NOT authenticated public discussion,” Proc. of Eurocrypt’97, LNCS 1233, Springer-Verlag, pp.209-225, 1997.

[2-34] U.M. Maurer and S. Wolf, “Unconditionally secure key agreement and the intrinsic conditional information,” IEEE Trans. on Information Theory, vol. 45, no. 2, pp.499-514, 1999.

[2-35] S. Obana and K. Kurosawa, “A^2-code = affine resolvable + BIBD,” Proc. of ICICS’97, LNCS 1334, Springer-Verlag, pp.118-129, 1997.

[2-36] R. Safavi-Naini and H. Wang, “New results on multi-receiver authentication codes,” Proc. of Eurocrypt’98, LNCS1403, pp.527-541, 1998.

[2-37] R. Safavi-Naini and H. Wang, "Broadcast authentication in group communication," Proc. of Asiacrypt'99, LNCS1716, Springer-Verlag, pp.399-411, 1999.

[2-38] R. Safavi-Naini and H. Wang, "Multireceiver authentication codes: models, bounds, constructions and extensions," Information and Computation, 151, pp.148-172, 1999.

[2-39] C.E. Shannon, "Communication theory of secrecy systems," Bell System Technical Journal, vol. 28, pp.656-715, 1949.

[2-40] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," SIAM J. Comp., 26, no.5, pp.1484-1509, 1997.

[2-41] G. J. Simmons, "Authentication theory/coding theory," Proc. of CRYPTO'84, LNCS 196, Springer-Verlag, pp.411-431, 1984.

[2-42] G. J. Simmons, "Message authentication with arbitration of transmitter/ receiver disputes," Proc. of Eurocyrpt'87, Springer-Verlag, pp.151-165, 1987.

[2-43] G. J. Simmons, "A Cartesian construction for unconditionally secure authentication codes that permit arbitration," Journal of Cryptology, 2, pp.77-104, 1990.

[2-44] D.R. Stinson, "On some methods for unconditionally secure key distribution and broadcast encryption," Designs, Codes and Cryptography, vol. 12, pp.215-243, 1997.

[2-45] R. Taylor, "Near optimal unconditionally secure authentication," Proc. of Eurocyrpt'94, LNCS 950, Springer-Verlag, pp.244-253, 1994.

[2-46] G.S. Vernam,"Cipher printing telegraph systems for secret wire and radio telegraphic communications," Journal of the American Institute for Electrical Engineers, vol. 55, pp. 109-115, 1926.

[2-47] Y. Wang and R. Safavi-Naini, "A^3-codes under collusion attacks," Proc. of Asiacrypt'99, LNCS 1716, Springer-Verlag, pp.390-398, 1999.

第3章　計算量的安全性に基づく方式の研究動向調査

3.1　ナップサック (Knapsacks) 問題に基づく方式

3.1.1　方式の説明

本節では単純な形の Merkle-Hellman 暗号 [3-17] を説明する。これに対して、より複雑な形の Merkle-Hellman 暗号（“iterated” Merkle-Hellman 暗号）も [3-17] において同時に提案されている。

正整数 $b_1, b_2, \ldots, b_n$ で

$$b_1 \approx 2^n,\ b_j > \sum_{i=1}^{j-1} b_i,\ (2 \leq j \leq n),\ b_n \approx 2^{2n} \tag{3.1}$$

をみたすものをとる。また、正整数 M, W を

$$\mathrm{GCD}(M, W) = 1, \qquad M > \sum_{j=1}^{n} b_j$$

をみたすようにとり、

$$a'_j = b_j \bmod M, \qquad 0 < a'_j < M, \qquad (1 \leq j \leq n)$$

を計算する。さらに、集合 $\{1, 2, \ldots, n\}$ 上のある置換 π に対して、

$$a_j = a'_{\pi(j)}, \qquad (1 \leq j \leq n)$$

を求める。

このとき、秘密鍵は $\{b_1, b_2, \ldots, b_n\}$, M, W, π であり、公開鍵は $\{a_1, a_2, \ldots, a_n\}$ である。

- 暗号化： 平文 $x = (x_1, x_2, \ldots, x_n)$ の暗号化は

$$c = \sum_{j=1}^{n} x_j a_j$$

で与えられる。

- 復号：暗号文 c に対して、まず、

$$s = cW^{-1} \bmod M, \qquad 0 \leq s \leq M$$

を計算する。このとき、

$$s = \sum_{j=1}^{n} x_j b_{\pi(j)} \bmod M$$

であり、$M > \sum_{j=1}^{n} b_j$, $0 \leq s < M$ より、

$$s = \sum_{j=1}^{n} x_j b_{\pi(j)}$$

がわかる。したがって、条件 (3.1) をみたすナップサック問題は容易に解けるため、平文 $x = (x_1, \ldots, x_n)$ を効率良く復号できる。

3.1.2 研究動向

これまでにNP困難問題 (NP-hard problem)、あるいはNP完全問題 (NP-complete problem）の困難性を利用した暗号系がいくつか提案されており、またそれらに対して効果的な攻撃法も提案されてきた。歴史的に見てみると、NP困難問題に基づいたはじめての暗号方式は、ナップサック (Knapsacks) 問題の困難性を利用する形で提案されている。

1978年、Merkle, Hellman [3-17] は比較的易しいナップサック問題を難しいと思われるようなナップサック問題に変換する形で、初めてNP困難問題に基づく公開鍵暗号方式を提案した。しかしながら、1982年、Shamir [3-22] により、単純な形のMerkle-Hellman暗号に対する攻撃法が提案された。ここで、Shamirはこの攻撃法の中で、Lenstraの論文 [3-14] の中に見られるアルゴリズムを利用している。また同年、Adleman [3-1] はLLLアルゴリズムを利用する形での攻撃法を提案した。その後、Brickell [3-4][3-5] によりこれらの攻撃法は、より複雑な形のMerkle-Hellman暗号（“iterated” Merkle-Hellman暗号）に拡張され、その結果、現実的なパラメータ設定においてはMerkle-Hellman暗号は安全でないことが示されている。

一方、密度 (density) がある程度小さなナップサック問題（Subset Sum問題）に対して、これをSVP(shortest vector problem) に帰着する方法が提案されており、これは 低密度攻撃 (Low-Density Attack) と呼ばれている。このアイデアは、密度 d が $d \leq 0.6463\cdots$ のときに初めてLagaris, Odlyzko [3-16] によって提案された。その後、この帰着法はCosterら [3-6] により、$d \leq 0.9408\cdots$ まで 拡張されている。ここで、元のナップサック問題を多項式時間で破るには、帰着されたSVPを多項式時間で解く必要がある。帰着後のSVPについて考えてみると、NP困難問題であるSVPは一般的に理論上は多項式時間では解けないと考えられるが、ある程度高い次元（ランク）までの (ナップサック問題から帰着される) SVPに対しては、ラティス基底縮小 (lattice reduction) アルゴリズムが実用上、望ましい結果を出してくれることが知られている。現実的には100 ∼ 200次元ぐらいまで

の低密度のナップサック問題にはこのアプローチは非常に効果的に働くことが報告されている [3-16] [3-25] [3-26]。ここで、低密度攻撃が、非常に大きな次元の低密度のナップサック問題に対して効果的に働くとは限らないが、いずれにせよ、この攻撃が効果的ではなくなるほど大きな n を取ると、ナップサック暗号の鍵サイズは非常に大きくなってしまい、もはや現実的なパラメータとしては適さなくなってしまう。

これらの経緯から、現実的なパラメータの範囲では Merkle-Hellman 暗号は安全でないことが示されたといえる。また、これまでに上記の攻撃法を考慮していくつかの工夫を施した方式も提案されてきたが、ナップサック問題を基にした方式の多くは現在まで生き残っているとは言い難い。この中で、ナップサック問題を基にした重要な暗号方式として、Chor-Rivest 暗号方式 [3-7] が挙げられるが、1997 年に Vaudenay [3-27] により破られている。

3.1.3 課題とまとめ

これまでの経緯からすると、ナップサック問題を安全性の根拠にしてきた暗号方式は、Merkle-Hellman 暗号提案以来いくつか提案されてきたが、十分な安全性を有するものが多いとは言い切れない。これまで、ナップサック問題を基にした暗号が提案される度に、何らかの攻撃法が提案されてきた。このような歴史的経緯を考慮すると、今後はナップサック問題を基にした暗号を提案するのに際し、ある妥当な仮定のもとでその安全性が証明される形で提案することが望ましいといえるだろう。

3.2 ラティス (Lattice) 問題に基づく方式

3.2.1 方式の説明

3.2.1.1 Ajtai-Dwork 暗号

まず、セキュリティパラメータ n に対して、

$$m = n^3,\ r_S = \frac{1}{n^3},\ r_B = 2^{O(n \log n)}$$

とおく。また、B, S をそれぞれ R^n における半径 r_S, r_B の n 次元球とする。

次に、R^n における n 次元単位球上の元 u をランダムに選ぶ。また、集合 $\{x \in B \mid \langle x, u \rangle\}$ からランダムに $a_1, a_2, \ldots, a_m$ を選ぶ。更に、S からランダムに $\delta_{i,j}$ $(1 \leq i \leq m,\ 1 \leq j \leq n)$ を選び、

$$\delta_i = \sum_{j=1}^{n} \delta_{i,j} \qquad (1 \leq i \leq m)$$

とおく。そして、

$$v_i = a_i + \delta_i, \qquad (1 \leq i \leq m)$$

とおく。このとき、$v_{i+1}, v_{i+2}, \ldots, v_{i+n}$ によって張られるラティス $\Lambda(v_{i+1}, v_{i+2}, \ldots, v_{i+n})$ に対して

$$\det \Lambda(v_{i+1}, \ldots, v_{i+n}) \geq \frac{r_B}{n^2}$$

をみたす最も小さな i を i_0 とする。そして、

$$w_j = v_{i_0+j} \qquad (1 \leq j \leq n)$$

とし、$\Lambda = \Lambda(w_1, \ldots, w_n)$ とおく。

秘密鍵は u であり、公開鍵は $v_1, \ldots, v_m$ 及び $i_0 \in \{1, \ldots, m\}$ である。

- 暗号化：Ajtai-Dwork 暗号はビットごとの暗号化方式であり、'0' 及び '1' は以下のようにして暗号化される。

  '0' を暗号化する場合には、まず $b_1, b_2, \ldots, b_m$ $(b_i \in \{0, 1\})$ をランダムに選び、

  $$c = \sum_{i=1}^{m} b_i v_i \bmod \Lambda$$

  を計算する。このとき、c が暗号文である。

  '1' を暗号化する場合には、Λ の中からランダムに c を選び、これを暗号文とする。

- 復号：暗号文 c に対して、$\tau = \langle c, u \rangle$ を計算する。このとき、もし τ がある整数から $1/n$ の範囲内にあれば '0' に復号し、そうでなければ '1' に復号する。

3.2.1.2 Goldreich-Goldwasser-Halevi 暗号

整数を成分とする $n \times n$ 正則行列を R とする。ただし、R の各行ベクトルはその長さが比較的短いもので考える（例えば、各成分が n の多項式オーダ）。また、L を R の各行ベクトルから生成される Z^n におけるラティスとする。さらに、B を R とは別の L の基底とし、いくつかのユニモジュラ行列を R にかけることにより生成されるものとする。 そして、σ をあるセキュリティパラメータとし、

$$\mathcal{E} = \{e = (e_1, e_2, \ldots, e_n) \in Z^n \mid e_i = \pm\sigma \ (1 \leq i \leq n)\}$$

とおき、また

$$\mathcal{M} = \{m = (m_1, m_2, \ldots, m_n) \in Z^n \mid -n \leq m_i \leq n \ (1 \leq i \leq n)\}$$

とする。

秘密鍵は R であり、公開鍵は B である。

- 暗号化：平文 $m \in \mathcal{M}$ に対して、その暗号化は

  $$c = mB + e$$

  で与えられる。ただし、$e \in \mathcal{E}$ はランダムに選ばれる。

- 復号： 暗号文 c に対して、Babai の Round-off アルゴリズム [3-3] により、平文

$$m = \left\lfloor cR^{-1} \right\rceil RB^{-1}$$

を復号する。

3.2.1.3 NTRU 暗号

p, q $(p < q)$ を $GCD(p, q) = 1$ をみたす正整数とし、$R = Z[X]/(X^N - 1)$ とおく。以下では、R 上の積を $*$ で表すものとする。更に、

$$\mathcal{L}_m = \{m \in R \mid m \text{ の係数は } -\frac{1}{2}(p-1) \text{ 以上} \frac{1}{2}(p-1) \text{ 以下 }\}$$

