応用数学 6 2011 年 1 月 14 日

暗号方式

講師: 安永憲司

1 公開鍵暗号方式

定義 1 (公開鍵暗号) 三つ組みの PPT アルゴリズム $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ が**公開鍵暗号方式**であるとは，以下を満たすときである．

1. Gen は鍵生成アルゴリズムであり，入力 1^n に対して，(pk, sk) を出力する．pk は公開鍵，sk は秘密鍵であり，pk, sk に対してメッセージ空間 $\mathcal{M}_n$ が関連づけられている．$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$ と表す．
2. Enc は暗号化アルゴリズムであり, 入力 pk と $m \in \mathcal{M}_n$ に対して, 暗号文 c を出力する．$c \leftarrow \mathsf{Enc}_{pk}(m)$ と表す．
3. Dec は復号アルゴリズムであり，入力 sk と c に対して，メッセージ $m \in \mathcal{M}_n \cup \{\perp\}$ を出力する．$m \leftarrow \mathsf{Dec}_{sk}(c)$ と表す．
4. 鍵生成された任意の $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$ と，任意の $m \in \mathcal{M}_n$ に対して，

$$\Pr[\mathsf{Dec}_{sk}(\mathsf{Enc}_{pk}(m)) = m] = 1$$

を満たしている．

4 つ目の性質は，m から作られた暗号文を復号すると必ず m に復元されることを保証している．メッセージ空間 $\mathcal{M}_n$ の情報は，公開鍵 pk に含まれていると考え，アルゴリズムへの入力として明記しないことが多い．また，復号の際に公開鍵 pk が必要な場合があるが，その場合も，秘密鍵 sk に pk が含まれていると考え，復号アルゴリズムへの入力に pk は明記しない．

例 2 RSA 暗号は，以下のように定義できる．

1. $\mathsf{Gen}(1^n)$: $p, q \xleftarrow{R} \Pi_n, N = pq, e \xleftarrow{R} \mathbb{Z}^*_{\varphi(N)}, d = e^{-1} \bmod \phi(N)$. $\mathcal{M}_n = \mathbb{Z}^*_N$. $(pk, sk) = ((N, e), d)$ を出力.
2. $\mathsf{Enc}_{pk}(m)$: $m^e \bmod N$ を出力.
3. $\mathsf{Dec}_{sk}(c)$: $c^d \bmod N$ を出力.

また，一般的に，落し戸付き置換 (TDP) $\mathcal{F} = \{f_i : \mathcal{D}_i \to \mathcal{R}_i\}_{i \in \mathcal{I}}$ は公開鍵暗号と見なすことができる．

1. $\mathsf{Gen}(1^n)$: $(f_i, f_i^{-1}) \leftarrow \mathsf{Gen}_T(1^n), \mathcal{M}_n = \mathcal{D}_i$. $(pk, sk) = (f_i, f_i^{-1})$ を出力.
2. $\mathsf{Enc}_{pk}(m)$: $f_i(m)$ を出力.
3. $\mathsf{Dec}_{sk}(c)$: $f_i^{-1}(c)$ を出力.

TDP を暗号方式と見なした場合，以下のような問題点がある．

- 部分情報が明らかになっている可能性がある．
- メッセージを一様ランダムに選ぶことを想定しているが，現実的にはそのような状況は考えにくい．
- メッセージ空間が小さい場合，全数探索によって安全性が破られてしまう．
- 同じメッセージを暗号化すると，同じ暗号文になる．

従って，TDP を暗号方式とみなした場合，十分な安全性をもっているとは言えないが，弱い安全性を満たしている．

定義 3 (一方向安全性) 公開鍵暗号方式 Π が一方向安全とは，任意の PPT アルゴリズム A に対して，無視できる関数 $\epsilon(\cdot)$ が存在し，すべての $n \in \mathbb{N}$ に対して，

$$\Pr[A(c) = m \mid (sk, pk) \leftarrow \mathsf{Gen}(1^n), m \xleftarrow{R} \mathcal{M}_n, c \leftarrow \mathsf{Enc}_{pk}(m)] \leq \epsilon(n)$$

