修士論文

識別子の特徴を考慮した
識別子利用暗号の拡張に関する研究

指導教官　　松浦 幹太　准教授

東京大学大学院 情報理工学系研究科 電子情報学専攻

66413　　北田 亘

平成20年2月4日提出

あらまし

公開鍵暗号における鍵のすり替えの問題を解消するために、ID ベース暗号という概念が提案されている。この方式では、各エンティティの ID 情報を公開鍵に用いるため、公開鍵のすり替えを防ぐことができるとされている。しかし、そのような特徴を持つ ID 情報というのは、自明に存在するわけではない。それにもかかわらず、これまで提案されてきた ID ベース暗号の方式は、どれもそのような ID が存在するとして議論を行っている。

本研究では、この点に注目し、本当の意味での ID ベース暗号の達成を目標とする。しかし、現在までに、このテーマに見合った先行研究はほとんどない。そのため、本研究では、理想的 ID ベース暗号の実現に向けた第一歩として、ID 情報を、従来とは異なった視点から捉え、その結果得ることができた方式に関して述べる。現在は、このように ID 情報の新しい利用法に関する研究となっているが、今後、前述のような、鍵のすり替えを防ぐことのできる、理想的な ID ベース暗号の構築を目標とする。

目次

あらまし **i**

1 序論 **1**
1.1 研究背景 …… 1
1.2 目的 …… 3
1.3 本稿の貢献 …… 3

2 基礎理論 **4**
2.1 情報セキュリティ …… 4
2.2 暗号 …… 5
2.3 公開鍵暗号 …… 5
2.3.1 アルゴリズム …… 5
2.4 証明可能安全性 …… 6
2.5 鍵管理問題 …… 6
2.6 識別子利用暗号 …… 7
2.6.1 ID ベース暗号 …… 8
2.6.2 属性ベース暗号 …… 8
2.6.3 従来の識別子利用暗号の再考 …… 10

3 柔軟な識別子評価可能な暗号化方式 **12**
3.1 概要 …… 12
3.2 定式化 …… 13
3.2.1 柔軟な識別子評価可能な暗号化方式 …… 13
3.3 構成 …… 15
3.3.1 鍵規定型の構成 …… 15
3.3.2 暗号文規定型の構成 …… 16
3.3.3 アクセス構造の構成法 …… 16
3.3.4 安全性証明 …… 17
3.4 評価 …… 19

4 ブラインド属性ベース暗号 **20**
4.1 意義 …… 20
4.2 関連研究 …… 21
4.2.1 Certificateless Encryption …… 21
4.2.2 鍵偽造の追跡可能な ID ベース暗号 …… 21
4.2.3 ブラインド ID ベース暗号 …… 21

4.3 定式化 …… 22
4.3.1 安全性定義 …… 22
4.3.2 ゼロ知識証明 …… 23
4.3.3 コミットメント …… 24
4.4 構成 …… 24
4.4.1 鍵生成プロトコル …… 26
4.5 考察 …… 27

5 情報理論的安全性を持つ Chaffing-and-Winnowing 29
5.1 概要 …… 29
5.1.1 目的 …… 29
5.1.2 Chaffing-and-Winnowing …… 29
5.1.3 関連研究 …… 29
5.2 用語定義 …… 30
5.2.1 情報理論的に安全な認証子 (A-code) …… 30
5.2.2 情報理論的に安全な暗号化方式 …… 30
5.3 情報理論的に安全な Chaffing-and-Winnowing …… 31
5.4 考察 …… 36
5.4.1 多項式オーダの空間サイズにおける秘匿性 …… 36

6 識別子の特徴に関する考察 38
6.1 考察 …… 38

7 結論 41
7.1 まとめ …… 41
7.2 今後の課題 …… 41

発表文献 42

参考文献 43

謝辞 46

Chapter 1　序論

1.1　研究背景

近年、社会の情報化は高度に進んできている。その結果、情報セキュリティ事故に対するリスクの増大、つまり、情報セキュリティ事故の発生率や発生時の被害の増大も著しい。そのため、そのリスクを抑えるための技術である、情報セキュリティの重要性は、ますます高まってきている。

情報セキュリティは、その性質上、扱う分野が非常に広い。技術的な面に絞った場合でも、システム設計の際の物理的なセキュリティ、ネットワークにおけるセキュリティ、データベースやWebアプリケーションなどのアプリケーションレベルのセキュリティといったように、それぞれの段階におけるセキュリティを考える必要がある。また、情報セキュリティ事故のリスクマネジメント、組織におけるセキュリティポリシー、セキュリティ教育など、情報通信技術では対応できない部分まで考慮する必要がある。

この情報セキュリティにおいて、情報通信技術は、主に、ネットワークセキュリティ、ソフトウェアセキュリティに関して貢献してきている。具体的には、ファイアウォール、アクセス制御、暗号技術、各種攻撃（サービス妨害攻撃、フィッシング、スパムメール、クロスサイトスクリプティング、インジェクション）への対応などがあげられる。

本研究は、この中の、暗号技術に関する研究である。暗号技術は、大きく分けると、暗号化技術、電子署名、暗号プロトコルの3つに分けることができる。暗号化技術は、鍵を持っていないエンティティに情報を見せないための、機密性を確保するための技術である。電子署名は、情報の改竄を検知し、完全性を保証するための技術である。暗号プロトコルは、単純に機密性や完全性を提供することを目的とせず、例えば、総資産をばらすことなくどちらがより金持ちかを調べる『金持ち比べ』などの、マジックプロトコルを実現するための技術である。

本研究は、この中の暗号化技術に関する研究である。暗号化技術は、大きく分けると、共通鍵暗号と公開鍵暗号の2種類に分けることが可能である。共通鍵暗号とは、送信者と受信者の間で共有している鍵に基づいて暗号化・復号を行う技術であり、代表的なものとして、AES(Advanced Encryption Standard)[14]がある。この方式は、処理が非常に速いが、送信者と受信者の間で鍵を共有しなければならず、その鍵配送が大きな問題となる。それに対し、公開鍵暗号は、暗号化用の鍵と復号用の鍵が異なっており、暗号鍵の情報から復号鍵の情報を得るのが困難な方式である。そのため、暗号鍵を公開することが可能となり、共通鍵暗号のときに問題であった鍵配送を容易に行うことが可能となる。

しかし、公開鍵暗号では、“鍵のすり替え”が問題となってしまう。これは、図1.1

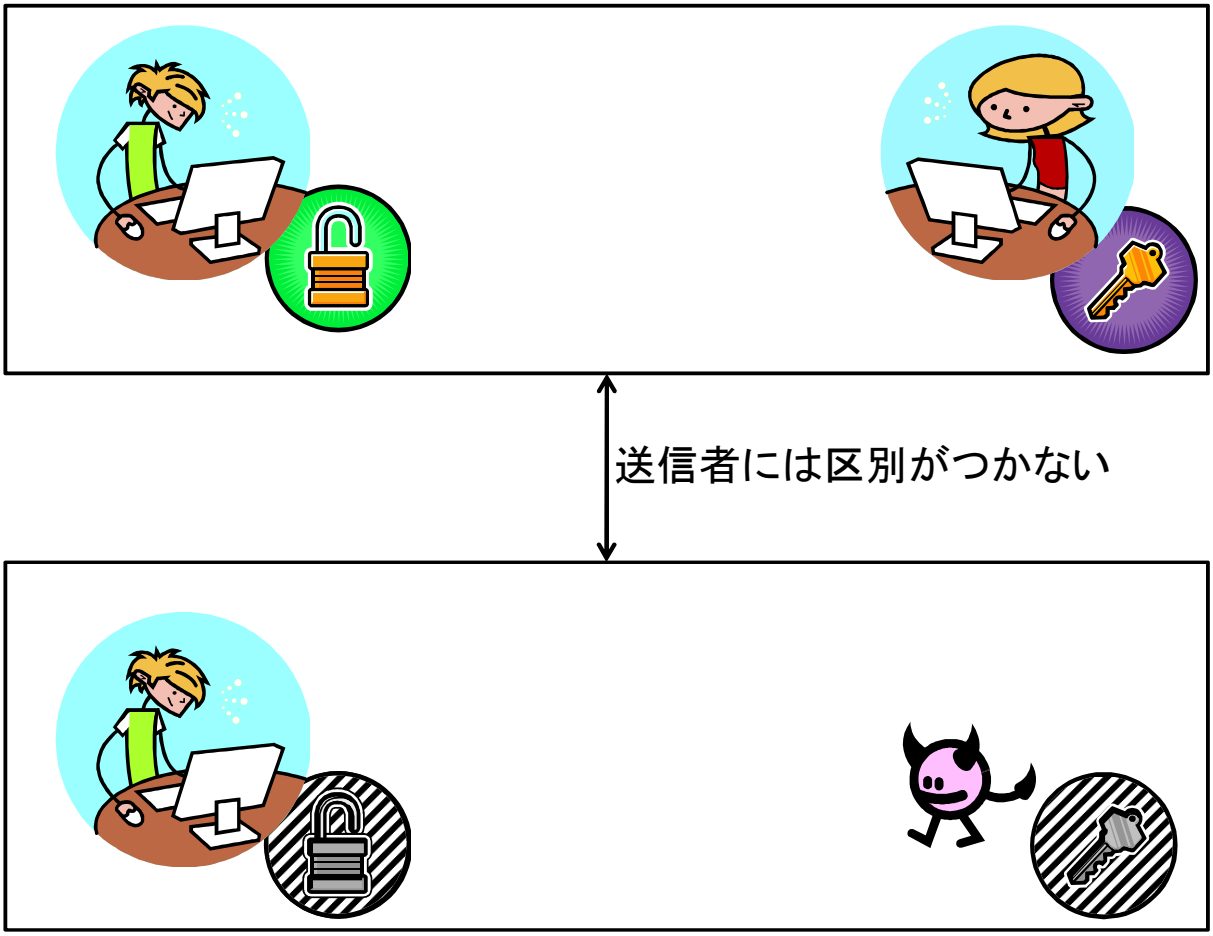

図 1.1: 鍵のすり替え問題

のように、受け取った公開鍵の正当性を判断することができないために、攻撃者の虚偽の鍵の広告により、鍵のすり替えが行われてしまうという問題である。現在では、この問題に対応するため、認証局がすべての公開鍵に、その正当性を保証する証明書を発行している。しかし、この方法では、認証局の負荷が大きくなってしまう。

そこで、この問題を解決するために、[32] では、ID ベース暗号という方式が提案されている。ID ベース暗号とは、各エンティティを表す識別子をそのまま公開鍵として用いるという方式である。このため、送信者は公開鍵である識別子からエンティティの判断を行うことができるため、鍵のすり替えを防ぐことができるようになる。

しかし、これまで提案されてきた ID ベース暗号 [5, 6, 33, 16] は、どれも ID として『任意長のビット列』を用いることができるとして扱われているだけである。これは、「各エンティティを表す識別子がどのようなものであっても、任意長のビット列を扱えるから、対応できる」ということしか実現してない。つまり、各エンティティの識別子として、どのような情報を用いるかということについては、何も述べられてないの

が現状である。

1.2　目的

1.1 で述べたように、鍵のすり替えの問題に対し、ID ベース暗号という概念が提案された。しかし、現在までに、実際に鍵のすり替えに対する十分な解決策は得られていない。そこで、本研究の最終的な目的として、鍵のすり替えを防ぐことができるような識別子を持った、ID ベース暗号の達成を目指す。

しかし、これまでこの識別子について議論した論文は数えるほどしかなく [25]、しかもそれらも十分に議論されているとは言えない状態である。そのため、本稿においては、識別子について深い議論を進める第一歩として、従来考えられていなかった識別子の利用について考える。

1.3　本稿の貢献

1.2 で述べたように、本稿は、識別子に対する議論を行う第一歩と位置づけられる。そして、本稿の貢献は、大きく分けると識別子の新たな評価と識別子の匿名性についての研究と言える。そのため、まず 2 章では、既存の情報セキュリティや暗号技術についてまとめ、なおかつ識別子の現状についてまとめ、識別子利用暗号の中で従来考えられていなかった識別子の特徴や利用に関する整理を行う。3 章では、識別子の新たな評価に関する研究について述べる。4 と 5 では、識別子の匿名性について議論を行う。6 では、これまでの結果を踏まえ、識別子に対する考察を行う。最後に 7 では、本稿のまとめと今後の課題について述べる。

Chapter 2　基礎理論

2.1　情報セキュリティ

情報システムが高度に発達し、社会に広く浸透した現代において、安全にシステムを利用する際、情報セキュリティは必要不可欠となっている。[34] では、情報セキュリティの目標として、次の３つの項目を挙げている。

- 機密性
- 完全性
- 可用性

ここで、これらについて簡単に説明を行う。機密性 (Confidentiality) とは、「アクセスを認可された者だけが、情報にアクセスできることを確実にすること」であり、機密性を失った場合には、「無権限者への情報の流出」が発生する可能性がある。完全性 (Integrity) とは、「情報およびその処理方法が、正確であり、かつ完全であることを保護すること」であり、完全性を失った場合には、「不当な改竄による損害」が発生する場合がある。可用性 (Availability) とは、「認可された利用者が、必要なときに、情報および関連する資産にアクセスできることを確実にすること」であり、可用性を失った場合には、「情報利用に対する不当な妨害」が発生する可能性がある。

また、情報セキュリティと一言にいっても、その扱う分野は広い。その分野を、大きく分ける場合、たとえば次のように分けることもできる。

- 暗号
- ネットワークセキュリティ
- セキュリティマネジメント

これらについて簡単に説明を行う。まず、暗号は情報に秘匿性や改竄防止性を持たせることで、情報の価値を維持・保証するための技術である。ネットワークセキュリティは、コンピュータネットワーク上での、種々の危険性の回避法を扱い、脅威から情報を守るための技術である。セキュリティマネジメントとは、システムを安全に運営するために、会社等の組織が組織として取るべき対策を扱う枠組みである。情報セキュリティの中には、これらにうまく括れない分野もあるが、基本的にはこれらの領域が合わさって、安全なシステムを構築するのを目的とする。

2.2　暗号

2.1 で簡単に説明したが、ここでは、暗号について詳しく説明する。

暗号は歴史が長く、紀元前から暗号が存在していたことが分かっている。例えば、古代ローマの軍人の名前に由来した “シーザー暗号” は、単純なシフト暗号ではあるが、現在知られている世界最古の暗号の 1 つである。現代では、コンピュータの発達により解析の速度が向上したため、暗号の安全性について厳密に評価しなければ意味がなくなってしまう。現代の暗号は、以下のように分類することができる。

- 暗号化
- 電子署名
- 暗号プロトコル

ここで、暗号化とは、情報に秘匿性を持たせ、情報の価値を維持するための技術である。電子署名とは、情報の改竄を検知することが可能になり、情報の価値を保証するための技術である。暗号プロトコルとは、複数人で特殊な情報をやり取りする技術である。

2.3　公開鍵暗号

公開鍵暗号 [15, 29, 18, 24] とは、暗号化の技術の 1 つである。もともと、暗号化は共通鍵暗号が最初に提案された。共通鍵暗号とは、送信者と受信者の間で、事前に鍵を共有しているという前提のもとで暗号通信を行う方式である。この際、鍵の配送の方法が困難であるという問題が生じる。この問題を解決するために、公開鍵暗号の概念が提案された [15, 29]。公開鍵暗号では、暗号化鍵と復号鍵が異なる。さらに、暗号鍵から復号鍵を計算することが、多くの場合計算量的に困難となるように設計されている。そのため、暗号鍵を公開することが可能となり、暗号文の送り手は容易に暗号鍵を得ることができる。これにより、共通鍵暗号のとき困難であった鍵配送を容易に行えることが分かる。

ここで、公開鍵暗号の一般的な構成について説明する。

2.3.1　アルゴリズム

公開鍵暗号は、鍵生成、暗号化、復号という 3 つのアルゴリズムから構成される。それぞれのアルゴリズムの役割は以下のようになる。

鍵生成アルゴリズム セキュリティパラメータに基づいた暗号鍵と復号鍵を生成するアルゴリズム

暗号化アルゴリズム 暗号化したい平文と暗号鍵から、暗号文を生成するアルゴリズム

復号アルゴリズム 暗号文と復号鍵から、平文あるいはエラーシンボル $\perp$ を出力するアルゴリズム

2.4　証明可能安全性

ここでは、公開鍵暗号の安全性を証明する際によく用いられる証明可能安全性という評価法について説明する。

証明可能安全性は、[17] などで用いられ始めた手法であり、以下のような手順で安全性の評価を行うのが一般的である。

1. 安全性の概念の定義
2. 定義した安全性を評価する試行（ゲーム）の定義
3. 試行の結果、安全性を破る確率を評価

以下に、それぞれのステップの簡単な説明を行う。

1. 安全性の定義

安全性を達成するために、暗号文や復号鍵が満たすべき条件を定義する。暗号化方式においては、一方向性、識別不可能性、頑強性などがよく用いられる概念である。特に、識別不可能性は非常に多く用いられる概念であり、これは「暗号文から平文の情報が全く漏れない」というものである。

