RSA 暗号の仕組み

村上（むらかみ）　仙瑞（せんずい）

§1. はじめに

整数の単元（数学A）の発展的な内容として，合同式と素数を用いたRSA暗号の解説を行った。このRSA暗号とは現代のネットワークセキュリティに欠かせない暗号技術で整数の性質が使われている。暗号の仕組みは極めてシンプルである。また，生徒の整数問題に対する興味づけのために格好の材料でもある。ただ，暗号の仕組みの証明はいろいろな書籍で紹介されているが，きわめて専門的で複雑である。しかし，証明の核心部分だけみればきわめてシンプルで，積の合同式の性質，不定方程式の整数解と解をもつための条件，フェルマーの小定理の3つだけであり，青チャートでも解説されている。合同式を理解するにはちょうどよい教材の内容となる。このレポートでは私が考えたシンプルな教科書の発展的レベルでおさまる証明を紹介する。

§2. 証明に必要な3つの整数の性質と定理

1. $a \equiv b \pmod{n} \Rightarrow ac \equiv bc \pmod{n}$
2. p が素数，x が p の倍数でないとき，
   $x^{p-1} \equiv 1 \pmod{p}$（**フェルマーの小定理**）
3. a と b が互いに素であるならば，不定方程式
   $ax+by=1$ は整数解をもつ。

この3つの条件だけでRSA暗号の仕組みを証明できる。

§3. RSA暗号の仕組みと証明

送信者Aから受信者Bにデータを送信するとき，第三者にデータが盗まれる可能性がある。だから暗号化してデータを送る必要がある。暗号化の仕組みは次のとおりである。

まず受信者Bが，2つの素数 p，q を用意して（**この2つの素数は秘密にしないといけない**），自然数 $n=pq$ と，$(p-1)$ かつ $(q-1)$ と互いに素な自然数 s を考える。この s の互いに素の条件は，証明の最後の方の式(3)で使う。また，それぞれ $p-1$，$q-1$ を考える理由は，後ほど出てくるフェルマーの小定理を使うためである。

受信者Bは，送信者Aに自然数 n と s を伝え（この n と s の数は公開しているので第三者にみられてもかまわない），Aが用意した値 a を s 乗して n で割った余りの値 b をBに伝えるように願える。このとき，送信者Aのデータ a に対して

$$a^s \equiv b \pmod{n}$$

が成り立っている。またこの操作を**暗号化**という。

次に暗号化された値 b を受信者Bが受け取り，ここである条件を満たす自然数 x を考え，x 乗して n で割った余りが送信者Aの元データ a となり，元データを受け取ることができる。

つまり，

$$b^x \equiv (a^s)^x \equiv a \pmod{n}$$

が成り立つ x，つまり，

$$a^{sx} \equiv a \pmod{n} \qquad \cdots\cdots(1)$$

を満たす x を求めればよい。

さて，受信者Bはどのようにして自然数 x（これを**復号鍵**という）を見つけるのだろうか。この x のみつけ方であるが，p，q が素数のとき，任意の自然数 a，y に対して，

$$a^{(p-1)(q-1)y+1} \equiv a \pmod{n} \qquad \cdots\cdots(2)$$

が成り立つことを利用する。まずこれを証明する。

(i) x が p，q と互いに素のとき

x は $n=pq$ と互いに素，

フェルマーの小定理より　$a^{p-1} \equiv 1 \pmod{p}$

すなわち　$(a^{p-1})^{q-1} \equiv 1^{q-1} \pmod{p}$

よって　$a^{(p-1)(q-1)} \equiv 1 \pmod{p}$

また，p と q を入れ替えても同様のことがいえて

$a^{(p-1)(q-1)} \equiv 1 \pmod{q}$

つまり，$a^{(p-1)(q-1)}-1$ は，p の倍数でもあり，q の

倍数でもあるから，$a^{(p-1)(q-1)}-1$ は $n=pq$ で割り切れるので，$a^{(p-1)(q-1)}\equiv 1 \pmod{n}$ がいえる。

よって，$(a^{(p-1)(q-1)})^y\equiv 1^y \pmod{n}$，

$a^{(p-1)(q-1)y}\equiv 1 \pmod{n}$ となり，両辺に a をかけて

$a^{(p-1)(q-1)y+1}\equiv a \pmod{n}$

(ii) a が p の倍数であり，q と互いに素のとき

フェルマーの小定理より　$a^{q-1}\equiv 1 \pmod{q}$

すなわち　$(a^{q-1})^{p-1}\equiv 1^{p-1} \pmod{q}$

よって　$a^{(p-1)(q-1)}\equiv 1 \pmod{q}$

両辺を y 乗して a をかけても，

$a^{(p-1)(q-1)y+1}\equiv a \pmod{q}$ が成り立つ。ここで，

$a^{(p-1)(q-1)y+1}-a$ は，a が p の倍数でもあるので，

p の倍数であり，q の倍数でもある。

よって，$n=pq$ でも割り切れるので

$a^{(p-1)(q-1)y+1}\equiv a \pmod{n}$

(iii) a が p かつ q の倍数のとき

一般に，s が m の倍数であるとき，任意の自然数 u に対して，$s^u\equiv s \pmod{m}$ が成り立つ。

よって，a は $n=pq$ の倍数であるので，

$a^{(p-1)(q-1)y+1}\equiv a \pmod{n}$ が成り立つ。

以上(i)から(iii)より，式(2)が証明された。

複号化をするための自然数 x の値は，(1)を満たす x であるが，証明した式(2)を利用して，

$$a^{sx}=a^{(p-1)(q-1)y+1}$$

を満たす x の値を求めればよい。

つまり，

$$sx=(p-1)(q-1)y+1$$

が成り立ち，つまり，

不定方程式　$sx-(p-1)(q-1)y=1$　……(3)