を満たすときである．

補題 4 TDP から構成される公開鍵暗号は，一方向安全である．

1.1 識別不能安全性

より望ましい安全性として，以下の安全性を定義する．

定義 5 (識別不能安全性) メッセージ空間 $\mathcal{M}_n$ である公開鍵暗号方式 Π が識別不能安全であるとは， 任意のメッセージ $m_0, m_1 \in \mathcal{M}_n$ に対して，以下の分布が計算量的に識別不能であるときである:

- $\{(pk, \mathsf{Enc}_{pk}(m_0)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$
- $\{(pk, \mathsf{Enc}_{pk}(m_1)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$

この定義では，m_0 の暗号文の分布と m_1 の暗号文の分布が区別できないことを保証している．

予測補題を用いると，上記の定義は次のように言い換えることができる．任意の PPT アルゴリズム A に対し，無視できる関数 $\epsilon(\cdot)$ が存在し， 任意の $n \in \mathbb{N}$, メッセージ $m_0, m_1 \in \mathcal{M}_n$ に対して，

$$\Pr[A(c, pk) = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(m_b)] \leq \frac{1}{2} + \epsilon(n)$$

この定義では，m_0 と m_1 それぞれの暗号文の分布のうち，どちらの分布からのサンプルか，予測できないことを保証している．

識別不能安全性は，敵のアルゴリズムを PPT に制限しない場合や，暗号化アルゴリズムが決定性の場合は達成できないことがわかる．

PPT に制限しない場合 敵の計算能力は十分に大きいので，二つの分布が識別できないのは，それらの分布が完全に一致する場合に限られる．しかし，それは不可能である．なぜならば，秘密鍵 sk をもっている人は，暗号文から一意にメッセージを復元できなければならない．つまり，暗号文分布のとる値（サポート）は，m_0 の場合と m_1 の場合で共通部分をもたない．したがって，分布が完全に一致することはできない．

暗号化が決定性の場合 もし決定性のアルゴリズムで暗号化している場合，敵は m_0 と m_1 を自分で暗号化し，どちらの暗号文が c と一致するかを調べることで簡単に区別できる．

1.1.1 識別不能安全な暗号方式

TDP をもとにした構成法を紹介する．ただし，$\mathcal{M}_n = \{0,1\}$，つまりメッセージが 1 ビットの場合である．TDP $\mathcal{F} = \{f_i : \mathcal{D}_i \to \mathcal{R}_i\}_{i \in \mathcal{I}}$ に対して，f_i のハードコアビットが h_i であるとき，

1. $\mathsf{Gen}(1^n) : (f_i, f_i^{-1}) \leftarrow \mathsf{Gen}_T(1^n)$. $\mathcal{M}_n = \{0,1\}$. $(pk, sk) = (f_i, f_i^{-1})$ を出力.
2. $\mathsf{Enc}_{pk}(m) : r \xleftarrow{R} \mathcal{D}_i$ を選び，$(f_i(r), h_i(r) \oplus m)$ を出力.

3. $\mathsf{Dec}_{sk}(c_1, c_2) : r \leftarrow f_i^{-1}(c_1)$ を計算し，$h_i(r) \oplus c_2$ を出力．

定理 6 上記の公開鍵暗号方式は，識別不能安全である．

証明: 識別不能でないと仮定して，矛盾を導く．予測補題を用いた別定義を満たさないと仮定すると，ある PPT アルゴリズム A が存在し，無限に多くの $n \in \mathbb{N}$ に対して，

$$\Pr[A(c, pk) = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(b)] \geq \frac{1}{2} + \frac{1}{p(n)},$$