2. 安全性を評価する試行の定義

安全性を評価するための試行を定義する。この試行のことを、証明可能安全性の分野においては “ゲーム” と呼んでいる。このゲームの定義は、暗号を破ろうとする “攻撃者” を考え、攻撃者の最終的な目的や、攻撃の途中で得られる情報などを定義する。ここで、攻撃者の目的が容易であればそれは高い安全性への試験を意味しており、攻撃者に与える情報が多ければそれも高い安全性への試験を意味している。

3. 安全性が破られる確率を評価定義したゲームを実行した結果、安全性が破られる確率を評価する。その確率の値が、セキュリティパラメータのネグリジブル関数に従うとき方式は安全であるといい、ネグリジブル関数におさえられないとき安全ではないという。

実際に、上記のように安全性を評価するには、多くの場合、他の問題に多項式時間帰着を行うことにより評価を行う。このとき、帰着させる問題は、NP 完全問題や NP 困難問題、あるいはそれに準ずる問題である。代表的なものとして、素因数分解問題や離散対数問題、ナップサック問題などがある。つまり、これらの問題を多項式時間で有効に解くことができない限り、暗号方式も安全であると結論付けられるのである (図 2.4)。

このように、多項式時間帰着の部分に誤りがない限り、定義された安全性は帰着された問題と同じ困難性を持つことになり、証明可能安全性は非常に強力な評価法であることが分かる。

2.5　鍵管理問題

前述のように、公開鍵暗号では暗号鍵を公開することにより鍵配送を行う。この際、悪意のあるユーザにより、偽物の鍵を広告された場合、間違った鍵で暗号化すること

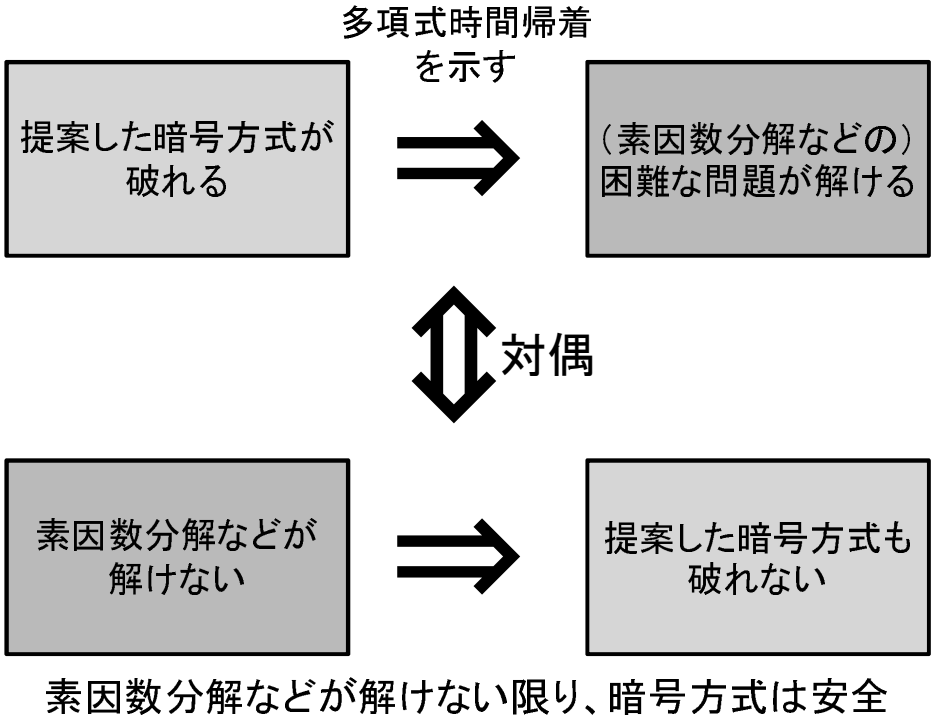

図 2.1: 証明可能安全性の概要

になる。

そのため、現在では認証局による公開鍵とエンティティの結びつきを保証する証明書を発行し、虚偽の広告による脅威を回避している。しかし、この方法の場合、認証局は各エンティティに対し、証明書の発行・管理を行わなければならない [35](図 2.5)。この際、その発行・管理の負荷が膨大になり、現在は認証局を階層化するなどして負荷分散を行って対処している。

2.6　識別子利用暗号

2.5 の問題を解決する方法として、各エンティティの識別子を利用するという方法が考えられた [32]。具体的には、識別子を公開鍵として暗号化・復号を行うというものである。この場合、識別子自身がエンティティを指し示すため、虚偽の広告を行うことが原理的に不可能になるので、証明書の発行を行う必要がなくなり、鍵管理問題が解消される。

具体的なシステムの構成は次のようになる。まず、エンティティは送信者と受信者の他に鍵生成センターが存在する。そして、鍵生成センターがシステム全体のパラメータ（公開パラメータ）を公開する。その際、そのパラメータ決定に利用した秘密情報（マスター鍵）はセンターが保持しておく。その上で、送信者はその公開情報と送り先の識別子を用いて暗号化を行う。暗号文を受け取った受信者は、まず鍵生成セ

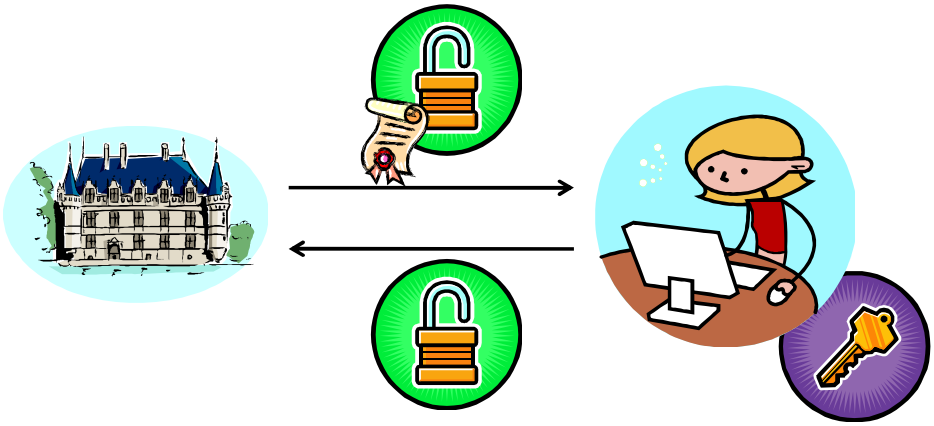

図 2.2: 鍵管理の問題
認証局は、すべての鍵に証明書を付与しなければならない

ンターに識別子情報を送る。識別子情報を受け取ったセンターは、その識別子に対応する復号鍵を生成し、受信者に送る。復号鍵を受け取った受信者は、その鍵を用いて暗号文の復号を行う。このような構成になっている (図 2.6)。

現在までに提案されている識別子利用暗号として、ID ベース暗号 [5, 6, 33, 16] と属性ベース暗号 [20, 11, 4, 12] の 2 つの方式が提案されている。

2.6.1　ID ベース暗号

各エンティティを唯一に特定する ID 情報を、公開鍵として用いる方法。ID ベース暗号は以下の 4 つのアルゴリズムから構成される。

セットアップアルゴリズム セキュリティパラメータに基づき、公開パラメータとマスター鍵を生成する。

暗号化アルゴリズム 送りたい平文と送り先の ID と公開パラメータから、暗号文を生成する。

鍵導出アルゴリズム ID を受け取り、その ID に対応する復号鍵を生成する。

復号アルゴリズム 復号鍵を用いて暗号文を復号し、平文あるいはエラーシンボル $\perp$ を出力する。

2.6.2　属性ベース暗号

[30] で概念が提案された方式であり、各エンティティの複数の属性情報と、属性情報に対するアクセス構造により暗号通信の制御を行う方式。ここで、アクセス構造とは、属性情報の集合が満たすべき条件を記述したものであり、アクセス構造を指定する方

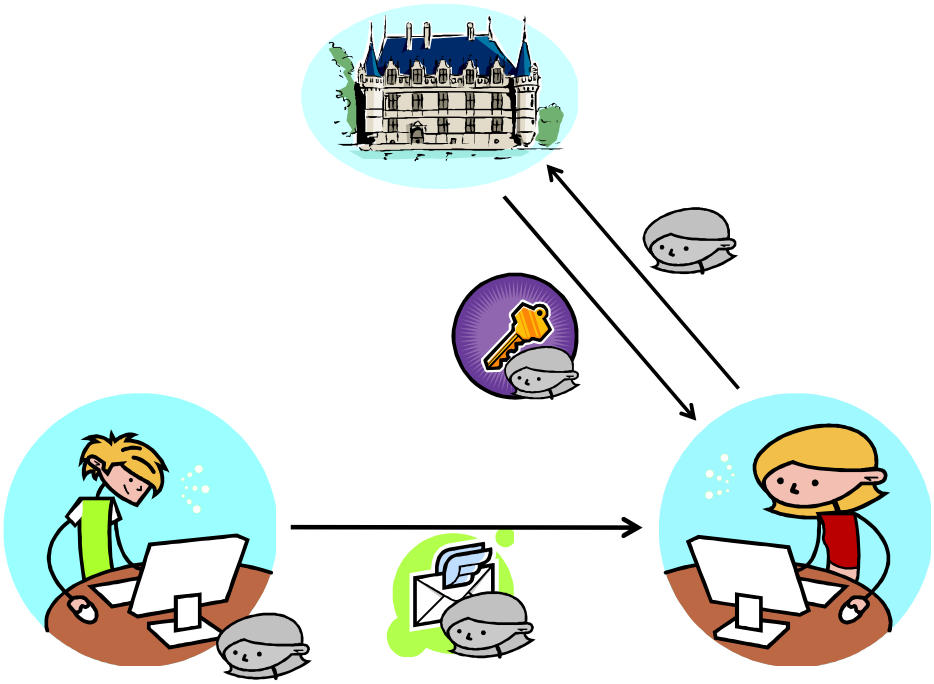

図 2.3: 識別子利用暗号の概要
公開鍵として、受信者の識別子を用いる

法により、鍵規定型 (Key-Policy) と暗号文規定型 (Ciphertext-Policy) に分類できる。それぞれ、以下のアルゴリズムから構成される。

鍵規定型

セットアップアルゴリズム セキュリティパラメータに基づき、公開パラメータとマスター鍵を生成する。

暗号化アルゴリズム 送りたい平文と送り先の属性集合と公開パラメータから、暗号文を生成する。

鍵導出アルゴリズム 暗号化の際の属性集合が満たすべきアクセス構造を受け取り、そのアクセス構造に対応する復号鍵を生成する。

復号アルゴリズム 復号鍵を用いて暗号文を復号し、平文あるいはエラーシンボル $\perp$ を出力する。

暗号文規定型

セットアップアルゴリズム セキュリティパラメータに基づき、公開パラメータとマスター鍵を生成する。

暗号化アルゴリズム 送りたい平文と送り先の属性集合が満たすべきアクセス構造と公開パラメータから、暗号文を生成する。

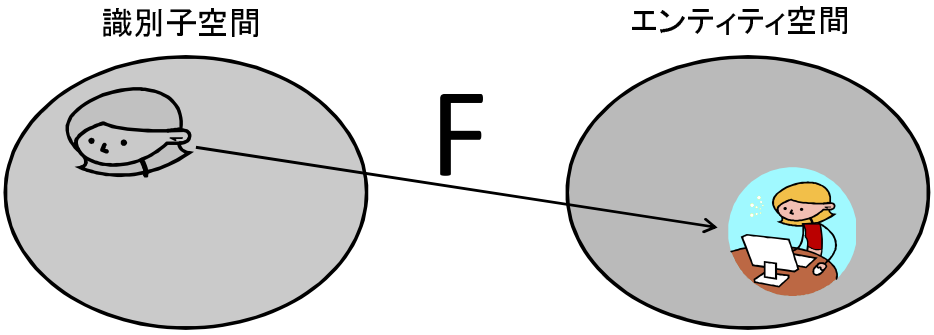

図 2.4: 識別子の性質
識別子が与えられると、それに対応するエンティティが決まる

鍵導出アルゴリズム 属性集合を受け取り、その属性集合に対応する復号鍵を生成する。

復号アルゴリズム 復号鍵を用いて暗号文を復号し、平文あるいはエラーシンボル $\perp$ を出力する。

この 2 つの分類は、アクセス構造の指定を、鍵導出アルゴリズムと暗号化アルゴリズムのどちらで行っているかにより分類されている。ここで、基本的に鍵導出のリクエストは受信者が行い、暗号化は送信者が行うものである。つまり、受信者と送信者のどちらがアクセス構造を指定するかにより、鍵規定型と暗号文規定型に分類できるとも考えることができる。

2.6.3　従来の識別子利用暗号の再考

ここで、従来の識別子利用暗号について考えてみる。

識別子とは、文字通りエンティティを識別するための情報である。言い換えると、識別子が与えられ、その値をもとにある写像を計算した結果、エンティティの部分集合が得られるのである (図 2.6.3)。

この際に、従来の方式は、識別子からエンティティの部分集合を誰でも特定可能としている。しかし、実際には識別子だけではエンティティの部分集合を特定できず、識別子集合からエンティティ集合への写像を知っている必要がある。つまり、システム全体に関する何らかの知識が必要であり、その知識をすべてのエンティティが共有しているという必要がある。この知識の共有方法に関しても議論を行わなければ、それまでの公開鍵暗号に対する利点も得られなくなる可能性がある。例えば、識別子集合からエンティティ集合への写像を与える方法として、識別子とエンティティとの対応表を用意することを考える。すると、その対応表の管理の負荷が大きくなってしまい、公開鍵暗号における鍵管理問題と本質的に同じ問題を抱えることになる。この場合、すでに識別子利用暗号の利点は消えてしまっていて、この方式を考える意味はなくなってしまう。この例からも、識別子利用暗号において、識別子に対する詳細な検討が必要なことが伺える。

ここで、現在の識別子利用暗号における識別子について、簡単に整理を行う。まず、従来の識別子は、識別子をもとに写像を計算するとエンティティの部分集合が得られると述べた。この写像の意味に関して考えると、与えられた識別子に『該当する』エンティティの集合を指し示すということを行っている。『該当する』とは、具体的にはエンティティがその識別子に属しているということである。視点を変えると、送信者が指定する送りたい相手の識別子と受信者の識別子が『一致する』か否かの判断を行っていることになる。これは、もともとIDベース暗号で1対1の通信を行うことが目的だったことからきていると考えられる。つまり、IDが一致するか否かを判断するため、それぞれの識別子が一致するか否かを判断できれば十分だったと考えられる。

また、識別子利用暗号における鍵導出についても考えてみる。従来の識別子利用暗号では、鍵導出の際、受信者は鍵生成センターに暗号通信に用いる識別子を送り、鍵生成センターはその識別子に対応する復号鍵を受信者に返している。この際、鍵生成センターは受信者の識別子情報を得ることが可能となってしまう。これは、プライバシー保護の観点から望ましくない。よって、鍵生成センターに対し、識別子情報の秘匿性を持たせる必要が生じることも考えられる。

ここで、秘匿性について、考察を行う。もともと、暗号化技術は他人には見られたくないメッセージ（平文）に秘匿性を持たせるための技術である。この際、平文空間のサイズは、基本的にセキュリティパラメータの指数オーダーのサイズで定義する。こうすることで、でたらめな推測により暗号文から平文を当てようとした際、その確率はセキュリティパラメータのネグリジブル関数に従うようになる。そのため、でたらめな推測により暗号が破られる確率は無視できるようになるため、安全性の証明を行えるようになる。しかし、識別子の秘匿性について考えると、方式によっては識別子の空間はセキュリティパラメータの多項式オーダーになっていることもあるため、従来の暗号における平文の秘匿性をそのまま適用するのは困難な場合がある。そのため、秘匿したい対象の空間の大きさが多項式オーダーに限られている状況を考える必要がある。

以上、

- 評価する識別子の関係
- 識別子の秘匿性
- 多項式オーダーサイズの空間における秘匿性

をまずは考える必要があると思い、本論文では、これらに関する成果を、各章で述べていく。

Chapter 3　柔軟な識別子評価可能な暗号化方式

3.1　概要

従来の識別子利用暗号では、送信者は送りたい相手の識別子をもとに暗号化し、受信者は自身の識別子を用いて復号する。そして、暗号化の際に指定した識別子と、復号の際に用いた識別子のすべて、あるいは一部が一致したときのみ元の平文に復号され、正しく通信を行うことができる。

本研究では、正しく通信を行える条件として、従来の「一致する」場合だけでなく、より多くの関係を対象とすることを考える。従来の ID ベース暗号では、１対１の通信の際の通信路の制御に主眼が置かれていたため、識別子が一致するか否かによる場合分けで十分であった。しかし、属性ベース暗号の登場により、識別子利用暗号は複数エンティティに対する通信路の制御を考慮に入れるようになった。そのため、従来のように「一致する」という識別子の関係の評価だけでなく、より多くの識別子の関係の評価を行うことにより、柔軟な通信路の制御を行えることが望ましい (図 3.1)。そこで、本研究では、「一致する」だけでなく、「異なる」、「(数値として見た際に) 大きい」、「小さい」といった関係を評価できるような方式を提案する。そのため、今後識別子について考察を行う際に、従来の ID ベース暗号や属性ベース暗号の枠組みでは不十分となった場合にも、本稿で提案する暗号方式を用いることで、柔軟に対応できることが期待できる.

