ホームページの広場

第46回

「HPKIのPKIを支えるRSA暗号について−その1 特徴−」

市立貝塚病院　顧問　放射線治療科　医師　兼
地方独立行政法人大阪府立病院機構　大阪急性期・総合医療センター
放射線治療科　医師（在職出向）　上杉 康夫

1. RSA暗号の特徴と安全性

HPKI(Healthcare Public Key Infrastructure：保健医療福祉分野公開鍵基盤)はPKI(Public Key Infrastructure：公開鍵基盤)上で運用されています。そのPKIでは公開鍵が公開鍵暗号方式や電子署名方式で用いられており、さらには公開鍵とその公開鍵の持ち主との対応関係を保証するための仕組みとなっています※1。公開鍵暗号方式には様々な方式がありますが、ここではPKIの情報セキュリティの基になったRSA(Rivest-Shamir-Adleman)暗号の特徴と安全性について述べたいと思います。

2. データのコード化

コンピュータのデータは数値として保存され、それらを私たちは様々に活用しています。種々の文字や数字、記号は、方式の差こそありますが、コード化され全て数値に置き換えられてコンピュータで処理されています。

主な文字コードにはUTF-8、Shift JIS、EUC、JISがあります。表１は「1000円OFFクーポンをもらった」という文字列を、Shift JISコードの10進数表記にしたものです。したがってコンピュータでのデータの暗号化とは、数値データの暗号化になります。

3. 情報セキュリティの安全性の分類

社会学的安全性と数学的安全性があります。数学的安全性には「情報理論的安全性」、「計算量的安全性」があります(図１)※2。

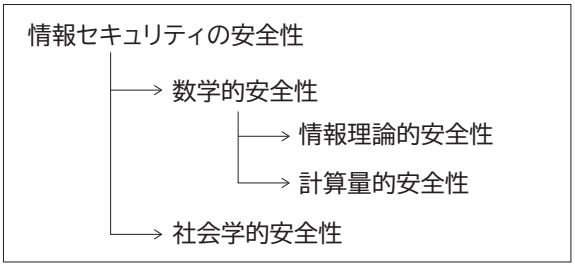

図１：情報セキュリティの安全性の分類※2

3-1. 社会学的安全性

まず、社会学的安全性は「人の一般行動に基づく安全性」を示すとしています。例としてパスワードの設定率があります。どのような高度の情報セキュリティを築いてもパスワードを設定しない人が大多数であれば、その効果は限定的です。

3-2. 数学的安全性

次に数学的安全性について述べます。

3-2-1. 情報理論的安全性

まず情報理論的安全性ですが、代表的な例としてはワンタイムパッド式暗号(以下ワンタイムパッド)があります。このワンタイムパッドは、1917(大正６)年にギルバート・バーナムが発案し、数学者クロード・エルウッド・シャノンによって安全性が示された暗号です。本方式は太平洋戦争で日本陸軍が利用したことで有名になりました。ワンタイムパッドは排他的論理和を用いて暗号化する暗号方式です。秘密鍵(暗号化をするための情報)は１回使用で破棄します。どれだけの暗号文を集めても、無限大の計算能力

文字	1	0	0	0	円	O	F	F	ク	ー	ポ	ン	を	も	ら	っ	た
Shift_JIS 10進数	049	048	048	048	137 126	079	070	070	131 078	129 091	131 124	131 147	130 240	130 224	130 231	130 193	130 189

表１：コード化
「1000円OFFクーポンをもらった」という文字列のShift JISコード化

をもってしても、解読できないとされています※3。メリットは暗号化も復号化も仕組みがシンプルです。乱数を用いることによって、暗号化の安全性が高くなっています。デメリットは暗号化前の平文と秘密鍵は同じ長さである必要があるため、平文が長くなると秘密鍵も長くなってしまうことです。送信者と受信者で事前に秘密鍵を共有しておく必要があります。上記の理由から安全性は高い一方で、手間がかかるため不便とされています。暗号文の解読は秘密鍵がなければ不可能ですが、実例として戦場で回収された秘密鍵による情報漏洩、本来１回使用で破棄する仕様であるはずの秘密鍵の使いまわしによる情報漏洩がありました。