ただし，$p(\cdot)$ は多項式である．この A を用いて，ハードコアビットを予測する A' を構成する．

A' は，入力 (pk, y) に対して，$c \xleftarrow{R} \{0,1\}$ を選び，$m \leftarrow A(pk, (y, c))$ を計算し，$c \oplus m$ を出力する．すると，

$$\begin{aligned}
&\Pr[A'(f_i, f_i(r)) = h_i(r) \mid (f_i, f_i^{-1}) \leftarrow \mathsf{Gen}_T(1^n), r \xleftarrow{R} \mathcal{D}_i] \\
&= \Pr[A(f_i, (f_i(r), c)) \oplus c = h_i(r) \mid (f_i, f_i^{-1}) \leftarrow \mathsf{Gen}_T(1^n), r \xleftarrow{R} \mathcal{D}_i, c \xleftarrow{R} \{0,1\}] \\
&= \Pr[A(f_i, (f_i(r), m \oplus h_i(r))) = m \mid (f_i, f_i^{-1}) \leftarrow \mathsf{Gen}_T(1^n), r \xleftarrow{R} \mathcal{D}_i, m \xleftarrow{R} \{0,1\}] \\
&= \Pr[A(pk, \mathsf{Enc}_{pk}(m)) = m \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), m \xleftarrow{R} \{0,1\}] \\
&\geq \frac{1}{2} + \frac{1}{p(n)}.
\end{aligned}$$

□

一般的に，1 ビットメッセージに対して識別不能安全な方式が存在したとき，それをもとにして，$\ell(n)$ ビットメッセージに対して識別不能安全な方式を構成することができる．1 ビットメッセージに対する暗号化方式を $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ とする．このとき，$\Pi' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ を以下のように定義する．

1. $\mathsf{Gen}'(1^n) = \mathsf{Gen}(1^n)$.
2. $\mathsf{Enc}'_{pk}(m_1 m_2 \cdots m_{\ell(n)}) = (\mathsf{Enc}_{pk}(m_1), \mathsf{Enc}_{pk}(m_2), \ldots, \mathsf{Enc}_{pk}(m_{\ell(n)}))$.
3. $\mathsf{Dec}'_{sk}(c_1 c_2 \cdots c_{\ell(n)}) = (\mathsf{Dec}_{sk}(c_1), \mathsf{Dec}_{sk}(c_2), \ldots, \mathsf{Dec}_{sk}(c_{\ell(n)}))$.

定理 7 Π が識別不能安全であるとき，Π' は識別不能安全性である．

1.2 零知識安全性

識別不能安全性は，どの二つのメッセージを暗号化しても，それらの暗号文分布が識別できないことを保証している．しかし，識別不能安全性では，暗号文から平文（もとのメッセージ）についての知識が漏れていないことを，直接的には保証していない．安全な暗号方式とは，暗号文から平文の知識が漏れていないことを保証しているものであって欲しい．

ここでは，暗号文から平文の知識が漏れていないことを保証する，零知識安全性を定義する．直感的に説明すると，敵が暗号文から計算できることは，その平文なしでも計算できるとき，零知識安全である．

定義 8 (零知識安全性) 暗号方式 Π が零知識安全であるとは，ある PPT アルゴリズム S (シミュレータと呼ぶ) が存在して，すべての $m \in \mathcal{M}_n$ について，以下の分布が計算量的に識別不能なときである:

- $\{(pk, \mathsf{Enc}_{pk}(m)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$
- $\{(pk, S(pk)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$

零知識安全性は，Shannon 秘匿性の概念を，計算量的な枠組みで捉えていると考えられる．Shannon 秘匿性は，任意のメッセージ m に対して，分布からメッセージ m' をサンプルして，m' の暗号文を見たという条件の下で，m' が m である確率と，暗号文を見ずに m' が m である確率が等しいことを保証している．つまり，暗号文を見ても，平文についての情報は，暗号文を見る前と変わってないことを保証している．この考え方を計算量的に，つまり，敵の能力が PPT に制限されているとして捉えているのが零知識安全性である．実際の零知識安全性では，暗号文を見て計算できることは，平文を見ずにも計算できることを保証している．

