情報化社会における暗号技術の役割

神田　雅透

情報化社会が安全に機能するための基盤技術の一つに暗号技術があることはよく知られている．しかし，縁の下の力持ち的存在であるがゆえに普段からあまり意識されずに利用されている一方で，その中身は難しいものと思われて興味の対象からは敬遠されがちである．しかし，暗号解読では確率分布などを利用したり，リスクコミュニケーションでは最適値問題としてOR的手法が必要となるなど，ORに近い分野の研究もある．そこで，本特集記事への橋渡しを兼ね，どのような暗号技術がどのような場面で使われているのかについて，最近の動向を交え，紹介する．

キーワード：暗号，セキュリティ，SSL，PKI，標準化

1．はじめに

インターネットに代表される情報化社会では，多くのユーザがオンラインショッピングやオンラインバンキング，あるいはオンラインゲームなどのサービスを通じて，様々な個人情報や金銭情報など価値ある情報をネットワーク上でやり取りしている．また，世界中でネットワークを介して多くの情報が瞬時にやり取りされるようになった結果，企業にとっても新たなビジネスチャンスが広がった一方，情報漏えいに伴う企業経営リスクが無視できなくなっている．

そこで，情報を必要とする人に安全かつ確実に送り届ける手段として暗号技術が大いに活用されるようになり，多くのセキュリティ製品が出回るようになった．

代表的なものでは，USB暗号化ソフトやシンクライアント，VPN（Virtual Private Network）など企業が導入する情報漏えい対策システムはもとより，個人が利用するSSL（Secure Socket Layer）や無線LANなどがある．このほかにも，ほとんど無意識のうちに使っているものとして，携帯電話の端末認証，SuicaやPASMOなどのICカード乗車券，Edyなどの電子マネー，B-CASやWOWOWといったスクランブル放送などもある．

これらのサービスすべてが暗号技術なしには成り立たないものばかりであるが，その中身は難しいものと思われて興味の対象からは敬遠されがちである．そこで，暗号技術の詳細な内容については以降の各特集記事をご覧いただくとして，本稿では全体の概観を紹介していく．

かんだ　まさゆき
NTT 情報流通プラットフォーム研究所
〒180-8585 武蔵野市緑町3-9-11

2．暗号技術の分類

情報セキュリティの特徴を端的に表す言葉として「CIA」がある．これは，無権限者への情報の秘匿性を表すConfidentiality，情報が正しいこと（完全性）を表すIntegrity，正当な権限者がいつでも利用できること（可用性）を表すAvailabilityの頭文字をとったものである．このうち，暗号技術は機能として主に「C」と「I」を受け持つことになる．

2.1　機能からの分類

機能からの分類とは，端的には「C」と「I」とへの分類ということになる（表1）．

「C」に相当するのが「暗号化」であり，盗聴や紛失・盗難への対策となる．一方，「I」に相当するのが「認証・署名」であり，データの改ざん・偽造の防止，成りすまし防止，行為否認の防止などを目的として利用される．

2.2　特徴からの分類

特徴からの分類では，暗号に関しては平文（もとのメッセージ）から暗号文に変換する暗号化と，暗号文から平文に変換する復号とで利用する鍵が同じか否かで大きく「共通鍵暗号」と「公開鍵暗号」とに分かれ

表1　機能分類

暗号化	権限が無い者から情報を秘匿
メッセージ認証	情報の完全性を確認
ユーザ認証	権限を有するユーザであることを確認
（電子）署名	署名者が情報の完全性を保証 （メッセージ認証+ユーザ認証）

る（表2）．

共通鍵暗号は，暗号化に使う鍵（暗号化鍵）と復号に使う鍵（復号鍵）とが同じ方式である．一般に，暗号化処理が高速であるため，主にコンテンツなどの大きなサイズのメッセージの暗号化に使われる．逆に，通信相手同士は何らかの手段で事前に鍵を共有する必要がある，使用する鍵はすべて秘密にする必要がある，通信相手ごとに異なる鍵を設定する必要があるなどの短所がある．