3-2-2. 計算量的安全性

解読のための計算量が多項式時間に収まらない(膨大な時間がかかる)場合の暗号の安全性は、計算量的に安全という考えに基づいています。この安全性は、「計算量的安全性」と呼ばれています。その代表的な暗号がRSA暗号です。

このRSA暗号では、桁数が大きい合成数の素因数分解が現実的な時間内で困難であると信じられていることを安全性の根拠としています。RSA暗号方式は、1977(昭和52)年に発明され、発明者であるロナルド・リベスト(Ronald Linn Rivest)、アディ・シャミア(Adi Shamir)、レオナルド・エーデルマン(Leonard Max Adleman)の頭文字をつなげてRSAと呼ばれています※4。

ところで、注意すべき特徴はこのRSA暗号では解読に「膨大な時間がかかる」という点です。これを正確に表現するのであれば「現在知りうる知識や使いうる技術を駆使しても、膨大な時間がかかる」とすべきで、数学的新発見や「ムーアの法則」(ムーアの法則：参考 拙著「ムーアの法則(Moore's law)」ホームページの広場31 大阪医科大学医師会会報 第49号(平成30年3月)pp18-20※5)に代表される計算機能力の向上があれば、解読に「膨大な時間がかかる」と言うことは成立しなくなる可能性があります。

とは言え、現在推奨される条件を満たしたRSA暗号の解読には現在数十億年を要するとされており、まさに「膨大な時間がかかる」ために「計算量的安全」だとされている由縁です。

3-3. 社会学的安全性と数学的安全性の相互作用

社会学的安全性と数学的安全性とは相互に作用しあい、そのことが情報セキュリティの安全性を左右します。

身近なところでは　ワンタイムパスワードがあります。ワンタイムパスワードの導入は多くの利用者にとって煩雑ながらも使用をやめるほどのことではなく、使用率も高くなっています。その一方でワンタイムパスワードは攻撃者のやる気を減弱させる効果は大きいとされています。この例は、使いやすい数学的安全性を導入すれば、社会学的安全性をより強固にする可能性があります。

その一方で、パスワードは長ければ安全と一般に考えられています。現在その長さは最大14文字から16文字の間が指定されることが多いですが、これが100文字とか200文字必要となれば、ルールを守る人は少なくなり、安全性が低下することは充分に予想されます。

これは先ほどとは逆に数学的安全性の強固が社会学的安全性の脆弱を導く例とされています。すなわち「パスワードを長くすること」が「パスワード設定を煩わしく感ずる使用者が増えて設定率が低下すること」を生み出す可能性があります※6。

この点PKI ではRSA暗号の技術を利用し高度なセキュリティが実現されている一方、ユーザーが意識することなくそれらを使いこなせるアプリが多く存在し、有用な情報セキュリティとされています※7。

4. RSA暗号の目指したところ

暗号化と復号化に共通鍵を使用する方式では逆演算が基本的に可能であり、共通鍵の配布と機密性の保持が困難でした。そこでこの発明者3人は、暗号化と復号化を異なる鍵で行うようにし、暗号作業から逆演算をなくした周期的な一方向性の演算と、さらに素数を使った暗号の

解読を困難にする新しい方式を考案しました。

後述するmod演算には結果の剰余だけからでは元の数を割り出すことは不可能という特徴があり、逆演算が無い一方向性関数の代表的なものとされ、この特徴を利用して暗号の逆演算を不可能にしました。

さらに整数論の業績として冪剰余(詳細 5-A-2.で後述)に周期性があることが、本暗号発明当時知られていましたので、これも利用しました。

そして冪剰余の周期は、割る数を素因数分解した数の性質で決まる(詳細 5-A-2.で後述)ので、この素因数分解を困難にすることで暗号の解読を困難にすることを考えつきました。

