In Laboratory Now

研究室訪問 3

現実的状況を考慮した暗号へ

田中 圭介 研究室〜数理・計算科学専攻

田中 圭介 准教授　　安永 憲司 助教

田中先生らの研究室では暗号理論の研究が行われている。暗号理論とは、暗号の計算手順や実際の利用手順、およびそれらの安全性などを研究する分野である。その研究範囲は広く、電子的に秘密のやりとりをするものは全て暗号理論に入ると考えてよいだろう。私たちがインターネット、金融などのシステムを安心して使えるのも、この研究分野があってこそといえる。

本稿では先生らの研究の中から、暗号の理論的研究が始まるきっかけとなった、秘密分散と公開鍵暗号に関するものを、それぞれ紹介する。

現実的な状況下での秘密分散

分割と復元の公平性

田中先生らが研究している暗号システムの一つに秘密分散というものがある。秘密分散とは、のちのち共有したい情報を分割し、それらを複数箇所に分散することで、情報を秘密にしたまま安全に保存する方法である。その基本的な仕組みは簡単で、次のようなものである。

秘密の情報を持っている人は、その情報を適当な方法で分割し、のちのちそれを共有したい複数の人（プレイヤー）に分散する。分割された情報のひとつひとつはシェアと呼ばれ、それぞれのシェアから秘密の情報を類推することは一切できない。そのためこの段階では、プレイヤーも含めて誰も秘密の情報を得ることはできないので、秘密の情報は安全に保存できる。秘密の情報を復元するときには、分散されたシェアを一定数以上集めて、適当な方法で結合する。つまり、秘密の情報を得たいプレイヤーがそれぞれのシェアを出し合って共有し、結合することで秘密の情報を復元するのだ。

この仕組みを実現するためには、秘密の情報を分割してシェアを作るときに、分割の方法に注意が必要となる。例えとして、宝の地図をそのまま分割する場合を考える。この地図には宝の位置を示す印が描かれているとする。すると、その印が描かれた地図の断片（シェア）のみで宝の場所が類推できるかもしれない。また、そのシェアは他のシェアよりも重要になるだろうから、復元の際にそのシェアを持つプレイヤーは優位になる可能性がある。このように、秘密の情報をそのまま分割すると、秘密の情報が安全に保存できなかったり、プレイヤー間に不公平が生じることがある。そのため、秘密の情報が類推できず、かつ公平であるようにシェアを作らなければならない。

実際の秘密分散では、秘密の情報は特別に加工され分割される。この加工によってシェアの公平性などを保証すると同時に、復元に必要なシェアの数を任意の閾値に設定できるようになる。つまり、プレイヤーどうしが協力して出し合ったシェアの総数が、閾値未満であれば秘密の情報は一切復元できないが、閾値以上であれば完全に復元できるようにすることが可能となる（**図 1**）。例えば各プレイヤーが一つずつシェアを持っていると

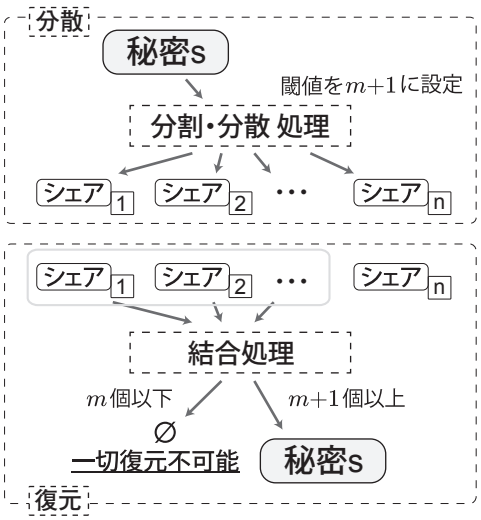
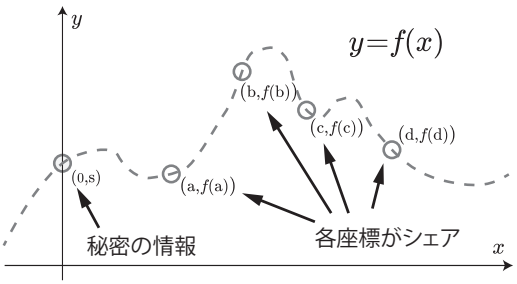