ここで、「識別子の関係」とは、２つの識別子 $\mathbb{L}_1$ と $\mathbb{L}_2$ があった場合に、$\mathbb{L}_1 = \mathbb{L}_2$、$\mathbb{L}_1 \neq \mathbb{L}_2$ のような関係や、あるいは、識別子を整数型のビット列と見た場合に、$\mathbb{L}_1 > \mathbb{L}_2$ の関係などが挙げられる.

具体的には、属性ベース暗号を用いて一般的に構成を行っている．その際、識別子をビット列とみなし、各ビットを ABE の１つの属性に対応するように構築することで、ビット単位の処理を可能にした．しかし、従来の ABE のほとんどは、AND と OR の２種類の演算のみ可能なため、そのまま適用すると、可能なビット演算は限られてしまう．そこで、すべてのビットに対し、'0' と '1' の両方を考慮に入れることで、NOT 演算を可能にした．これにより、加法標準形あるいは乗法標準形に変形可能な（組み合わせ回路として表現できる）論理演算をすべて行うことを可能にした.

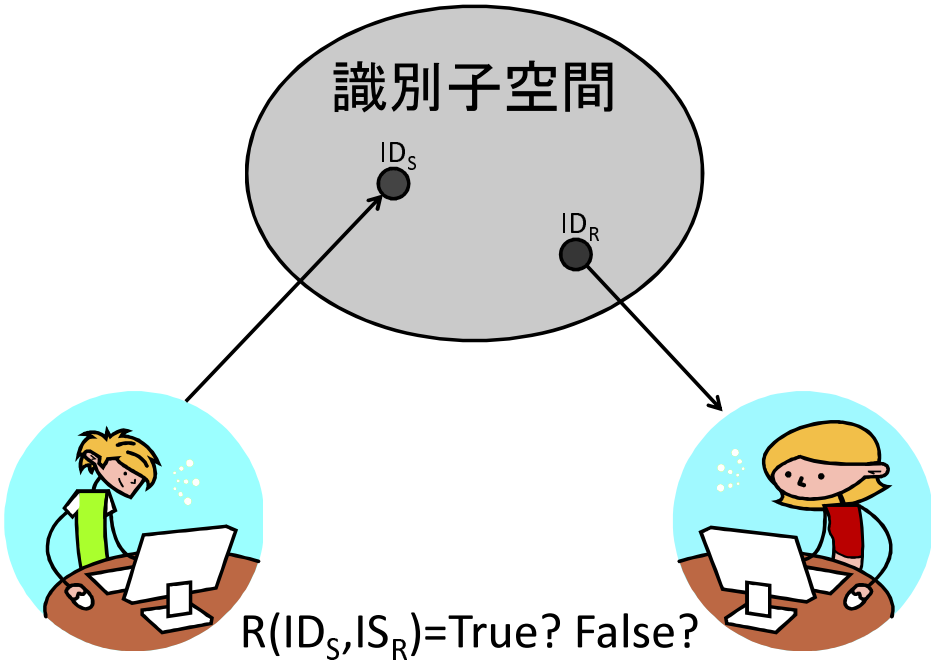

図 3.1: 識別子の関係
両者の指定する識別子が関係 R を満たすか否か

3.2　定式化

3.2.1　柔軟な識別子評価可能な暗号化方式

この節では、本稿で提案する、柔軟な識別子評価可能な暗号化方式について、説明する．**識別子の「関係」**
本稿では、識別子 $\mathbb{L}_1$ と $\mathbb{L}_2$ の関係を $R(\mathbb{L}_1, \mathbb{L}_2)$ で表し、$\mathbb{L}_1$ と $\mathbb{L}_2$ が関係 R を満たす場合には $R(\mathbb{L}_1, \mathbb{L}_2) = 1$、満たさない場合には $R(\mathbb{L}_1, \mathbb{L}_2) = 0$ と表すことにする．

アルゴリズム
提案方式は鍵規定型 (Key-Policy) と暗号文規定型 (Ciphertext-Policy) の 2 種類に分けられる．まず、鍵規定型の場合は、以下の 4 つのアルゴリズムから構成される．

$Setup(1^k)$ ：公開パラメータ prm とマスタ鍵 msk を出力する．

$Ext(\mathsf{prm}, \mathsf{msk}, \mathbb{L}, R)$ ： 公開パラメータ prm とマスタ鍵 msk と識別子 $\mathbb{L}$ と関係 R を受け取り、各エンティティの秘密鍵 $d_{\mathbb{L},R}$ を出力する．

$Enc(M, \mathsf{prm}, \mathbb{L})$ ： 平文 M と公開パラメータ prm と識別子 $\mathbb{L}$ を受け取り、暗号文 C を出力する．

$Dec(C, d_{\mathbb{L},R})$: 暗号文 C と秘密鍵 $d_{\mathbb{L},R}$ を受け取り、平文 M またはエラーシンボル $\perp$ を出力する.

次に、暗号分規定型の場合は以下の４つのアルゴリズムから構成される.

$Setup(1^k)$: 公開パラメータ prm とマスタ鍵 msk を出力する.

$Ext(\mathsf{prm}, \mathsf{msk}, \mathbb{L})$: 公開パラメータ prm とマスタ鍵 msk と識別子 $\mathbb{L}$ を受け取り、各エンティティの秘密鍵 $d_{\mathbb{L}}$ を出力する.

$Enc(M, \mathsf{prm}, \mathbb{L}, R)$: 平文 M と公開パラメータ prm と識別子 $\mathbb{L}$ と関係 R を受け取り、暗号文 C を出力する.

$Dec(C, d_{\mathbb{L}})$: 暗号文 C と秘密鍵 $d_{\mathbb{L}}$ を受け取り、平文 M またはエラーシンボル $\perp$ を出力する.

安全性定義
鍵規定型の IND-sLabel-CPA の安全性を評価する際には、以下のゲームを用いる.

鍵規定型における IND-sLabel-CPA ゲーム

初期化 攻撃者 $\mathcal{A}$ は、攻撃対象とする $\mathbb{L}^*$ をチャレンジャー $\mathcal{B}$ に送る. $\mathcal{B}$ は $\mathbb{L}^*$ を受け取った後、公開パラメータ prm を $\mathcal{A}$ に送る.

第１段階 $\mathcal{A}$ は $\mathcal{B}$ に $(\mathbb{L}_i, R_i)$ $(R_i(\mathbb{L}^*, \mathbb{L}_i) \neq 1)$ を送り、$\mathcal{B}$ はそれに対応する $d_{\mathbb{L}_i,R_i}$ を $\mathcal{A}$ に送る.

チャレンジ $\mathcal{A}$ はメッセージ空間 $\mathcal{M}$ から M_0, M_1 を選び、$\mathcal{B}$ に送る. $\mathcal{B}$ は M_0, M_1 を受け取った後、b を $\{0,1\}$ からランダムに選び、$\mathbb{L}^*$ を用いて、M_b を暗号化して、得られた暗号文 C を $\mathcal{A}$ に送る..

第２段階 「第１段階」と同様の試行を繰り返す.

推測 最後に、$\mathcal{A}$ は $b' \in \{0,1\}$ を出力する.

定義 3.1. 上述のゲームにおいて、すべての t 時間アルゴリズム $\mathcal{A}$ に対し、$\left|\Pr[b = b'] - \frac{1}{2}\right| < \epsilon$ のとき、その方式は (t, ϵ)-*IND-sLabel-CPA* 安全であるという.

次に、暗号文規定型の IND-sLabel-CPA の安全性を評価する際には、以下のゲームを用いる.

暗号文規定型における IND-sLabel-CPA ゲーム

初期化 攻撃者 $\mathcal{A}$ は、攻撃対象とする $(\mathbb{L}^*, R^*)$ をチャレンジャー $\mathcal{B}$ に送る．$\mathcal{B}$ は $(\mathbb{L}^*, R^*)$ を受け取った後、公開パラメータ prm を $\mathcal{A}$ に送る．

第１段階 $\mathcal{A}$ は $\mathcal{B}$ に $\mathbb{L}_i$ $(R^*(\mathbb{L}^*, \mathbb{L}_i) \neq 1)$ を送り、$\mathcal{B}$ はそれに対応する $d_{\mathbb{L}i}$ を $\mathcal{A}$ に送る．

チャレンジ $\mathcal{A}$ はメッセージ空間 $\mathcal{M}$ から M_0, M_1 を選び、$\mathcal{B}$ に送る．$\mathcal{B}$ は M_0, M_1 を受け取った後、b を $\{0,1\}$ からランダムに選び、$\mathbb{L}^*, R^*$ を用いて、M_b を暗号化して、得られた暗号文 C を $\mathcal{A}$ に送る..

第２段階 「第１段階」と同様の試行を繰り返す.

推測 最後に、$\mathcal{A}$ は $b' \in \{0,1\}$ を出力する．

定義 3.2. 上述のゲームにおいて、すべての t 時間アルゴリズム $\mathcal{A}$ に対し、$\left|\Pr[b = b'] - \frac{1}{2}\right| < \epsilon$ のとき、その方式は (t, ϵ)-*IND-sLabel-CPA* 安全であるという．

3.3　構成

ここでは、鍵規定型と暗号文規定型それぞれの構成について示す。

3.3.1　鍵規定型の構成

鍵規定型の属性ベース暗号 $(ABE.Setup, ABE.Ext, ABE.Enc,$
$ABE.Dec)$ から一般的に鍵規定型の柔軟な識別子評価可能な暗号化方式を構成する方法を示す．この際、元の属性ベース暗号の属性空間を $\mathcal{U}$ とし、識別子のビット長を n とする．

$Setup(1^k)$ ： まず、$2n$ 個の乱数 $a_1, \ldots, a_n, b_1, \ldots, b_n \in \mathcal{U}$ をランダムに選ぶ．
次に、$(\mathsf{prm}', \mathsf{msk}') \leftarrow ABE.Setup(1^k)$ とする．そして、公開パラメータ $\mathsf{prm} =$ $(\mathsf{prm}',$
$\{a_i\}_{1 \leq i \leq n}, \{b_i\}_{1 \leq i \leq n})$、$\mathsf{msk} = \mathsf{msk}'$ として出力する．

$Ext(\mathsf{prm}, \mathsf{msk}, \mathbb{L}, R)$ ： $\mathsf{prm} = (\mathsf{prm}', \{a_i\}_{1 \leq i \leq n}, \{b_i\}_{1 \leq i \leq n})$ とみなす．$\mathbb{L}$ の i 番目のビットを $\mathbb{L}^i$ で表したとき、$\gamma = \{a_i\}_{\forall i \mathbb{L}^i = 0} \cup \{b_j\}_{\forall j \mathbb{L}^j = 1}$ とする．また、$\mathbb{L}, R$ から、アクセス構造 $\mathbb{A}$ を構築する（詳細は後述）．その後、$d_{\mathbb{L},R} \leftarrow ABE.Ext(\mathsf{prm}', \mathsf{msk}, \gamma, \mathbb{A})$ として、$d_{\mathbb{L},R}$ を出力する．

$Enc(M, \mathsf{prm}, \mathbb{L})$ ： Ext アルゴリズムと同様に、$\gamma = \{a_i\}_{\forall i \mathbb{L}^i = 0} \cup \{b_j\}_{\forall j \mathbb{L}^j = 1}$ とする．その後、$C \leftarrow ABE.Enc(M, \mathsf{prm}', \gamma)$ として、C を出力する．

$Dec(C, d_{\mathbb{L},R})$ ： $M \leftarrow ABE.Dec(C, d_{\mathbb{L},R})$ として、その出力結果 M（$\perp$ を含む）を出力する．

3.3.2　暗号文規定型の構成

暗号文規定型の属性ベース暗号 ($ABE.Setup, ABE.Ext, ABE.Enc, ABE.Dec$) から一般的に暗号文規定型の柔軟な識別子評価可能な暗号化方式を構成する方法を示す．この際、元の属性ベース暗号の属性空間を$\mathcal{U}$とし、識別子のビット長をnとする．

$Setup(1^k)$: まず、$2n$ 個の乱数 $a_1, \ldots, a_n, b_1, \ldots, b_n \in \mathcal{U}$ をランダムに選ぶ．
次に、$(\mathsf{prm}', \mathsf{msk}') \leftarrow ABE.Setup(1^k)$ とする．そして、公開パラメータ $\mathsf{prm} = (\mathsf{prm}', \{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n})$、$\mathsf{msk} = \mathsf{msk}'$ として出力する．

$Ext(\mathsf{prm}, \mathsf{msk}, \mathbb{L})$: $\mathsf{prm} = (\mathsf{prm}', \{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n})$ とみなす．$\mathbb{L}$ の i 番目のビットを $\mathbb{L}^i$ で表したとき、$\gamma = \{a_i\}_{\forall i \mathbb{L}^i=0} \cup \{b_j\}_{\forall j \mathbb{L}^j=1}$ とする．その後、$d_{\mathbb{L}} \leftarrow ABE.Ext(\mathsf{prm}', \mathsf{msk}, \gamma)$ として、$d_{\mathbb{L}}$ を出力する．

$Enc(M, \mathsf{prm}, \mathbb{L}, R)$: Ext アルゴリズムと同様に、$\gamma = \{a_i\}_{\forall i \mathbb{L}^i=0} \cup \{b_j\}_{\forall j \mathbb{L}^j=1}$ とする．また、$\mathbb{L}, R$ から、アクセス構造 $\mathbb{A}$ を構築する（詳細は後述）．その後、$C \leftarrow ABE.Enc(M, \mathsf{prm}', \gamma, R)$ として、C を出力する．

$Dec(C, d_{\mathbb{L},R})$: $M \leftarrow ABE.Dec(C, d_{\mathbb{L},R})$ として、その出力結果 M（$\perp$ を含む）を出力する．

3.3.3　アクセス構造の構成法

この節では、アクセス構造の構成法について説明を行う．

今回我々が提案した方式では、二つの識別子間の関係を評価する際、それぞれのビット同士の論理式に展開した結果、その式を加法標準形や乗法標準形に書き表すことができる、つまり組み合わせ回路で表現できるすべての関係を評価可能となる．例えば、二つの識別子 $\mathbb{L}_1$ と $\mathbb{L}_2$ が「等しい $\left(\wedge_{\forall i}(\mathbb{L}_1^i = \mathbb{L}_2^i)\right)$」や「異なっている $\left(\vee_{\forall i}(\mathbb{L}_1^i = \neg\mathbb{L}_2^i)\right)$」や「３番目と５番目のビットが等しい $\left(\wedge_{i=3,5}(\mathbb{L}_1^i = \mathbb{L}_2^i)\right)$」などの関係が可能である．

ここでは、例として、関係 “>” を用いてアクセス構造の構成法を説明する．$n = 8$ として、Ext アルゴリズムに入力された識別子 $\mathbb{L}_R$ ='10011010' とする．このとき、Enc アルゴリズムで用いられる識別子 $\mathbb{L}_S$ が $\mathbb{L}_S > \mathbb{L}_R$ を満たすときに、復号可能となるようにアクセス構造 $\mathbb{A}$ を構成する．このとき $\mathbb{L}_S$ に求められる条件は、'11.....', '1.1.....', '1..111..', '1..11.11' のいずれかとなる（'.' は任意の１文字を表す）．このことから、Enc アルゴリズムで $\mathbb{L}_S$ から作られる γ が満たすべき条件は以下のようになる．

$$(b_1 \wedge b_2) \vee (b_1 \wedge b_3) \vee (b_1 \wedge b_4 \wedge b_5 \wedge b_6) \vee (b_1 \wedge b_4 \wedge b_5 \wedge b_7 \wedge b_8)$$