このことにより、一方向関数としてのmod演算、冪剰余の周期性、解読困難性としての素因数分解の困難性、この3つを有した強固な暗号理論としてRSA暗号は完成されました※8。

これには、桁数が大きい合成数の素因数分解が現実的な時間内で困難であると信じられていることが安全性の大きな根拠となっています。
さらに具体的に以下に記載します。

5. RSA暗号の3要素について

5-A-1. 一方向関数とmod演算(剰余演算)

数学にはガウスが発明した mod演算(剰余演算)というものがあります。これはある数字で除算(割り算)した時の剰余(余り)に着目するものです。もともとは整数論として確立されました。例えば50を7で除算した時の剰余は1、324を7で除算した時の剰余は2ですから

50 mod 7 = 1

324 mod 7 = 2

と表現します。割る数を「法」と呼びます。

このmod演算の一方向性について観てみます。8 mod 5 = 3、そして11 mod 5 = 1です。これらの演算は容易です。ですが逆演算はどうでしょうか? mod演算の結果が3であったとしても、元の数はわかりません。逆演算は不可能です。なぜならx mod 5 = 3のxは…8,13,18, …で、一般に$x=5\times n+3$(nは整数)となり、無限に存在するからです。

mod演算(剰余演算)は一方向関数の代表的なものとされています。このmod演算の一方向性の特徴がRSA暗号では利用されています。

5-A-2. 冪剰余の周期性

正の整数bの整数e乗(冪乗)を正の整数m(法)で割った余りを、「mを法とするbのe-冪剰余」と呼んでいます。数式で表現しますと

$C=b^e$ mod m のCが相当します。

フェルマー、オイラーやガウスの研究によって、冪剰余には周期性が発見され、その特に顕著な成果はガウスの整数論※9で発表されています。

それら成果によると冪剰余の周期は存在し、またこの冪剰余の周期は、ある数mが素数pとqとでm=p×qならば、mを法とした時に、冪剰余に現れる周期は、p-1とq-1との最小公倍数になることが発見されていました。この冪剰余の周期性の特徴がRSA暗号では利用されています※10、11。

5-A-3. 素因数分解の困難性

巨大な素数の積からなる合成数の素因数分解は、極めて困難とされています。RSA暗号では冪剰余の周期性を決める法mの決定に巨大な素数の積である合成数を使い暗号の計算量的安全性を担保しています。この巨大合成数の素因数分解の困難性の特徴がRSA暗号の暗号化と復号化とに利用されています※12、13。

5-B. RSA暗号の3要素の概念図

図2にRSA暗号の3要素の概念図を示します。mod演算を用いた冪剰余を使っていますので、mod演算の一方向性のため逆演算は存在しないとして、図では反時計回転方向だけに進みうる周期性をもつ関数として描いています。

重要なのはmodの法mです。このmは、2つの相異なる奇素数pとqとの積である合成数(m=p×q)で定められます。そしてm=p×qならmを法とするbのe冪剰余(b^e mod m)の周期Sはp-1とq-1の最小公倍数に決まります。そして暗号として利用するために、この周期Sを秘密にします。