とし、

$$\mathcal{L}(d_1, d_2) = \{F \in R \mid F \text{ の係数のうち } d_1 \text{個は } 1, d_2 \text{個は } -1, \text{それ以外は } 0\}$$

とする。このとき、正整数 d_f, d_g, d に対して、

$$\mathcal{L}_f = \mathcal{L}(d_f, d_f - 1),\ \mathcal{L}_g = \mathcal{L}(d_g, d_g),\ \mathcal{L}_\psi = \mathcal{L}(d_\psi, d_\psi)$$

とおく。

秘密鍵は $f \in \mathcal{L}_f$ 及び $g \in \mathcal{L}_g$ であり、ここで $f \bmod p$ 及び $f \bmod q$ はそれぞれ逆元をもつようにとる。このとき、公開鍵は、$h = g/f \bmod q$ である。

- 暗号化： 平文 $m \in \mathcal{L}_m$ に対して、その暗号化は、

$$e = p\psi * h + m \bmod q$$

で与えられる。ただし、$\psi \in \mathcal{L}_\psi$ はランダムに選ばれる。

- 復号： 暗号文 e に対して、$a = f * e \bmod q$ を計算する。ただし、a の係数は $-\frac{q}{2}$ 以上 $\frac{q}{2}$ 以下の範囲でとるものとする。次に、$a/f \bmod p$ を計算する。ただし、係数は $-\frac{p}{2}$ 以上 $\frac{p}{2}$ 以下の範囲でとるものとする。これにより、平文 m が復号される。

3.2.2 研究動向

1997 年に Ajtai-Dwork 暗号 [3-2] は提案され、その安全性は u-SVP (unique shortest vector problem) を基にしている。Ajtai-Dwork 暗号では ‘0’ の暗号文はいつでも ‘0’ に正しく復号されるが、‘1’ の暗号文は誤って確率 $1/n$ で ‘0’ に復号されてしまう。そこで、‘1’ の暗号文に対してもいつでも正しく ‘1’ に復号されるような工夫が Goldreich,Goldwasser,Halevi[3-10] により提案されている。

Ajtai-Dwork 暗号方式が特に注目されたのは、その安全性の証明が u-SVP の最悪の場合 (worst-case) を基にしていた点である。つまり、もしランダムなインスタンスに対し

て、この暗号方式において '0' と '1' の暗号文が多項式時間で区別できるとすれば、多項式時間で最悪の場合の u-SVP が解けてしまうことが [3-2] で示された。また逆に、$n^{0.5-\epsilon}$ 以内で SVP を近似できるか、もしくは $n^{1.33}$ 以内で CVP を近似できるオラクルがあれば、Ajtai-Dwork 暗号における解読が多項式時間で可能なことが Nguyen, Stern [3-20] により示されている。更に、[3-20] では実際にヒューリスティックな立場から秘密鍵の解読を行っており、少なくとも $n = 32$ ぐらいまではほぼ攻撃が成功するという結果を報告している。

1997 年に提案された Goldreich-Goldwasser-Halevi 暗号 は McElice 暗号のラティス版とみなすことができ、この公開鍵暗号の安全性は、CVP（closest vector problem）, SBP (smallest basis problem) を基にしている。また、この暗号方式に対して、Micciancio [3-18] によりいくつかの工夫が提案されている。しかしながら、現在、現実的な鍵サイズを用いた Goldreich-Goldwasser-Halevi 暗号は、少なくとも初めの形では、Nguyen [3-19] により破られたと考えられる。実際、Goldreich-Goldwasser-Halevi 暗号における鍵サイズは $O(n^2 \log n)$ であり、特に $n = 200, 250, 300, 350, 400$ のチャレンジ問題が Goldreich, Goldwasser, Halevi によって公開されたが、[3-19] により $n = 400$ 以外はすべて解かれている。

NTRU 暗号は, Hoffstein, Pipher, Silverman [3-12] によって提案された公開鍵暗号であり、その安全性は SVP, CVP を基にしている。現在のところ NTRU 暗号に対する効果的な攻撃法としては、ラティス基底縮小に基づく方法が挙げられる。この種の攻撃法は、Coppersmith, Shamir [3-8] により初めて提案されたが、[3-12] によると N を適切に選ぶことで、この種の攻撃法はそれほど効果的ではないと述べられている。また、NTRU 暗号における鍵サイズは $O(N \log q)$ であり、これはナップサック問題及び、他のラティス問題に基づいた方式よりも短い鍵サイズとなっている。

また安全性に関して、[3-12] における提案どおりの NTRU 暗号の形では、十分な安全性（選択暗号文攻撃に対する強秘匿性（semantic security）、頑強性 (non-malleability)）が満たされないことが報告されている [3-13]。そこで、[3-11] では、NTRU 暗号に対して適切なパディング (padding) を行うことで、選択暗号文攻撃に対する耐性を持たせることが提案されている。

3.2.3 課題とまとめ

Ajtai-Dwork 暗号方式は、その安全性が u-SVP の最悪の場合の困難性を仮定して証明された暗号方式である。この方式は、ランダムなインスタンスを解読することは、その安全性の基となる（困難とされる）問題の最悪の場合を解くのと同じくらい難しいことが示された、初めての公開鍵暗号方式である。そのため、理論的には非常に意味深いといえる。しかしながら、現実的なパラメータ、例えば $n = 32$ ぐらいまでは [3-20] によると、実際にヒューリスティックな立場から秘密鍵を解読する攻撃がほぼ成功するという結果が報告されている。したがって、大きな改良点が見つからない限りは、この暗号は実用的とはいえないであろう。

また、現在、現実的なパラメータ設定においては、[3-19] により、少なくとも初めの形

の Goldreich-Goldwasser-Halevi 暗号は破られたと考えられている。

NTRU 暗号における鍵サイズは $O(N \log q)$ であり、これはナップサック問題及び、他のラティス問題に基づいた方式よりも短い鍵サイズとなっている。このため、現実的なパラメータ設定として比較的大きな N を選ぶことが可能となる。このとき、例えば [3-8] によるラティス基底縮小を基にした方法では、あまりに高次元のラティス問題に対応してしまうため、ほとんど効果的ではなくなってしまう。今のところ、（現実的なパラメータ設定の範囲内で）大きな N を選んだとき、NTRU 暗号が破れたということは報告されていないが、今後は更なる安全性の研究調査が必要であろう。

3.3 線形符号の復号問題に基づく方式

線形符号の復号問題とは、任意に与えられた線形符号 $\mathcal{C}$ および受信語 y から、$y = c \oplus e$ を満たす最小の重みを持つベクトル e および符号語 c を求める問題である。この問題は符号 $\mathcal{C}$ の代数的な構造が見えなければ、e の重みが大きくなるにつれ指数関数的に難しくなる。

線形符号の復号問題を応用した暗号系としては McEliece 公開鍵暗号と Niederreiter 公開鍵暗号があるが、両者共現実的な攻撃方法が知られているため実用には向かない。しかしながら、最近これらの攻撃方法を全て回避し、なおかつ強秘匿性（平文に関するわずかな情報すら得ることが難しいこと）を満たす研究が進んでいる。

本章では McEliece 公開鍵暗号および Niederreiter 公開鍵暗号の安全性改良に関する研究動向をまとめる。

3.3.1 方式の説明

3.3.1.1 McEliece 公開鍵暗号

鍵生成: 以下の三つの行列 G、S、P を生成する。

G: 2 元 (n, k) 線形符号の $k \times n$ 生成行列

S: $k \times k$ のランダムな正則行列

P: $n \times n$ のランダムな置換行列

G、S、P を掛け合わせて G' 生成する。

$$G' = SGP \tag{3.2}$$

そして、G で生成される符号語の最小距離を d_{min} として $\left\lceil \frac{d_{min}-1}{2} \right\rceil$ を満たす正数を t とする。

秘密鍵: (S, G, P)

公開鍵: (G', t)

暗号化: 平文 m を 2 元 k 次元ベクトルとする。暗号文 c は以下により計算される。

$$c = mG' \oplus z \tag{3.3}$$

ここで、z は重みが t である（1 を t 個含む）2 元 n 次元ベクトルである。

復号: 平文 m は以下により求まる。まず、P の逆行列 P^{-1} を暗号文に掛ける。

$$cP^{-1} = (mS)G + zP^{-1} \tag{3.4}$$

この値 CP^{-1} は G で生成される符号語に t ビットの誤りが乗っている状態と等しいため、G に対応する誤り訂正アルゴリズム EC を cP^{-1} に適用することで mS が得られる。

$$mS = EC(cP^{-1}) \tag{3.5}$$

この mS に対して S の逆行列 S^{-1} を掛けることで平文 m は求まる。

$$m = (mS)S^{-1} \tag{3.6}$$

3.3.1.2 Niederreiter 公開鍵暗号

鍵生成: 以下の三つの行列 H、S、P を生成する。

H: 2 元 (n, k) 線形符号の $n \times (n-k)$ パリティ検査行列

S: $(n-k) \times (n-k)$ のランダムな正則行列

P: $n \times n$ のランダムな置換行列

H、S、P を掛け合わせて H' 生成する。

$$H' = PHS \tag{3.7}$$

そして、H で訂正できる誤りのシンボル数を t とする。

秘密鍵: (S, H, P)

公開鍵: (H', t)

暗号化: 平文 m を重み t の n 次元ベクトルとする。暗号文 c は以下により計算される。

$$c = mH' \tag{3.8}$$

復号: 平文 m は以下により求まる。まず、S の逆行列 S^{-1} を暗号文に掛ける。

$$cS^{-1} = (mP)H \tag{3.9}$$

この値 cS^{-1} は H を用いて得られるシンドロームと等しいため、H に対応する誤り訂正アルゴリズム EC を cS^{-1} に適用することで mP が得られる。

$$mP = EC(cS^{-1}) \tag{3.10}$$

この mP に対して P の逆行列 P^{-1} を掛けることで平文 m は求まる。

$$m = (mP)P^{-1} \tag{3.11}$$

3.3.1.2.1 重み t のランダム誤りベクトル生成方法 McEliece 暗号の暗号化では重み t の誤りベクトルを生成する必要があり、Niederreiter 暗号ではメッセージを重み t の誤りベクトルに変換する必要がある。これらの処理は、以下で述べる処理（これを変換関数 $Conv$ と定義する）により行える。

まず、重み t の 2 元 n 次元誤りベクトル z の総数は $N = C(n,t)$ 通りあり、その内、一番左の値が 1 であるものの数は $C(n-1,t-1)$ 個あり、一番左の値が 0 であるものの数は $C(n-1,t)$ 個ある。よって、

$$C(n,t) = C(n-1,t-1) + C(n-1,t) \tag{3.12}$$

が成立する。つまり、0 以上 N 未満の正数 $\bar{z}$ に対して、$\bar{z} < C(n-1,t)$ が成立すれば z の一番左のビットを 0 に設定し、$\bar{z} \geq C(n-1,t)$ が成立すれば 1 に設定する。そして、1 を設定した場合は $\bar{z} := \bar{z} - C(n-1,t)$ とする。同様の処理を左から二番目のビットから順に適用していくことで z の全てのビットが求まる。$Conv$ の逆関数 $Conv^{-1}$ は上記の処理を逆にたどることで行える。

3.3.2 研究動向

3.3.2.1 安全性と攻撃手法に関する研究動向

3.3.2.1.1 McEliece 暗号と Niederreiter 暗号との安全性の等価性 パラメータサイズを適当に選択すれば McEliece 暗号と Niederreiter 暗号との安全性は等価となることが知られている [3-39]。理由は以下のとおりである。McEliece 暗号の公開行列および暗号文をそれぞれ G'、c_M とし、Niederreiter 暗号の公開行列および暗号文をそれぞれ H'、c_N とする。公開行列 G' および H' のデュアル[1]を求めることは簡単であり、それらをそれぞれ H''、G'' とする。McEliece 暗号の暗号文 c_M は $c_N = c_M H''$ により Niederreiter 暗号の公開鍵 H'' に対応した暗号文 c_N に変換できる。Niederreiter 暗号において平文が分かるということは誤りベクトル z が分かるということであり、$z \oplus c_M$ により McEliece 暗号の暗号文 c_N の誤りベクトルを削除することができる。一旦、誤りベクトルを削除することができれば、以下で説明する情報組復号により McEliece 暗号の平文を求めることができる。