図１　分割・分散と復元

き、その閾値以上の数のプレイヤーが協力して初めて、秘密の情報を復元することができる。

ここで秘密分散の実現方法の一例としてShamir's Secret Sharingを紹介しよう（**図２**）。これは多項式を用いる非常に簡単な方法だ。まず、適当な m 次多項式 $y=f(x)$ を用意する。このとき、その多項式の y 切片 $f(0)$ を秘密の情報とする。次に、切片を除く座標 $(x,f(x))$ をシェアとして各プレイヤーに配分する。この時シェアはいくつでも作り出せるので、プレイヤーは何人でも構わない。m 次多項式である $f(x)$ は、$m+1$ 個以上の通る点（シェア）が得られれば多項式としてただ一つに定まり、その切片である秘密の情報が復元できる。ここで確認して欲しいのは、どのシェアも公平であることと、閾値である $m+1$ 個未満のシェアでは切片は全く定まらず、秘密の情報は一切復元できないという点である。

秘密分散が実際に使われるときには、シェアの公平性とは別に、復元の際にも公平性が求められる。すなわち、秘密の情報の復元に協力したプレイヤー全員が、公平に秘密の情報を得られる事も保証される必要がある。これまでに述べたとおり、秘密分散は秘密の情報を分割・分散したあとで、複数のプレイヤーがシェアを出し合うことで秘密の情報を復元する。しかし、これだけの簡単な仕組みでは、復元に協力した全員が秘密の情報を得られるという公平性を保証できないことが知られている。そのことを議論するためにゲーム理論というものを用いるケースを紹介する。



m次多項式は$m+1$個以上の通る点（シェア）で切片が定まり、秘密の情報が復元可能。
m個以下では切片は区間$(-\infty,+\infty)$で全く定まらない。

図２　Shamir's Secret Sharing

現実的なプレイヤーの振る舞い

暗号理論による秘密分散の議論では、プレイヤーの行動に比較的単純な仮定をおく。しかし実際には、プレイヤーは意思を持って行動するため、単純な仮定では不十分なことがある。そこで、秘密分散の議論にゲーム理論というものを用いることで、より現実的な設定のもとでプレイヤーの振る舞いを考えることができる。

ゲーム理論は、複数の意思決定主体が存在し、それぞれの行動が相互に影響する場合を考えるのに適した理論である。秘密分散においては、各プレイヤーがゲーム理論の意思決定主体にあたる。ゲーム理論を用いての議論では、プレイヤーにある種の合理性をおいて、行動選択の基準を明確にする。そして各プレイヤーは、それぞれ自身の利得が最大になるように行動を選択する。

では実際に、秘密分散のプレイヤーにある種の合理性、すなわちゲーム理論としての仮定を加えて考えてみる。

仮定1.［秘密の情報を得られないよりも得られる方が利得は高い］

仮定2.［仮定1のもとで、秘密の情報を得る人が少ないほど利得は高い］

この二つの仮定を認めるものとする。すると、特に仮定2により、現実的なプレイヤーの振る舞いを考えることができる。

これまでの議論では、秘密の情報を復元する際

に全てのプレイヤーは協力的であるという暗黙の了解があった。これを無くし、先の合理性のみに従ってプレイヤーの行動を決定する。すると、復元の際に自分のシェアを出さないという非協力的な行動が選ばれうるようになる。この選択肢によって、復元に協力したプレイヤー全員が秘密の情報を得られるということを保証できなくなる。ある状況を仮定することで、それを説明しよう。

秘密の情報を復元する際、他のプレイヤー全てが協力的である中、ただ一人のプレイヤーのみが先の合理性に従う状況を考えよう。秘密の情報の復元に必要なシェア数の閾値が$m+1$であり、全$m+1$人のプレイヤーが復元に参加するとき、合理性に従うプレイヤーは仮定2を満たす行動を選択する。すなわち、他のm人のプレイヤーが実際に各自のシェアを出す中、自分だけシェアを出さないという選択を取る。これにより、シェアを出した協力的なプレイヤーはそれぞれ、合理性に従ったプレイヤーのシェアを除くm個のシェアを持つことになる。一方で、合理性に従い、非協力的な行動を選択したプレイヤーは、自身のシェアを含め、ちょうど$m+1$個のシェアを持つことになる。