そして，零知識安全性と識別不能安全性は等価であることがわかる．

定理 9 暗号方式 Π は，識別不能安全であるならば，かつそのときに限り零知識安全である．

証明: 両方向をそれぞれ証明する．

識別不能 $\Rightarrow$ 零知識

以下のシミュレータ $S(pk)$ を考える:

1. $m' \in \mathcal{M}_n$ を選ぶ.
2. $c \leftarrow \mathsf{Enc}_{pk}(m')$ を出力.

すると，シミュレータの出力は，識別不能安全性より，任意の $m \in \mathcal{M}_n$ について，分布 $\{(pk, \mathsf{Enc}_{pk}(m)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$ と識別できない.

零知識 $\Rightarrow$ 識別不能

矛盾を導くため，識別不能安全でないと仮定する．つまり，ある PPT アルゴリズム D が存在して，無限に多くの $n \in \mathbb{N}$ について，あるメッセージ $m_1, m_2 \in \mathcal{M}_n$ が存在して，D は

- $H_n^1 = \{(pk, \mathsf{Enc}_{pk}(m_1)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$
- $H_n^2 = \{(pk, \mathsf{Enc}_{pk}(m_2)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$

を確率 $1/p(n)$ で区別する．ただし，$p(\cdot)$ は多項式.

零知識安全のシミュレータ S が存在するので,

- $H_n^3 = \{(pk, S(pk)) \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n)\}$

と定義する．ハイブリッド補題より，D は H_n^1 と H_n^3 もしくは，H_n^2 と H_n^3 を確率 $\frac{1}{2p(n)}$ で区別することができる．これは，零知識安全性に矛盾する． □

1.3 選択平文安全性

前節で定義した識別不能安全性では，メッセージ空間が $\mathcal{M}_n$ である暗号方式を考えていた．しかし，実際の暗号方式では，メッセージ空間が公開鍵 pk を選んだ後に決まる場合もある．

また，識別不能安全性では，メッセージのペア (m_0, m_1) を選んだ後，公開鍵 pk が生成され，敵はそれをもとにして与えられた暗号文が m_0 か m_1 かを推測していた．敵は，公開鍵 pk を入力として受け取っているが，その情報を利用してメッセージを推測しているとは言い切れない.

上記の問題点を解決するため，以下のような定義を考える．まず始めに公開鍵 pk と秘密鍵 sk，メッセージ空間 $\mathcal{M}_n$ を生成する．敵は $pk, \mathcal{M}_n$ を受け取った後，二つのメッセージ $m_0, m_1 \in \mathcal{M}_n$ を出力する．そして，ランダムに選んだ $b \in \{0, 1\}$ に対して m_b を暗号化し，その暗号文を見て，敵はどちらを暗号化したか，つまり b を推測する．先ほどと異なり，メッセージ m_0, m_1 は任意に選べる訳でなく、敵である PPT アルゴリズムによって選ぶことができるものに限られる．もし任意に選ぶことができてしまうと，$m_0 = sk$ となる

メッセージを選ぶことができる．すると，m_0 を暗号化した場合，敵は秘密鍵を手に入れることができ，必ず推測することができてしまう．

上記の攻撃は，選択平文攻撃と呼ばれており，この安全性は，選択平文攻撃に対する識別不能安全性 (indistinghishability against chosen plaintext attacks; IND-CPA) と呼ばれている．

定義 10 (選択平文攻撃に対する識別不能性安全性 (IND-CPA)) 暗号方式 Π, PPT アルゴリズム A, $n \in \mathbb{N}$, $b \in \{0,1\}$ に対して以下の実験を考える．