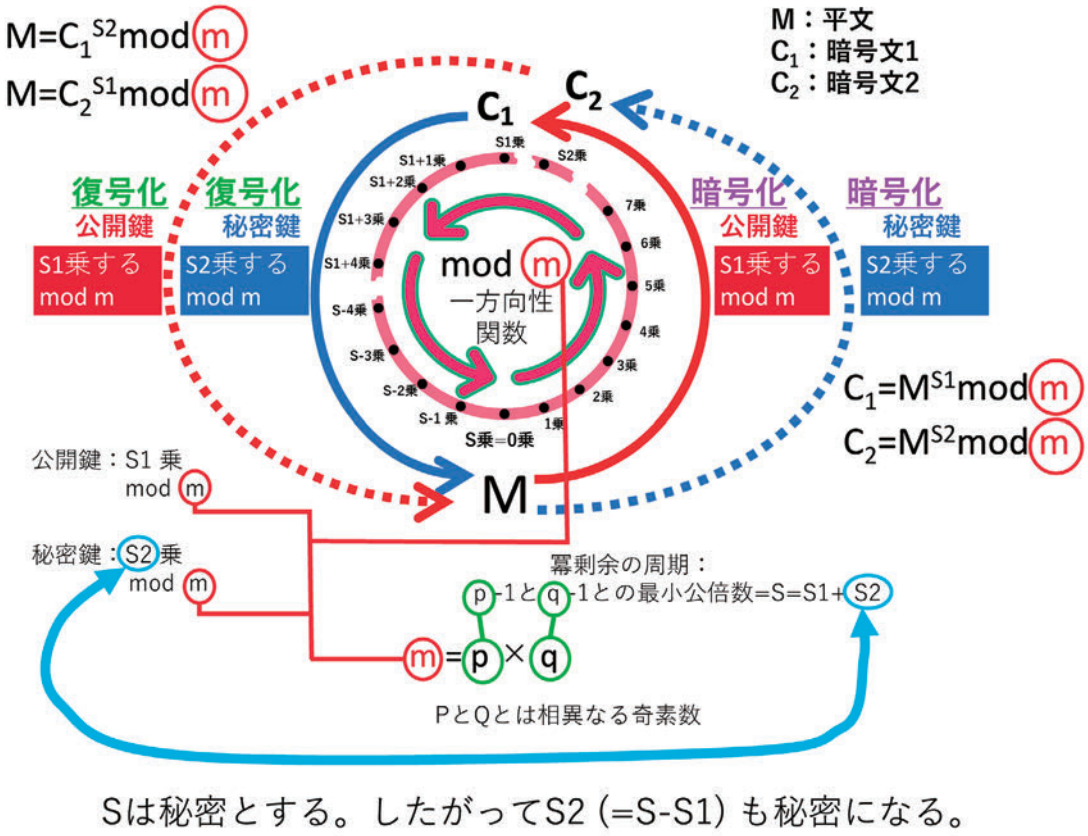

図2：RSA暗号の概念図
重要なのはmodの法mである。このmは、2つの相異なる奇素数pとqとの積である合成数(m=p×q)で定められている。そしてm=p×qならb^e mod mの冪剰余の周期Sはp-1とq-1の最小公倍数に決まる。mの素因数分解を極めて困難としておけばpとqは計算量的安全性が成立し、その結果周期が求められないこととなり、解読を阻止している。
またmod mは一方向性関数なので、逆演算は不可能である。本図を用いて説明すればmod mによって反時計回転方向にのみ演算可能で、時計回転方向の逆演算は不可能となっており、逆演算を使う暗号解読を不可能にしている。
周期性を利用して、①M→公開鍵で暗号化→C1→秘密鍵で復号化→M(実線)と、②M→秘密鍵で暗号化→C2→公開鍵で復号化→M(破線)の2通りの経路が成立する。

図の平文M→暗号文C1→平文Mを例にして述べますと、次いで暗号化するために任意の冪乗の数S1を決めます。復号化するには、冪剰余の周期性を利用して、図で例えると元の位置に周回して戻るような冪乗数S2を決めます。すなわちS=S1+S2となるようなS2を求めることになります。その結果S2=S-S1となりますのでSを秘密にしていますからS2も秘密になります。

こうして公開鍵としてS1とmを公開し、秘密鍵としてS2を秘密で保持します。

以上で平文M の暗号化はM^{S1} mod mを計算することで達成し、暗号文C_1ができます。C_1から元の文に復号するためには、mと秘密鍵S2とでC_1^{S2} mod mを計算し周期を進めて元の平文Mにします。この場合、復号はS2を知らなければできません。S2を秘密にしていますから、解読は阻止されます。