これまでに述べたとおり、閾値が$m+1$であるとき、m個のシェアでは秘密の情報を一切復元できないが、$m+1$個のシェアでは秘密の情報を完全に復元することができる。結局、この復元の際、実際に秘密の情報を得られるのは、合理性に従った非協力的なプレイヤーのみとなってしまう。

このような復元に協力した全員が秘密の情報を得ることを保証できない点を含め、秘密分散にはまだいくつかの課題が存在する。実は、現時点で秘密分散はあまり実用の段階には達していない。実用に耐えうるものにするためには、非協力的なプレイヤーの存在時でも、あるいは偽のシェアを出すような、より攻撃性の高いプレイヤーの存在時でも、うまく機能するようにしなければならない。

そのときに厳しい制約になるのは、同時性が保証できないことである。秘密の情報を復元する際に、あるプレイヤーが他のプレイヤーと同時に、それも他のプレイヤーにシェアを出させるような強制力を持って行動を起こすのは極めて困難だ。その理由は、秘密分散が実際に使われるのはコンピュータネットワークを介してになると考えられるため、プレイヤーどうしが実際に対面しないことにある。そのため、同時性を保証できない場合でも機能させられる秘密分散の手法が研究されており、実際にいくつか存在する。しかし、そうした手法にも別の問題点があり、現実的な状況での利用に耐えうるものは、未だ考え出されていない。

田中先生らは、プレイヤーの振る舞いをここで紹介した以外にも想定し、ゲーム理論によって、より現実的な状況においても機能する秘密分散を構成しようとしている。秘密分散の仕組みは広く応用できるため、研究が進み実用の段階に至れば、秘密分散は様々な技術のベースになるだろう。

情報漏洩のもとでの暗号システム

続いて紹介する研究は、実用化されている暗号システムに対する研究である。これまで暗号理論ではさまざまな暗号システムが研究され、安全なものが構成されてきた。そして今では、私たちの意識していないところでもさまざまな形で暗号が用いられている。現在広く使われている暗号システムの一つに、公開鍵暗号というものがある。暗号理論により、公開鍵暗号の安全性もいくつかの前提条件のもとで保証されている。しかし、それらの前提条件が崩れてしまった場合の安全性については、これまで十分に議論されてこなかった。そこで、田中先生らの研究室では、そのような場合でも安全性を保証できる公開鍵暗号を構成することを研究の一つとしている。

公開鍵暗号の安全性要件

公開鍵暗号は現在広く使われている暗号システムの一つで、情報の暗号化と復号に別の鍵を用いるのが特徴だ。今では電子署名や秘匿通信の多くがこの公開鍵暗号を用いて成り立っている。例えば、ウェブブラウザを使ってhttpsで始まるアドレスに接続する際には、接続相手が正しいかチェックする電子署名と、それに続く秘匿通信の開始に、それぞれ公開鍵暗号が使われている。