$\mathsf{IND}_b(\Pi, A, n)$
$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$
$(m_0, m_1, st) \leftarrow A(pk)$
$c \leftarrow \mathsf{Enc}_{pk}(m_b)$
(st, c) を出力

暗号方式 Π が選択平文攻撃に対して識別不能安全である，または，IND-CPA 安全であるとは，

$$\{\mathsf{IND}_0(\Pi, A, n)\} \approx_c \{\mathsf{IND}_1(\Pi, A, n)\}$$

を満たすときである．

アルゴリズム A が出力する st は計算の状態 (state) を表しており，A が出力する際に記憶していた情報をすべて含むと考える．上記の場合，st には pk, m_0, m_1 などが含まれていると考える．これは，分布を識別する識別者が，メッセージを選ぶアルゴリズムである A の計算過程すべてを知ることができることを意味する．

予測補題を用いると，上記の定義は以下と等価である．暗号方式 Π が IND-CPA 安全であるとは，任意の PPT アルゴリズム A_1, A_2 に対して，無視できる関数 $\epsilon(\cdot)$ が存在し，

$$\Pr\left[b = b' \;\middle|\; \begin{array}{l}(pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_0, m_1, st) \leftarrow A_1(pk), \\ b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(m_b), b' \leftarrow A_2(st, c)\end{array}\right] \leq \frac{1}{2} + \epsilon(n)$$

を満たすときである．

IND-CPA 安全性は，前節で定義した識別不能安全性よりも真に強い安全性であることを示すことができる．

定理 11 暗号方式が IND-CPA 安全であるとき，その方式は識別不能安全でもある．

定理 12 暗号方式として，識別不能安全であるが IND-CPA 安全でないものが存在する．

これまでに紹介した識別不能安全な方式は，IND-CPA 安全でもある．その意味で，識別不能安全と IND-CPA 安全は，非常に近い安全性であると言える．

IND-CPA 安全性に対応する，零知識安全性も存在する．ここではその安全性を，選択平文攻撃に対する零知識性 (Zero Knowledge against Chosen Plaintext Attacks; ZK-CPA) と呼ぶことにする．

定義 13 (選択平文攻撃に対する零知識安全性 (ZK-CPA)) 暗号方式 Π，PPT アルゴリズム A, S に対して以下の実験を考える．

$\mathsf{Real}(\Pi, A, n)$
$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$
$(m, st) \leftarrow A(pk)$
$c \leftarrow \mathsf{Enc}_{pk}(m)$
(m, st, c) を出力

$\mathsf{Sim}(\Pi, A, S, n)$
$(pk, sk) \leftarrow \mathsf{Gen}(1^n)$
$(m, st) \leftarrow A(pk)$
$c' \leftarrow S(pk, st)$
(m, st, c') を出力

暗号方式 Π が ZK-CPA であるとは，任意の PPT アルゴリズム A に対して， PPT アルゴリズム S が存在し，

$$\{\mathsf{Real}(\Pi, A, n)\} \approx_c \{\mathsf{Sim}(\Pi, A, S, n)\}$$

を満たすときである．

定理 14 暗号方式 Π は，IND-CPA 安全であるならば，かつそのときに限り ZK-CPA 安全である．

証明: 両方向をそれぞれ証明する．

IND-CPA $\Rightarrow$ ZK-CPA

ZK-CPA 安全でないと仮定する．すなわち，ある PPT アルゴリズム A が存在し，任意の PPT シミュレータ S に対して，

$$\{\mathsf{Real}(\Pi, A, n)\} \not\approx_c \{\mathsf{Sim}(\Pi, A, S, n)\}$$

である．つまり，予測補題を用いると，ある PPT アルゴリズム A' と多項式 $p(\cdot)$ が存在して，無限に多くの $n \in \mathbb{N}$ について，

$$\Pr[A'(m, st, c) = b \mid b \xleftarrow{R} \{0,1\}, (m, st, c) \leftarrow X_n^b] \geq \frac{1}{2} + \frac{1}{p(n)}$$

を満たしている．ただし，$X_n^0 = \mathsf{Real}(\Pi, A, n)$, $X_n^1 = \mathsf{Sim}(\Pi, A, S, n)$ である．

ここで，以下の PPT シミュレータ $S(pk, st)$ を考える．

1. $(m', st') \leftarrow A(pk)$ を計算.
2. $c' \leftarrow \mathsf{Enc}_{pk}(m')$ を計算.
3. c' を出力.