を満たす x の値を求めればよいのである。この不定方程式が解をもつための条件は，**x の係数 s と y の係数 $(p-1)(q-1)$ が互いに素である必要がある。**最初に s を $(p-1)$ かつ $(q-1)$ と互いに素であるようにしたのもこのためである。

以上で RSA 暗号の仕組みが解説できたので，次は具体的な数字で本当に成り立つか考えてみる。

§4. 誕生日をあてよう

たとえば，送信者Aさんの誕生日 8 月 31 日を $a=831$ と表すことにする。受信者Bさんは，2 つの素数 $p=59$ と $q=167$ を用意して，

$n=59\cdot 167=9853$ と $p-1=58$ と $q-1=166$ に互いに素な自然数の 1 つ $s=11$ をつくり，送信者Aさんに A さんの誕生日を 11 乗して 9853 で割った余りをBさんに伝えるようにいう。これを式で表すと，

$$831^{11}\equiv 8752 \pmod{9853}$$

つまり，B さんは 8752 をA さんから受け取る。

ここでBさんは，(3)を利用して，

不定方程式　$11x-9628y=1$

を解く。この解は n を整数とすると，

$x=9628n+6127$，$y=11n+7$ を得るので，x として $x=6127$ を選び，A さんから受け取った 8752 を 6127 乗して 9853 で割った余りを考える。すると，831 を得て，A さんの誕生日が 8 月 31 日であることがわかる。式で表せば，

$$8752^{6127}\equiv 831 \pmod{9853}$$

となる。

さて，この一連の流れで，どこで秘密性が保たれているかというと，実は最初に考えた 2 つの素数 p と q である。この 2 つの素数は公開せず，受信者Bは秘密にしておかないといけない。送信者Aに 2 つの素数の積 $n=pq$ を公開して伝えるので，第 3 者は n を 2 つの素数に素因数分解すれば暗号が解読できるとわかっているのであるが，実際に使われている 2 つの素数は 200 桁で，2 つの素数の積 n は 400 桁，この 400 桁の自然数を 2 つの素数の積に分解することは，現代の一般のコンピューターでは一生かかるらしい。だから，ネットで検索してすぐに出てくるような 2 つの素数ではセキュリティが守られない。そうした現在のコンピューターの性能事情を利用した暗号システムが RSA 暗号なのである。逆に，2 つの巨大な素数がわかっていると，積 pq は計算することは簡単にできる。積は簡単だが，素因数分解は難しい。そうした計算事情を活用しているのである。**だからあるセキュリティ会社では，巨大な素数一覧表を厳重に金庫に保管している。素数はネットワーク社会の財産なのである。**

もし素数の規則性がわかってしまうかコンピューターの性能が飛躍的にのびると，素数一覧表を作ることができる。そうすると簡単に巨大な自然数 n を 2 つの素数に分解できるようになり，現在のセキュリティシステムが一気に崩壊してしまうという危険性もはらんでいる。それが現在の暗号システムなのである。素数の規則性の発見やコンピューターの性能の向上は誰もが望んでいることだが，それができ

てしまうとセキュリティシステムが崩壊する。なんとも矛盾にみちた悩ましい問題なのである。

§5. WOLFRAM CLOUD を活用する

最後に WOLFRAM CLOUD を紹介する。最近までは,『Mathematica』といえば高価な数学ソフトであったが, 時代の流れからか, Web 上でかなりの計算をフリーで利用することができる。今回, §4. の計算は, すべて

https://ja.wolframalpha.com/

にアクセスして行った。

たとえば, 不定方程式 $11x-9628y=1$ を解くのも, ページの入力フォームに式 $11x-9628y=1$ と入力して, Shift+Enter キーを押すだけである。Mathematica で計算するときは **Shift キーを忘れてはいけない**ことに注意する。また, 8752 を 6127 乗して 9853 で割った余りを計算するときは, 入力フォームに Mod[8752^{6127}, 9853] と入力して Shift+Enter キーを押すだけで, 瞬時に 831 と計算してくれる。単純に 8752^{6127} の計算結果も興味があるのであれば, 入力フォームに 8752^6127 と入力して Shift+Enter キーを押す。すると, 巨大な数字の計算結果が現れる。自然数 5673215567942 の素因数分解の結果を求めたければ, FactorInteger [5673215567942] と入力して, Shift+Enter キーを押すと簡単に, 2×13×157×5813×239087 と表示される。

ちなみに家庭用コンピューターで簡単に計算できる数字は RSA 暗号で用いてはいけない。

§6. 最後に

最近, 大学での純粋数学分野が直接社会に役立たないという理由で予算が削られていくという話をよく聞く。よく考えてみていただきたい。この RSA 暗号の仕組みは, フェルマーの小定理という名前からもわかるようにフェルマーが生きた時代(1600 年代前半)の純粋数学分野が, 1980 年に応用され暗号システムとして開発された。つまり, 直接社会に役立たなさそうにみえる素数の性質などが約 400 年経ってネットワーク時代に応用され欠かせないものとなったというのである。つまり, いまは直接役に立たないかもしれないがいつどこでそれが応用されるかわからないという可能性も残しているのである。目先のことばかり考えるのではなく, 基礎研究も大事であるということを RSA 暗号の仕組みを解説するときは生徒に説いている。実は私が RSA 暗号の授業を行うに当たってこのことが一番いいたかったことである。