また、暗号化と復号化とは逆演算の関係にあるのではなく、M→C_1→Mのように平文→暗号文→平文の関係で、何周目でもよいのでMではない他の冪剰余を暗号文として暗号化し、暗号文から冪乗を繰り返し周回して、何周目でもよいので元の平文Mの位置にくることで復号化が達成されています。

RSAの特徴は、公開鍵と秘密鍵の1組を使うことにより、平文→公開鍵→暗号文→秘密鍵→平文で周回して元の位置に戻るようになっていますから、秘密鍵で暗号化した場合、公開鍵で復号化できても、秘密鍵で復号できない理由

です(図2)。よく見るキャッチコピー「鍵Aで暗号化したら、鍵Bでしか復号できない。鍵Bで暗号化したら、鍵Aでしか復号できない」※14はRSA暗号が上記の冪剰余の周期性を利用しているためにM→公開鍵で暗号化→C1→秘密鍵で復号化→MとM→秘密鍵で暗号化→C2→公開鍵で復号化→Mの2通りの経路が成立すると言う特徴を示しています。

6. 解読に関する数学的な容易さと計算量的安全性

RSA暗号では、法mは公開を前提にして、暗号の送信側に教えます。ところで冪剰余の周期Sは法mを素因数分解してm=p×q とできる2つの奇素数pとqとが求まれば、p-1とq-1の最小公倍数としてSを求めることが可能です。ということはmを素因数分解して所定の計算をすれば、難なくSが分かるはずです。Sがわかれば、暗号化の乗数S1は公開されていますから、復号化する乗数S2はS2=S-S1と計算して求まり、復号可能になります。数学的には、何の問題もなく、このRSA暗号は解読可能ということになります。

しかしながら、数学的に可能というのは、有限回の操作で解決可能ということを示しているだけで、計算に何時間かかるかは考慮されていません。ここまで読まれた方はもうお気づきと思います。暗号解読を目的として周期Sを知るためには「法mを素因数分解する」操作が必要でした。RSA暗号ではこの素因数分解を極めて困難にして周期の解読を阻み計算量的安全性を担保しているのです。

素数の分布の規則性は充分には解明されておらず、ある数がどの素数で割り切れるかを事前に予測する簡単な方法の存在は証明されていません。したがって総当たりで割り算を試していく方法しかなく、数が大きくなると試すべき素数の数も爆発的に増えます。このため、膨大な計算時間が必要になります。

RSA暗号は先の章で記載した計算量的安全性が保たれている良い例です。

7. RSA暗号と素因数分解の世界記録

RSA暗号での暗号鍵の長さ(鍵長)は、2048ビットや4096ビットが現在主流です。2048ビットは10進数で約617桁、4096ビットは10進数で約1234桁になります※15。

RSA Securityは1991(平成3)年から2007(平成19)年までRSA Factoring Challengeと称するコンテストを開催していました。このRSA Factoring Challengeでは様々な桁数の数が与えられ、その数の素因数分解が求められました。RSA鍵の解読の実際的な難しさを促進することを目的としていました。それぞれの数には、その難易度に応じて賞金が授与され2007(平成19)年に公式のコンテストは終了しましたが、その時点で8万ドルを超える賞金が授与されていました。コンテストの公式終了後も、残されたデータは数論研究者によって因数分解法のさらなる発展のために利用され続けています※16、17。

一例として数RSA-250と数RSA-260とについて記載いたします。RSA-250は10進数250桁(829ビット)で、素因数分解の発表は2020(令和2)年2月28日に行われました。次にはRSA-260がありますが10進数260桁(862ビット)で、本稿記載時点で素因数分解は、なされていません※18、19。

RSA暗号での暗号鍵の長さ(鍵長)の主流が、10進数約617桁(2048ビット)や10進数約1234桁(4096ビット)ということであれば、前述の素因数分解の結果をみても、RSA暗号に使われているmod演算の法mの素因数分解の困難性は十分であり、RSA暗号の計算量的安全性が担保されていると判断されます。

RSA暗号の解読の困難さの本質は『周期が分かる場合、つまり素因数分解ができる場合は、元に戻す乗数を算出できるが、素因数分解ができない場合は、周期が分からず、元の値に戻す乗数の算出はまず不可能である。』という素因数分解の成否による点にあると思われます。

8. 計算量的安全性は永遠なのか?