公開鍵暗号は，暗号化鍵と復号鍵が異なる方式である．暗号化鍵は公開することができ，復号鍵は一人しか持たない．このため，事前に鍵共有を行う必要がなく，不特定多数の暗号通信に適している．その一方，暗号化処理が遅いため，コンテンツなどの暗号化には向かず，実際には共通鍵暗号で使う秘密鍵の配送手段として使われることが多い．

ところで，公開鍵暗号では復号鍵を一人しか持たないことから，その鍵による暗号文は一人しか作れないともいえる．この概念を利用したものがデジタル署名である．つまり，メッセージに対し，秘密に持つ鍵（署名鍵）で暗号化を行い，公開されている鍵（検証鍵）で復号することによって，そのメッセージの完全性と署名者を検証できる．

なお，デジタル署名では，メッセージを圧縮したダイジェストに対して署名を行うことが一般的であるため，ダイジェストを作るためのハッシュ関数が重要な関連技術として挙げられる．

3. 暗号技術はどのように使われているか

暗号技術の実際の利用場面では大きく分けて二つのやり方がある．

一つは，B-CASやICカード乗車券などに代表されるシステムである．図1にB-CASの例を示す．

これらのシステムでは，共通鍵暗号だけで構成されているため，高速処理性に優れている．その一方，サービス提供者が秘密鍵をICカードにあらかじめ封印してユーザに配布し，実際のサービスを受けるためにユーザはICカードをサービス提供者に提示するというやり取りが一般的に行われる．つまり，ICカードを介することによって，サービス提供者とユーザとの間での鍵共有を実現している．

このことは，サービスを受ける前にサービス提供者とユーザとの間で事前に何らかの契約行為が結ばれ，その契約に基づいてサービス提供者がユーザ管理をしながらICカードを発行していることを意味している．

もう一つは，SSLやVPNなどに代表されるシステムである．これらのシステムでは，公開鍵暗号と共通鍵暗号によるハイブリッド暗号として構成される（図2）．

このシステムの最大の特長は，公開鍵暗号を使って共通鍵暗号で利用する秘密鍵（セッション鍵）を配送することにより，事前にセッション鍵を共有していないユーザ同士であってもすぐに暗号通信を始められる

表2　特徴からの分類

共通鍵暗号	暗号化と復号とで同じ秘密鍵を利用	
ブロック暗号	データを一定の長さに区切って暗号化	AES, Camellia, Triple DES
ストリーム暗号	擬似乱数生成器を利用して暗号化	RC4
公開鍵暗号	暗号化と復号とで異なる鍵を利用	
公開鍵暗号	データ（主に秘密鍵）の暗号化に利用	RSA
ハイブリッド暗号	公開鍵暗号と共通鍵暗号を組み合わせた方式	SSL等で採用 (RSA+RC4)
（デジタル）署名	電子署名を実現する一形態	RSA, DSA, ECDSA
ハッシュ関数	任意のデータを一定長のデータに圧縮	MD5, SHA-1, SHA-2, AHS