同様に Niederreiter 暗号の暗号文 c_N は $c_M G'' = c_N$ を満たす c_M を探すことにより McEliece 暗号の公開鍵 G'' に対応した暗号文 c_M に変換できる。McEliece 暗号において平文が分かるということはその際の誤りベクトルが分かるということであり、それは Niederreiter 暗号の暗号文 c_N の平文が求まったことと等しい。

以下では McEliece 暗号に対する攻撃方法について述べるが、上記で述べたように McEliece 暗号に適用可能な攻撃アルゴリズムは同様に Niederreiter 暗号に対しても適用可能である。

3.3.2.1.2 公開鍵への攻撃に関する研究動向 弱公開鍵（G を生成する際に利用したゴッパ多項式の全係数が部分体に含まれ、なおかつそれらの全探索が可能であるような公開

[1] $k \times n$ 行列 G' のデュアル H'' とは $G' \times H'' = 0$ となる $n \times (n-k)$ 行列のことをいう。

鍵）に対する攻撃として以下のような方法が知られている [3-40]。ゴッパ多項式を一つ選択してそれに対応する公開鍵 G''' を生成する。G''' と公開鍵 G' との同等性を SSA(Support Splitting Algorithm)[3-46] 等を用いることにより判定する。同等でなければ、別のゴッパ多項式を選択して同様の処理を繰り返す。同等であればそのゴッパ多項式を用いて復号を行うことができる。この攻撃方法は結局ゴッパ多項式の全探索を行う必要があるため、ゴッパ多項式の候補数が十分大きければ脅威とならない。

3.3.2.1.3 暗号文への攻撃に関する研究動向 上記の攻撃以外に公開鍵 G' から秘密鍵 (S, G, P) を求める効率的な攻撃方法は現在のところ知られていない [3-42]。以下で紹介する攻撃方法は、全て暗号文から直接平文を求める攻撃である。単にパラメータサイズを大きくすることにより防げる攻撃を「致命的でない攻撃」、防ぎきれない攻撃を「致命的な攻撃」と分類することにする。

致命的でない攻撃方法

一般化情報組復号攻撃 まず、ベースとなる情報組復号攻撃について述べる。公開鍵 G'、暗号文 c および誤りベクトル z から互いに対応する k 列を取り出したものをぞれ G'_k、c_k、z_k とする。これらには

$$c_k = mG'_k \oplus z_k \tag{3.13}$$

という関係があるため、G'_k が正則でありなおかつ $z_k = 0$ である場合には

$$m = c_k G'^{-1}_k \tag{3.14}$$

により平文が求まる [3-28]。$z_k = 0$ となる k 列がたまたま取り出せる確率は n 個から k 個取り出す組み合わせの数を $C(n, k)$ として、

$$\frac{C(n-t, k)}{C(n, k)} \tag{3.15}$$

で与えられる。

$z_k = 0$ であるか否かは、以下の関数を評価することで判定できる。

$$Eval_0(c_k, G'_k) = c_k G'^{-1}_k G' \oplus c \tag{3.16}$$

$Eval_0$ の重みが t となった場合の $c_k G'^{-1}_k$ が解読したい暗号文 c の平文となる。

McEliece が当時示した推奨パラメータ $n = 1024$、$t = 50$、$k \geq 524$ を $k = 524$ として (3.15) に代入してみるとおよそ $2^{-53.6}$ となる。ちなみに、(3.15) を最も小さくするパラメータの組み合わせは $n = 1024$ の場合、$n = 1024$、$t = 38$、$k \geq 644$ であり [3-45]、$k = 524$ とするとおよそ $2^{-56.1}$ となる。$n = 2048$ の場合、$t = 69$、$k = 1289$ においておよそ $2^{-101.7}$ となり、$n = 4096$ では $t = 128$、$k = 2560$ においておよそ $2^{-186.1}$ となる。

情報組復号攻撃は n 個の座標から z_k の重みが 0 となる k 個の座標を取り出す攻撃方法であるが、これを一般化して取り出した z_k の重み t_k が v_1 以上 v_2 以下の範囲にあるとみなし、この範囲にある $\sum_{t_k=v_1}^{v_2} C(k,t_k)$ とおりの z_k の候補から正しい z_k を探すことも可能である [3-38]。この攻撃方法を一般化情報組復号攻撃とよぶことにする。

推測した z_k の値が正しいかどうかは、以下の評価関数の出力の重みが t となっているか否かにより判定できる。

$$Eval_1(z_k, c_k, G'_k) = (c_k \oplus z_k)G'^{-1}_k G' \oplus c \tag{3.17}$$

z_k が分かれば以下により平文 m を求めることができる。

$$m \quad = \quad (c_k \oplus z_k)G'^{-1}_k \tag{3.18}$$

取り出した z_k の重み t_k がたまたま v_1 以上 v_2 以下の範囲にある確率 $Pr(v_1 \le t_k \le v_2)$ は

$$Pr(v_1 \le t_k \le v_2) = \sum_{t_k=v_1}^{v_2} \frac{C(n-t, k-t_k)C(t,t_k)}{C(n,k)} \tag{3.19}$$

で与えられるため、平均して $1/Pr(v_1 \le t_k \le v_2)$ とおりの k 列を試すことで、z_k の重み t_k は v_1 以上 v_2 以下の範囲に入る。この範囲の重みを持つ z_k の数は全部で $\sum_{t_k=v_1}^{v_2} C(k,t_k)$ 個あるため、合計 $1/Pr(v_1 \le t_k \le v_2)\sum_{t_k=v_1}^{v_2} C(k,t_k)$ 回 (3.17) を評価することで正しい z_k を見つけることができる。その際の値を (3.18) に代入して得られた値が平文 m である。

t/n が小さな場合、従来の逆行列攻撃と比べ評価関数を評価しなければならない回数自体は大きくなる場合もある。しかしながら、評価関数の計算で最も時間の掛る処理は逆行列の演算である。一般化方式では一度計算した逆行列の値を $\sum_{t_k=v_1}^{v_2} C(k,t_k)$ 回利用できるため、全体の計算量は小さくなる。

低重み符号語探索攻撃 この攻撃は任意の線形符号の生成行列から、その生成行列を用いて生成される符号語の内、低重みの符号語を探すアルゴリズムを利用する [3-48, 31] 。以下の $(k+1) \times n$ 生成行列から生成される最小重みの符号語は誤りベクトルと等しいためこのアルゴリズムが利用できる。

$$\begin{bmatrix} G' \\ c \end{bmatrix} \tag{3.20}$$

なお、ここで $c = mG' \oplus z$ であり、z は誤りベクトルを意味する。

しかしながら、この攻撃方法における繰り返し回数の期待値の漸近的な下界は、各繰り返しが独立であると仮定すると $C(n,k+1)/C(n-t,k+1)$ で与えられるため、パラメータを適切に設定すればこの攻撃はかわすことができる。各繰り返しの従属性を考慮した場合の計算コストは [3-31] で評価されており、McEliece により [3-41] において示されたオリジナルのパラメータサイズ $(n,k,t) = (1024, 524, 50)$ は $2^{64.2}$ のワークファクターで解読できるが、適切なパラメータサイズ例えば $n \ge 2048$ においてはこの攻撃を適用することは困難であることが示されている。

致命的な攻撃方法

既知部分平文攻撃 平文の一部が既知である場合、以下の方法により残りの平文を求めることができる。攻撃者は平文 m の右側 k_r ビットの値 m_r と m の暗号文 c を知っていたとする。すると、m の左側 k_l ビットの値 m_l は以下により求まる。まず、公開行列 G' の 1 行目から k_l 行目までの $k_l \times n$ 行列を G'_l とし、残りの $k_r \times n$ 行列を G'_r とする。これらには、以下の関係が成立する。

$$c = m_l G'_l \oplus m_r G'_r \oplus z \tag{3.21}$$

G'_l、G'_r、C および Z から互いに対応する k_l 列を取り出したものをぞれぞれ G'_{l,k_l}、G'_{r,k_l}、c_{k_l}、z_{k_l} とする。これらには

$$c_{k_l} = m_l G'_{l,k_l} \oplus m_r G'_{r,k_l} \oplus z_{k_l} \tag{3.22}$$

という関係があるため、G'_{l,k_l} が正則でありなおかつ $z_{k_l} = 0$ である場合には

$$m_l = (c_{k_l} \oplus m_r G'_{r,k_l}) G'^{-1}_{l,k_l} \tag{3.23}$$

により m_l は求まる。$z_{k_l} = 0$ であるかどうかは、復号した値を m'_l として

$$(m'_l || m_r) \oplus G' \oplus c \tag{3.24}$$

により誤りベクトルを求めればよい。その重みが t となっていれば正しい平文が求まったことになる。

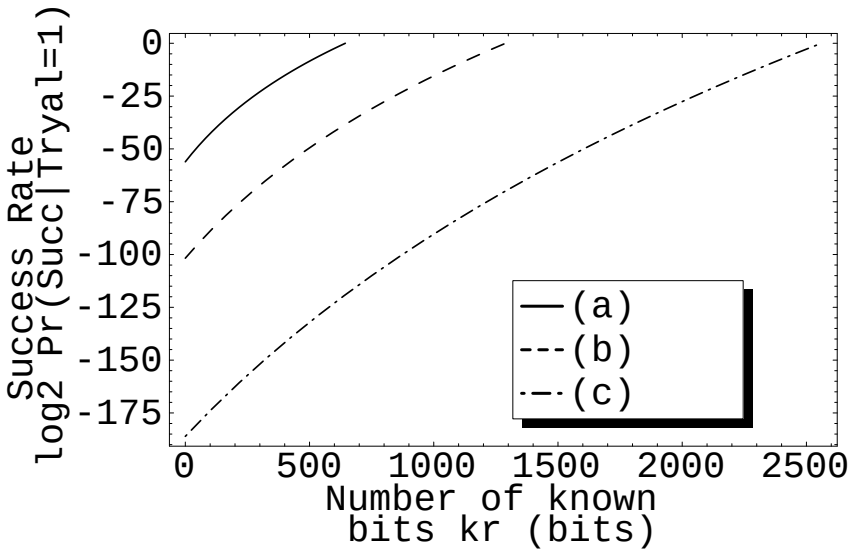
なお、重みが t となれば正しい平文が求まることは以下より明らかである。まず、正しい値 m_l 以外に m''_l という値がこの条件を満たすと仮定する。このことは、c からハミング距離で t 以下の範囲に 2 つの符号語が存在することを意味し、符号間の距離が $2t$ 以下であることを意味する。しかしながら、t の定義（$t = \left\lceil \frac{d_{min}-1}{2} \right\rceil$）より符号語間の最小距離 d_{min} は $2t + \{1 \text{ or } 2\}$ であるため t の定義と矛盾する。よって、(3.24) の重みを t とする m'_l は 2 つ以上存在しない。また、暗号文 c および m_r が正しい値であれば c に対応する m'_l は必ず 1 つ存在する。

結局、m_l を求めるには以下の評価関数 $Eval_2$ を構成して、この関数を k_l 列の取り方を変えながら評価してやればよい。$Eval_2$ の重みが t となった場合には (3.23) により m_l が求まる。

$$\begin{aligned} & Eval_2(c_{k_l}, G'_{r,k_l}, G'_{l,k_l}) \\ = \ & (c_{k_l} \oplus m_r G'_{r,k_l}) G'^{-1}_{l,k_l} G'_l \oplus m_r G'_r \oplus c \end{aligned} \tag{3.25}$$

ここで、$m_r G'_r \oplus c$ は一度計算するだけでよく、$m_r G'_{r,k_l}$ は $m_r G'_r$ から k_l 列取り出すことで高速化できる。

攻撃者がこの $Eval_2$ を一回評価して、z_{k_l} がたまたま 0 となる確率（m_l を求めることのできる確率）を $Pr(Succ|Tryal = 1)$ とすると $Pr(Succ|Tryal = 1) = C(n-t, k_l)/C(n, k_l)$



(a) $(n, k, t) = (1024, 644, 38)$
(b) $(n, k, t) = (2048, 1289, 69)$
(c) $(n, k, t) = (4096, 2560, 128)$

図 3.1: 既知ビット数 k_r と確率 $Pr(Succ|Tryal = 1)$ の関係

となる。図 3.1 に $Pr(Succ|Tryal = 1)$ と攻撃者の知っている平文のビット数 k_r との関係を示す。例えば、暗号文および平文のビットサイズがそれぞれ $n = 2048$, $k = 1289$ である McEliece 暗号において攻撃者が平文の $k_r = 789$ ビットを知っていたとする。残りの $k - k_r = 500$ ビットは、2^{500} 通りの平文に G' を掛けてそれと暗号文 c との排他的論理和をとることでも求まるが、拡張方式 1 を用いれば、2^{50} 回程度 (3.25) を評価することで残りの $k - k_r = 500$ ビットが求まることが図 3.1 よりわかる。つまり、平文のフォーマットなどにより平文の一部が攻撃者に分かっている場合、McEliece 暗号の完全解読に対する耐性は著しく減少することになる。