現在推奨される条件を満たしたRSA暗号の解読には現時点で数十億年かかるとされており、

またさらに計算機の計算能力の増大はある程度予測可能であるため、新たな暗号標準を採用することによって、計算機能力の増大に備えた十分なマージンを持たせ、暗号解読が所定時間内に現実化されることのない値を選択するといった対策がなされています。

しかしながら、新しい数学理論が見出されることや新型のコンピュータが登場し計算機能力は強化されることによって、膨大な時間がかかることはなくなる可能性があります。

これらの現実化は、計算量的安全性で安全を担保している暗号への脅威となりえます。なぜなら、現時点でも計算時間を無限にすれば、原理的には計算量的安全性で安全を担保している暗号は全て解読可能だからです。

また、そのような脅威になり得ることがいくつか報告されています。

8-1. リーマン予想(Riemann hypothesis)

数学理論であるリーマン予想には、アメリカのクレイ数学研究所が設定した100万ドル(約1億数千万円)の懸賞金※20が懸けられています。

リーマン予想とは「ゼータ関数ζ(s)が0になるようなsは、全てs=1/2 + i×t (tは実数)となるだろう」とのことです。

ドイツの数学者リーマンは自分の研究の中で、ある自然数N以下の素数の個数を計算する公式を作っていました。

$$N\text{以下の素数の個数} = \frac{N}{\log N} + (\text{ゼータ関数のゼロ点で書ける式})$$

正確な式は省略しますが、ざっくり上の式のような感じです。

ゼータ関数の情報がなくとも、N/logNだけで大雑把には素数の個数を数えることができますが、どうしても誤差が出てしまいます。その誤差を解消するのに「ゼータ関数のゼロ点」の情報が必要になるわけです。

このゼロ点にきれいな性質があると分かれば、この誤差の式が書けるようになる⇒素数の個数を正確に数えられる⇒素数の規則性が分かる。

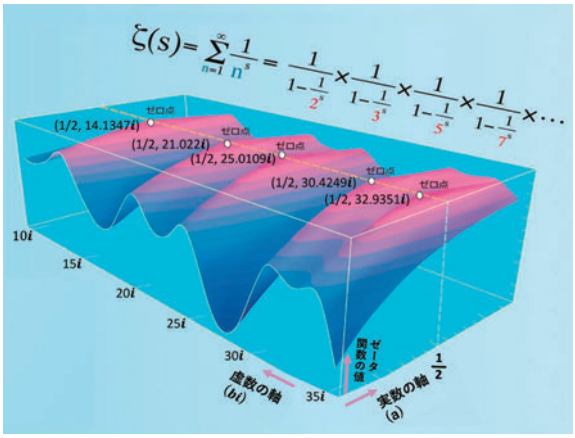

図3：リーマン予想の概念図 – ゼロ点は全て、一直線上に並ぶ※21、22、28
1859年、リーマンは、「ゼータ関数ζ(S)の虚数領域のゼロ点(○)は、無限に存在し、全て一直線上に並んでいる。直線の位置は、実数部分が1/2となる場所である。」と予想した。ゼータ関数ζでの1-5番目のゼロ点を示す(※21の図を一部改変)。

そしてそのゼロ点は図3※21、22、28のように「ゼータ関数を使って非自明なゼロ点は全て、一直線上に並ぶはずだ※23、24。」というのが、リーマン予想です(図3)。

素数の規則性が判明してしまうとRSA暗号はより簡単に解ける可能性があります。したがって「もしリーマン予想が解かれて素数の規則性が判明してしまうと、素数を使っている RSA暗号は簡単に破られてしまう! 通信の安全性の危機だ!!」は煽りすぎですが、解読がより簡単になるのは確実です※25。