そして，IND-CPA においてメッセージを選ぶ PPT アルゴリズム $A_1^{\mathsf{IND}}(pk)$ として，以下を考える．

1. $(m, st) \leftarrow A(pk)$ を計算.
2. $(m', st') \leftarrow A(pk)$ を計算.
3. $(m, m', (m, st))$ を出力.

アルゴリズム A_1^{IND} は，A を二回実行し，それぞれで出力したメッセージ m, m' を選んでいる．

このとき，

$$\begin{aligned}
&\Pr[A'(m, st, c) = b \mid b \xleftarrow{R} \{0,1\}, (m, st, c) \leftarrow X_n^b] \\
&= \Pr\left[A'(m, st, c) = b \;\middle|\; \begin{array}{l} b \xleftarrow{R} \{0,1\}, (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m, st) \leftarrow A(pk), \\ (m', st') \leftarrow A(pk), (m_0, m_1) = (m, m'), c \leftarrow \mathsf{Enc}_{pk}(m_b) \end{array}\right] \\
&= \Pr\left[b = b' \;\middle|\; \begin{array}{l} (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_0, m_1, (m_0, st)) \leftarrow A_1^{\mathsf{IND}}(pk), \\ b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(m_b), b' \leftarrow A'(m_0, st, c) \end{array}\right]
\end{aligned}$$

である．IND-CPA においてメッセージを当てるアルゴリズムを，$A_2^{\mathsf{IND}}((m, st), c) = A'(m, st, c)$ とすれば，A_2^{IND} は，確率 $1/2 + 1/p(n)$ 以上で，m_0 と m_1 のどちらの暗号文であるかを当てている．したがって，IND-CPA 安全ではない．

ZK-CPA $\Rightarrow$ IND-CPA

IND-CPA 安全でないと仮定する．つまり，ある PPT アルゴリズム A_1, A_2 と多項式 $p(\cdot)$ が存在して，無限に多くの $n \in \mathbb{N}$ について，

$$\Pr\left[b = b' \;\middle|\; \begin{array}{l} (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_0, m_1, st) \leftarrow A_1(pk), \\ b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(m_b), b' \leftarrow A_2(st, c) \end{array}\right] \geq \frac{1}{2} + \frac{1}{p(n)}$$

を満たしている．ここで，$A_1(pk)$ を，以下の動作をする $A'_1(pk)$ に置き換えても，上記の不等式は成り立つ．

1. $(m_0, m_1, st) \leftarrow A_1(pk)$ を計算．
2. $st' = (m_0, m_1, st)$．
3. (m_0, m_1, st') を出力．

このとき，ZK-CPA における PPT アルゴリズム $A(pk, st)$ として以下を考える．

1. $(m_0, m_1, st') \leftarrow A'_1(pk)$ を計算．
2. $b \xleftarrow{R} \{0,1\}$ を選ぶ．
3. (m_b, st') を出力．

すると，

$$\begin{aligned}
&\Pr\left[b = b' \;\middle|\; \begin{array}{l}(pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_0, m_1, st') \leftarrow A'_1(pk), \\ b \xleftarrow{R} \{0,1\}, c \leftarrow \mathsf{Enc}_{pk}(m_b), b' \leftarrow A_2(st', c)\end{array}\right] \\
&\quad = \Pr[A_2(st', c) = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_b, st') \leftarrow A(pk), c \leftarrow \mathsf{Enc}_{pk}(m_b)] \\
&\quad = \Pr[D(m_0, st', c) = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_b, st') \leftarrow A(pk), c \leftarrow \mathsf{Enc}_{pk}(m_b)]
\end{aligned}$$