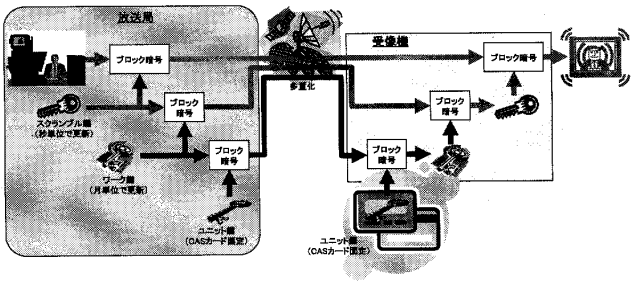

図1　B-CASの例

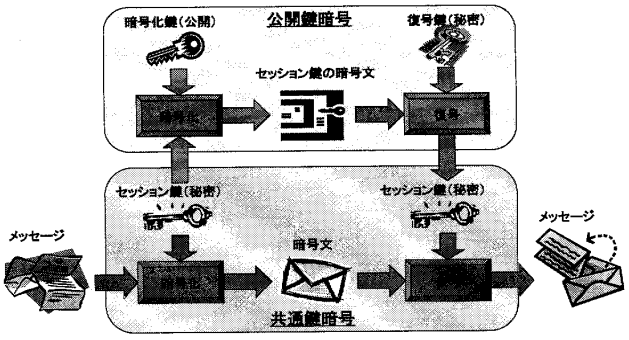

図2　ハイブリッド暗号

ところである．つまり，オンラインショッピングなどでは不特定多数のユーザを相手にビジネス展開をすることができるようになる．

以下では，SSLを例にインターネット上でどのように暗号技術が使われているかを見ていく．

3.1 SSLサーバ証明書

公開鍵暗号では，どのようにして公開されている公開鍵が正しいことを保証するかという問題がある．例えば，ユーザCがユーザBの公開鍵であると偽って自分の公開鍵をユーザAに提示できてしまうと，AはBに送るつもりのメッセージをCに送ってしまうことになるからである．

そこで，Bの公開鍵が確かにBのものであるということを保証する手段として，Bの本人性と，Bの公開鍵と秘密鍵の正当性を確認した信頼できる第三者TTP（Trusted Third Party）がBの公開鍵に自らのデジタル署名を付与した公開鍵証明書を用いる．AはBの公開鍵証明書に正当なTTPのデジタル署名が付いていることを根拠に，Bの公開鍵が正しいものと判断し，Bとの暗号通信を始めることができる．

ここで，TTPのことを認証局CA（Certification Authority）といい，CAによって公開鍵の正当性が保証される全体のスキームのことを公開鍵認証基盤PKI（Public Key Infrastructure）という（図3，4，5）．

SSLに当てはめてみると，上記Aに相当するのがユーザ，Bに相当するのがSSLサーバ，CAに相当するのがVeriSignなどの民間認証局ということになる．SSLで利用する公開鍵証明書のことをSSLサーバ証明書と呼び，当該SSLサーバへ暗号文を送る際に利用する公開鍵が正しいものであることを保証する．

つまり，SSLサーバ証明書はフィッシングサイトへの誘導を防止するなどユーザが騙されないための保険である．したがって，有効期限切れや信頼できない認証局のSSLサーバ証明書では意味がないことに注意されたい．

3.2 SSL通信の開始

SSL通信では，実際の暗号通信を始める前にサーバとブラウザの間で，SSLサーバ証明書の検証，利用する暗号アルゴリズムの決定，およびセッション鍵の共有が行われる（図6）．この処理はハンドシェイクと呼ばれる．

図6からわかるように，SSL通信は，SSLサーバ証明書の正当性を実現するための「署名」，セッション鍵をブラウザとサーバとで共通するために暗号化されたPre Master Secretを送信する「公開鍵暗号」，そして実際の暗号通信を行う「共通鍵暗号」といったすべての暗号技術を組み合わせて実現されたセキュリ