この論理式を $\mathbb{A}$ とすることで、$\mathbb{L}_S$ が上記の条件を満たした時のみ、復号が可能となる．

ここで、与えられる論理式が加法標準形の場合に、アクセス構造を構成可能であることを説明する．加法標準形の論理式は、AND, OR, NOT の３種類の論理演算によ

$==$	$!=$
$\wedge_{1\leq i\leq n}(\alpha_i == \beta_i)$	$\vee_{1\leq i\leq n}(\alpha_i != \beta_i)$
$>$	$<$
$count = 1$ $for(i = 1; i <= n; i++)\{$ $if(\beta_i == 0)\{$ $A_{count} \leftarrow (\wedge_{1\leq j\leq i-1}(\alpha_j == \beta_j)) \wedge (\alpha_i != \beta_i)$ $count++$ $\}$ $\}$ $\vee_{1\leq i\leq count-1} A_i$	$count = 1$ $for(i = 1; i <= n; i++)\{$ $if(\beta_i == 1)\{$ $A_{count} \leftarrow (\wedge_{1\leq j\leq i-1}(\alpha_j == \beta_j)) \wedge (\alpha_i != \beta_i)$ $count++$ $\}$ $\}$ $\vee_{1\leq i\leq count-1} A_i$

図 3.2: 各関係の一般的構成法

り記述される．そして、NOT は各変数にのみかかる．つまり、$\neg(X \wedge Y)$ のような形は存在しない．よって、任意の AND, OR と、各変数についての NOT の計算が可能ならば良い．ここで、AND, OR はすべての ABE で実現されているため、ABE の機能を利用することにより、可能となる．また、各変数にかかる NOT は、提案手法では'0' に対応する $\{a_i\}$ と、'1' に対応する $\{b_i\}$ を用意しているため、$\neg a_i = b_i,\ \neg b_i = a_i$ という関係が成り立ち、すべての変数に対する NOT を実現できるようになる．以上から、加法標準形で表わされる論理式をすべて表現できることが分かる．また、このことから、提案手法は任意の組み合わせ回路を実現可能であり、２つの識別子のみに依存する関係は全て扱うことができることが分かる．

参考までに、$==, !=, <, >$ の一般的な構成法を、図 3.3.3 に示す。

3.3.4　安全性証明

ここでは、先ほど構成した提案方式の安全性評価を行う．

定理 3.1. 提案した鍵規定型の方式は、元になる属性ベース暗号が $(t_{ABE}, \epsilon_{ABE})$-*IND-sAtt-CPA* 安全ならば、
$(t_{ABE}, \epsilon_{ABE})$-*IND-sLabel-CPA* 安全となる．

Proof. (t_L, ϵ_L)-IND-sLabel-CPA 攻撃者 $\mathcal{A}$ を用いて、属性ベース暗号の IND-sAtt-CPA 攻撃者を以下の手順で構成する．

初期化 ：まず、$2n$ 個の乱数 $\{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n}$ を選ぶ．そして、$\mathcal{A}$ が出力した $\mathbb{L}^*$ に対し、Ext アルゴリズムと同様にして $\gamma^*(= \{a_i\}_{\forall i \mathbb{L}^{*i}=0} \cup \{b_j\}_{\forall j \mathbb{L}^{*j}=1})$ を生成し、チャレンジャー $\mathcal{B}_{ABE}$ に送る．$\mathcal{B}_{ABE}$ が公開パラメータ prm' を返してきたら、$\mathcal{A}$ に公開パラメータとして、
$(\mathsf{prm}', \{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n})$ を送る．

第１段階：$\mathcal{A}$ の $(\mathbb{L}_i, R_i)$ に対する Ext クエリを受け、Ext アルゴリズムの処理と同様にアクセス構造 $\mathbb{A}_i$ を構成する．そして、$\mathbb{A}_i$ を $\mathcal{B}_{ABE}$ に送り、d_i を受け取る．その後、$d_{\mathbb{L}_i,R_i}$ として、d_i を $\mathcal{A}$ に送る．

チャレンジ：$\mathcal{A}$ からチャレンジクエリ M_0, M_1 が送られてきたら、それを $\mathcal{B}_{ABE}$ へのチャレンジとして、そのまま送る．$\mathcal{B}_{ABE}$ から暗号文 C^* が返ってきたら、それをそのまま $\mathcal{A}$ に送る．

第２段階：**第１段階**と同様の操作を行う．

推測：最後に、$\mathcal{A}$ が b' を出力したら、それを自分の推測として、$\mathcal{B}_{ABE}$ に b' をそのまま送る．

ここで、$\mathcal{A}$ が攻撃に成功するとき、$\mathcal{A}$ は $\mathcal{B}_{ABE}$ が b として $0, 1$ のどちらを選んだかを正しく推測したことになる．そのため、$\mathcal{A}$ が攻撃に成功するとき、上述のシミュレータによる $\mathcal{B}_{ABE}$ への攻撃も成功することになり、$\mathcal{A}$ と、今回構成したシミュレータの成功確率は等しくなる．よって $\mathcal{A}$ が (t_L, ϵ_L)-IND-sLabel-CPA 攻撃者なら、上述のシミュレータは (t_L, ϵ_L)-IND-sAtt-CPA 攻撃者となる．　□

定理 3.2. 提案した暗号文規定型の方式は、元になる属性ベース暗号が $(t_{ABE}, \epsilon_{ABE})$-*IND-sAtt-CPA* 安全ならば、
$(t_{ABE}, \epsilon_{ABE})$-*IND-sLabel-CPA* 安全となる．

Proof. (t_L, ϵ_L)-IND-sLabel-CPA 攻撃者 $\mathcal{A}$ を用いて、属性ベース暗号の IND-sAtt-CPA 攻撃者を以下の手順で構成する．

初期化：まず、$2n$ 個の乱数 $\{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n}$ を選ぶ．そして、$\mathcal{A}$ が出力した $\mathbb{L}^*, R^*$ に対し、Ext アルゴリズムと同様にして $\gamma^*(= \{a_i\}_{\forall i \mathbb{L}^{*i}=0} \cup \{b_j\}_{\forall j \mathbb{L}^{*j}=1})$ を生成し、チャレンジャー $\mathcal{B}_{ABE}$ に送る．$\mathcal{B}_{ABE}$ が公開パラメータ prm' を返してきたら、$\mathcal{A}$ に公開パラメータとして、
$(\mathsf{prm}', \{a_i\}_{1\leq i\leq n}, \{b_i\}_{1\leq i\leq n})$ を送る．

第１段階：$\mathcal{A}$ の $(\mathbb{L}_i)$ に対する Ext クエリを受けたら、$R^*(\mathbb{L}^*, \mathbb{L}_i) = 0$ を確認してから、$\mathbb{L}_i$ に対応する $\gamma^*(= \{a_i\}_{\forall i \mathbb{L}^{*i}=0} \cup \{b_j\}_{\forall j \mathbb{L}^{*j}=1})$ を生成する．そして、γ_i を $\mathcal{B}_{ABE}$ に送り、d_i を受け取る．その後、$d_{\mathbb{L}_i}$ として、d_i を $\mathcal{A}$ に送る．

チャレンジ：$\mathcal{A}$ からチャレンジクエリ M_0, M_1 が送られてきたら、それを $\mathcal{B}_{ABE}$ へのチャレンジとして、そのまま送る．$\mathcal{B}_{ABE}$ から暗号文 C^* が返ってきたら、それをそのまま $\mathcal{A}$ に送る．

第２段階：**第１段階**と同様の操作を行う．

推測：最後に、$\mathcal{A}$ が b' を出力したら、それを自分の推測として、$\mathcal{B}_{ABE}$ に b' をそのまま送る．

ここで、$\mathcal{A}$ が攻撃に成功するとき、$\mathcal{A}$ は $\mathcal{B}_{ABE}$ が b として $0, 1$ のどちらを選んだかを正しく推測したことになる．そのため、$\mathcal{A}$ が攻撃に成功するとき、上述のシミュレータによる $\mathcal{B}_{ABE}$ への攻撃も成功することになり、$\mathcal{A}$ と、今回構成したシミュレータの成功確率は等しくなる．よって $\mathcal{A}$ が (t_L, ϵ_L)-IND-sLabel-CPA 攻撃者なら、上述のシミュレータは (t_L, ϵ_L)-IND-sAtt-CPA 攻撃者となる．　□

3.4　評価

ここでは、関係'>' を例に取り、従来の IBE を用いて'>' を達成する場合と、今回の構成とを比較する．ここで、今回の比較においては、識別子は各エンティティにつき 1 つのみとするため、属性ベース暗号ではなく ID ベース暗号で考えれば十分である．

ID ベース暗号を用いて、'>' を達成するには、例えば ID 空間が $\{0, 1, \ldots, 15\}$ の場合に、'8' より大きい ID に対して復号可能にするためには、ID$9, 10, 11, 12, 13, 14, 15$ に対して復号できるようにすれば良い．よって、ID 空間に要素が t 個あり、s より大きいものを復号するためには、$t - s$ 個の ID に対する鍵が必要になり、平均で $\frac{t}{2}$ 個の鍵が必要になる．

このような方式と今回提案した方式を比較するために、ID ベース暗号として、最も簡単なランダムオラクルを用いた [5] で提案されたものを用い、属性ベース暗号として、最初に提案された [20] を用いる．そして、識別子のビット長を n とする．このとき、ID 空間のサイズは 2^n となるため、必要な鍵の数は $\frac{2^n}{2} = 2^{n-1}$ となる．このような条件で復号鍵のサイズ、暗号文のサイズ、暗号化にかかるコスト、復号にかかるコストを比較した結果は、表 3.4 のようになる．

表 3.1: '>' を実現する方式の平均サイズ・コストの比較

	from IBE	提案方式
復号鍵サイズ	$\underline{2^{n-1}\mathbb{G}}$	$\underline{\frac{n^2}{2}\mathbb{G}}$
暗号文サイズ	$\mathbb{G} + \mathbb{G}_T$	$n + n \cdot \mathbb{G} + \mathbb{G}_T$
暗号化コスト	$\mathbb{P} + 2\mathsf{PM} + \mathsf{exp}_{\mathbb{G}_T}$	$n \cdot \mathsf{PM} + \mathsf{exp}_{\mathbb{G}_T}$
復号コスト	$\mathbb{P}$	$\frac{n^2}{2}\mathbb{P} + \mathsf{exp}_{\mathbb{G}_T}$

$\mathbb{G}$: 双線形写像の元となる群の要素のサイズ．
$\mathbb{G}_T$: 双線形写像により作られる群の要素のサイズ．
（双線形写像 e:$\mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$）
PM: $\mathbb{G}$ 上の Point Multiplication にかかるコスト．
$\mathbb{P}$:双線形写像の計算にかかるコスト．

表 3.4 から、復号鍵サイズに関して、ID ベース暗号から構成した方式では n の指数オーダーのサイズが必要になり、n が大きくなると実現不可能となる．一方、提案方式では復号鍵サイズは n の 2 乗のオーダーになっており、他も n の多項式オーダーとなっているため、現実的であることが分かる．

Chapter 4　ブラインド属性ベース暗号

4.1　意義

識別子利用暗号方式は，従来，鍵導出の際に必要な情報を直接鍵生成センターに送って，その識別子に対応する秘密鍵の導出を行っている．この場合，鍵生成センターは，各エンティティがどのような情報を用いて鍵を生成したか分かるという問題が生じる．

この問題を解決する上で有効と考えられるのが，Green らによって 2007 年の ASIACRYPT で発表された “Blind Identity Based Encryption(BIBE)” という概念である [21]．これは，鍵導出の際に，エンティティの ID をセンターに知らせることなく，エンティティの ID に対応した秘密鍵を生成させることができるというものである．[21] は，実際には BIBE を用いて Oblivious Transfer(OT)[27] を構成するという研究で，BIBE は基本的には OT を構成するためのツールとして用いられている．

本研究では，BIBE を匿名性を持たせた識別子利用暗号のために利用することを考える．しかし，BIBE を直接用いると，エンティティは好きな ID に対する秘密鍵を作ることが可能になってしまう (図 4.1)．そこで，本研究では，エンティティは自身の ID はセンターに送り本人証明を行うが，エンティティがセンターに見せたくない情報は隠すことができるようにする方式を提案する．具体的には，本研究においては，暗号文規定型の方式に属性の匿名性を与えた方式を示す (図 4.1)．

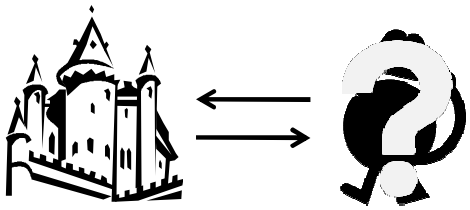

図 4.1: ブラインド IBE の場合 (センターはユーザに関する情報が全く分からないため，鍵を渡す相手の認証ができない．)

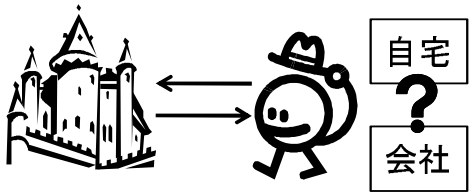

図 4.2: 提案方式 (センターはユーザの特定はできるが，鍵導出の際の付帯情報は分からない．)

4.2　関連研究

ここでは、センターに対する安全性について議論している関連研究について、説明を行う。

4.2.1　Certificateless Encryption

Certificateless Encryption[1, 2, 3, 23] は、公開鍵暗号における鍵のすり替えという問題と、ID ベース暗号における鍵供託問題の両者を防ぐ方式として、提案された [1]。この方式は、鍵導出の際に、センターは受信者に部分秘密鍵と呼ばれる秘密鍵の一部のみを生成し、最後に受信者自身が乱数を選び、秘密鍵を完成させるため、センターも秘密鍵のすべての秘密情報を得ることはできない。そのため、ID ベース暗号のように鍵供託の問題は発生しない。また、公開鍵として、受信者の ID と公開鍵という 2 種類の鍵を用いるため、攻撃者による鍵のすり替えも防ぐことが可能となる。

また、この方式では、センターに暗号文を解読されないという安全性を議論していて、センターに識別子を見られないという安全性については何も言及していないため、本研究とは目的が異なっている。

4.2.2　鍵偽造の追跡可能な ID ベース暗号

この方式は、受信者の秘密鍵が漏洩した際に、鍵を漏洩したのは受信者自身か、センターが偽造したものかを区別できるというものである [19]。これは、前述の Certificateless Encryption より弱い安全性だが、その分効率の高い方式となっている。また、この方式も、センターに ID を見せるか否かは言及していないため、本研究とは目的が異なっている。

4.2.3　ブラインド ID ベース暗号

Green らによって今年 (2007 年) の ASIACRYPT で提案された概念である．実際には，Green らは IBE に匿名性を付与するのが主な目的ではなく，Blind IBE があれば，Oblivious Transfer を構成できるという主張をするために，その構成要素として Blind IBE という概念を提案した．ここで，Blind IBE を鍵導出の際に匿名性を付与するために利用しようとすると，センターは自分がどの識別子に対し鍵導出をしているか分からないため，逆にクライアント側に，ID の詐称をされる可能性がある．そのため，Blind IBE は実際に IBE の匿名性を確保するものではなかったため,「IBE に匿名性を付与した」という主張はされなかったと考えられる．

そこで本稿では，センターがクライアントの認証をするために，鍵生成の際に識別子は受け取るものとし，それ以外の情報を付加して鍵生成を行う場合に，匿名性が付与できるようにすることを考えた．ここで，鍵生成の際の付加情報として，クライアントの居場所や利用端末の種類など，各エンティティのプライバシーに関わるものも考えられるので，今回の提案方式はプライバシーの保護にもつながると考えられる．

4.3　定式化

4.3.1　安全性定義

他の多くの公開鍵暗号方式と同様に，安全性の証明を行う際に，攻撃者とチャレンジャーの間でゲームを行うことを考える．CP-ABE 方式の IND-sAtt-CPA の安全性を評価する際には，以下のゲームを用いる．

CP-ABE における IND-sAtt-CPA ゲーム

初期化 攻撃者 $\mathcal{A}$ は，攻撃対象とする γ^* をチャレンジャー $\mathcal{B}$ に送る．$\mathcal{B}$ は γ^* を受け取った後，公開パラメータ prm を $\mathcal{A}$ に送る．

第１段階 $\mathcal{A}$ は $\mathcal{B}$ に $(\gamma_i, \mathbb{A}_i)$ $(\gamma^* \notin \mathbb{A}_i)$ を送り，$\mathcal{B}$ はそれに対応する d_i を $\mathcal{A}$ に送る．

チャレンジ $\mathcal{A}$ はメッセージ空間 $\mathcal{M}$ から M_0, M_1 を選び，$\mathcal{B}$ に送る．$\mathcal{B}$ は M_0, M_1 を受け取った後，b を $\{0,1\}$ からランダムに選び，γ^* を用いて，M_b を暗号化して，得られた暗号文 C を $\mathcal{A}$ に送る．