再送攻撃　送信者がたまたま同じ平文 m を二度暗号化した、あるいは攻撃者が送信者に同じ平文 m を二度暗号化させたとする。その際の暗号文および誤りベクトルをそれぞれ c_1、c_2、z_1、z_2 とすると、それらの間には以下の関係が成立する。

$$c_1 = mG' \oplus z_1 \tag{3.26}$$

$$c_2 = mG' \oplus z_2 \tag{3.27}$$

$$c_1 \oplus c_2 = z_1 \oplus z_2 \tag{3.28}$$

つまり、$c_1 \oplus c_2$ において 0 となっている部分に誤りが乗っている可能性は小さいことが分かるため、この部分から公開鍵 G' および暗号文 c の対応する k 列を取り出すことで一般化情報組復号攻撃が行いやすくなる [3-30]。

関連平文攻撃 この攻撃は平文再送攻撃を一般化したものである。攻撃者は平文間に成り立つ線形な関係を知っているとする。例えば、攻撃者は平文 m_1 と平文 m_2 の値は知らないが、それらの差分値 δm を知っているとする。

$$\delta m = m_1 \oplus m_2 \tag{3.29}$$

$z_1 \oplus z_2$ の値は以下により与えられるため、

$$\begin{aligned} c_1 &= m_1 G' \oplus z_1 && (3.30)\\ c_2 &= m_2 G' \oplus z_2 && (3.31)\\ c_1 \oplus c_2 &= \delta m G' \oplus z_1 \oplus z_2 && (3.32) \end{aligned}$$

平文再送攻撃と同様に一般化情報組復号攻撃を効率よく行うことができる [3-30]。

改ざん攻撃 この攻撃方法は、任意の平文 m に対する暗号文 $c = \mathcal{E}^{McEliece}(m, z)$ に対して、その平文 m を知らない攻撃者が $m' = m \oplus \delta m$ を満たす暗号文 $c' = \mathcal{E}^{McEliece}(m', z)$ を任意の δm に対して求めることのできる攻撃方法である [3-35, 49]。

具体的な攻撃方法は以下のとおりである。まず、δm を GF(2) 上の k 次元行ベクトルとし、GF(2) 上の $k \times n$ 行列 G' の i 行目のベクトルを $G'[i]$ とする。δm の列の値が 1 である列の位置 i の集合を $\{i_1, i_2, \cdots\}$ とすると、$c = mG' \oplus z$ に対して $m' = m \oplus \delta m$ を満たす暗号文 c' は以下で与えられる。

$$\begin{aligned} c' &= c \oplus G'[i_1] \oplus G'[i_2] \oplus \cdots \\ &= (m \oplus \delta m)G' \oplus z \\ &= m'G' \oplus z && (3.33) \end{aligned}$$

つまり、McEliece 暗号は受動的攻撃（公開鍵および暗号文 c のみが与えられた攻撃者が自分だけの力で暗号文の解読を試みる攻撃）に対して頑健性 (non-malleability)[3-32] （任意の平文 m に対する暗号文 $c = \mathcal{E}(m)$ に対してその平文 m を知らない攻撃者がいかなる関数 F に対しても $m' = F(m)$ を満たす暗号文 $c' = \mathcal{E}(m')$ を求めることが困難であること）を満たさない。また、このことは同時に McEliece 暗号が選択暗号文攻撃（解読を試みようとしている暗号文 c 以外の任意の暗号文に対応する平文を攻撃者が入手できることを想定した攻撃）に対して完全解読（平文全体を求めること）が可能であることを意味する。実際暗号文 c に対して平文反転攻撃により c' を求め、選択暗号文攻撃により c' に対する平文 m' を入手することで、c の平文 $m = m' \oplus \delta m$ を求めることができる。

リアクション攻撃 この攻撃は、（適応的）選択暗号文攻撃に分類することができる。しかしながら、典型的な選択暗号文攻撃と異なり、攻撃者は選択暗号文に対応する平文を得る必要は無い。攻撃者は、解読を試みる暗号文 c を 1 ビット反転[2]し、その暗号文を正当

[2]暗号文を数ビット反転させる類似の攻撃方法が [3-49] において提案されている。この攻撃方法は暗号文を復号オラクルに復号させる必要がある。

m に対する暗号化処理:			c に対する復号処理:		
r	:=	$Rand$	z, y_1	:=	$\mathcal{D}^{McEliece}(Msb_n(c))$
$\bar{m}$	:=	$Prep(m)$	$\bar{z}$	:=	$Conv^{-1}(z)$
$\bar{z}$	:=	$Hash_z(r\|\|\bar{m})$	$(r\|\|\bar{m})$	:=	$Gen(\bar{z}) \oplus (y_1\|\|y_2)$
$(y_1\|\|y_2)$	:=	$Gen(\bar{z}) \oplus (r\|\|\bar{m})$	If		$\bar{z}$=$Hash_z(r\|\|\bar{m})$
z	:=	$Conv(\bar{z})$			return $Prep^{-1}(\bar{m})$
c	:=	$\mathcal{E}^{McEliece}(y_1, z)\|\|y_2$	Otherwise		reject c
return		c			

図 3.2: コンバージョン α

な受信者に復号させる。そして、その受信者の反応を観測することにより誤りベクトルに関する情報を得る [3-34]。

リアクション攻撃は、攻撃者の反転したビット位置に応じて受信者の反応が異なることを利用している。受信者の反応としては主に以下の二種類が挙げられる。

反応１： 意味のある平文が復号できなかったか、復号自体が行えなかったため、暗号文の再送を要求する。

反応２： 意味のある平文が復号できたため何の反応もしない。あるいは、正しく平文を受け取れたことの確認情報を返す。

もし、反転ビットが誤りビットに乗った場合、誤りの数は少なくなるため受信者は必ず後者の反応をすることになる。誤りビットに乗らなかった場合、受信者は必ず前者の反応をすることになる。

受信者が前者の反応を示した場合には、その反転ビットの位置には誤りが乗っていないことは確実であるため、この情報を利用することで一般化情報組復号攻撃を効率よく適用することができる。

3.3.2.2 安全性の改善に関する研究動向

前節でまとめたとおり、McEliece 暗号および Niederreiter 暗号に対しては致命的な攻撃方法が知られているため、それらをそのまま利用するのは危険である。しかしながら、僅かな改良を加えることでこれらの暗号の安全性を格段に向上させる方法に関する研究が進んでいる。

安全性が向上する理由は以下に起因する。

- 致命的な攻撃方法は、平文に関する知識、あるいは復号オラクルを利用するため、これらの利用を防止できれば致命的な攻撃方法を防ぐことができる。

m **に対する暗号化処理:**			c **に対する復号処理:**		
r	:=	$Rand$	y_4	:=	$Msb_{Len(c)-n}(c)$
$\bar{m}$	:=	$Prep(m)$	z, y_3	:=	$\mathcal{D}^{McEliece}(Lsb_n(c))$
y_1	:=	$Gen(r) \oplus \bar{m}$	$(y_2 \| \| y_1)$	:=	$(y_4 \| \| y_3)$
y_2	:=	$r \oplus Hash(y_1)$	r	:=	$y_2 \oplus Hash(y_1)$
$(y_4 \| \| y_3)$	:=	$(y_2 \| \| y_1)$	$\bar{m}$	:=	$Gen(r) \oplus y_1$
z	:=	$Conv(Hash_z(r))$	If		$Conv^{-1}(z)$=$Hash_z(r)$
c	:=	$y_4 \| \| \mathcal{E}^{McEliece}(y_3, z)$			return $Prep^{-1}(\bar{m})$
return		c	Otherwise		reject c

図 3.3: コンバージョン β

- その他の攻撃方法は全て致命的な攻撃方法に分類されるため、これらはパラメータを適切に設定することにより防ぐことができる。

さらに、改良方式はランダムオラクルモデル[3]のもとで適応的選択暗号文攻撃に対して識別困難性[4]を満たすことが証明できる [3-37]。

[3-37] において提案されている方式を図 3.2 ~ 3.4 に示す。なお、各記号の意味は以下のとおりである。また、図 3.2 において、$Len(y_1) = k$ であり、$Len(y_2) = Len(r||\bar{m}) - k$ である。y_2 は $Len(r||\bar{m}) = k$ となる場合には削除してよい。図 3.3 において、$Len(y_3) = k$ であり、$Len(y_4) = Len(r||\bar{m}) - k$ である。y_4 は $Len(r||\bar{m}) = k$ となる場合には削除してよい。図 3.4 において、$Len(y_3) = k$ であり、$Len(y_4) = \lfloor \log_2 C(n,t) \rfloor$、$Len(y_5) = Len(\bar{m})+Len(Const)+Len(r)-Len(y_4)-k$ である。y_5 は $Len(\bar{m})+Len(Const)+Len(r) = Len(y_4) + k$ となる場合には削除してよい。

[3] 利用されるハッシュ関数や引き伸ばし関数が理想的な乱数性を有するとの仮定のもとで行われる証明モデル。

[4] 攻撃者が任意に選択した 2 種類の平文の一方を第三者に暗号化してもらい、攻撃者はその暗号文とその二つの平文が与えられ、どちらの平文がその暗号文に対応するかをあてる問題を考えた場合、その攻撃者が有意な確率でその問題を解くことが難しいこと。

m に対する暗号化処理:	c に対する復号処理:
$r := Rand$	$y_5 := Msb_{Len(c)-n}(c)$
$\bar{m} := Prep(m)$	$z, y_3 := \mathcal{D}^{McEliece}(Lsb_n(c))$
$y_1 := Gen(r) \oplus (\bar{m}\|\|Const)$	$\bar{z} := Conv^{-1}(z)$
$y_2 := r \oplus Hash(y_1)$	$y_4 := Lsb_{\lfloor \log_2 C(n,t) \rfloor}(\bar{z})$
$(y_5\|\|y_4\|\|y_3) := (y_2\|\|y_1)$	$(y_2\|\|y_1) := (y_5\|\|y_4\|\|y_3)$
$z := Conv(y_4)$	$r := y_2 \oplus Hash(y_1)$
$c := y_5\|\|\mathcal{E}^{McEliece}(y_3, z)$	$(\bar{m}\|\|Const') := y_1 \oplus Gen(r)$
return c	If $Const'$=$Const$ return $Prep^{-1}(\bar{m})$
	Otherwise reject c

図 3.4: コンバージョン γ

表記

$C(n,t)$: n 個の要素から t 個を取り出す組合せの数。
$Prep(m)$: 平文 m への可逆な前処理。コード変換、データ圧縮、パディング等を含む。逆変換を $Prep^{-1}()$ で表す。
$Hash(x)$: 任意の長さの系列 x から固定長の系列へ変換する一方向性ハッシュ関数。なお、$N = C(n,t)$ として値域が Z_N である場合は $Hash_z(x)$ と表現する。
$Conv(\bar{z})$: 整数 $\bar{z} \in Z_N$ を誤りベクトル z に変換する関数。逆関数を $Conv^{-1}()$ で表す。
$Gen(x)$: 初期値 x を真性乱数と計算量的に区別することが困難であるような擬似乱数系列に引き伸ばす関数。
$Len(x)$: 系列 x のビット長。
$Msb_{x_1}(x_2)$: 系列 x_2 の左 x_1 ビット。
$Lsb_{x_1}(x_2)$: 系列 x_2 の右 x_1 ビット。
$Const$: 予め決められ、公開されている定数。
$Rand$: 真性（あるいは計算量的に安全な擬似）乱数を生成する乱数源。

m に対する暗号化処理:

$$
\begin{aligned}
r, r' &:= Rand \\
\bar{m} &:= Prep(m) \\
z &:= Conv(Hash_z(\bar{m}||r)) \\
y_1 &:= \mathcal{E}^{McEliece}(r', z) \\
y_2 &:= Gen(r') \oplus (\bar{m}||r) \\
c &:= y_1||y_2 \\
\text{return} &\quad c
\end{aligned}
$$

図 3.5: Pointcheval の包括的コンバージョン + McEliece 暗号

m に対する暗号化処理:

$$
\begin{aligned}
r &:= Rand \\
\bar{m} &:= Prep(m) \\
z &:= Conv(Hash_z(\bar{m}||r)) \\
y_1 &:= \mathcal{E}^{McEliece}(r, z) \\
y_2 &:= Gen(r) \oplus \bar{m} \\
c &:= y_1||y_2 \\
\text{return} &\quad c
\end{aligned}
$$