現実にはリーマン予想が正しいとされても、あくまで、そして、かなり誇張しても「素数がどういう規則で並んでいるか」ということだけですので、RSA暗号の安全性を担保している「素因数分解の困難さ」は、素数の並び方の法則とは別に、「現実的な計算時間で効率よく素因数分解を行えるアルゴリズム」が開発されない限りRSA暗号の安全性は破られません。

とは言え、リーマン予想の解決が全くRSA暗号の安全性を脅かさないというわけでもないです。素数の規則性が分かれば、上記の「試し割り」に使う素数の候補を厳選することが可能となり、計算回数を緩和できるようになると思われるからです。

8-2. 量子コンピュータ

RSA暗号は、巨大な整数の素因数分解の困難性を安全性の根拠とする公開鍵暗号方式です。特にRSA-2048の規格では、2048ビット長の合成数(10進数で約600桁)を利用しており、現代のスーパーコンピューターをもってしても、既存の素因数分解アルゴリズムでは現実的な時間内での素因数分解は困難です。しかし、もし非常に大規模な量子コンピュータが実現すれば、飛躍的に高速な素因数分解アルゴリズムが実行可能になると予想されています。

では、RSA-2048を解読するためには、どの程度の規模の量子コンピュータが必要になるのでしょうか?

2019(令和元)年、GidneyとEkeråは、誤り率0.1%以下の2000万量子ビットを持つ量子コンピュータがあれば、RSA-2048を8時間で解読できるという推定を発表しました[※26]。

このような状況となればRSA暗号の計算量的安全性への脅威は、無視しえなくなると思われます[※24、25]。

9. 思い出

RSA暗号の存在は、小生が大阪医科大学の2年生であった1980(昭和55)年にコンピュータ雑誌のbit[※27]で初めて知りましたが、理解しようと挑戦するもできませんでした。

その頃は、Youtubeもインターネットもなく、WordもExcelもなく、キットを半田ごてで作ったZ80のCPUでRAMを20KB、ディスプレイはモノクロの10型CRTでASCII文字と半角カナ表示で、カセットテープが記憶媒体の8ビットパーソナルコンピュータMZ-80Kを所有していた時代です。市販ソフトはほとんどなく、自前でソフトを作成していました。冪乗表を作るのも煩わしく、また資料を集めようにも、文献検索は手動、コピーで取り寄せ、用語についても調べるのに一苦労でした。その後、何度か挑戦しましたが、できませんでした。

現在では、情報工学の発展によって国内外の情報検索も驚異的な桁数の冪乗計算も瞬時で可能となりました。RSA暗号を理解するためのいくつかの課題も何とかこなせて何とか文章になりました。

いつかはあの雲の上に聳え立つ峰にという願いが、必ずしも充分とは言えないにせよ何とか叶えられた思いがします。

しかし、RSA暗号はまさに大山であり、会報1号では記載尽くすことができない量があり、次号以降でもRSA暗号の計算例等について引き続き記載いたしたいと思います。

今回はRSA暗号の特徴について記載いたしました。

素因数分解の困難さが、RSA暗号の安全性を支えています。

参考文献

※1：公開鍵暗号 - Wikipedia
https://ja.wikipedia.org/wiki/公開鍵暗号

※2：社会学的安全性と数学的安全性
https://www.jnsa.org/jnsapress/vol30/2_tokusyu.pdf

※3：計算量的安全性を持つ暗号 - Wikipedia
https://ja.wikipedia.org/wiki/計算量的安全性を持つ暗号

※4：RSA暗号 – Wikipedia
https://ja.wikipedia.org/wiki/RSA暗号

※5：ホームページの広場31：「ムーアの法則」大阪医科大学医師会会報 第49号(平成30年3月)pp18-20
https://www.ompu.ac.jp/u-deps/ompuda/report/pdf/report_49_p18-20.pdf

※6：強力な暗号「ワンタイムパッド」とは? miracleave Tech Blog
https://www.miracleave.co.jp/contents/1958/post-1958/#index_id3