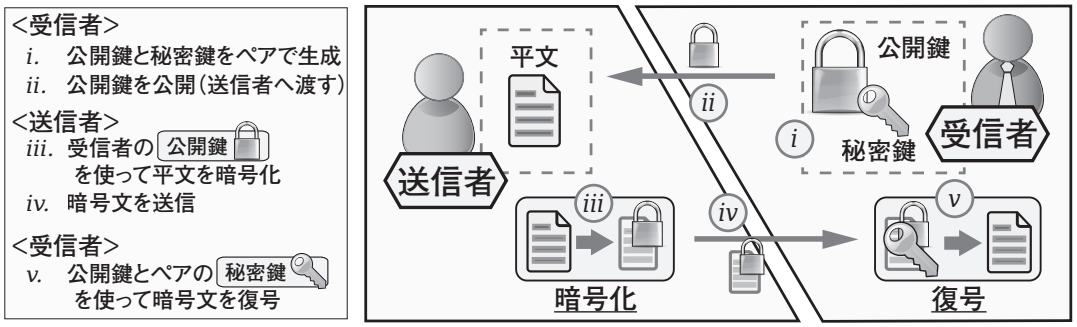


図3　公開鍵暗号

公開鍵暗号を実際に使うときの流れを説明しよう（**図3**）。暗号化のためには、最初に鍵を用意する必要がある。公開鍵暗号で用いられる鍵は二種類あり、ひとつを公開鍵、もうひとつを秘密鍵と呼ぶ。公開鍵は暗号化用の鍵で、秘密鍵は復号用の鍵となる。

これら二つの鍵は、暗号化された情報を受け取りたい受信者がペアで生成する。公開鍵は名前のとおり公開してよい鍵であり、誰でもこれを受け取ってよい。そのため、送信者は受信者の公開鍵を使って、送りたい情報（平文）を暗号化して送ることができる。平文が暗号化されたもの（暗号文）を受け取った受信者は、自身の秘密鍵でそれを復号し、もとの平文を得ることができる。また、その復号が許されるのは、二つの鍵を生成した受信者本人のみである。そのため、受信者は秘密鍵を秘密に保管し、漏洩させてはいけない。

このように暗号化用の鍵と復号用の鍵を別にすることで、平文の内容を知られることなく情報を送ることができる。しかし、これだけでは公開鍵暗号の安全性としては不十分だ。

もし同じ公開鍵を使って、同じ平文を暗号化すると、毎回同じ暗号文が作られる。そして、たとえ平文の内容が分からなくても、もとは同じ平文が複数回送られているという情報自体が、第三者にとって有益になりうる。また特に、平文の候補が絞られているとき、第三者は受信者の公開鍵を使ってそれら全てを実際に暗号化してみることで、全ての暗号文と平文の組み合わせを知ることもできる。これでは後に述べる、公開鍵暗号における基本的な安全性要件を満たすことはできない。

そこで送信者は平文を暗号化する際、受信者から与えられた公開鍵のみを用いるのではなく、自身で生成した乱数も同時に用いる。このとき、暗号化の度に別の乱数を用いるようにする。すると、暗号化に用いる鍵が毎回変わるようなものなので、たとえ同じ平文を暗号化しても異なる暗号文が作られるようになる。一方で受信者は、計算をうまく工夫することにより、乱数によらず同じ秘密鍵で復号できる。

このように乱数も利用することで、公開鍵暗号における基本的な安全性要件を満たしている。そのため、受信者にとっての秘密鍵と同様に、送信者は暗号化の際の乱数を漏洩させてはいけない。

構成できることが既に知られている公開鍵暗号の一つに、Indistinguishability under chosen-plaintext attack（IND-CPA）という安全性要件を満たした公開鍵暗号（IND-CPA 安全な公開鍵暗号）がある。田中先生らは、この IND-CPA 安全な公開鍵暗号の研究を行った。IND-CPA 安全な公開鍵暗号は、公開鍵暗号の安全性クラスの中

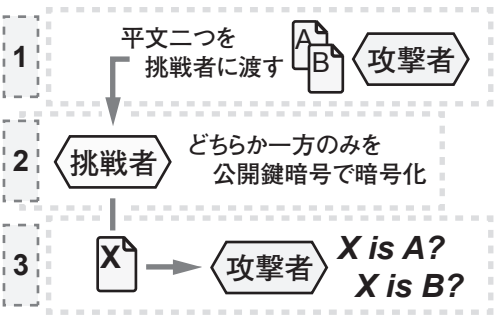


図4　IND-CPA

で最も弱いクラスに属し、他の公開鍵暗号の基本となっている。そのため研究の成果は、既存のより高い安全性クラスの公開鍵暗号に応用可能となる。これから簡単にIND-CPAの説明をした上で、その研究内容を紹介する。

IND-CPAとは、公開鍵暗号を構成する上で基本となる要件の一つである。この要件は、次のような攻撃者と挑戦者のゲームにおいて、攻撃者が1/2より有意に高い確率で答えを当てることができないというものである（**図4**）。