図 3.6: 藤崎-岡本の包括的コンバージョン + McEliece 暗号

m に対する暗号化処理:

$$
\begin{aligned}
r, \bar{z} &:= Rand \\
\bar{m} &:= Prep(m) \\
z &:= Conv(\bar{z}) \\
c_1 &:= \mathcal{E}^{McEliece}(r, z) \\
c_2 &:= Gen(r) \oplus \bar{m} \\
c_3 &:= Hash(r, c_1, \bar{m}, c_2) \\
c &:= (c_1, c_2, c_3) \\
\text{return} &\quad c
\end{aligned}
$$

図 3.7: REACT + McEliece 暗号

m に対する暗号化処理:

$$
\begin{aligned}
r &:= Rand \\
\bar{m} &:= Prep(m) \\
y_1 &:= (\bar{m}||Const) \oplus Gen(r) \\
y_2 &:= r \oplus Hash(y_1) \\
z &:= Conv(y_2||y_1) \\
c &:= \mathcal{E}^{Niederreiter}(z) \\
\text{return} &\quad c
\end{aligned}
$$

図 3.8: OAEP コンバージョン + Niederreiter 暗号

$\mathcal{E}^{McEliece}(x, z)$: 誤りベクトルを z とする McEliece 暗号による x の暗号化。
$\mathcal{D}^{McEliece}(x)$: McEliece 暗号による x の復号。
$\mathcal{E}^{Niederreiter}(x)$: Niederreiter 暗号による x の暗号化。
$\mathcal{D}^{Niederreiter}(x)$: Niederreiter 暗号による x の復号。

また、McEliece 暗号に対して包括的なコンバージョンを適用した場合を図 3.5〜 3.8 に示し、それらとの比較を表 3.1 に示す。表からも分かるように McEliece 暗号に対しては包括的なコンバージョンを適用するより [3-37] において提案されている方式を適用したほうが、冗長なデータ量が少なくなることが示されている。なお、Pointcheval の方式は部分落し戸一方向性関数[5]に適用可能な方式であり、McEliece 暗号も部分一方向性関数の範

[5]入力として 2 変数 x、y を持ち $v = f(x, y)$ となる v を出力する関数であり、v から x および y を求めることが難しいが、ある秘密情報を用いれば x を求めることが可能となる関数。ElGamal 暗号などが含ま

表 3.1: McEliece暗号におけるコンバージョンと冗長データ量の比較

Conversion Scheme	Conversion Type	Complexity*2	$\geq 2^{56.3}$	$\geq 2^{101.9}$	$\geq 2^{186.2}$
		Data Redundancy*1 = Ciphertext Size - Plaintext Size			
		(n,k)	(1024, 644)	(2048, 1289)	(4096, 2560)
		t	38	69	128
Pointcheval's [3-44]	Generic to PTOWF	$n + Len(r)$	1184	2208	4256
Fujisaki -Okamoto's [3-33]	Generic to OWE	n	1024	2048	4096
REACT [3-43]	Generic to OWE*3	$n + Len(Hash())$	1184	2208	4256
Kobara-Imai's α [3-37]	Specific	$n - k + Len(r)$	540	919	1696
Kobara-Imai's β [3-37]	Generic to PTOWF	$n - k + Len(r)$	540	919	1696
Kobara-Imai's γ [3-37]	Specific	$n - k + Len(r)$ $+Len(Const)$ $-\lfloor \log_2 C(n,t) \rfloor$	470	648	1040
Original McEliece	None	n-k	380	759	1536

*1: 具体的な数値はそれぞれ以下の設定において求めた。$Len(r) = 160$、$Len(Const) = 160$、$Len(Hash()) = 160$。
*2: 低重み符号語探索攻撃を用いて暗号文から平文を求める際に要求される繰り返し回数の期待値の漸近的な下界。厳密な計算量は [3-31] において見積もられている。
*3: 平文検査攻撃（暗号文と平文の組を与えるとその暗号文がその平文に対応するものであるか否かを判定してくれるオラクルを利用できるという状況における攻撃）に対して一方向性を満たす必要がある。

疇に含まれるため適応できる。Crypto'99で提案されている藤崎-岡本の方式は選択平文攻撃に対して一方向性が満たされる任意の公開鍵暗号に適用できる。McEliece暗号も選択平文攻撃に対して一方向性が満たされるため適用できる。REACTは平文検査攻撃（暗号文と平文の組を与えるとその暗号文がその平文に対応するものであるか否かを判定してくれるオラクルを利用できるという状況における攻撃）に対して一方向性を満たす公開鍵暗号に適用できる。McEliece暗号においては、平文検査攻撃に対する一方向性と選択平文攻撃に対する一方向性との間に等価性が成り立つため、平文検査攻撃に対しても一方向性が成り立ち、その方式を適用することができる。

また、Niederreiter暗号に対するコンバージョンとそれらの比較を表3.2に示す。OAEPは、OAEPの逆像を求める際に必要な部分（図3.8でいうy_2。これを重要部分とよぶことにする）に対して一方向性が満たされるような落とし戸付き置換関数に対して適用できる（このような置換関数をCPOWTP(Critical Part One-Way Trapdoor Permutation)とよぶことにする）。Niederreiter暗号を図3.8のように利用した場合、y_2に対して一方向性が成り立つため、OAEPはNiederreiter暗号に対しても適用できる。OAEP+およびOAEP++は一方向性落とし戸置換に対して適用でき、Niederreiter暗号は一方向性落とし

れる。

表 3.2: Niederreiter 暗号におけるコンバージョンと冗長データ量の比較

Conversion Scheme	Conversion Type	Complexity*2	$\geq 2^{56.3}$	$\geq 2^{101.9}$	$\geq 2^{186.2}$
		Data Redundancy*1 = Ciphertext Size - Plaintext Size			
		(n,k)	(1024, 644)	(2048, 1289)	(4096, 2560)
		t	38	69	128
Fujisaki -Okamoto's [3-33]	Generic to OWE	$n-k+Len(Hash())$	540	919	1696
REACT [3-43]	Generic to OWE*3	$n-k+Len(Hash())$	540	919	1696
OAEP [3-29]	Generic to CPOWTP	$n-k-\lfloor \log_2 C(n,t) \rfloor$ $+Len(r)+Len(Const)$	- (0)*4	648 (111)*4	1040 (496)*4
OAEP+ [3-47]	Generic to OWTP	$n-k-\lfloor \log_2 C(n,t) \rfloor$ $+Len(r)+Len(Const)$	- (0)*4	648 (111)*4	1040 (496)*4
OAEP++ [3-36]	Generic to OWTP	$n-k-\lfloor \log_2 C(n,t) \rfloor$ $+Len(r)+Len(Const)$	470	648	1040
Original Niederreiter	None	$n-k$ $-\lfloor \log_2 C(n,t) \rfloor$	150	328	720

*1: 具体的な数値はそれぞれ以下の設定において求めた。$Len(r) = 160$、$Len(Const) = 160$、$Len(Hash()) = 160$。
*2: 低重み符号語探索攻撃を用いて暗号文から平文を求める際に要求される繰り返し回数の期待値の漸近的な下界。厳密な計算量は [3-31] において見積もられている。
*3: 平文検査攻撃（暗号文と平文の組を与えるとその暗号文がその平文に対応するものであるか否かを判定してくれるオラクルを利用できるという状況における攻撃）に対して一方向性を満たす必要がある。
*4: 暗号化できるビットサイズの上限。他の方式は任意の長さを暗号化可能。

戸置換であるため、OAEP+および OAEP++も Niederreiter 暗号に対して適用できる。

なお、OAEP+は図 3.8 の $y_1 := (\bar{m}||Const) \oplus Gen(r)$ の代わりに $y_1 := (\bar{m} \oplus Gen(r))||Hash(\bar{m}||r)$ とする方式であり、OAEP++は図 3.8 において $Len(y_3) = k$、$Len(Gen(r)) \geq k$ となるように $(y_4||y_3) := (y_2||y_1)$ を付け加え、$z := Conv(y_2||y_1)$ の代わりに $z := Conv(y_3)$ とする方式である。

3.3.3 課題とまとめ

線形符号の復号問題に基づく暗号系として McEliece 公開鍵暗号と Niederreiter 公開鍵暗号の研究動向についてまとめた。これらの公開鍵方式は、現実的な攻撃方法が知られているが、これらの攻撃方法を全て回避し、なおかつ強秘匿性を満たす方法が提案されている。そのため、安全性の面から言えばこれら改良版の使用は問題無いと思われる。また、これらの方式は暗号化および復号処理が極めて単純であるため、処理の高速化が期待できる。問題点としては、公開鍵のサイズが大きいことが挙げられる。例えば、$(n,k,t) = (2048, 1289, 69)$ の場合、Niederriter 暗号の公開鍵サイズは 122KB 程度となり、1536 ビット RSA と比べると 630 倍くらい大きくなってしまう。ただし、記憶領域は年々高密度化と低価格化の方向に進みつつあるため、サイズに関しては将来あまり問題とならないかもしれない。

線形符号の復号問題に基づき、安全でかつ効率のよい電子署名方式は現在のところ知られていない。線形符号の復号問題に基づく署名方式を提案することが今後の課題として挙げられる。

参考文献

[3-1] L. M. Adleman, “On breaking generalized knapsack public key cryptosystems”, In Proc. of 15th STOC, pp. 402-412, ACM, 1983.

[3-2] M. Ajtai and C. Dwork, “A public-key cryptosystem with worst-case/average-case equivalence”, Proc. of 29th STOC, ACM, 284-293, 1997.

[3-3] L. Babai, “On Lovasz lattice reduction and nearest lattice point problem”, in Combinatorica, vol. 6, pp. 1-13, 1986.

[3-4] E. F. Brickell, “Solving low density knapsacks”, In Proc. of Crypto’83, Plenum Press, 1984.

[3-5] E. F. Brickell, “Breaking iterated knapsacks”, In Proc. of Crypto’84, Lecture Notes in Computer Science 196, Springer-Verlag, 1985.

[3-6] M. J. Coster, A. Joux, B. A. LaMacchia, A. M. Odlyzko, C. P. Schnorr and J. Stern, “Improved low-density subset sum algorithms”, Comput. Complexity, 2, pp. 111-128, 1992.

[3-7] B. Chor and R. L. Rivest, “A knapsack-type public key cryptosystem based on arithmetic in finite fields”, IEEE Trans. Information Theory, 34, 1988.

[3-8] D. Coppersmith and A. Shamir, “Lattice attacks on NTRU”, In Proc. of Eurocrypt ’97, Lecture Notes in Computer Science, Springer-Verlag, 1997.

[3-9] O. Goldreich, S. Goldwasser and S. Halevi, “Public-key cryptosystems from lattice reduction problems”, In Proc. of Crypto ’97, Lecture Note in Computer Science, Springer-Verlag, 1997.

[3-10] O. Goldreich, S. Goldwasser and S. Halevi, “Eliminating decryption errors in the Ajtai-Dwork cryptosystem”, Electronic Colloquium on Computational Complexity, Reports Series 1997.

[3-11] J. Hoffstein and J. H. Silverman, “Protecting NTRU against chosen ciphertext and reaction attacks”, NTRU Cryptosystems Technical Report, Report # 016, Version 1, June, 2000.

[3-12] J. Hoffstein, J. Pipher and J. H. Silverman, “NTRU: A ring-based public key cryptosystem”, In Proc. of Algorithmic Number Theory Symposium (ANTS-3), Lecture Notes in Computer Science, pp. 267-288, Springer=Verlag, 1998.

[3-13] E. Jaulmes and A. Joux, “A chosen-ciphertext attack against NTRU”, In Proc. of Crypto 2000, Lecture Note in Computer Science 1880, pp. 20-35, Springer-Verlag, 2000.

[3-14] H. W. Lenstra, Jr., “Integer programming with a fixed number of variables”, Math. Oper. Res., 8 (4), pp. 538-548, 1983.

[3-15] A. K. Lenstra, H. W. Lenstra and L. Lovasz, “Factoring polynomials with rational coefficients”, Mathematishe Ann. 261, 513-534, 1982.

[3-16] J. C. Lagarias and A. M. Odlyzko, “Solving low-density subset sum problems”, Journal of the Association for Computing machinery, January 1985.

[3-17] R. Merkle and M. Hellman, “Hiding information and signatures in trapdoor knapsacks”, IEEE Trans. Information Theory, IT-24, pp.525-530, September, 1978.