第２段階 「第１段階」と同様の試行を繰り返す．

推測 最後に，$\mathcal{A}$ は $b' \in \{0,1\}$ を出力する．

定義 4.1. すべての t 時間アルゴリズム $\mathcal{A}$ に対し，$\left|\Pr[b = b'] - \frac{1}{2}\right| < \epsilon$ のとき，*CP-ABE* 方式は (t, ϵ)-*IND-sAtt-CPA* 安全であるという．

ブラインドネス

本稿では，[21] と同様に，鍵導出アルゴリズムを変更することで，匿名性 (Blindness) を付与することを考える．この際，実際にはセンターとクライアントの間で実行するプロトコルに変えて属性に匿名性を付与する．センターとクライアントは，以下のように，それぞれ $\mathcal{C}$ と $\mathcal{U}$ を実行し鍵生成を行う．

$BlindExt(\mathcal{C}(\mathsf{prm}, \mathsf{msk}), \mathcal{U}(\mathsf{prm}, \mathsf{Att}))$ ：$\mathcal{C}$ は公開パラメータ $sfprm$ とマスター鍵 msk を用いて $\mathcal{U}$ とプロトコルを走らせ，結果的には出力として何も得ない．$\mathcal{U}$ は公開パラメータ prm とセンターに秘匿にしたい属性 Att を用いて $\mathcal{C}$ とプロトコルを走らせ，その結果として Att に対応する秘密鍵 d を得る．

そして，このブラインドネスを検証するため，属性ベース暗号方式のブラインドネスのゲームを以下のように定義する．

1. 攻撃者 $\mathcal{A}$ は，攻撃対象とする属性 $\mathsf{Att}_1, \mathsf{Att}_2$ をチャレンジャー $\mathcal{B}$ に送る．

2. $\mathcal{B}$ は $Setup(1^k)$ アルゴリズムを実行し prm を $\mathcal{A}$ に送った後，$b \in \{0,1\}$ をランダムに選ぶ．

3. $\mathcal{B}$ は $\mathcal{U}_0(\mathsf{prm}, \mathsf{Att}_b), \mathcal{U}_1(\mathsf{prm}, \mathsf{Att}_{1-b})$ を，$\mathcal{C}(\mathsf{prm}, \mathsf{msk})$ と実行する．

4. プロトコルを実行した結果，$\mathcal{B}$ は $d_0 \leftarrow \mathcal{U}_0, d_1 \leftarrow \mathcal{U}_1$ を得て，(d_0, d_1) を $\mathcal{A}$ に送る．

5. 最後に $\mathcal{A}$ は $b' \in \{0, 1\}$ を選ぶ．

定義 4.2 (**ブラインドネス**)．上記のゲームで，すべての多項式時間アルゴリズム $\mathcal{A}$ に対し，$\left|\Pr[b = b'] - \frac{1}{2}\right|$ が k のネグリジブル関数となっているとき，その属性ベース暗号方式はブラインドネスを持つという．

Leak Freeness

前述のブラインドネスがセンターに対する安全性とすると、Leak Freeness はクライアントに対する安全性である。これは、鍵導出をプロトコルに変更した場合でも、クライアントが得られる情報は従来の鍵導出アルゴリズムにより得られる情報と変わらない。つまり、「鍵導出プロトコルを通して、クライアントが得られる情報は復号鍵についての情報のみである」ことを保証する安全性である (図 4.3.1)。この安全性は以下のように定義する。

1. まず、現実的なゲーム (G_0) と、理想的なゲーム (G_1) を考える。

2. G_0 では、実際に鍵導出プロトコルを実行し、その過程におけるクライアントを観察する。

3. G_1 では、プロトコルを実行する代わりに、信用できる第三者に直接属性情報を渡し、その情報をもとに鍵導出アルゴリズムを実行し、得られた復号鍵をクライアントに返すという状況を考える。そして、この際のクライアントの観察を行う。

4. ここで、G_0、G_1 ともにクライアントを観察しているが、そのクライアントのみを観察していた場合に、G_0 と G_1 の区別をできる識別者が存在する場合、安全性は破られたとし、識別することができないとき、安全であるということにする。

4.3.2　ゼロ知識証明

今回用いるゼロ知識証明は，以下の 3 種類である．

1. 素数を法とした離散対数の知識の証明 [31]

2. コミットされた値が既定の範囲の中であることの証明 [10, 8, **?**]

3. 既出の 2 つの任意の論理和・論理積であることの証明 [13]

これらは，2. の中には強 RSA 仮定を要求するものもあるが，基本的には離散対数仮定だけですべて証明可能である．

これらの証明について，本稿では [9] の記述法に従うものとする．つまり，$PoK\{(x, r) : y = g^x \cdot h^r \wedge (1 \leq x \leq n)\}$ が，x と r が $y = g^x \cdot h^r$ かつ $1 \leq x \leq n$ であるという知識のゼロ知識証明を表すとする．

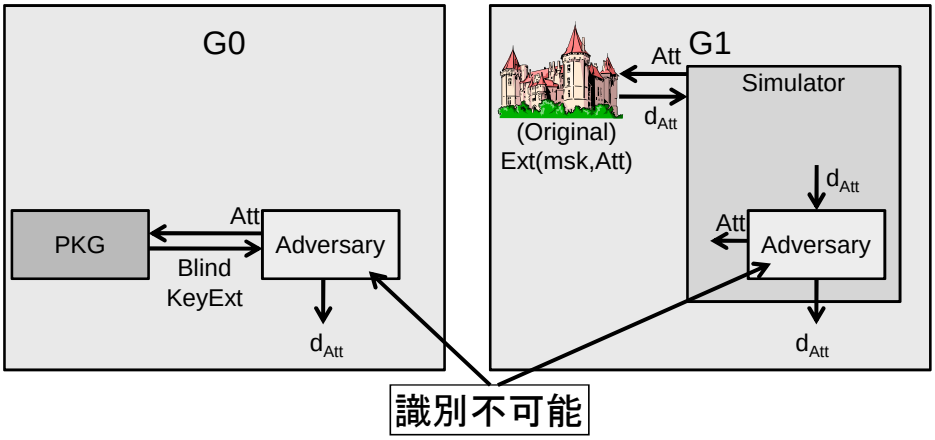

図 4.3: Leak Freeness の概要
G_0 と G_1 の区別がつかない

4.3.3　コミットメント

コミットメントは以下の 3 つのアルゴリズムから構成される．

$CSetup(1^k)$ ：公開パラメータ prm を生成する．

$Commit(\mathsf{prm}, M)$ ：公開パラメータ prm と平文 M を受け取り，$(\mathcal{C}, \mathcal{D})$ の組を出力する．

$Decommit(\mathsf{prm}, M, \mathcal{C}, \mathcal{D})$ ：公開パラメータ prm，平文 M，$(\mathcal{C}, \mathcal{D})$ を受け取り，$\mathcal{D}$ により $\mathcal{C}$ が M にデコミットされたら 1 を返し，それ以外の場合には 0 を返す．

今回，我々は [21] と同様に離散対数仮定に基づいた Pederson の方式 [26] を用いる．この方式を簡単に説明すると，素数オーダー q の群とその生成元 (g_0, g_1) を公開パラメータとして，$v \in \mathbb{Z}_q$ をコミットすることを考える．このとき，$r \in \mathbb{Z}_q$ をランダムに選び，$\mathcal{C} = g_0^r \cdot g_1^v, \mathcal{D} = r$ とするという方式である．この方式は，$D = r$ という知識を証明するために，[31] のテクニックを効率よく用いることができる．

4.4　構成

まず，本研究で基にした暗号文規定型の属性ベース暗号方式を説明する．本研究では，Bethencourt らによって提案された方式をもとにしている [4]．そのアルゴリズムは以下のようになっている．

$Setup(1^k)$ ：素数オーダー p の双線形写像の群 $\mathbb{G}_0$ を選び，その生成元を g とする．また $H\colon \{0,1\}^* \to \mathbb{Z}_p$ をランダムオラクルとする．α, β を $\mathbb{Z}_p$ からランダムに選ぶ．
そして，
$\mathsf{prm} = (\mathbb{G}_0, H, g, h = g^\beta, f = g^{1/\beta}, e(g,g)^\alpha)$
$\mathsf{msk} = (\beta, g^\alpha)$
とする．

$Ext(\mathsf{msk}, \gamma)$: r, r_i $(i \in \gamma)$ を $\mathbb{Z}_p$ からランダムに選ぶ．そして，γ に対応する秘密鍵は
$d = (g^{(\alpha+r)/\beta},\ \forall i \in \gamma \colon g^r \cdot h^{H(i)\cdot r_i}, g^{r_i})$
とする．

$Enc(M, \mathsf{prm}, \mathbb{A})$: 公開パラメータを $\mathsf{prm} = (\mathbb{G}_0, g, h, f, Z)$ とみなす．まず，アクセス構造 $\mathbb{A}$ の各ノード x に対して，多項式 q_x をランダムに選ぶ．この際，q_x の次数 d_x は，そのノードのしきい値を k_x としたときに，$d_x = k_x - 1$ となるように定める．そして，ルートノード R から多項式を選ぶが，その際，R では，s を $\mathbb{Z}_p$ からランダムに選び，$q_R(0) = s$ となるようにする．他のノード x については，$q_x(0) = q_{\mathsf{parent}(x)}(\mathsf{index}(x))$ を満たすように選ぶ．Y を $\mathbb{A}$ の葉ノードの集合とする．この時，暗号文 C は，

$$C = (\mathbb{A}, M \cdot Z^s, h^s, \\ \forall\, y \in Y \colon g^{q_y(0)}, h^{H(\mathsf{att}(y))\cdot q_y(0)})$$

となる．

$Dec(C, d)$: 暗号文を $C = (\mathbb{A}, C_1, C_2, \{C_{y1}, C_{y2}\})$，秘密鍵を $d = (D, \{D_{j1}, D_{j2}\})$ とみなす．まず，$\mathbb{A}$ の葉ノード y については，以下の計算を行う．
$F_y = \frac{e(D_{j1}, C_{y1})}{e(D_{j2}, C_{y2})}$
次に，それ以外のノード x では，その子ノードの計算結果を F_z(葉ノードの計算結果 F_y も含む) としたとき，そのノードのしきい値 k_x だけ F_z を集め (その集合を S_x とする)，次の計算を行う．$(i = \mathsf{index}(z), S'_x = \{\mathsf{index}(z) \colon z \in S_x\})$

$$\begin{aligned} F_x &= \prod_{z \in S_x} F_z^{\Delta_{i,S'_x}(0)} \\ &= e(g,g)^{r \cdot q_x(0)} \end{aligned}$$

この計算を繰り返すと，ルートノード R では $F_R = e(g,g)^{r\cdot q_R(0)} = e(g,g)^{rs}$ が得られる．そこで次の計算を行い，もとの平文を得る．
$C_1/(e(C_2, D)/e(g,g)^{rs}) = C_1/e(g,g)^{\alpha\cdot s}$
この式から，正しく暗号化された暗号文が送られて来た場合，元のメッセージを得ることができる．

上記のアルゴリズムは [?] で，IND-sAtt-CPA を満たすことが証明されている．なお，[?] では，ランダムオラクルは $H \colon \{0,1\}^* \to \mathbb{G}_0$ としていたが，証明の際に，$H' \colon \{0,1\}^* \to \mathbb{Z}_p$ というランダムオラクルを用いて，$H(x) = g^{H'(x)}$ とするため，本稿における変更は，計算速度を多少遅くさせるのみで，安全性に影響を及ぼさない．本研究では，このアルゴリズムの「鍵導出」の部分のみ改変し，ブラインドネスを付与する．実際には，鍵導出の際，センターとクライアントとの間でプロトコルを実行することにより，ブラインドネスを達成している．

4.4.1　鍵生成プロトコル

鍵生成の際に，センターとクライアントの間で走らせるプロトコルについて，説明を行う．これは，[?] とほぼ同じ方法で構成することができる．概要としては，まずクライアントはセンターに知らせたくない属性を選び，コミットメントとゼロ知識証明を用いてセンターに属性を知らせずにその属性に対応する鍵を導出させるという流れである．具体的には，図 4.4.1 ようなプロトコルを実行する．このプロトコルを走らせた

$\mathcal{C}(\mathsf{prm}, \mathsf{msk})$	$\mathcal{U}(\mathsf{prm}, \mathsf{Att})$
	1. y をランダムに選ぶ．
	2. $g' = h^{H(\mathsf{Att})} \cdot g^y$ を $\mathcal{C}$ に送る．
	3. $PoK\{(y, H(\mathsf{Att})) \colon g' = h^{H(\mathsf{Att})} \cdot g^y\}$ を実行する．
4. 証明が失敗した場合，停止する．	
5. r' をランダムに選ぶ．また，r を D を計算する際に用いた乱数とする．	
6. $d_1' = g^r \cdot g'^{r'}$ $d_2' = g^{r'}$ を $\mathcal{U}$ に送る．	
	7. $e(h, D) \cdot e(d_2', g')/e(g, g)^{\alpha} = e(g, d_1')$ を満たすかどうか検証する．
	8. 正しく検証ができたら，z をランダムに選ぶ．
	9. $d_1 = d_1'/(d_2'^{y}) \cdot h^{H(\mathsf{Att})+z}$，$d_2 = d_2' \cdot g^z$ とする．
	10. $d = (d_1, d_2)$ として出力する．

図 4.4: 提案した $BlindExt$

結果,

$$
\begin{aligned}
d_1 &= \frac{d_1'}{d_2'^{y}} \cdot h^{H(\mathsf{Att})+z} \\
&= \frac{(g^r \cdot (h^{H(\mathsf{Att})} \cdot g^y)^{r'})}{g^{r' \cdot y}} \cdot h^{H(\mathsf{Att}) \cdot z} \\
&= g^r \cdot h^{H(\mathsf{Att}) \cdot r'} \cdot h^{H(\mathsf{Att}) \cdot z} \\
&= g^r \cdot h^{H(\mathsf{Att}) \cdot (r'+z)} \\
d_2 &= d_2' \cdot g^z \\
&= g^{r'+z}
\end{aligned}
$$

となり，[4] の鍵生成アルゴリズムで選ぶ r_i の代わりに $r' + z$ となって，d_1, d_2 が生成されることが分かる．さらに，上記のプロトコルは [21] で提案されているプロトコル

と同じ形とみなすことができ，ブラインドネスが保証されることが分かる．以下に証明の概要を示す．

3.3 で定義したブラインドネスのゲームを行うことを考える．このとき，$\mathcal{U}$ で選ぶ Att に関して，コミットメントからクライアント側では途中で Att の値を変更することはできないことが保証され，ゼロ知識証明からセンター側には Att の情報が漏れていないことが保証される．その上で，$\mathcal{C}$ は d'_1, d'_2 を計算し，(d'_1, d'_2) を $\mathcal{U}$ に返す．この時点では，センターは $\mathcal{U}_0$,$\mathcal{U}_1$ から受け取った情報の区別はつけることができる（送られてきた値の区別であり，Att を直接知ることはできない）．その後，$\mathcal{U}$ は z を選び，d'_1, d'_2 をそれぞれ群の要素の z 乗でマスクした形となっている．そのマスクしたものをチャレンジとして $\mathcal{C}$ に送るので，離散対数仮定を置けば，$\mathcal{C}$ は $\mathcal{U}_0$ と $\mathcal{U}_1$ それぞれの出力の見分けがつかなくなる．

4.5　考察

以上から，暗号文規定型の属性ベース暗号方式に匿名性を付与することができた．従来の識別子利用暗号方式では，センターは直接鍵生成を行い，なおかつマスター鍵を持っている．そのため，各エンティティの保持している鍵や通信路上を流れている暗号文の元の平文といった秘密情報を全て知ることが可能となる (key-escrow)．この問題を解決するために，今までは Certificateless Encryption(CLE)[?] や鍵生成センターに必要な信頼度を下げる方法 [19] などが考えられてきた．CLE は，センターに「なり済まし」も「盗聴」もさせない強い方式であるが，その分完全な安全性を達成するのは難しい．それに対し，[19] はセンターの「盗聴」は防げないが,「なり済まし」や「鍵の漏えい」に関して安全な方式となっており，CLE よりも構成が容易になっている．

今回我々が提案した方式は，このようなセンターに対する安全性としては，[?] と同等の安全性を確保することが可能となる．つまり，センターはマスター鍵を用いれば任意の暗号文を復号することが可能となる．しかし，鍵生成の際に，クライアント側がセンターに見えない情報を付加して鍵生成を行うために，センターが鍵を偽造する，つまり「なり済まし」を行うのは困難となる．実際には，この安全性を達成するためには，センターが通信路上の暗号文を見て，暗号化の際に用いられた属性（アクセス構造）が分からないという性質が必要になるが，これは見せたくない属性のみ暗号文に乗せないという方法で比較的容易に達成可能と考えられる．さらに，今回の方式では，上記のような安全性を確保するだけでなく，より多くの応用があると考えられる．その根拠として，鍵生成の際に，クライアントは自分の ID を見せるだけで，他の属性情報はセンターに秘匿のまま任意に導出することが可能となる (図 4.5)．