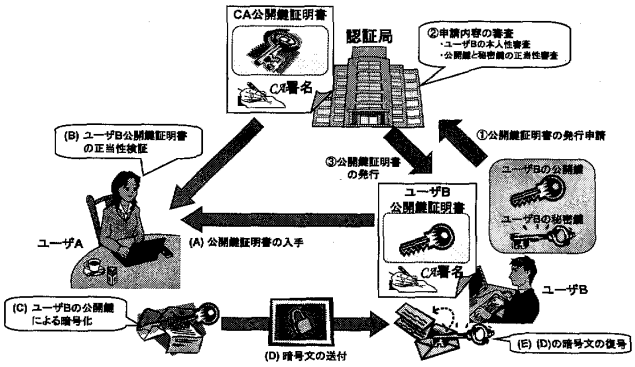

図3　PKIとPKIを利用した暗号通信例

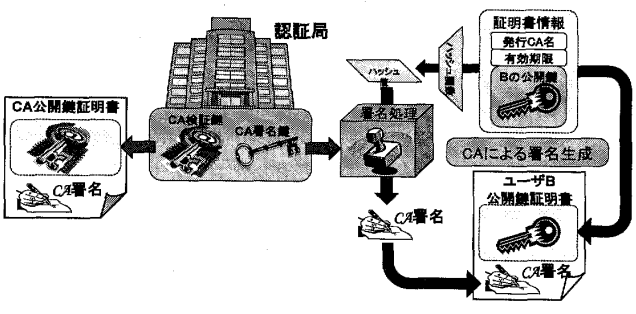

図4　CAによる公開鍵証明書の発行

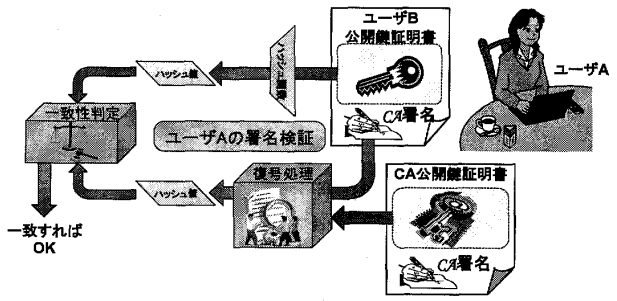

図5　ユーザBの公開鍵証明書の正当性検証

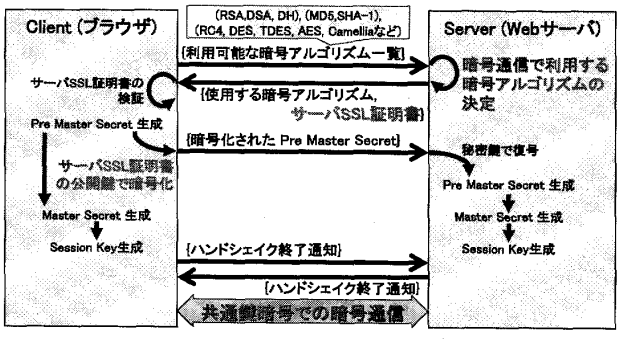

図6　SSL通信

ティプロトコルである．ちなみに，ハッシュ関数は，SSL サーバ証明書の検証処理と暗号通信時の誤り検出処理に用いられる．

4. 解読技術の進展—安全性の概念—

一言で暗号技術の安全性といっても，暗号研究者らが使う安全性の定義は，署名，公開鍵暗号，共通鍵暗号，ハッシュ関数とで少しずつ異なる．詳しくは，以降の本特集記事を参照していただくとして，ここでは概念的な説明をする．特に注目すべきは，暗号技術が安全であるかを判定する基準として二つの要素があることである．ひとつは「暗号解読手法に対する安全性」であり，もう一つは「計算機能力の向上に対する安全性」である．

4.1 暗号解読手法に対する安全性

一般に暗号研究者らが対象とするのは，暗号解読手法に対する安全性であることが多い．

この場合，ある特定の暗号アルゴリズムに対して何らかの解読手法を適用した結果，あらかじめ決められている安全性の定義に照らして，「その定義を満たしているかどうか」によって安全であるかどうかを判断する．つまり，新たな暗号解読手法が見つかり安全性の定義を大きく下回る結果が導かれると，その暗号アルゴリズムの安全性が急激に低下したことを意味する．

ただし，安全性の定義や攻撃者の想定能力は理論的に起こりうる可能性がある最悪のケースを想定して決められていることが多く，その想定が現実的かどうかは問われないのが一般的である．そして，理論的にある解読手法が有効であり，安全性の定義を少しでも下回ることを示せれば「その暗号アルゴリズムが破れた」と暗号研究者は位置づける．つまり，少しでも脆弱性があるような暗号アルゴリズムは（学術的に）よくないと判断する．