[3-18] D. Micciancio, “Lattice based cryptography: a global improvement”, Technical Report, Theory of Cryptography Library, Report 99-05, 1999.

[3-19] P. Nguyen, “Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from Crypto ’97”, In Proc. of Crypto’99, Lecture Notes in Computer Science 1666, pp. 288-304, Springer-Verlag, 1999.

[3-20] P. Nguyen and J. Stern, “Cryptanalysis of the Ajtai-Dwork cryptosystem”, In Proc. of Crypto ’98, Lecture Notes in Computer Science 1462, pp. 223-242, Springer-Verlag, 1998.

[3-21] P. Nguyen and J. Stern, “Lattice reduction in cryptology: an update”, In Proc. of Algorithm Number Theory Symposium (ANTS-4), Lecture Notes in Computer Science 1838, pp. 85-112, Springer-Verlag, 2000.

[3-22] A. Shamir, “A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem”, In Proc. of 23rd FOCS, pp. 145-152,1982.

[3-23] C. P. Schnorr, “A hierarchy of polynomial lattice basis reduction algorithms”, Theoretical Computer Science 53, 201-224, 1987.

[3-24] C. P. Schnorr, “A more efficient algorithm for lattice basis reduction”, Journal of Algorithms 9 (1), 47-62, 1988.

[3-25] C. P. Schnorr and M. Euchner, “Lattice basis reduction: improved practical algorithms and solving subset sum problems”, Math. Programming 66, pp. 181-199, 1994.

[3-26] C. P. Schnorr and Hörner, “Attacking the Chor-Rivest cryptosystem by improved lattice reduction”, In Proc. of Eurocrypt’95, Lecture Notes in Computer Science 921, pp. 1-12, Springer-Verlag, 1995.

[3-27] S. Vaudenay, “Cryptanalysis of the Chor-Rivest cryptosystem”, In Proc. of Crypto’98, Lecture Notes in Computer Science 1462, Springer-Verlag, 1998.

[3-28] C. M. Adams and H. Meijer. “Security-Related Comments Regarding McEliece’s Public-Key Cryptosystem”. In *Proc. of CRYPTO ’87, LNCS 293*, pages 224–228. Springer–Verlag, 1988.

[3-29] M. Bellare and P. Rogaway. “Optimal Asymmetric Encryption”. In *Proc. of EUROCRYPT ’94, LNCS 950*, pages 92–111, 1995.

[3-30] T. Berson. “Failure of the McEliece Public-Key Cryptosystem Under Message-Resend and Related-Message Attack”. In *Proc. of CRYPTO ’97, LNCS 1294*, pages 213–220. Springer–Verlag, 1997.

[3-31] A. Canteaut and N. Sendrier. “Cryptoanalysis of the Original McEliece Cryptosystem”. In *Proc. of ASIACRYPT ’98*, pages 187–199, 1998.

[3-32] D. Dolve, C. Dwork, and M. Naor. “Non-Malleable Cryptography”. In *Proc. of the 23rd STOC*. ACM Press, 1991.

[3-33] E. Fujisaki and T. Okamoto. “Secure Integration of Asymmetric and Symmetric Encryption Schemes”. In *Proc. of CRYPTO ’99, LNCS 1666*, pages 535–554, 1999.

[3-34] C. Hall, I. Goldberg, and B. Schneier. “Reaction Attacks Against Several Public-Key Cryptosystems”. In *Proc. of the 2nd International Conference on Information and Communications Security (ICICS’99), LNCS 1726*, pages 2–12, 1999.

[3-35] K. Kobara and H. Imai. “Countermeasure against Reaction Attacks (in Japanese)”. In *The 2000 Symposium on Cryptography and Information Security : A12*, January 2000.

[3-36] K. Kobara and H. Imai. “OAEP++ – Another Simple Bug Fix in OAEP –”. In *Rump Session at Asiacrypt 2000: http://imailab-www.iis.u-tokyo.ac.jp/kobara/Material/OAEP++.pdf*, 2000.

[3-37] K. Kobara and H. Imai. “Semantically Secure McEliece Public-Key Cryptosystems –Conversions for McEliece PKC–”. In *Proc. of PKC ’01*. Springer–Verlag, 2001.

[3-38] P. J. Lee and E. F. Brickell. “An Observation on the Security of McEliece’s Public-Key Cryptosystem”. In *Proc. of EUROCRYPT ’88, LNCS 330*, pages 275–280. Springer–Verlag, 1988.

[3-39] Y. X. Li, R. H. Deng, and X. M. Wang. “On the Equivalence of McEliece’s and Niederreiter’s Public-Key Cryptosystems”. *IEEE Trans. on IT*, 40:271–273, 1994.

[3-40] P. Loidreau and N. Sendrier. “Some weak keys in McEliece public-key cryptosystem”. In *Proc. of IEEE International Symposium on Information Theory, ISIT ’98*, page 382, 1998.

[3-41] R. J. McEliece. “A Public-Key Cryptosystem Based on Algebraic Coding Theory”. In *Deep Space Network Progress Report*, 1978.

[3-42] A. J. Menezes, P. C. Oorschot, and S. A. Vanstone. “McEliece public-key encryption”. In *“Handbook of Applied Cryptography”*, page 299. CRC Press, 1997.

[3-43] T. Okamoto and D. Pointcheval. “REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform”. In *Proc. of RSA Conference ’01*, 2001.

[3-44] D. Pointcheval. “Chosen-Ciphertext Security for Any One-Way Cryptosystem”. In *Proc. of PKC 2000, LNCS 1751*, pages 129–146. Springer–Verlag, 2000.

[3-45] B. Schneier. “McEliece”. In *“Applied Cryptography, Second Edition”*, pages 479–480. John Wiley & Sons, 1996.

[3-46] N. Sendrier. “The Support Splitting Algorithm”. *Rapport de recherche: ISSN0249-6399*, 1999.

[3-47] V. Shoup. “OAEP Reconsidered”. 2000.

[3-48] J. Stern. “A method for finding codewords of small weight”. In *Proc. of Coding Theory and Applications , LNCS 388*, pages 106–113. Springer–Verlag, 1989.

[3-49] H. M. Sun. “Further Cryptanalysis of the McEliece Public-Key Cryptosystem”. *IEEE Trans. on communication letters*, 4:18–19, 2000.

第4章　時間的安全性に基づく方式の研究動向調査

4.1　タイムスタンプ方式

4.1.1　方式の説明

タイムスタンプ方式とは、あるデータが特定の時刻に存在し、かつ、その時刻以降データが改竄されていないことを第三者に証明する方式である。

近年インターネット上での電子商取引が活発化しているが、計算能力・解読技術の進歩や事故等によって現在の基盤における暗号技術の安全性が低下することにより、過去の取引や契約情報が危険に冒される恐れがある。例えば電子債権や電子手形の場合、債務者がそれらに対してディジタル署名を作成してから一定期間の後に換金される。この期間にディジタル署名の安全性が著しく低下し、署名生成鍵が漏洩してしまうかもしれない。そのような場合、第三者が不正に債権を偽造し、払い戻しを要求することも考えられる。このような脅威に対して、正規の所有者は自分の持っている債権が正しいものであること、つまり過去に正しく取引されたものであることを証明できなくてはならない。

また、近年紙ベースの文書を電子媒体に置き換えて管理する電子文書管理を採用する動きも広がっているが、例えば特許などの場合、いつ申請されたかという時間情報が重要な意味を持ち、将来にわたってその事実が揺るがないよう固定化しなければならない。

タイムスタンプ方式はこれらのニーズを背景に、現在特に注目されている技術である。本節ではこのタイムスタンプ方式の概要とその研究動向について述べる。

タイムスタンプ方式は以下のエンティティにより構成され、各エンティティ間でのプロトコル実行によりタイムスタンプの生成・検証を行う。(1) **発行依頼者**：あるデータに対してタイムスタンプを発行するよう発行機関に依頼し、タイムスタンプを受け取る。(2) **発行機関**：依頼されたデータに対してタイムスタンプを発行する。同時にタイムスタンプの検証に用いられる情報 (以下、検証情報) を生成し、しかるべきところに保管する。(3) **証明者**：データが特定の時刻に存在していたことを検証者に証明する。一般的には、発行依頼者と同一のエンティティ。(4) **検証者**：証明者から入手したタイムスタンプ等の情報と、発行機関から入手した検証情報を用いて、データが特定の時刻に存在したか否かを確認する。

この他、発行機関とは別のエンティティが検証情報、または発行機関の保管する検証情報の一貫性を保証する情報を保管し、検証者の求めに応じてその情報を提供する方式など

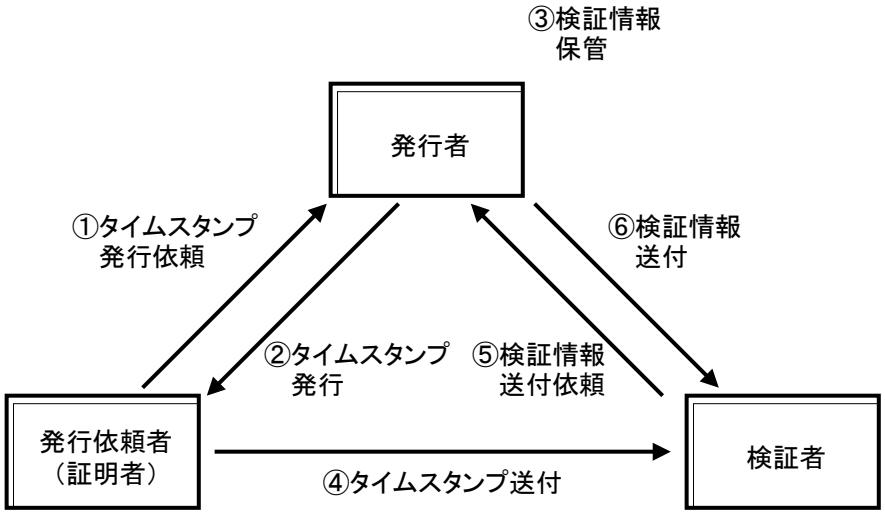


図 4.1: タイムスタンプ生成・検証手順

も提案されている。

4.1.2　研究動向

ISO13888[4-2] では、前記プロトコルにおいて発行機関が十分信頼できることを前提としたタイムスタンプ方式が規定されている。この方式はシステム構成が単純なため、コスト面を重視するアプリケーションには向いていると言える。

しかし、タイムスタンプ方式では発行機関が重要な位置を占めており、発行機関に絶対の信頼をおくことは安全上問題がないとは言い切れない。例えば、発行機関が不正を行い、特定の人に有利なタイムスタンプを発行する恐れがある。また、発行機関の秘密鍵が漏洩することによりシステム全体が利用できなくなる危険性も考えられる。そこで、発行機関が信頼できない (不正を行う可能性がある、または事故等によりダウンする可能性がある) 場合でもタイムスタンプの安全性を確保するためにはどうしたら良いか、という観点からさまざまな方式が提案されている。これらの提案方式を大別すると、連鎖型と分散型に分けられる。

連鎖型 他のタイムスタンプに含まれる情報を用いてタイムスタンプを生成する方式で、タイムスタンプの時系列的な順序を保証する。タイムスタンプの時系列が一貫していることを確認するための情報を第三機関等で安全に保管する、または大勢の人目に触れるメディア等に発表することにより、発行機関の不正を困難にする。連鎖型タイムスタンプ方式としては、Bayer らの方式 [4-4]、Cuculus[4-5]、TIMESEC[4-6]、PKITS[4-7] などが挙げられる。

分散型 複数の発行機関を設け、一定数の発行機関が結託しない限りタイムスタンプの偽造・改竄が困難な方式。ただし、タイムスタンプを発行する際、一定数以上の発行機関に発行を依頼しなければならない。分散型タイムスタンプ方式としては、秘密分散共有法を用いた NTT の分散時刻証明システム [4-8] が挙げられる。

一方、各方式の安全性評価に関する動きも活発になっている。タイムスタンプ方式に対する攻撃としては、前述の発行機関のタイムスタンプ偽造以外にも、発行依頼者同士が結託してタイムスタンプを偽造する場合、発行依頼者と検証情報を保管する第三機関が結託する場合などが考えられる。宇根・松本 [4-11, 9] は、各方式の安全性を体系的に評価するため、タイムスタンプに含まれる情報、検証情報の取得先、検証手続きなどによる従来方式の分類方法を提案し、連鎖型方式に対する偽造攻撃およびバイパス攻撃についての安全性評価おこなっている [4-12]。