そして，暗号化を行う際には，センターに知られていない属性情報によるアクセス構造で制御が可能なため，センターに情報を漏らさずに，識別子利用暗号による通信が行える．このことは，センターに対する安全性を保持したまま，属性ベース暗号の登場により達成されてきている「鍵の制御」を行うことができるということに繋がると考えられる．

本稿では，属性ベース暗号方式の鍵導出の際に，属性に秘匿性を持たせる方法を提

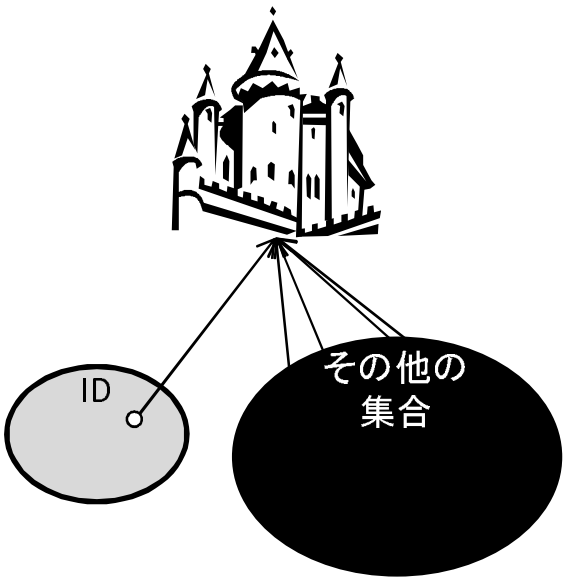

図 4.5: センターから見える鍵集合 (センターからは ID のみ見え，他の属性情報に関しては，クライアントがいくつの属性に関して鍵導出を行ったかのみ分かる．)

案した．これにより，識別子利用暗号の弱点であったセンターに対する安全性を確保する 1 つの方法を示したことになる．なお，今回は暗号文規定型の方式に対してブラインドネスを達成したため，今後の課題として，鍵規定型の属性ベース暗号方式にブラインドネスを持たせた方式を構築したいと思う．また，5 でも触れたが，この方式では，鍵導出の際にセンターに対して秘匿性を持たせたが，実際に秘匿性を実現するためには，センターが通信路上の暗号文を見た際にもその暗号文に使われた属性（アクセス構造）が分からないという条件が必要となるので，この点についても，今後検討を行っていきたいと思う．

Chapter 5　情報理論的安全性を持つ Chaffing-and-Winnowing

5.1　概要

5.1.1　目的

本章では、平文空間が多項式オーダーのサイズに制限されている場合の秘匿性に関する研究について説明する。この研究により、識別子空間が多項式オーダーのサイズの識別子利用暗号における、識別子の秘匿性を論じることが可能となる。具体的には、平文空間がある程度制限されている場合における、情報理論的安全性について論じる、Chaffing-and-Winnowing という技術に焦点を当てた研究である。Chaffing-and-Winnowing とは、情報理論的安全性を持つ暗号技術と同等の機能を達成する手法である。ただ、復号の際に、平文空間すべての要素に対して処理を行わなければならない。そのため、その平文空間はセキュリティパラメータの多項式オーダーに限られてしまう。この状況における、平文の匿名性を論じることは、多項式オーダーに空間が限定された場合の識別子の匿名性につながる。

5.1.2　Chaffing-and-Winnowing

Chaffing-and-Winnowing は、従来の暗号化とは異なるアプローチで秘匿性を持たせる方法として提案された [28]。この方法は、処理の重い従来の暗号化を用いずに秘匿性を持たせるというものである。その代わり、復号の際に、平文空間のすべての要素に対して操作を行わなければならない。本研究では、鍵交換頻度を低く抑えることが可能な Chaffing-and-Winnowing の構成を目指す。

5.1.3　関連研究

情報理論的安全性を持つ Chaffing-and-Winnowing を、任意の認証子から構成可能であることが既に示されている [22]。しかし、この方法では、1 つのメッセージを送るたびに新たな鍵を用意しなければならず、鍵交換のコストが高くなってしまうという欠点があった。そこで、本研究では、鍵交換の頻度を低くすることが可能な方式提案する。

5.2　用語定義

5.2.1　情報理論的に安全な認証子 (A-code)

認証子は、あるメッセージが改ざんされることなく、オリジナルのメッセージが届いたことを保証するための技術である。これは、送信者 S、受信者 R がいる状況を考える。まず、秘密鍵 u と v を生成し、u を R が、v を S が保持するようにする。S がメッセージ M を送る際、まず M と u から認証子 α を生成し、(M, α) を R に送る。R は、M と v から α の検証を行う。

本章では、1 回の鍵交換で、n 回 A-code を用いることを考える。この際、A-code の安全性として、“spoofing” という概念を考える。この定義は以下のようになっている。

定義 5.1. 秘密鍵空間を $\mathcal{U}, \mathcal{V}$、平文空間を $\mathcal{M}$、認証子空間を $\mathcal{A}$ とする。また、それぞれの空間から得られる変数をそれぞれ u, v, m, α とする。このとき、$(\mathcal{U}, \mathcal{V}, \mathcal{M}, \mathcal{A})$ が、以下の条件を満たすとき、(p, n)-*spoofing* に対して安全 *(*(p, n)-Spf*)* であるという。

A-code を n 回利用する状況で、ある確率 p について、

$$\forall i \in \{0, \ldots, n\}, \max_{\{(m_j, \alpha_j)\}_{0 \le j \le i}} \max_{(m,\alpha)(\notin \{(m_j,\alpha_j)\}_{0 \le j \le i})} \Pr[R \textit{ accepts } (m, \alpha) | (m_0, \alpha_0), \ldots, (m_i, \alpha_i)] \le p,$$

が成り立つ。ただし、(m_i, α_i) $(i = 1, \ldots, n)$ は、S によって送られたメッセージを表し、(m_0, α_0) は *1* 回もメッセージが送られていない状態を表すものとする。

ここから、$f : \mathcal{M} \times \mathcal{U} \to \mathcal{A}$ は、$f(m, u) = \alpha$ を得る写像とする。

5.2.2　情報理論的に安全な暗号化方式

ここでは、n 回のメッセージの送信を行う情報理論的に安全な暗号化方式について説明する。まず、送信者 S、受信者 R を想定する。そして、暗号鍵 e を S が、復号鍵 d を R が保持するようにする。i 回目 $(1 \le i \le n)$ のメッセージ M_i の送信について考える。S は e を用いて M_i を暗号化し、得られた暗号文 C_i を R に送る。R は d を用いて C_i を復号する。

なお、暗号化の安全性の定義の際、ある乱数空間 $\mathcal{X}$ に対し、$H(\mathcal{X})$ を $\mathcal{X}$ のエントロピーを表すものとする。また、$X := \{x | \Pr[\mathcal{X} = x] > 0\}$ なる X について、$|X|$ を X の濃度を表すものとする。

定義 5.2. i 回目の暗号鍵空間を $\mathcal{E}$、復号鍵空間を $\mathcal{D}$、平文空間を $\mathcal{M}_i$、暗号文空間を $\mathcal{C}_i$ とし、それぞれの変数を e, d, m_i, c_i とする。以下の条件を満たすとき、$(\mathcal{E}, \mathcal{D}, \{\mathcal{M}_i, \mathcal{C}_i\}_{1 \le i \le n})$ は n-*perfect secrecy (*n-PS*)* を満たすとする。

1. すべての i について、R は c_i から m_i を正しく復号できる。つまり、$\forall i \in \{1, \ldots, n\},\ H(\mathcal{M}_i | \mathcal{C}_i, \mathcal{D}) = 0$。

2. すべての i について、どんな攻撃者も、c_i から m_i に関する情報を全く得られない。つまり、

$$H(\mathcal{M}_1|\mathcal{C}_1) = H(\mathcal{M}_1),$$

かつ

$$\forall i \in \{2, \ldots, n\},$$
$$H(\mathcal{M}_i|\mathcal{C}_i, (\mathcal{M}_1, \mathcal{C}_1), \ldots, (\mathcal{M}_{i-1}, \mathcal{C}_{i-1})) = H(\mathcal{M}_i|(\mathcal{M}_1, \mathcal{C}_1), \ldots, (\mathcal{M}_{i-1}, \mathcal{C}_{i-1})).$$

また、*n-perfect-non-malleability(*n-NM*)* は、以下のように定義する。

3. どんな攻撃者も、n 回全体を通して、ある平文 m と意味のある関係のある平文に対応する暗号文を作ることができない。つまり、

$$\forall i \in \{1, \ldots, n\},$$
$$H(\hat{\mathcal{M}}|\hat{\mathcal{C}}, (\mathcal{M}_1, \mathcal{C}_1), \ldots, (\mathcal{M}_i, \mathcal{C}_i)) = H(\hat{\mathcal{M}}|(\mathcal{M}_1, \mathcal{C}_1), \ldots, (\mathcal{M}_i, \mathcal{C}_i)),$$

ここで、$\hat{c}(\notin \{c_i\}_{1\leq i\leq n})$ を本来とは異なる暗号文とし、$\hat{m}(\notin \{m_i\}_{1\leq i\leq n})$ はそれに対応する平文とする。また、$\hat{\mathcal{C}}$ と $\hat{\mathcal{M}}$ はそれぞれの空間とする。

ここで、$(\mathcal{E}, \mathcal{D}, \{\mathcal{M}_i, \mathcal{C}_i\}_{1\leq i\leq n})$ が n-PS と n-NM の両者を満たすとき、n-PS&NM を満たすと表記する。

5.3　情報理論的に安全な Chaffing-and-Winnowing

ここでは、1 つの鍵で n 回の利用に対して安全な、情報理論的安全性を持つ Chaffing-and-Winnowing(USCW) の構成について説明する。

本研究では、説明を簡単にするために、オプティマルな (p, n)-Spf な A-code を用いることにする。つまり、$p = 1/|A| = 1/|U|^{1/(n+1)}$ を満たす A-code を用いる。

---

n-PS&NM な情報理論的に安全な A-code の構成

鍵生成基にする A-code を $(\mathcal{U}, \mathcal{V}, \mathcal{M}, \mathcal{A})$ とする。$u \in U, v \in V$ をそれぞれ暗号鍵と復号鍵とし、平文空間を M とする。u と v を、それぞれ S と R に保持させる。S は $|M|$ 個の異なる鍵 $u_1, ..., u_{|M|}$ を、$U\backslash\{u\}$ から次のように選ぶ。

$$\forall u_i, u_j(\neq u_i),\ \forall m \in M,\ f(m, u_i) \neq f(m, u_j).$$

暗号化平文を、$m^* \in M$ とする。S は、$\alpha := f(m^*, u)$ を計算し、$f(m^*, u_i) = \alpha$ となる u_i を見つける。その上で、S は $c := (m||\alpha_m)_{m\in M}$ を R に送る。

復号 R が受け取った暗号文を $c' := (m||\alpha_m)_{m\in M}$ とする。R は、α が v によって正しく検証される m' を探し、m' をもとの平文として出力する。

ここで、安全性の証明を行う前に、構成の条件を満たす $u_1, ..., u_{|M|}$ が $(1/|M|, n)$-Spf の A-code に常に存在することを示す。

補題 5.1. $(\mathcal{U}, \mathcal{V}, \mathcal{M}, \mathcal{A})$ が $(1/|M|, n)$-Spf のとき、すべての $u \in U$ について、$u_i, u_j(\neq u_i) \in \{u_1, ..., u_{|M|}\} \forall m \in M, f(m, u_i) \neq f(m, u_j)$ を満たす $u_1, ..., u_{|M|} \in U\backslash\{u\}$ が存在する。

Proof. 補題 5.1 を示す。まず、u_1 を $U\backslash\{u\}$ からランダムに選ぶ。そして、すべての m について、$U_{1,m} := \{u|f(m, u_1) = f(m, u)\}$ とする。もとの A-code が $(1/|M|, n)$-Spf であるため、$|U_{1,m}| \leq |M|^n$ は明らかである。このことから、

$$|U\backslash \cup_{m\in M} U_{1,m}| \geq |M| - 1.$$

が分かる。次に、$u_2, ..., u_{|M|} \in U\backslash \cup_{m\in M} U_{1,m}$ をランダムに選び、$i = 2, ..., |M|, m \in M$ について、$U_{i,m} := \{u|f(m, u_i) = f(m, u)\}$ とする。ここで、もしある $m_0 \in M$ について、$f(m_0, u_{i_0}) = f(m_0, u_{i_1})$ となるような $u_{i_0}, u_{i_1}(\neq u_{i_0}) \in \{u_2, ..., u_{|M|}\}$ が存在した場合、つまり、$U_{i_0,m_0} = U_{i_1,m_0}$ となる場合、これは $|\cup_{2\leq i\leq |M|} U_{i,m_0}| \leq (|M| - 2)|M|^n$ も意味することになる。しかし、$U = \cup_{\alpha\in A}\{u|f(u, m_0) = \alpha\}$ であり、その結果

$$|U| = |\cup_{\alpha\in A} \{u|f(u, m_0) = \alpha\}| = |\cup_{1\leq i\leq |M|} U_{i,m_0}| \leq (|M| - 1)|M|^n.$$

となる。しかし、これは $|U| = |M|^{(n+1)}$ から矛盾となる。よって、すべての $i_0, i_1(\neq i_0), m \in M$ について、$f(m_0, u_{i_0}) \neq f(m_0, u_{i_1})$ となる。 □

次に、このような $u_1, ..., u_{|M|}$ と任意の $m \in M$ に対し、$f(m, u) = f(m, u_i)$ となる u_i がただ 1 つだけ存在することを示す。

補題 5.2. 任意の $u \in U$、$u_1, ..., u_{|M|}$、$m \in M$ について、$|\{u_i|f(m, u_i) = f(m, u),\ u_i \in \{u_1, ..., u_{|M|}\}\}| = 1$

Proof. この補題が偽だと仮定する。このとき、$f(m, u_{i_0}) = f(m, u_{i_1})$ を満たす $u_{i_0}, u_{i_1} \in \{u_1, ..., u_{|M|}\}$ が存在することになる。しかし、これは矛盾である。よって題意は示された。 □

補題 5.1、5.2 から、任意の u と m について、提案方式は実行可能であることが分かる。

次に、安全性の証明を行う。安全性を証明するために、提案方式が $H(\mathcal{M}_1) = H(\mathcal{M}_1|\mathcal{C}_1)$ と n-NM を満たすことを示す。

補題 5.3. 提案方式は $H(\mathcal{M}_1) = H(\mathcal{M}_1|\mathcal{C}_1)$ を満たす。ここで、$\mathcal{M}_1$ は m_1 の空間である。

Proof. $\mathcal{U}_i$ を $f(m_1,u) = f(m_1,u_i)$ を満たす u_i の空間とする。ここで、$H(\mathcal{C}_1|\mathcal{U}_i) = 0$ から、

$$H(\mathcal{M}_1|\mathcal{U}_i) - H(\mathcal{M}_1|\mathcal{C}_1,\mathcal{U}_i) = H(\mathcal{M}_1|\mathcal{U}_i) - H(\mathcal{M}_1|\mathcal{U}_i) = 0.$$

を得る。ここで、$H(\mathcal{M}_1|\mathcal{U}_i) = H(\mathcal{M}_1)$ のとき、

$$\begin{aligned} H(\mathcal{M}_1) - H(\mathcal{M}_1|\mathcal{C}_1) \leq & H(\mathcal{M}_1) - H(\mathcal{M}_1|\mathcal{C}_1,\mathcal{U}_i) \\ = & H(\mathcal{M}_1|\mathcal{U}_i) - H(\mathcal{M}_1|\mathcal{C}_1,\mathcal{U}_i) \\ = & 0. \end{aligned}$$

となる。よって、これから、$H(\mathcal{M}_1|\mathcal{U}_i) = H(\mathcal{M}_1)$ を示せば、題意が証明されることになる。

ある u_i について、ある m_1 について、$u \notin \{u_1,...,u_{|M|}\}, f(m_1,u) = f(m_1,u_i)$ が成り立つことが分かっている。このことから、$u \in \{u|\exists m \in M,\ f(m,u) = f(m,u_i),\ u \in U\backslash\{u_i\}\} = \cup_{m\in M} U_{i,m}\backslash\{u_i\}$ となる。よって、$m \in M$ について、$|U_{i,m}| = |M|^n$ が成り立ち、

$$\max_{m_1,u_i}\max_{m'} \Pr[m' = m_1|u_i] = \frac{|U_{i,m'}\backslash\{u_i\}|}{|\cup_{m\in M} U_{i,m}\backslash\{u_i\}|} = \frac{|M|^n - 1}{|M|(|M|^n - 1)} = \frac{1}{|M|}.$$