まず攻撃者は、このゲームで挑戦者が使用する公開鍵を受け取る。次いで攻撃者は、異なる二つの平文を作成する。これはどのような平文でもよいが一般に、平文作成時において攻撃者にとってのちのちゲームを有利に進めるのに最も都合の良い平文を作る。攻撃者はその二つの平文をそのまま挑戦者に渡す。挑戦者は、攻撃者に知られないように、その二つの平文のどちらか一つをランダムに選ぶ。選んだ平文を公開鍵暗号で、すなわち公開鍵と乱数で暗号化する。そして出来上がった暗号文を攻撃者に渡す。最後に、攻撃者は受け取った暗号文が、もとはどちらの平文が暗号化されたものなのかを当てる。

このとき、攻撃者が正解できる確率が1/2ならば、攻撃者は暗号文からは、それが二つの平文のどちらが暗号化されたものなのかを一切識別できないことを意味する。なぜならば、二つの平文はもともと攻撃者にとって都合良く作ることもできるにもかかわらず、ランダムに当てようとして正解する確率と等しいからだ。実際には厳密な定義を行った上での議論を要するが、簡単には以上が、公開鍵暗号における基本的な安全性要件である。

乱数漏洩時の安全性

IND-CPA安全な公開鍵暗号は、秘密鍵と乱数の二つの情報が漏洩しないという前提条件のもとで、安全な暗号システムとして構成される。それでは、もしそれらの情報が漏洩したときはどうなるのだろうか。実はこれまでの暗号理論においては、公開鍵暗号の、情報漏洩が生じた際の理論的安全性について一切の保証がなかった。

田中先生らの研究室では、情報漏洩時においても理論的安全性を保証できる公開鍵暗号を構成することを研究の一つとしている。その研究の中から、乱数が漏洩した場合の理論的安全性の保証について、田中先生らの研究結果を、秘密鍵の漏洩に関する外部の先行研究と合わせて紹介する。

秘密鍵あるいは乱数の漏洩が生じた際のIND-CPAに関しては、漏洩の程度と、漏洩のタイミングを考えなくてはならない。秘密鍵と乱数の漏洩はどちらも同じように考えることができるので、ここではIND-CPAのゲームにおける乱数の漏洩で考えていく。

まず、漏洩の程度については次のように考える。乱数が全て漏洩した場合は、それを全く用いていないのと同じなので、IND-CPA安全は達成できない。なぜならば、挑戦者が暗号化に使った公開鍵と乱数があれば、攻撃者は二つの平文を自分で暗号化し、どちらの暗号文も知ることが出来るからだ。そこで、漏洩は適当な仮定に従うものとして、漏洩する情報量を制限する。例えば、漏洩は乱数の半分相当に限るなどの仮定をおく。

続いて、漏洩のタイミングを考える。漏洩のタイミングは四通りで、攻撃者が公開鍵を受け取る前、公開鍵を受け取り二つの平文を作成する前、平文を作成した後、暗号文を受け取った後である。

漏洩する情報量を制限しても、もし漏洩する部分があらかじめ決まっていて変わらないと、意味のある議論を行えない。なぜならば、挑戦者は単に漏洩しないと分かっている部分を用いて暗号化を行えばIND-CPA安全を達成できるからだ。そこで研究では漏洩が生じるとき、漏洩する情報量を制限した上で、攻撃者は乱数のどの部分が漏洩し入手できるかを自由に決めることが許されると仮定している。

この仮定により攻撃者は、漏洩のタイミングが遅いほどIND-CPA安全を崩しやすくなる。なぜならば、漏洩のタイミングが遅いほど攻撃者が持つことのできる情報が多くなり、より都合の良い乱数を入手できるからだ。例えば、漏洩のタイミングが暗号文受け取り後の場合を考える。すると攻撃者は、漏洩した乱数は平文作成には利用できないが、挑戦者から受け取った暗号文を見た後に、漏洩し入手する乱数の部分を決めることができる。そのため、他のタイミングよりもIND-CPA安全を崩すために都合の良い乱数を入手できる。