4.1.3 課題とまとめ

宇根・松本 [4-11] によってタイムスタンプ方式の概念整理および分類が行われたことにより、安全性の評価・比較を行う動きが活発化している。しかし、全ての方式および攻撃方法に対する安全性評価はいまだ不十分といわざるを得ない。これらの安全性評価を厳密なものとするとともに、安全性証明可能な方式について検討していくことも必要であろう。

一方、タイムスタンプ方式とは異なった方法でデータの時間情報を保証する方式も提案されている。松本ら [4-14] により提案されたヒステリシス署名では、署名者が過去に生成した署名の履歴情報を新しい署名に埋め込むことにより、ある署名が過去に生成されたものか否かをその履歴情報から検証することができる。

繰返しになるが、現在の基盤におけるデジタル署名、ハッシュ関数等の暗号技術は、計算能力・解読技術の進歩や事故等によって安全性が低下する可能性がある。そのような場合でも、過去のデータの有効性・安全性が低下しないような方式の研究・評価を行う必要がある。その意味でタイムスタンプ方式の研究、およびヒステリシス署名に代表される新たな方式を模索する必要があると考えられる。

4.2 Witness-base 署名方式

4.2.1 Witness-base 署名方式とは

Witness-base 署名方式は、計算量的安全性に依らず、かつ、既存のインフラを用いて実現できる署名方式として唯一のものである。原理を簡単に説明すると、署名者と契約書の受領者が契約書に署名する際 (この署名には計算量的安全性に基づく署名方式を用いる)、署名された契約書のコピーを witness というグループに送る。こうすることによって、後に論争が起こった場合に、裁判所に witness を召喚することによって正しい契約書を参照することができる。しかし、この方法では witness が不正に契約書を改ざんできてしまうため、witness に高い信頼性が要求されるという問題がある。そこで、Witness-base 署名方式では匿名通信プロトコルを用いることにより、署名者と受領者を、witness グループから特定できなくする。署名者や受領者がわからなければ、witness が不正を働く可能性は低くなる。また、データの保全性を高めるため、一人の witness だけに契約書を保持させるのではなく、秘密分散法を用いて契約書を複数の秘密片に分け、そのコピーを witness グループに送信する。さらに、こうすることによって witness に対する信頼性を分散させることもできる。

次に，署名者と受領者の契約書が一致しない場合の論争解決の手段を説明する．論争の際には、まず裁判所が署名者と受領者に秘密片を提出させる。そして、それぞれの秘密片の 50%の部分を witness グループに公開し、Witness グループの中からこれらの秘密片と一致するものを持っている witness を探し、その秘密片を提出させる。秘密片を全て公開しないのは、全て公開してしまうと，witness が全ての秘密片を手に入れて、契約書が復元されてしまうからである。裁判所は、witness から集めた秘密片より正しい署名を復元し、その署名が生成された時間を特定することができる。さらに裁判所は、署名者と受領者から提出させた秘密片より、それぞれの契約書を復元し、これを witness から手に入れた契約書と比較してどちらが不正を働いたかを判定する。

4.2.2 匿名通信プロトコル ‘Crowds’ に関して

匿名通信プロトコルに関しては、Reiter と Rubin によって提案された Crowds[4-15] プロトコルを利用している。図 4.2 に Crowds を用いた匿名通信の仕組みを示す。

匿名通信を行いたいユーザはまず、Blender に自分の IP アドレスを伝えて crowd のメンバー (jondo) となる。jondo となるとユーザは他の jondo の IP アドレスのリストを渡される。

crowd のメンバーとなったユーザが匿名通信を試みる場合、直接サーバにメッセージを送るのではなく、ランダムに選んだ他の jondo にメッセージの送信を依頼する。メッセージの送信を依頼されたユーザはサーバにメッセージを送信するか、他のメンバーに送信を依頼するか決める。他のユーザに送信を依頼する場合は、ランダムに選んだ crowd のメンバーにメッセージを送信する。これを繰り返すことにより、最終的にサーバがメッセージを受け取ったときに、サーバはランダムに選ばれたユーザからメッセージを受信するの

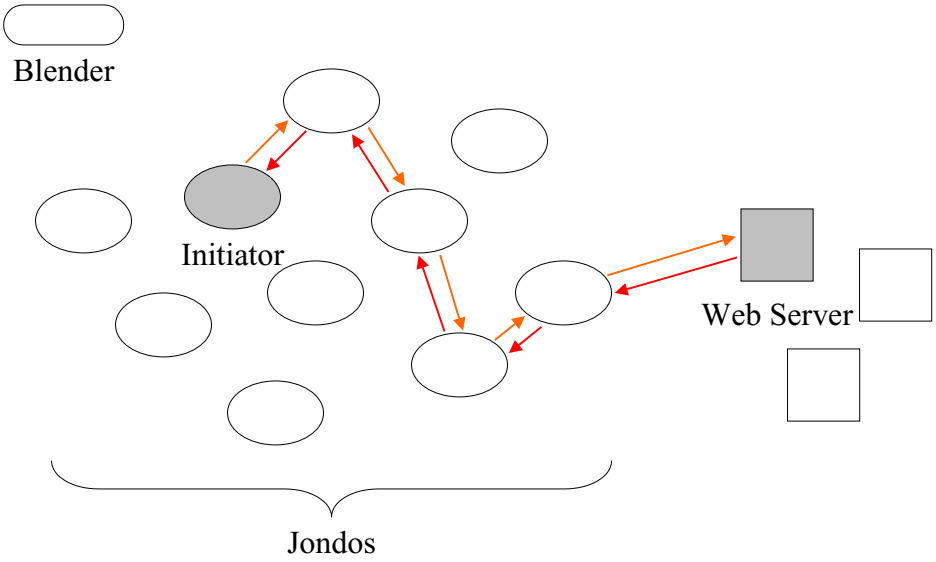


図 4.2: Crowds を用いた匿名通信

で、メッセージの生成者が誰であるか特定できないことになる。また、Blender は、プロトコル中ではメンバーの登録、脱退手続きのときのみ登場するので、Blender に要求される信頼性はそれほど高くはないというメリットがある。

4.2.3 研究動向

Witness-base 署名方式の詳しい内容は、文献 [4-17] に述べられている。Witness-base 署名方式に関する研究は地道ではあるものの今後の進展が期待される。また、Witness-base 署名方式は、匿名通信プロトコルに Crowds 方式を採用しているが、crowd の中にメッセージの送信を拒否したり、メッセージを改変して転送したりという不正を働くメンバーがいた場合、サービス全体に影響が出てくることになる。この問題を取り扱った研究も行われており [4-16]、Witness-base 署名方式と関連する研究として動向が注目される。

4.2.4 課題とまとめ

Witness-base 署名方式の利点を整理してみると、

- 公開鍵暗号を利用した署名方式と異なり計算量的安全性に依らないので長期間に渡り、署名の安全性が保たれる。
- TCP-IP 上で動作することを前提としており、既存のもの以外に特別なインフラを

必要としないので、実装が比較的簡単である。

逆に、Witness-base 署名方式に残されている課題として、

- 秘密片から元の契約書を復元する際の計算量や witness のメモリサイズという点からよりよい秘密分散法はないか。
- 裁判所に召喚される witness に対して匿名性を与えることはできないか。
- TCP-IP プロトコルを詳細に調査し、Witness-base 署名方式を実装する際に安全上問題となり得る個所をみつけだし、対処する。

が挙げられる。

実用化する際に解決すべき問題も多く残されており、これらを解決し、シミュレーションを用いたパフォーマンスの評価や実装を行うことが今後の大きな目標となるだろう。

参考文献

[4-1] C.Adams, P.Cain, D.Pinkas and R.Zuccherato, Internet X.509 Public Key Infrastructure Time Stamp Protocol (TSP)," January 2001.

[4-2] ISO/IEC JTC1/SC27, "ISO/IEC 13888-1, Information technology - Security techniques - Non-repudiation - Part 1: General," 1997.

[4-3] S.Haber and W.S.Stornetta, "How to Time-Stamp a Digital Document," Journal of Cryptology, Vol.3, No.2, pp.99-111, 1991.

[4-4] Dave Bayer, Stuart Haber and Scotto Stornetta, "Improving the Efficiency and Reliability of Digital Time-Stamping," Sequences II: Methods in Communication, Security and Computer Science, pp.329-334, Springer-Verlag, 1992.

[4-5] Ahto Buldas, Peeter Laud, Helger Lipmaa and Jan Villemson, "Time-Stamping with Binary Linking Schemes," Proc. of CRYPTO'98, LNCS 1462, pp.486-501, Springer-Verlag, 1999.

[4-6] Jean Jacques Quisquater, Henri Massias, J.S.Avila, Bart Preneel and Bart Van Rompay, "Specification and Implementation of Timestamping System," TIMESEC Technical Report 4, 1999.

[4-7] Fabrica Nacional de Moneda y Timbre, PKITS: Deliverable D4 Time-Stamping Service Functional Specification and Protocols for Unstructured Data Revision Number 16, July 30, 1998.

[4-8] A.Takura, S.Ono and S.Naito, "Secure and Trusted Time Stamping Authority," Proc. of IWS'99, pp.123-128, 1999.

[4-9] 宇根正志, "電子文書の送受信証明を行うためのプロトコルの研究動向と安全性評価," 日本銀行金融研究ディスカッションペーパーシリーズ 2000-J-33, 2000.

[4-10] 宇根正志, 松浦幹太, 田倉昭, "最近のディジタルタイムスタンプ技術の現状と課題", 金融研究第 19 巻別冊第 1 号, pp.105-154, 2000.

[4-11] 宇根正志, 松本勉, "連鎖型タイムスタンプの検証に用いられる情報の管理", コンピュータセキュリティシンポジウム 2000(CSS2000) 論文集, 情報処理学会, pp.25-30, 2000.

[4-12] 宇根正志, 松本勉, “タイムスタンププロトコル Cuculus と PKITS の安全性に関する一考察”, 情報処理学会研究報告, 20000-CSEC-11, 情報処理学会, pp.55-60, 2000.

[4-13] 宇根正志, 松本勉, “タイムスタンプの安全性と検証手続きとの関連性”, 2001 年暗号と情報セキュリティシンポジウム (SCIS2001) 予稿集, 電子情報通信学会, pp.629-634, 2001.

[4-14] 松本勉, 岩村充, 佐々木良一, 松木武, “暗号ブレイク対応電子署名アリバイ実現機構 (その 1)”, 情報処理学会研究報告, 2000-CSEC-8, 情報処理学会, pp.13-17, 2000.

[4-15] M. Reiter and A. Rubin, Crowds: Anonymity for Web Transactions; ACM Transactions on Information and System Security, (April, 1998).

[4-16] 野尻　大祐, Anderson Nascimento, 今井　秀樹 : “Dealing with Malicious Users in Crowds” SCIS2001, 神奈川県大磯 (2001-01)

[4-17] Nascimento A., Mueller-Quade J., Imai H., “Improving Digital Signatures by Use of Witnesses”, ISEC 11, Technical Meeting, Tokyo, 2000

第5章　耐タンパー性に基づく方式の研究動向調査

5.1　システム鍵方式

5.1.1　耐タンパー装置を用いたIDに基づく暗号方式

普通、耐タンパー装置を使用する目的は、他人が不正に秘密の情報を読み出したり、不正な書き込みを防ぐことであるが、そのような装置は、正当な使用者自身に対しても、秘密の情報の読出しや不正な書換えを許さないように構成することもできる。このような耐タンパー装置を用いると、IDに基づく暗号方式が可能となる。ただし、これには、選択平文攻撃に対しても安全な共通鍵暗号方式が必要である。また、信用できる管理センターが必要である。