もっとも，この段階では，ただちに現実的な被害を生じるような解読手法であるかどうかとは別問題であることに注意が必要である．

4.2 計算機能力の向上に対する安全性

この安全性は，「実際に計算機を回して暗号解読をすることができたかどうか」によって安全であるかどうかを判断する．

もちろん用意できる計算機能力は，政府・諜報機関レベルなのか，企業や大学の研究室レベルなのか，個人レベルなのかによって大きく異なり，それによって暗号解読できる範囲も違う．ただ，計算機で解読できたと発表するのは，ほとんどの場合，企業や大学（あるいはそれらが中心となったプロジェクト）であるため，ある程度の現実的なコストをかければ実際に解読できることの証左となる．このことから，政府・諜報機関などが本気になれば，企業や大学が用意できる計算機能力をはるかに上回る能力を投入でき，もっと簡単に暗号解読を実現できると考えられる．

また，解読に成功した事例をもとに，その解読に使った計算機環境やムーアの法則による計算機能力の向上などを加味して，安全性に関する中長期的な予測が出されることもある．この予測は，極めて効果的な暗号解読手法が新たに見つからない限り，かなり正確な推移を示す．

例えば，公開鍵暗号として代表的な RSA 暗号の安全性の根拠となっている素因数分解問題の困難性に対して，素因数分解が可能になるビット数を予測曲線として表したものと，実際に素因数分解をされたビット数をプロットしたものを図 7 に示す．この予測曲線は，2000 年に Lenstra や Brent らによって作られたもの[1][2]であるが，素因数分解問題に対して一般数体ふるい法より効果的な手法がいまだに見つかっておらず，現在でも計算機能力の向上に伴ってその予測通りに素因数分解可能なビット数が推移していることが分かる．

このことから，RSA 暗号の安全性が素因数分解問題の困難性に依拠している以上，図 7 は RSA 暗号の安全性が徐々に低下していくこと，逆にいえば同じ安全性を維持しようとすれば鍵長を伸ばしていく必要があることを示している．

5. 暗号技術にどのように向き合うか

5.1 暗号アルゴリズム 2010 年問題とは

汎用目的で利用する暗号アルゴリズムでは，安全性だけではなく，処理性能も設計上の重要な要求条件となる．したがって，開発当時の計算機環境における十分な安全性と処理性能のバランスを考慮してアルゴリ

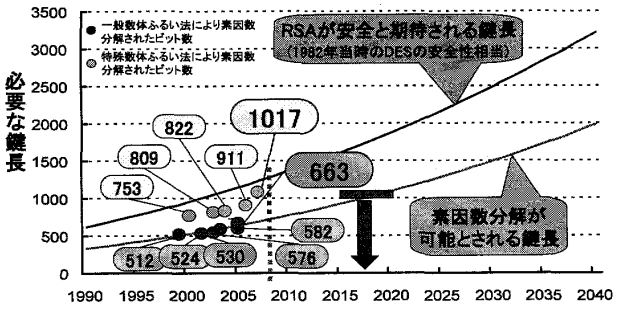

図 7 素因数分解問題の困難性に対する予測

ズムを設計するのが一般的である．

しかし，長期間使い続けるうちに計算機環境が大きく変わり，暗号解読に使える計算機能力が大幅に向上していく．その結果，ある暗号アルゴリズムに対する解読手法が現実的なものになってくることがある．

例えば，現在もっともよく使われているRSA暗号は鍵長が1024ビットのものであり，1990年代中頃から利用され始めた．当時の計算機環境では500ビットの素因数分解を行うことさえ難しいため，鍵長が1024ビットというのは十分な安全性を持つといえる．その一方，鍵長を2048ビットにしてしまうと，処理速度があまりにも遅くなりすぎたり，ICカードでは実装できないなどの性能上の問題を引き起こす．そのため，当時としては鍵長を1024ビットとしたことに合理性があった．

ところが，図7からわかるように，計算機能力の向上によって遅くとも2015～2020年頃に1024ビットの素因数分解が可能になると予測されている．つまり，今から10年たたずして鍵長1024ビットのRSA暗号が解読できるようになる恐れがあることを意味している．暗号技術監視委員会CRYPTRECが公表しているCRYPTREC Report 2007でもほぼ同様の調査結果を示しており，2010～2020年の間に素因数分解される可能性が高いとしている[3]．