※7：公開鍵暗号方式とは?仕組みやメリットなどをわかりやすく解説 | セキュリティ NOW! サイバーセキュリティ情報ポータルサイト
https://www.sms-datatech.co.jp/securitynow/articles/blog/sec_public-key-cryptography/#公開鍵暗号方式の身近な活用事例

※8：暗号解読(下)(新潮文庫) Kindle版 サイモン・シン (著), 青木薫 (翻訳) 新潮社 (2007/7/1) 第Ⅵ章　アリスとボブは鍵を公開する PP149-175

※9：Disquisitiones Arithmeticae - Wikipedia
https://ja.wikipedia.org/wiki/Disquisitiones_Arithmeticae

※10：【図解】素数とRSA暗号/署名の仕組み ~わかりやすい計算例とシーケンス,アルゴリズム~ | SEの道標
https://milestone-of-se.nesuke.com/nw-basic/tls/rsa-summary/

※11：基礎から学ぶ整数論-RSA暗号入門 長嶋 祐二 ,福田 一帆. P5 コロナ社 (2020/9/25)

※12：RSA暗号の仕組み
https://kobayashi.hub.hit-u.ac.jp/topics/rsa.pdf

※13：公開鍵暗号(RSA暗号)解説その2　仕組みについて
https://ndk.co.jp/wp-content/uploads/2023/11/tp_20231102-2.pdf

※14：電子署名と認証の役割
https://www.jmaca.med.or.jp/hpki/role/

※15：暗号鍵はどれくらいの長さが必要?伸ばすメリット・デメリットを解説 | ITトレンド
https://it-trend.jp/encryption/article/64-0088

※16：RSA Factoring Challenge - Wikipedia
https://en.wikipedia.org/wiki/RSA_Factoring_Challenge

※17：RSA Factoring Challenge – Wikipedia (de)
https://de.wikipedia.org/wiki/RSA_Factoring_Challenge

※18：RSA numbers - Wikipedia
https://en.wikipedia.org/wiki/RSA_numbers#RSA-250

※19：Números RSA - Wikipedia, la enciclopedia libr
https://es.wikipedia.org/wiki/N%C3%BAmeros_RSA

※20：【未解決問題(懸賞金1億円超)】解けぬ謎と挑戦の先にある学び～ボトムアップ的学習アプローチ～|松田達哉 | AI×教育企業CEO
https://note.com/tatsuyamatsuda/n/nfffb23c8e0a8

※21：Newton 素数のふしぎ Kindle版　株式会社ニュートンプレス (2015/11/26)位置No.20/22

※22：courtesy of Wolfram Research, Inc.
http://wolfram.com

※23：中学数学で習う「最も基本的な数」はナゾだらけ…天才数学者たちを悩ませ続ける「素数」の未解決問題 このナゾが解ければ「宇宙の真理」に近づく (2ページ目) | PRESIDENT Online(プレジデントオンライン)
https://president.jp/articles/-/76528?page=2

※24：リーマン予想って何だろう? - ちょぴん先生の数学部屋
https://stchopin.hatenablog.com/entry/2020/03/13/202050

※25：RSA暗号 ～素数は世の中でメチャクチャ役に立っている～ - ちょぴん先生の数学部屋
https://stchopin.hatenablog.com/entry/2022/07/21/144051

※26：量子コンピュータでRSA-2048を解くためには? (2025年5月版)
https://zenn.dev/nttdata_tech/articles/df651ec88cb08d

※27：A. Lempel: Cryptology in transition, Computing Serveys, 11(4), 1979, 285-303 〔西村和夫　訳:暗号学の変遷、コンピュータサイエンス, bit別冊,共立出版,1980, 109-125].
https://dl.ndl.go.jp/pid/12627352/1/61

※28：ZetaZero[{1, 2, 3,4,5,6}] -- N - Wolfram-Alpha
https://www.wolframalpha.com/input?i=ZetaZero[{1, 2, 3,4,5,6}] // N&lang=ja