以上のように、乱数の漏洩は程度のみならず、そのタイミングによってもIND-CPA安全を達成できるかが変わる。すなわち、漏洩のタイミングも公開鍵暗号の安全性をどれほど脅かすのかに影響するのである。これは秘密鍵の漏洩に関しても同様となる。それでは、田中先生らの乱数漏洩に関する研究結果と、秘密鍵漏洩に関する先行研究の結果を示そう。

まず、田中先生らによる乱数漏洩の研究では、表中の左に示す結果となった。公開鍵受取り前であれば、暗号手法に変更を加えることで、これまでに述べた一定の乱数漏洩時においても、IND-CPA安全な公開鍵暗号を構成できる。しかし、平文作成前の場合、攻撃者にある制約条件を加えないと、乱数漏洩時においてもIND-CPA安全な公開鍵暗号は構成できない。その後の二つのタイミングでは、IND-CPA安全な公開鍵暗号を構成することは不可能となる。

次に、秘密鍵漏洩に関する先行研究の結果は、表中の右に示す結果となっている。公開鍵受け取り前であれば、同様にIND-CPA安全な公開鍵暗号を構成できる。その後の二つのタイミングでは、Hash Proof Systemという暗号学的要素を用いて、IND-CPA安全な公開鍵暗号を構成できる。暗号文受け取り後では、IND-CPA安全な公開鍵暗号を構成することは不可能となる。

この結果のとおり、秘密鍵や乱数はそれぞれあるタイミング以降に漏洩が生じると公開鍵暗号の安全性を脅かす。田中先生らの研究結果から分かった興味深い点は、秘密鍵よりも乱数の漏洩の方が早い段階でその安全性を脅かすことである。公開鍵暗号を用いた実際の暗号システムにおいて、秘密鍵を漏洩させないよう注意することは、ある意味当然である。しかしこの研究結果が示すとおり、漏洩という観点からは、むしろ乱数を漏洩させないことの方が、公開鍵暗号の理論的安全性を保証する上では重要であることが分かった。

表　研究結果

	乱数漏洩	秘密鍵漏洩
公開鍵受け取り前	IND-CPA安全構成可能	IND-CPA安全構成可能
二つの平文作成前	条件付きで構成可能	Hash Proof System から構成可能
二つの平文作成後	構成不可能	
暗号文受け取り後	構成不可能	構成不可能

実はこの研究のように、暗号システムにおける、情報漏洩のもとでの理論的安全性が注目され、研究されるようになったのはごく最近のことである。田中先生らは、研究が広まる前の早い段階からその安全性に着目していた。そして公開鍵暗号については、ここで示した研究結果のとおり、乱数漏洩が平文作成前に生じても、攻撃者にある制約条件があれば、IND-CPA安全な公開鍵暗号を構成できる事を証明した。

しかし今のところ、その制約条件は現実的に考えるとかなり厳しいものとなっている。その理由は、そもそも漏洩の仕方として、攻撃者はどの部分が漏洩し入手できるかを自由に決められるという、攻撃者にとって非常に有利な仮定をおいていることにある。この仮定のもとでは制約条件を加えざるを得ない。そこで、田中先生らがこれから進めようとしているのは、漏洩の仕方としてより現実的な仮定をおくことで、乱数漏洩が生じても、より遅いタイミングまでIND-CPA安全な公開鍵暗号を構成することである。

暗号理論は、もっぱら数学的な興味のもとで研究されることが多いが、その数理的な研究が情報セキュリティという現実の問題に直接応用可能となる珍しい分野である。そのため、暗号理論の進歩は情報セキュリティとしての発展につながる。本稿で取り上げた二つの研究を含め、田中先生らが行っている研究により、今後さらに安全な暗号システムが構成されていくことを期待する。

最近では、個々人でもクラウドストレージを利用し、また既に日常的に通信やファイルの暗号化を行っています。一端ではありますが、この執筆のため田中先生らの研究内容を伺うことで、それら情報システムの安全性には、まさに暗号理論が必要不可欠であることを実感しました。

取材に快く応じてくださり、専門的な知識のない私たちにも分かりやすい説明をしてくださった田中先生、安永先生に深謝申し上げます。

（寺坂　欣也）