これと同様のことが，共通鍵暗号やハッシュ関数でも起こってきている．例えば，衝突計算攻撃によるSHA-1の安全性低下[4]やMD5を利用した公開鍵証明書の偽造[5]，DESの鍵全数探索攻撃による解読，RC4の脆弱性を利用したWEPの解読[6]などが指摘されている．

しかし，これらの暗号アルゴリズムはいずれも現在広く使われているものであるが，規格化されたのは1970年代後半から1990年代前半頃である．つまり，20～30年前の計算機環境において十分な安全性を持つように設計されたものであるとはいえ，今後の利用を考えた場合，現在の計算機環境では十分な安全性を有しているとはいえず，長期的に利用するのは適切ではない．

米国商務省・国立技術標準研究所NIST（National Institute of Standards and Technology）は，2010年末を目途に，米国政府システムが使う暗号アルゴリズムについてより安全なものへの移行を進めることを公表し，そのガイドラインを示している（図8）[7]．このガイドラインに従って欧米企業では暗号アルゴリズムの世代交代への対応をすでに進めており，日本では「暗号アルゴリズム2010年問題」として紹介される基となった動きである[8][9]．

日本でも，内閣官房情報セキュリティセンターが2013年を目途とした「政府機関の情報システムにおいて使用されている暗号アルゴリズムSHA-1及びRSA 1024に係る移行指針」を公表している（表3）[10]．この指針はGPKIを基本とする政府機関情報システムを対象とするものだが，暗号世代交代に関する日本政府として初めての公式な移行指針といえるものである．

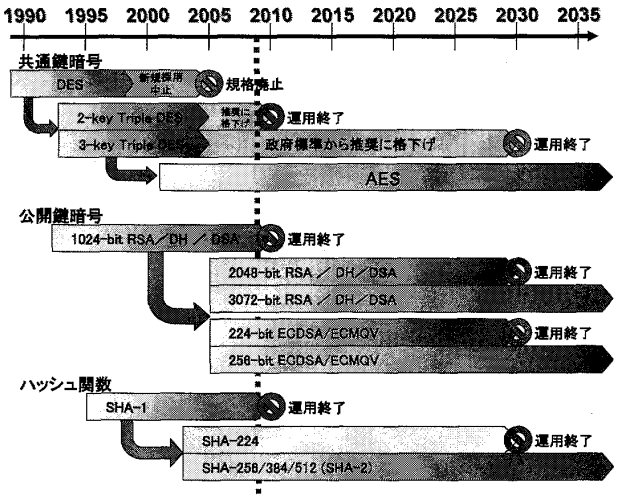

図8　NISTの暗号アルゴリズム移行ガイドライン

表3　政府機関における移行指針

	政府認証基盤(GPKI)及び商業登記認証局	GPKIに依存する情報システム	その他の情報システム
構成要件	電子証明書の発行・検証に使用する暗号を複数から選択可能	文書ファイルへの電子署名・検証に使用する暗号を複数から選択可能	別の暗号への変更が速やかに対応できる措置を事前に用意
用意する暗号	RSA2048 with SHA-1, RSA2048 with SHA-256を含める エンドユーザ用はRSA1024, RSA2048を含める	SHA-1, SHA-256, RSA1024, RSA2048を含める	複数の暗号を導入する際はSHA-256相当以上、RSA1152相当以上のものを含める
その他の要件	電子証明書発行は特定時期に切替可能 検証では開始・終了時期を設定可能	開始・終了時期を設定可能	必要なときのみRSA1024 with SHA-1を利用可能
	緊急避難的に、電子証明書の失効・再発行等を行うことで、業務継続性を確保		

5.2 暗号アルゴリズムの国際標準化

2000年以前，暗号アルゴリズムは武器の一種として輸出管理対象となっており，必ずしも世界中で自由に強度の高い安全な暗号が使えるわけではなかった．しかし，インターネットなどが普及し，オンラインビジネスが立ち上がるにつれ，安全な暗号がインフラ基盤として必要になってきた．