となる．ここで，アルゴリズム $D(m_0, st', c) = A_2(st', c)$ である．ZK-CPA であると仮定すると，A に対して PPT シミュレータ S が存在し，

$$\begin{aligned}
&\Pr[D(m_0, st', c) = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_b, st') \leftarrow A(pk), c \leftarrow \mathsf{Enc}_{pk}(m_b)] \\
&\quad \Pr[D(m_0, st', c') = b \mid (pk, sk) \leftarrow \mathsf{Gen}(1^n), (m_b, st') \leftarrow A(pk), c' \leftarrow S(pk, st')] \\
&\quad \geq \frac{1}{2} + \frac{1}{p(n)}
\end{aligned}$$

となるが，最後の不等式を達成するような D は存在しない．なぜならば，D が受け取る $(m_0, (m_0, m_1, st), c')$ には，b に関する情報が全く含まれていないため，b を $1/2$ より大きな確率で当てることはできないからである．したがって矛盾が導かれた． □

2 効率的な公開鍵暗号方式

これまでに見てきた安全な公開鍵暗号方式の構成は，効率の面で十分でなく，実用的ではない．効率的な方式は，TDP の存在という一般的な仮定ではなく，ある特定の計算量的仮定にもとづいて構成することが多い．ここでは，Diffie-Hellman 問題の困難性にもとづく，ElGamal 暗号方式を紹介する．この方式は，有名な Diffie-Hellman 鍵交換をもとにしている．

2.1 Diffie-Hellman 鍵交換

鍵交換とは，二者間でランダムな鍵を共有するための方式である．ランダムな鍵が共有できれば，使い捨て鍵暗号などの秘密鍵暗号方式を利用することができる．

Diffie-Hellman 鍵交換は以下の手続きで行われる．

1. Alice と Bob は，位数 q の巡回群 G を選び，その生成元 g をランダムに選ぶ．
2. Alice は，$a \in \mathbb{Z}_q$ をランダムに選び，$\alpha = g^a$ を計算し，α を Bob に送る．
3. Bob は，$b \in \mathbb{Z}_q$ をランダムに選び，$\beta = g^b$ を計算し，β を Alice に送る．

4. Bob は，α を受け取った後，$s_B = \alpha^b$ を計算し，s_B を共有鍵とする．
5. Alice は，β を受け取った後，$s_A = \beta^a$ を計算し，s_A を共有鍵とする．

Alice と Bob が計算した s_A と s_B が一致することは，

$$s_A = \beta^a = (g^b)^a = g^{ab} = (g^a)^b = \alpha^b = s_B$$

から簡単に確認できる．二人の通信を盗聴している Eve は，q, G, g, α, β を知ることができる．これらを知ったとしても，共有した秘密 $s = s_A = s_B$ がランダムに見えれば，鍵交換は成功したことになる．

以下に定義する Diffie-Hellman 判定仮定は，上記の鍵交換が安全に行われることを仮定している．

予想 15 (Diffie-Hellman 判定 (DDH) 仮定) 以下の二つの分布は，計算量的に識別できない．

- $\{(q, G, g, g^a, g^b, g^{ab}) \mid (q, G) \leftarrow \mathsf{Gen}(1^n), g \xleftarrow{R} \mathrm{G}_q, a, b \xleftarrow{R} \mathbb{Z}_q\}$
- $\{(q, G, g, g^a, g^b, g^c) \mid (q, G) \leftarrow \mathsf{Gen}(1^n), g \xleftarrow{R} \mathrm{G}_q, a, b, c \xleftarrow{R} \mathbb{Z}_q\}$