となる。以上から、$H(\mathcal{M}_1|\mathcal{U}_i) = H(\mathcal{M}_1)$ となり、題意が証明された。 □

補題 5.4. 提案方式は、n-NM となる。つまり、

$$\begin{aligned} &\forall i \in \{1,\ldots,n\}, \\ &\qquad H(\hat{\mathcal{M}}|\hat{\mathcal{C}},(\mathcal{M}_1,\mathcal{C}_1),\ldots,(\mathcal{M}_i,\mathcal{C}_i)) = H(\hat{\mathcal{M}}|(\mathcal{M}_1,\mathcal{C}_1),\ldots,(\mathcal{M}_i,\mathcal{C}_i)), \end{aligned}$$

ここで、$\hat{c}(\neq c_i)$ は本来とは異なる暗号文とし、$\hat{m}(\neq m_i)$ はそれに対応する平文とする。そして、$\hat{\mathcal{C}}$ と $\hat{\mathcal{M}}$ はそれぞれに対応する空間とする。

Proof. まず、$k \in \{1,\ldots,n\}$ について、

$$H(\hat{\mathcal{M}}|\hat{\mathcal{C}},(\mathcal{M}_1,\mathcal{C}_1),\ldots,(\mathcal{M}_k,\mathcal{C}_k)) = H(\hat{\mathcal{M}}|(\mathcal{M}_1,\mathcal{C}_1),\ldots,(\mathcal{M}_k,\mathcal{C}_k)),$$

を示す。M_k と C_k は与えられる平文と暗号文の集合とする。つまり、$M_k = \{m_1,\ldots,m_k\}, C_k = \{c_1,\ldots,c_k\}$ となる。M_k と C_k からは、攻撃者は、$u \in \cap_{m\in M_k} U_{i,m}\backslash\{u_i\}$ しか知ることができない。ここで、$u \in \cap_{m\in M_k} U_{i,m}\backslash\{u_i\}$ を知られている状況下でも、すべての $\tilde{m} \in M\backslash\{M_k\}$ が同様に確からしいことを示す。

ここで、$\tilde{m} \in M$ は $|(\cap_{m\in M_k} U_{i,m}) \cap U_{j,\tilde{m}}| \neq 0$ のときのみ平文となり得る。

主張 5.1. すべての $\tilde{m} \in M\backslash M_k$ について、$|(\cap_{m\in M_k} U_{i,m}) \cap U_{j,\tilde{m}}| = |M|^{n-k}$ となる。

Proof. まず、$k=1$ の場合を考える。つまり、ある m^* について、$|U_{i,m^*}\cap U_{j,\tilde{m}}|=|M|^{n-1}$ であることを示す。

$|U_{i,m^*}\cap U_{j,\tilde{m}}|<|M|^{n-1}$ と仮定する。すると、攻撃者は $(\tilde{m},\tilde{\alpha})$ を用いて spoofing の攻撃を行うことが可能となる。ここで、$\tilde{\alpha}$ は、$A\backslash\{f(\tilde{m},u_j)\}$ からランダムに選ぶものとする。また、$|\cap_{1\le j'\le|M|}U_{j',\tilde{m}}|=|U|=|M|^{n+1}$、$|\cap_{1\le j'\le|M|}(U_{j',\tilde{m}}\cap U_{i,m^*})|=|U_{i,m^*}|=|M|^n$ から、

$$\begin{aligned}\Pr[R\text{ accepts }(\tilde{m},\tilde{\alpha})|(m^*,\alpha^*)]&\ge\frac{|\cap_{1\le j'\le|M|,j'\ne j}(U_{j',\tilde{m}}\cap U_{i,m^*})|}{|\cap_{1\le j'\le|M|,j'\ne j}U_{j',\tilde{m}}|}\\&=\frac{|M|^n-|U_{i,m^*}\cap U_{j,\tilde{m}}|}{|M|^{n+1}-|U_{j,\tilde{m}}|}\\&>\frac{|M|^n-|M|^{n-1}}{|M|^{n+1}-|M|^n}=\frac{1}{|M|}.\end{aligned}$$

を得る。これは、もとの A-code が $(1/|M|,n)$-Spf であることと矛盾する。よって、$|U_{i,m^*}\cap U_{j,\tilde{m}}|\ge|M|^{n-1}$ となる。

次に、$|U_{i,m^*}\cap U_{j,\tilde{m}}|>|M|^{n-1}$ と仮定する。このとき、攻撃者は $(\tilde{m},\tilde{\alpha})$ を用いて spoofing の攻撃を行うことが可能となる。ここで、$\tilde{\alpha}=f(\tilde{m},u_j)$ とする。また、U_{i,m^*} の中に、$\tilde{m}$ の正しい認証子を $f(\tilde{m},u_j)$ とする鍵が少なくとも $|M|^{n-1}+1$ 個あることから、

$$\Pr[R\text{ accepts }(\tilde{m},\tilde{\alpha})|(m^*,\alpha^*)]\ge\frac{|U_{i,m^*}\cap U_{j,\tilde{m}}|}{|U_{i,m^*}|}\ge\frac{|M|^{n-1}+1}{|M|^n}>\frac{1}{|M|},$$

を得る。これは矛盾となる。よって、$|U_{i,m^*}\cap U_{j,\tilde{m}}|\le|M|^{n-1}$ となる。以上から、$|U_{i,m^*}\cap U_{j,\tilde{m}}|=|M|^{n-1}$ となる。

次に、$k=\ell$ $(2\le\ell\le n)$ の場合を考える。このとき、$k=\ell-1$ の場合については成り立つと仮定する。この状況のもとで、以下を示す。

$$\begin{aligned}&|(\cap_{m\in M_{\ell-1}}U_{i,m})\cap U_{j,\tilde{m}}|=|M|^{n-\ell+1}\\&\Rightarrow|(\cap_{m\in M_\ell}U_{i,m})\cap U_{j,\tilde{m}}|=|M|^{n-\ell}.\end{aligned}$$

まず、$|(\cap_{m\in M_\ell}U_{i,m})\cap U_{j,\tilde{m}}|>|M|^{n-\ell}$ と仮定する。このとき、攻撃者は $(\tilde{m},\tilde{\alpha})$ を用いて spoofing の攻撃を行うことが可能となる。ここで、$\tilde{\alpha}=f(\tilde{m},u_j)$ とする。また、$\cap_{m\in M_\ell}U_{i,m}$ の中に、$\tilde{m}$ の正しい認証子を $f(\tilde{m},u_j)$ とする鍵が少なくとも $|M|^{n-\ell}+1$ 個あることから、

$$\begin{aligned}|\cap_{m\in M_\ell}U_{i,m}|&=|(\cap_{m\in M_{\ell-1}}U_{i,m})\cap U_{j,m'}|\\&=|M|^{n-\ell+1}.\end{aligned}$$

を得る。その結果、

$$\begin{aligned}\Pr[R\text{ accepts }(\tilde{m},\tilde{\alpha})|(m_1,\alpha_1),\ldots,(m_\ell,\alpha_\ell)]&\ge\frac{|(\cap_{m\in M_\ell}U_{i,m})\cap U_{j,\tilde{m}}|}{|\cap_{m\in M_\ell}U_{i,m}|}\\&\ge\frac{|M|^{n-\ell}+1}{|M|^{n-\ell+1}}>\frac{1}{|M|},\end{aligned}$$

を得るが、これは矛盾である。よって、$|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j,\tilde{m}}| \geq |M|^{n-\ell}$ となる。

次に、$|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j,\tilde{m}}| < |M|^{n-\ell}$ と仮定する。前提から、

$$|\cap_{m\in M_\ell} U_{i,m}| = |M|^{n-\ell+1},$$

となるため、以下を満たす $(U_{j^*,\tilde{m}})(j^* \neq j)$ が存在しなければならない。

$$|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j^*,\tilde{m}}| > |M|^{n-\ell}.$$

このとき、攻撃者は $(\tilde{m}, \tilde{\alpha})$ を用いて spoofing の攻撃を行うことが可能となる。ここで、$\tilde{\alpha}$ は、ある $u_{j^*} \in U_{j^*,\tilde{m}}$ を用いて、$\tilde{\alpha} = f(\tilde{m}, u_{j^*})$ と求められるものとする。以上から、

$$\begin{aligned}\Pr[R \text{ accepts } (\tilde{m},\tilde{\alpha})|(m_1,\alpha_1),\ldots,(m_\ell,\alpha_\ell)] &\geq \frac{|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j^*,\tilde{m}}|}{|\cap_{m\in M_\ell} U_{i,m}|} \\ &\geq \frac{|M|^{n-\ell}+1}{|M|^{n-\ell+1}} > \frac{1}{|M|},\end{aligned}$$

を得るが、これは矛盾である。よって、$|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j,\tilde{m}}| \leq |M|^{n-\ell}$ となる。

以上から、$|(\cap_{m\in M_\ell} U_{i,m}) \cap U_{j,\tilde{m}}| = |M|^{n-\ell}$ となる。

これらのことから、主張 5.1 が成り立つ。 □

主張 5.2. すべての $\tilde{m}_i \in M$ について、$|\cap_{\tilde{m}_0,\ldots,\tilde{m}_{n-k}} U_{j,\tilde{m}_i}| = |M|^k$ が成り立つ。

Proof. $\tilde{U}_k :=\cap_{\tilde{m}_0,\ldots,\tilde{m}_{n-k}} U_{j,\tilde{m}_i}$ とする。$|\tilde{U}_n| = |M|^n$ から、$|\tilde{U}_n| = |U_{j,\tilde{m}_0}| = |M|^n$ を得る。ここで、$|\tilde{U}_k| = |M|^k$ のときを考え、$|\tilde{U}_{k-1}| = |M|^{k-1}$ を証明する。つまり、以下のことを証明する。

$$|\tilde{U}_k| = |M|^k \Rightarrow |\tilde{U}_{k-1}| = |M|^{k-1}.$$

まず、$|\tilde{U}_{k-1}| > |M|^{k-1}$ と仮定する。このとき、攻撃者は $(\tilde{m}_{n-k+1}, \tilde{\alpha}_{n-k+1})$ を用いて spoofing の攻撃を行うことが可能となる。ここで、$\tilde{\alpha}_{n-k+1} = f(\tilde{m}_{n-k+1}, \tilde{u})$ と $\tilde{u}$ は $\tilde{U}_k$ からランダムに選ぶものとする。また、$\tilde{U}_k$ の中に、$\tilde{m}_{n-k+1}$ の正しい認証子を $f(\tilde{m}_{n-k+1}, u_j)$ とする鍵が少なくとも $|M|^{k-1}+1$ 個あることから、

$$\begin{aligned}\Pr[R \text{ accepts } (\tilde{m}_{n-k+1},\tilde{\alpha}_{n-k+1})|(\tilde{m}_0,\tilde{\alpha}_0),\ldots,(\tilde{m}_{n-k},\tilde{\alpha}_{n-k})] &\geq \frac{|\tilde{U}_k \cap U_{j,\tilde{m}_{n-k+1}}|}{|\tilde{U}_k|} \\ &\geq \frac{|M|^{k-1}+1}{|M|^k} > \frac{1}{|M|},\end{aligned}$$

を得るが、これは矛盾である。よって、$|\tilde{U}_{k-1}| \leq |M|^{k-1}$

次に、$|\tilde{U}_{k-1}| < |M|^{k-1}$ と仮定する。$|\tilde{U}_k| = |M|^k$ から、$|\tilde{U}_k \cap U_{j,\tilde{m}^*}| > |M|^{k-1}$ を満たすような $\tilde{m}^*$ が存在する。このことから、攻撃者は $(\tilde{m}^*, \tilde{\alpha}^*)$ を用いて spoofing の攻

撃を行うことが可能となる。ここで、$\tilde{\alpha}^* = f(\tilde{m}^*, \tilde{u})$ と $\tilde{u}$ は $\tilde{U}_k$ からランダムに選ぶものとする。このことから、

$$\begin{aligned}\Pr[R \text{ accepts } (\tilde{m}^*, \tilde{\alpha}^*) | (\tilde{m}_0, \tilde{\alpha}_0), \ldots, (\tilde{m}_{n-k}, \tilde{\alpha}_{n-k})] &\geq \frac{|\tilde{U}_k \cap U_{j,\tilde{m}^*}|}{|\tilde{U}_k|} \\ &> \frac{|M|^{k-1}}{|M|^k} = \frac{1}{|M|},\end{aligned}$$

を得るが、これは矛盾である。よって、$|\tilde{U}_{k-1}| \geq |M|^{k-1}$ となる。

以上から、

$$|\tilde{U}_k| = |M|^k \Rightarrow |\tilde{U}_{k-1}| = |M|^{k-1}.$$

となり、主張 5.2 は証明された。 □

主張 5.1、5.2 から、平文と認証子の組が k 個与えられた場合の、ランダムな推測により平文を求められる確率の最大値を考える。ここで、その確率を鍵の個数から評価する。つまり、確率の分子は正しい認証子が得られる鍵の数とし、分母は可能な鍵の候補の数として求める。すると、求める確率は

$$\begin{aligned}&\max_{\hat{m}, \forall m_i \in M_k, (m_i, u_i)} \max_{\tilde{m}} \Pr[\tilde{m} = \hat{m} | (m_1, u_1), \ldots, (m_k, u_k)] \\ =&\frac{|U_{j,\hat{m}} \cap (\bigcap_{m \in M_k} U_{i,m}) \backslash (\bigcup_{m \in (M_k \cup \{\hat{m}\})} \{u_i\})|}{|\bigcup_{m' \in M \backslash M_k} (U_{j,m'} \cap (\bigcap_{m \in M_k} U_{i,m}) \backslash (\bigcup_{m \in (M_k \cup \{m'\})} \{u_i\}))|} \\ =&\frac{|M|^{n-k+1} - (k+1)}{(|M| - k)(|M|^{n-k+1} - (k+1))} \\ =&\frac{1}{|M| - k},\end{aligned}$$

となる。そして、これはすべての $k \in \{1, \ldots, n\}$ について成り立つため、補題は証明された。 □

5.4　考察

5.4.1　多項式オーダの空間サイズにおける秘匿性

本章では、平文空間のサイズがセキュリティパラメータの多項式オーダの際のメッセージの秘匿性について議論を行ってきた。その結果、Chaffing-and-Winnowing の研究の分野では、暗号文を与えられた際に、それに対応する平文を当てられる確率が、平文空間分の $1(\frac{1}{p})$ ならば、秘匿性が達成されていると考えることにしている。これは、全く情報が与えられていない状態での推測と同じ確率である。つまり、

$$H(M|C) = H(M)$$

となっている。これは、One Time Pad などのパーフェクトセキュリティと同じ定義になっているため、妥当な定義と考えられる。

以上の議論をもとに、本研究の目的となっている、識別子空間が多項式オーダの際の、識別子の秘匿性について考察を行う。具体的には、センターに鍵導出の要求を出した際に、識別子の情報がセンターにどれだけ漏れるかということについて、考える。この際、識別子空間を $\mathcal{S}$、鍵導出を行いたい識別子を $s \in \mathcal{S}$、鍵導出の際にセンターに送る情報を $t \in \mathcal{T}$ とする。このとき、パーフェクトセキュリティと同様の安全性、つまり、t から s の情報が全く漏れないという条件は、以下のようになる。

$$H(s|t) = H(s)$$

この条件を満たすとき、センターは、クライアントからの情報 t から、クライアントが鍵導出しようとしている識別子 s の情報は全く漏れていないことになる。これを、本章と同様に確率で議論をする場合には、すべての多項式時間アルゴリズムを $\mathcal{A}$ として、

$$max_{\forall s \in \mathcal{S}, t \in \mathcal{T}, \mathcal{A}} \ \Pr[\mathcal{A}(t) = s | t(\text{corresponding to } s)] \leq \frac{1}{|\mathcal{S}|}$$

のように定義できる。

上記の条件を満たすのが理想的だが、本研究のように、計算量的安全性に基づく場合には、ほとんどの場合、このような完全な安全性は満たすことが困難である。そのため、少し安全性を弱めることを考える。そして、扱いやすい識別不可能性の形で表現すると以下のようになる。

$$\begin{aligned} &max_{\forall s_1, s_2 \in \mathcal{S}, t_1, t_2 \in \mathcal{T}} \\ &\Big| \Pr[\mathcal{A}(t_1) = s_1 | t_1(\text{corresponding to } s_1)] - \Pr[\mathcal{A}(t_2) = s_1 | t_2(\text{corresponding to } s_2)] \Big| \\ &\leq \epsilon \end{aligned}$$