図5.1にIDに基づく暗号方式のための耐タンパー装置を示す。この図で、E、Dは共通

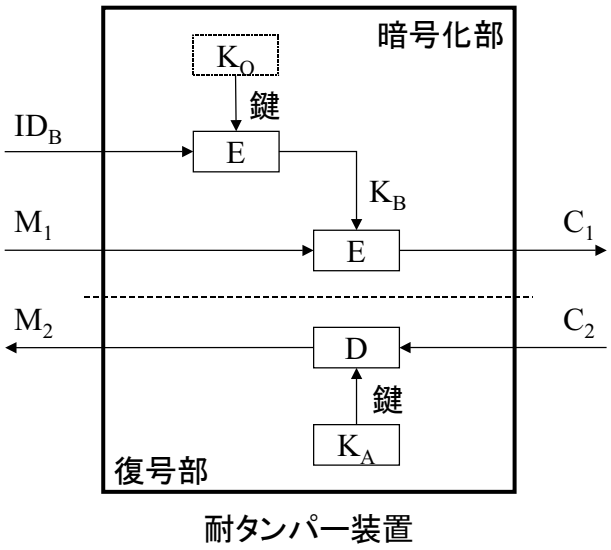


図 5.1: 耐タンパー装置を用いるID暗号方式

鍵暗号方式の暗号器と復号器を表す。K_0 と K_A は秘密鍵である。このうち、K_0 はシステ

ム全体に共通の秘密鍵で、これが漏れるとシステム全体が破れてしまう。K_A はこの耐タンパー装置の所有者である Alice の秘密鍵で、管理センターで、Alice の名前 ID_A を鍵 K_0 で暗号化して作る。Alice が Bob に暗号文を送りたい場合は、この耐タンパー装置の暗号化部に Bob の名前 ID_B と平文 M_1 を入力する。すると、相手の秘密鍵 K_B で平分 M_1 を暗号化した暗号文 C_1 が出力される。Alice が受け取った暗号文 C_2 を復号する場合は、これを耐タンパー装置の復号部に入れれば、Alice の秘密鍵で、復号され平文 M_2 が得られる。

5.1.2　耐タンパー装置によるディジタル署名

前述の耐タンパー装置と選択平文攻撃に強い共通鍵暗号方式を用いた暗号方式は一種の公開鍵暗号方式とみなすことができる。ID を入力するだけで、相手の暗号文を復号できるので相手の公開鍵を入手する必要がないという点で公開鍵暗号方式よりも優れているが、各自が勝手に鍵を生成することができないという点では不利である。そして、公開鍵暗号方式とみなせるということは、電子署名にも使えるということである。

図 5.2 に電子署名を生成するための耐タンパー装置を示す。Alice がメッセージ M_2 の署

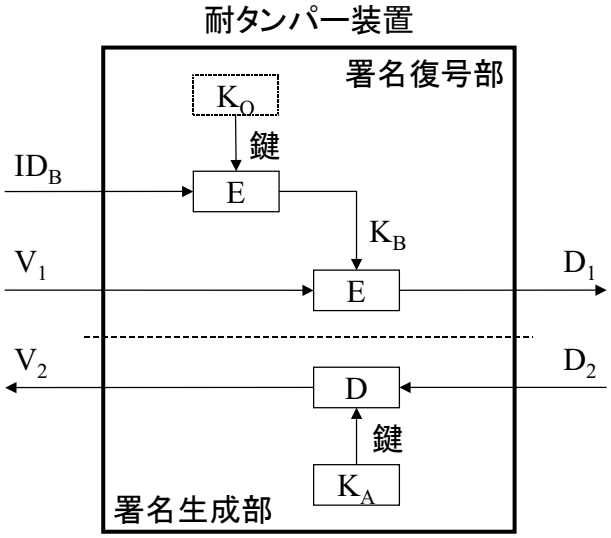


図 5.2: 耐タンパー装置を用いた署名方式

名を生成する際は、M_2 をまずハッシュ関数で圧縮しメッセージダイジェスト D_2 を作る。これを署名生成部に入力し、Alice の秘密鍵 K_A で暗号化することにより、署名 V_A を生成することができる。Alice が Bob から受け取ったメッセージ M_1 に対する署名 V_B を検証する場合は、V_B を Bob の ID_B とともに、耐タンパー装置の署名復号部に入力する。すると、メッセージダイジェスト D_1 が出力されるので、これを Alice が自分で計算した M_1

のメッセージダイジェストと比較することにより署名の検証が行われる。

5.1.3 研究動向と今後の課題

耐タンパー装置の概念が登場して以来、この耐タンパー装置を用いた暗号方式や署名方式が研究されてきたが、完全に安全な耐タンパー装置は実現困難であるということが理解され始めてからは、耐タンパー性自体に関する研究は縮小傾向にあるといえる。しかし、後述する KPS(鍵事前配送システム) と耐タンパー装置を組み合わせた方式を用いれば、耐タンパー装置が破られたとしてもシステムの安全性を保つことができる。KPS はユーザ間で必要になる鍵を予め全て配送しておき必要に応じて二者間の鍵を生成する鍵共有システムであるが、この分野に関しての研究は盛んであり、今後も耐タンパー装置と組み合わせた署名方式の研究は進められていくと思われる。

5.2 KPS 方式

5.2.1 耐タンパー装置と KPS による暗号方式

前述の方式では選択平文攻撃に対して安全な共通鍵暗号方式が必要であったが、KPS(鍵事前配送システム) を用いることにより、この条件を取り除くことができ、情報量的に安全な電子署名システムを築くことができる。

図 5.3 に耐タンパー装置と KPS を用いた署名方式を示す。

各ユーザは署名用と検証用の 2 種類の ID を持っている。Alice が Bob に署名付きの文章を送る場合、署名をつけたいメッセージのダイジェスト D_1 とともに Bob の検証用の ID($ID_{B'}$) を入力すると、KPS 鍵 $K_{AB'}$ によって暗号化された署名 V_1 が出力される。

次に Alice が Bob の署名を検証する場合を考える。Alice と同様に、Bob は Alice の検証用の ID($ID_{A'}$) を入力して署名を生成する。Bob のメッセージダイジェスト D_2 は KPS 鍵 $K_{A'B}$ によって暗号化され、署名 V_2 となる。Alice は、Bob から送られてきたメッセージダイジェスト D_2、その署名 V_2 とともに、Bob の署名用の ID(ID_B) を装置に入力する。装置内では、Bob の署名 V_2 が $K_{A'B}$ で復号化され、さらにメッセージダイジェスト D_2 と比較され、結果が出力される。

KPS があれば耐タンパー装置がなくても同様の署名方式が実現できるが、各ユーザが所有する KPS 関数 $f(ID, x)$ を耐タンパー装置に入れることにより、ユーザ間の結託攻撃を防ぐことができる。KPS では何人のユーザが結託すれば結託攻撃によってシステムを破ることができるかを表す結託閾値というパラメータが存在する。当然、結託閾値は大きい方が望ましいのであるが、結託閾値を大きくすればするほど各ユーザに必要とされるメモリサイズも大きくなってしまう。しかし、耐タンパー装置を用いてユーザ間の結託を難しくすれば、結託閾値を小さくすることができるので、ユーザに必要なメモリサイズも小さくできるというメリットがある。

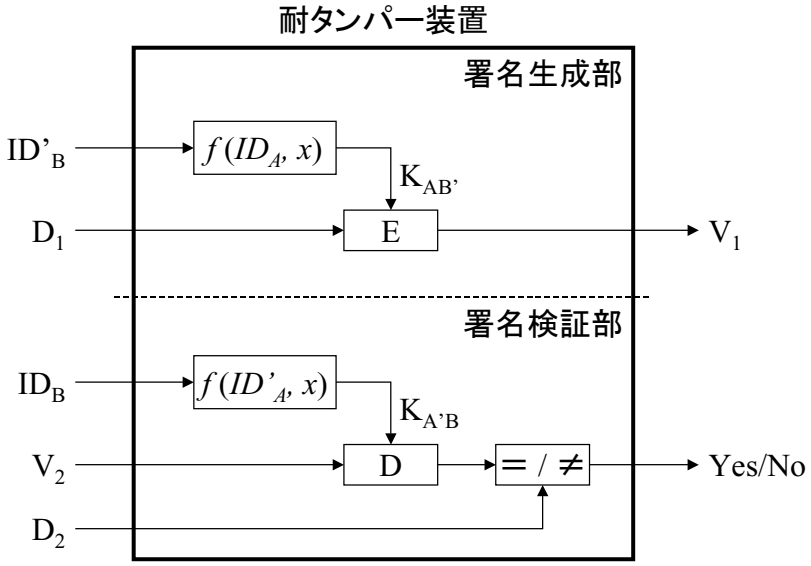


図 5.3: 耐タンパー装置と KPS を用いた署名方式

5.2.2　研究動向

前述の耐タンパー装置と KPS を組み合わせた方式では情報量的に安全な署名方式が実現できたが、署名者が意図した受信者しか署名を検証できないという大きなデメリットがある。この問題を取り扱った研究に ISSUSG[5-2] がある。ISSUSG では署名者が署名できる回数に制限があるものの、誰でもその署名を検証できるというより現実的な署名方式を提案している。ISSUSG も KPS に基づいた方式であるため、ユーザ間の結託攻撃が問題となるが、耐タンパー装置と ISSUSG を組み合わせれば、この問題を解決することができ、必要なメモリサイズも小さくできる。

また、KPS に関する研究としては [5-1] などがある。これは KPS の鍵配送センターを複数に分けて階層的に配置することにより、効率的に鍵配送を行ったり、センターがユーザの不正を監視する際の負担を軽くしようというものである。

5.2.3　課題とまとめ

耐タンパー装置と KPS を組み合わせた電子署名方式のメリットは、

- 情報量的に安全な署名方式である。
- ユーザの結託攻撃を耐タンパー装置を用いることにより防ぐことができる。
- システム全体の秘密を暴くには結託閾値以上の KPS 関数が必要であるため、一つの

耐タンパー装置を破るだけでは、システム全体を暴くことができない。

などが挙げられるが、前述した署名回数の制限の問題や、KPS のためのインフラを必要とするので実装が困難であるという点は今後改善されるべき点であろう。また、現時点では耐タンパー装置の安全性はそれほど高いとは言えず、耐タンパー装置の研究の動向も注目していかなければならない。

参考文献

[5-1] D. Nojiri, G. Hanaoka and H. Imai : "A Practical Implementation of Hierarchically Structured Key Predistribution System and Its Evaluation" ISW2000, LNCS 1975 Information Security ISW2000 Proceedings pp.224-236, Wollongong, Australia (2000-12)

[5-2] G. Hanaoka, J. Shikata, Y. Zheng and H. Imai, "Unconditionally Secure Digital Signature Schemes Admitting Transferability," Advances in Cryptology - ASIACRYPT2000, LNCS 1976, Springer-Verlag, pp.130-142, 2000.

おわりに

素因数分解問題あるいは離散対数問題と独立な問題の難しさに基づく方式として、情報量的安全性に基づく方式、計算量的安全性に基づく方式、時間的安全性に基づく方式、耐タンパー性に基づく方式をとりあげ、それらの研究動向を調査した。

情報量的安全性に基づく方式に関しては、情報量的に安全な鍵共有方式および認証方式の調査を行い、これらの性質を明らかにした。これらの方式は、理論上、将来にわたる確実な安全性を保証することが可能であり、長期的な安全性が必要となるアプリケーションに対してはとくに有効であることが示された。これらの方式は必要となる記憶容量が大きく、計算機環境によっては問題となりうるが、技術の進歩や記憶装置の大容量化に伴い、このような問題は解決するものと思われる。今後も、情報量的安全性に基づく各方式において必要となる記憶容量の削減方法に重点をおいて調査を続けていく必要がある。

計算量的安全性に基づく方式に関しては、素因数分解問題あるいは離散対数問題に頼らず、なおかつ安全で効率のよい公開鍵暗号方式の存在が明らかとなった。しかしながら、素因数分解問題あるいは離散対数問題に頼らず、かつ安全で効率のよい電子署名方式は現在のところ知られていない。そのような電子署名方式を提案するための調査研究を今後も続けていく必要がある。

時間的安全性に基づく方式に関しては、過去の情報を保護する必要が高まっていることにより、研究が活発化していることが分かった。本稿で説明したタイムスタンプ方式では、今後、タイムスタンプ発行者の不正を含む、既存方式に対する安全性の評価が必要と言える。また、計算量的安全性に依らず、かつ、既存のインフラを用いて実現できる署名方式として唯一のものといえる Witness-base 署名方式について説明した。Witness-base 署名方式は、匿名通信プロトコルに Crowds 方式を採用しているが、crowd の中に不正を働くメンバーがいた場合サービス全体に影響が出てくることになる。今後はこうした不正ユーザの対策に関する研究が行われて行くであろう。

耐タンパー装置と KPS を用いた署名方式に関しては、耐タンパー装置と KPS を組み合わせることにより、耐タンパー性を KPS の結託耐性によって補うことができることを示した。また、耐タンパー装置の耐タンパー性を利用することによって KPS のメモリサイズの問題も解決されることを示した。しかし、署名回数の制限の問題や、KPS のためのインフラを必要とするので実装が困難であるという点は今後改善されるべきであろう。

情報量的安全性、計算量的安全性、時間的安全性、耐タンパー性をバランスよく組み合わることにより、より安全でより使い易い暗号・認証基盤が構築されること、また、その構築の際に本報告書が役立つことを期待する。