ただし，Gen は，n ビットの素数 q を位数とする巡回群 G を生成するアルゴリズムであり，G_q は G の生成元の集合である．

上記に示した方式では，どのような巡回群 G を利用すればよいか明示していない．例えば，素数 p に対して，$\mathbb{Z}_p^*$ は，法 p のもとでの乗法演算について巡回群となる．しかし，巡回群 $\mathbb{Z}_p^*$ に対して DDH 仮定は成り立たない．盗聴者である Eve は，$g^{ab} \in \mathbb{Z}_p^*$ が平方剰余であるかどうかを，簡単に計算できるのである．ここで，$a \in \{1, \ldots, p-1\}$ が平方剰余であるとは，$a = x^2 \bmod p$ を満たす $x \in \{1, \ldots, p-1\}$ が存在するときである．また，生成元 $g \in \mathbb{Z}_p^*$ に対して, $a = g^z \in \mathbb{Z}_p^*$ であるとき，(1) a が平方剰余であること，(2) z が偶数であること，(3) $a^{(p-1)/2} = 1 \mod p$ であることは等価であることが知られている．つまり，$a \in \mathbb{Z}_p^*$ が平方剰余であるとき，$a = g^z$ である z は偶数である．Eve は，g^a, g^b が平方剰余であるかどうかを (3) を利用して計算することができ，その結果，g^{ab} が平方剰余であるかを計算できる．これは，g^c と識別できることを意味する．なぜなら，ランダムに c を選んだとき，c が偶数である確率は 1/2 であり，g^c が平方剰余である確率は 1/2 である．一方，$a, b \in \mathbb{Z}_p^*$ をランダムに選んだとき，ab が偶数である確率は 3/4 であり，g^{ab} が平方剰余である確率は 3/4 である．つまり，平方剰余が計算できると，g^{ab} と g^c を識別できるのである．

上記の問題点を克服する方法として，$p = 2q + 1$ を満たす素数 p, q を利用する方法がある．巡回群 G として，$\mathbb{Z}_p^*$ ではなく，その部分群である，平方剰余の集合を考える．つまり，$\mathbb{Z}_p^* = \{1, 2, \ldots, p-1\}$ に対して，$G = \{1^2 \bmod p, 2^2 \bmod p, \ldots, (p-1)^2 \bmod p\}$ とする．このとき，G の位数は $(p-1)/2$ であることが知られている．つまり，$p = 2q + 1$ であるとき，G の位数は素数 q となる．また，$\mathbb{Z}_p^*$ の生成元 h に対して，$g = h^2 \bmod p$ は G の生成元であり，$\mathbb{Z}_p^*$ の生成元をランダムにサンプルできれば，G の生成元もランダムにサンプル可能である．

上記で利用した，$p = 2q + 1$ を満たす素数 p, q は，Sophie Germain 素数と呼ばれており，無限に存在するかどうかについては未解決である．特に，p は安全素数 (safe prime) と呼ばれている．

2.2 ElGamal 暗号方式

ElGamal 暗号方式は以下のように定義される．

1. $\mathsf{Gen}(1^n) : (q, G) \leftarrow \mathsf{Gen}_G(1^n), g \xleftarrow{R} \mathrm{G}_q,\ a \xleftarrow{R} \mathbb{Z}_q.\ \mathcal{M}_n = G.\ (pk, sk) = ((q, G, g, g^a), a)$ を出力．
2. $\mathsf{Enc}_{pk}(m)\ b \xleftarrow{R} \mathbb{Z}_q.\ (g^b, g^{ab} \cdot m)$ を出力．
3. $\mathsf{Dec}_{sk}(c_1, c_2) : c_2 \cdot c_1^{-a}$ を出力．

この方式は，暗号化と復号のアルゴリズムが効率的である．実際に，どちらも，ベキ剰余演算二回以下で計算できる．また，暗号文のサイズは，平文の二倍で抑えられている．

定理 16 DDH 仮定のもとで，ElGamal 暗号方式は，IND-CPA 安全である．