ただし、ここで ϵ は $|\mathcal{S}|$ のネグリジブル関数に従うものとする。この形にすると、公開鍵暗号の分野でよく用いられる識別不可能性のゲームの定義が容易にできる。そのため、今後、多項式オーダの空間サイズの識別子の匿名性について議論を行う場合に、この式から匿名性のゲーム等の定義を行って行くことが可能と考えられる。

Chapter 6　識別子の特徴に関する考察

6.1　考察

これまで、識別子の特徴に考慮して、従来の識別子利用暗号では考えられていなかった識別子の利用法について述べてきた。本章では、これまでの結果を踏まえ、識別子に関する考察を行う。

もともと、識別子利用暗号は、ID ベース暗号として、[32] で提案されたものである。[32] では、「各エンティティを特定する ID 情報を公開鍵として用いることで、従来の公開鍵暗号で問題であった鍵のすり替えを防ぐことができる」という発想であった。ここでは、まず従来の公開鍵暗号における公開鍵の特徴についてまとめ、ID ベース暗号、およびその拡張方式である属性ベース暗号における識別子との違いについて議論を行う。

まず、従来の公開鍵暗号も、『暗号技術』の 1 つであることを考えると、機密性を保つために、「ある公開鍵で暗号化された暗号文は、それに対応する秘密鍵でのみ復号可能である」という性質が必要である。つまり、送信者は、公開鍵に対応する復号鍵の持ち主であるエンティティにのみメッセージを送りたいので、機密性を保つためには、それ以外のエンティティに読まれる（他の復号鍵で復号される）ことは許されないのである。また、公開鍵暗号は、「共通鍵暗号における鍵配送の問題を解決する」という目的のもとに提案された方式なので、「公開鍵が公開可能」という性質も必要である。ここで、「公開鍵が公開可能」であるということは、つまり「公開鍵の情報から復号鍵に関する情報が漏れない」ということである。そして、問題点としては、公開鍵はランダムなビット列であるため、公開鍵とエンティティとをひも付けする必要があるという点が挙げられる。そのため、現在では、認証局が、正当性を保証するために、すべての公開鍵に対する証明書を発行・管理している。ここで、認証局が信頼できない場合には、認証局が勝手に作った公開鍵に対する証明書の発行をし、それを広告することで、鍵のすり替えと同じ問題が発生してしまう。そのため、我々は認証局を信頼できるものとみなさなければ、公開鍵の正当性を納得することはできない。

以上をまとめると、以下のようになる。

- 特徴
  - ある公開鍵で暗号化された暗号文は、それに対応する秘密鍵でのみ復号可能
    - 機密性を保証するため
  - 公開鍵の情報から復号鍵に関する情報が漏れない
    - 鍵を公開可能にし、鍵配送を容易にするため
- 問題点

- 公開鍵とエンティティのひも付けが必要
  - 認証局による証明書の発行により回避

ここから、識別子利用暗号での公開鍵にあたる識別子は、前述の特徴をどのように反映しているか検討を行う。

まず、1つ目の特徴、「ある公開鍵で暗号化された暗号文は、それに対応する秘密鍵でのみ復号可能」について考える。つまり、識別子利用暗号と公開鍵暗号でのそれぞれの機密性について比較を行う。識別子利用暗号では、送信者は識別子をもとにメッセージを送りたい相手の指定を行う。そして、暗号文を復号できるのは、送信者が指定した識別子の示すエンティティのみであることを保証するのが識別子利用暗号である。そのため、識別子利用暗号は、公開鍵暗号と同じ機密性を有していると考えられる。しかし、属性ベース暗号の登場により、エンティティの指定の方法が、公開鍵暗号の場合と異なってきている。公開鍵暗号の場合には、ある公開鍵を用いて暗号化を行うと、その公開鍵に対応する復号鍵を持っているエンティティを指定したことになる。つまり、1回の暗号化で指定できるエンティティは、1つの鍵で指定できるエンティティに限られる。それに対し、属性ベース暗号では、複数の属性情報を用いてエンティティの指定が可能となる。つまり、1回の暗号化で、複数の鍵にまたがってエンティティの指定を行うことが可能となる。さらに、第3章の結果は、すでに述べたように、そのエンティティの指定を、より柔軟に行うことを可能にした方式である。結果として、属性ベース暗号、および第3章の方式により、機密性を定義するために必要な、『アクセスを許可されたエンティティ』の指定を、従来よりも効率的に行うことを可能としている。

次に、2つ目の特徴、「公開鍵の情報から復号鍵に関する情報が漏れない」という点に関して考える。これは、鍵配送を容易にするために必要な特徴である。識別子利用暗号では、公開鍵である識別子に対応する復号鍵は、鍵生成センターがすべて発行する。鍵生成センターは、鍵導出の際に、自身の秘密情報を用いて復号鍵を導出する。そして、その秘密情報がない限り、識別子情報からそれに対応する復号鍵を計算するのは困難となっている。つまり、識別子から自由に復号鍵を得ることができるのは、必要な秘密情報を持っている鍵生成センターだけである。ここで、鍵導出について考える。鍵導出は、センターに識別子を送り、それに対応する復号鍵を得るプロトコルである。これは、「公開鍵から復号鍵を得る」という操作を意味している。つまり、センター以外のエンティティは、自身では識別子から復号鍵を得るのは困難だが、『鍵導出』を通して、その作業が可能となる。そのため、鍵導出を行う際の、制限が必要となる。具体的には、その識別子に対する復号鍵を得る資格があるエンティティにのみ、鍵導出を行えるようにしなければならない。これは、従来の識別子利用暗号の方式の中では議論されていない。そのため、第4章での、「鍵導出の際にセンターが必要な情報と、ユーザが送るべき情報」に対する考察は、識別子利用暗号が「公開鍵から復号鍵の情報が漏れない」という性質を満たすための第一歩であると考えられる。この問題への対応は、現在は、鍵導出の際にクライアントの認証を行うなど、運用の際に対処するのが良いと思われるが、今後、第4章での議論を深め、識別子利用暗号の方式の枠組みの中で捉えられるようにしていく必要があると考えられる。

そして、公開鍵暗号の際に問題であった、「公開鍵とエンティティのひも付けが必要」ということは、識別子利用暗号でどのように改善されたか考える。[32] では、この問題を解決するために ID ベース暗号を提案したため、本来ならば、この問題は解消されているはずである。しかし、実際には、第 1 章でも述べたように、現在の識別子利用暗号は、「公開鍵として任意長のビット列を扱える」という方式を指しているに過ぎない。そのため、本質的に、この問題を解決しているとは考えられない。

以上、公開鍵暗号と識別子利用暗号における公開鍵の特徴について比較を行った。その結果、もともと識別子利用暗号の目的として掲げられていた、「公開鍵とエンティティのひも付けを不要とする」という目標は依然として達成されているとはいえない状況である。しかし、本研究は、識別子利用暗号の機密性に関する機能の向上や、鍵配送に要求される性質への議論などを含んでいることが分かった。そして、公開鍵暗号では達成されていた、鍵配送の容易化は、識別子利用暗号では議論の余地があることが分かり、今後は、この点も含め、識別子利用暗号に関する議論を深めていかなければならないと考えられる。

Chapter 7　結論

7.1　まとめ

本研究では、識別子利用暗号の識別子の特徴に考慮した方式の提案を中心に、識別子利用暗号における識別子に関する議論を行ってきた。まず、第 1 では、識別子利用暗号が提案された背景についてまとめ、本研究の目的について述べた。その上で、第 3 章では、従来より柔軟にエンティティの指定が行える識別子利用暗号の方式の提案を行った。そして、第 4 章、第 5 章では、センターに対する匿名性について議論を行い、鍵空間が多項式オーダの場合の、匿名性に関する考察及び定式化を行った。さらに、第 6 章では、それまでの結果をもとに、識別子利用暗号における公開鍵としての識別子について考察を行い、[32] で提唱された機能が達成されていないだけでなく、公開鍵暗号では達成されていた鍵配送の容易化も、慎重に議論を行わなければならないという結論を導いた。

7.2　今後の課題

まず、[32] で提案された機能を、完全に達成する方式を考える必要がある。その際、第 6 章でも議論したように、鍵導出に関する深い議論も必要になると考えられる。

また、比較的短期の課題としては、第 4 章で達成できなかった、鍵規定型の属性ベース暗号に関する匿名性の付与を行いたい。その際、第 5 で議論した、鍵空間が多項式オーダの場合の匿名性に関する定式化を用いることが予想される。

発表文献

国際会議

[i] Wataru Kitada, Goichiro Hanaoka, Kanta Matsuura and Hideki Imai, "Unconditionally Secure Chaffing-and-Winnowing for Multiple Use", *International Conference on Information Theoretic Security(ICITS'07)*, 2007.

国内会議

[i] 北田 亘, 松浦 幹太, "ブラインド属性ベース暗号", 2008 年 暗号と情報セキュリティシンポジウム (SCIS'08), 2008.

[ii] 北田 亘, 松浦 幹太, "柔軟な識別子評価可能な暗号化方式", 第 30 回情報理論とその応用シンポジウム (SITA'07), 2007.

[iii] 北田 亘, Nuttapong Attrapadung, 花岡 悟一郎, 松浦 幹太, 今井 秀樹, "IBE-PKE 変換の広がりの限界への更なる考察", 2007 年 暗号と情報セキュリティシンポジウム (SCIS'07), 2007.

[iv] 北田 亘, 花岡 悟一郎, Nuttapong Attrapadung, 張 鋭, 松浦 幹太, 今井 秀樹, "BDDH 仮定と Square BDDH 仮定の関係の考察", 第 29 回情報理論とその応用シンポジウム (SITA'06), 2006.

参考文献

[1] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography", *Advances in Cryptology – Asiacrypt'03*, LNCS 2894, pp. 452-473, 2003.

[2] S. S. Al-Riyami and K. G. Paterson, "CBE from CL-PKE: A generic construction and efficient schemes", *Public Key Cryptography (PKC'05)*, LNCS 3386, pp. 398-415, 2005.

[3] J. Baek, R. Safavi-Naini and W. Susilo, "Certificateless Public Key Encryption without Pairing", *Proc. of the 8th Information Security Conference (ISC 2005)*, LNCS 3650, pp. 134-148, 2005.

[4] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext-Policy Attribute-Based Encryption", *Proc. of the 2007 IEEE Symposium on Security and Privacy*, pp. 321-334, 2007.

[5] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing", *Advances in Cryptology – Crypto'01*, LNCS 2139, pp. 213-229, 2001.

[6] D. Boneh and X. Boyen, "Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles", *Advances in Cryptology – Eurocrypt'04*, LNCS 3027, pp. 223-238, 2004.

[7] F. Boudot, "Efficient Proofs that a Committed Number Lies in an Interval," *Advances in Cryptology – Eurocrypt'00*, LNCS 1807, pp. 431-444, 2000.

[8] J. Camenisch and M. Michels, "Proving in Zero-Knowledge that a Number n Is the Product of Two Safe Primes", *Advances in Cryptology – Eurocrypt'99*, LNCS 1592, pp. 107-122, 1999.

[9] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups," *Advances in Cryptology – Crypto'97*, LNCS 1296, pp. 410-424, 1997.

[10] A. Chan, Y. Frankel and Y. Tsiounis, "Easy Come - Easy Go Divisible Cash", *Advances in Cryptology – Eurocrypt'98*, LNCS 1403, pp. 561-575, 1998.

[11] M. Chase, "Multi-Authority Attribute Based Encryption", *Proc. of TCC'07*, LNCS 4392, pp. 515-534, 2007.

[12] L. Cheung and C. Newport, "Provably Secure Ciphertext Policy ABE", *Proc. of CCS'07*, pp. 456-465, 2007.

[13] R. Cramer, I. Damgard and B. Schoenmakers, "Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols", *Advances in Cryptology – Crypto'94*, LNCS 839, pp. 174-187, 1994.

[14] J. Daemen and V. Rijmen, "AES Proposal: Rijndael", *AES Algorithm Submission*, 1999.

[15] W. Diffie and M. E. Hellman, "New Directions in Cryptography", *IEEE Transactions on Information Theory*, vol. IT-22, no. 6, pp. 644-654, 1976.

[16] C. Gentry, "Practical Identity-Based Encryption Without Random Oracles", *Advances in Cryptology – Eurocrypt'06*, LNCS 4004, pp. 445-464, 2006.

[17] S. Goldwasser and S. Micali, "Probabilistic Encryption and How to Play Mental Poker Hiding All Partial Information", *Proc. of the 14th Annual ACM Symposium on the Theory of Computing*, pp. 365-377, 1982.

[18] S. Goldwasser and S. Micali, "Probabilistic Encryption", *Special issue of Journal of Computer and Systems Sciences*, vol. 28, no. 2, pp. 270-299, 1984.

[19] V. Goyal, "Reducing Trust in the PKG in Identity Based Cryptosystems", *Advances in Cryptology – CRYPTO'07*, LNCS 4622, pp. 430-447, 2007.

[20] V. Goyal, O. Pandey, A. Sahai and B. Waters, "Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data", *Proc. of CCS'06*, pp. 89-98, 2006.

[21] M. Green and S. Hohenberger, "Blind Identity-Based Encryption and Simulatable Oblivious Transfer", *Advances in Cryptology – Asiacrypt'07*, LNCS 4833, pp. 265-282, 2007.

[22] G. Hanaoka, Y. Hanaoka, M. Hagiwara, H. Watanabe and H. Imai, "Unconditionally secure chaffing-and-winnowing: a relationship between encryption and authentication", *Proc. of AAECC'06*, LNCS 3857, pp. 154-162, 2006.

[23] B. Libert and J.J. Quisquater, "On constructing certificateless cryptosystems from identity based encryption", *Public Key Cryptography (PKC'06)*, LNCS 3958, pp. 474-490, 2006.

[24] P. Paillier, "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes", *Advances in Cryptology – Eurocrypt'99*, LNCS 1592, pp. 223-238, 1999.

[25] K. G. Paterson, "Cryptography from Pairings: A Snapshot of Current Research", *Information Security Technical Report*, vol. 7(3), pp. 41-54, 2002.

[26] T. P. Pedersen, “Non-Interactive and Information-Theoretic Secure Verifiable Secret Sharing”, *Advances in Cryptology – Crypto'92*, LNCS 576, pp. 129-140, 1992.

[27] M. Rabin, “How to exchange secrets by oblivious transfer”, *Technical Report TR-81*, 1981.

[28] R. Rivest, “Chaffing and Winnowing: Confidentiality without Encryption”, `http://people.csail.mit.edu/rivest/publications.html`.

[29] R. L. Rivest, A. Shamir and L. M. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”, *Communications of the ACM*, vol. 21, Issue 2, pp. 120-126, 1978.

[30] A. Sahai and B. Waters, “Fuzzy Identity-Based Encryption”, *Advances in Cryptology – Eurocrypt'05*, LNCS 3494, pp. 457-473, 2005.

[31] C. P. Schnorr, “Efficient Signature Generation for Smart Cards”, *Journal of Cryptology*, vol. 4(3), pp. 239-252, 1991.

[32] A. Shamir, “Identity-Based Cryptosystems and Signature Schemes”, *Advances in Cryptology – Crypto'84*, LNCS 196, pp. 47-53, 1984.

[33] B. Waters, “Efficient Identity-Based Encryption Without Random Oracles”, *Advances in Cryptology – Eurocrypt'05*, LNCS 3494, pp. 114-127, 2005.

[34] JIS X 5080:2002, 情報技術–情報セキュリティマネジメントの実践のための規範.

[35] RFC4210, Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP).

謝辞

本研究は、東京大学情報理工学系研究科 松浦 幹太准教授のご指導のもとに遂行いたしました。松浦 幹太准教授には、研究の遂行にあたり多大なる助言をいただきました。記して深く感謝申し上げます。また、本研究室の秘書 仲野 小絵さん、すでにご退職されましたが鶴山 陽子さんには、日常の細かなことまで面倒見ていただいたことに、記して感謝申し上げます。同じく、本研究室の博士過程学生の楊 鵬さん、Jacob Schuldt さんには、打ち合わせなどを通し、適切な助言をいただいたことに感謝申し上げます。そして、同期として 2 年間ともに頑張ってきた松田 隆宏さん、Phan Thi Lan Anh さん、Vadim Jefte Zendejas Samano さんには、研究だけでなく、研究室の日常におけるさまざまなことを通して、多くのことを学びました。記して、深く感謝申し上げます。また、修士課程 1 年の渡邉 悠君には、日頃から鋭い指摘を頂いたことに、感謝申し上げます。その他、研究室の OB や共同研究者の方々など、本研究を遂行するにあたり手助けいただいた方々に、記して感謝申し上げたいと思います。