その流れを受け，ISO/IECでは安全な暗号アルゴリズムとしてISO/IEC国際標準暗号を策定した（表4）．また，インターネットの標準規格を決めるIETFでも，SSLやIPsec等のセキュリティプロトコルが作られたときに採用したRC4やTriple DESといった標準暗号よりも安全な暗号アルゴリズムであるAESやCamelliaを新たに追加している．

このほかにも，各国政府が自国政府の情報システムや重要インフラを保護することを目的に，使用すべき暗号アルゴリズムを強制的に指定する政府標準規格や，使用が強制されているわけではないが利用することが望ましい政府推奨規格を決める場合もある．前者の例としては米国政府標準暗号[11]や韓国政府標準暗号が，後者の例としては日本の電子政府推奨暗号[12]や欧州連合推奨暗号[13]がある．

6. まとめ

本稿では，暗号アルゴリズムについての概要と最近の動向について説明してきた．

暗号アルゴリズム2010年問題に代表されるように，安全性の観点から，暗号研究者らはより安全な暗号アルゴリズムへの世代交代を進めるようアナウンスしている．

ところが，システム提供者・ベンダからみると，暗号研究者らがいう“暗号解読”により実害が引き起こされる発生リスクや被害リスクが実際にどのぐらいあるか，リスク回避のために本当に移行する必要があるのか，あるいは世代交代に伴う移行コストを誰が負担するのか，といった問題を解決しなければ実際の世代交代はなかなか進まない．

その意味では，利害対立をもつ関係者間の合意形成を導くため，「(暗号アルゴリズムの選択・移行に対する）投資対リスク」問題の解法を見つけることは大きな課題といえる．こういった場面でOR的手法が使われ，課題解決に向けた大きな原動力になることが期待される．

表4 暗号技術に関する主なISO/IEC国際標準規格

種別	規格番号
暗号方式	ISO/IEC 18033 Encryption algorithms（Part 2：公開鍵暗号、Part 3：ブロック暗号、Part 4：ストリーム暗号）
デジタル署名	ISO/IEC 9796 Digital signature schemes giving message recovery ISO/IEC 14888 Digital signatures with appendix ISO/IEC 15946 Cryptographic techniques based on elliptic curves
ハッシュ関数	ISO/IEC 10118 Hash-functions

参考文献

[1] A. K. Lenstra and E. Verhaul, “Selecting cryptographic key sizes,” PKC 2000, LNCS 1751, Springer, 2004.

[2] R. P. Brent, “Recent progress and prospects for integer factorization algorithm,” COCOON 2000, LNCS 1858, Springer, 2000.

[3] CRYPTREC, “CRYPTREC Report 2007,” 2008.

[4] X. Wang, Y. L. Yin and H. Yu, “Finding Collisions in the Full SHA-1,” CRYPTO 2005, LNCS 3621, Springer, 2005.

[5] A. K. Lenstra and B. Weger, “On the possibility of constructing meaningful hash collisions for public keys,” ACISP 2005, LNCS 3574, Springer, 2005.

[6] 寺村亮一, 曽谷紀史, 仲神秀彦, 朝倉康夫, 大東俊博, 桑門秀典, 森井昌克, “WEPの現実的な鍵導出法（その2），” CSS 2008, 2008.

[7] NIST, “Recommendation for Key Management,” Special Publications SP 800-57, 2007.

[8] 宇根正志, 神田雅透, “暗号アルゴリズムにおける2010年問題について，” 金融研究25巻別冊第1号, 2006.

[9] 神田雅透, “デファクトスタンダード暗号技術の大移行，” アットマーク・アイティ, 2006.

[10] 内閣官房情報セキュリティセンター, “政府機関の情報システムにおいて使用されている暗号アルゴリズムSHA-1及びRSA 1024に係る移行指針，” 第17回情報セキュリティ政策会議, 2008.

[11] NIST, “Cryptographic Toolkit.”

[12] 総務省, 経済産業省, “電子政府推奨暗号リスト，” 2003.

[13] NESSIE, “Portfolio of recommended cryptographic primitives,” 2003.