卒業研究論文

反復回数改ざんを利用した楕円曲線暗号に対するフォールト攻撃

Fault attacks to elliptic curve cryptosystems by tampering the number of iterations

学籍番号　12D8101003C
渡邊　千尋
Chihiro Watanabe

中央大学理工学部情報工学科
趙研究室

2016年3月

概論

現在, あらゆる情報機器に暗号が実装されており, 暗号の安全性がセキュリティの根拠となっている. 暗号装置への実装における安全性解析において注目を集めている手法にフォールト攻撃がある. フォールト攻撃とは, 暗号装置への物理的干渉により意図的に機器にエラーを起こさせ, 秘密情報を推測・特定する攻撃手法である.

本研究では, 楕円曲線暗号におけるスカラー倍算中のループ部分の反復回数をフォールトの導入対象とした, 新しい攻撃手法を提案する. さらに, NIST 推奨パラメータを用いた楕円曲線暗号への攻撃を行い, 攻撃の有効性を示した.

キーワード

- 楕円曲線暗号
- 離散対数問題
- フォールト攻撃

目次

第 1 章

序論

近年, インターネットの普及と発展が急速に進み, ネットワーク上であらゆる情報のやり取りが行われている. また, スマートカードやパソコンといった比較的計算機性能の制限された装置間でのデータのやりとりも増加している. それに伴い, このような環境に適した情報セキュリティ技術がさかんに研究されており, その根本的な技術となっているのが暗号技術である.

暗号技術には, 共通鍵暗号方式と公開鍵暗号方式の 2 つの方式がある. 共通鍵暗号方式は, 暗号化と復号時に同じ鍵を用いる. 鍵を送信者と受信者の二者間でしか共有しないため, 安全な通信となり処理が比較的高速である. しかし, 通信相手への安全な鍵の受け渡し方法や, 複数の相手とやり取りをする際に増える鍵の管理方法が課題となる. 一方, 公開鍵暗号方式は, 相手が公開している鍵を用いて暗号化し, 相手は非公開の秘密鍵で復号を行うといったように, 暗号化と復号時に異なる鍵を用いる. 鍵の配送が容易であるが, 暗号化や復号の処理が重いという課題がある.

公開鍵暗号方式の安全性は, 安全性の根拠としている数学的問題を解く手法の中で, 最も効率の良い手法を用いた場合に要する計算時間 (計算量) によって評価される. 鍵長を長くしていった際の計算量の増加の程度により, 解くのに必要な時間は, 指数関数時間, 準指数関数時間, 多項式時間, という 3 つのカテゴリに分類できる. 数学的問題を解くのに要する時間が指数関数時間, または準指数関数時間のとき, その暗号は計算量的に安全とされる.

インターネットで広く使われてきた暗号方式として, 公開鍵暗号方式の RSA 暗号がある. RSA 暗号は, 素因数分解の求解の困難性が安全性の根拠となっている. この従来の RSA 暗号に代わって注目され, 利用が進んできているのが楕円曲線暗号である. その理由は, 主に次の 2 つが挙げられる.

1. RSA 暗号と比較して, 約 1/10 程度の鍵長で同程度の安全性が保証されるため, 計算機性能 (計算能力やメモリ等) が制限された環境 (IC カードや組込み機器等) での利用に適している
2. RSA 暗号と鍵生成の仕組みが異なる点から, 脆弱な鍵を発行しやすい等, 安全性に関わる運用時の問題が生じにくい

楕円曲線暗号の離散対数問題は, 一部の曲線を除いて, 指数関数時間で求解可能な攻撃手法は一つ (ρ 法) しか知られていない. よって, 離散対数問題を解くのに必要な計算量を同程度に設定した場合, RSA 暗号のような素因数分解問題に依拠した暗号アルゴリズムよりも相対的に短い鍵長で済む.

楕円曲線暗号に対する効率的な攻撃手法はまだ見つかっていないため, 楕円曲線暗号を搭載した実装部分への攻撃の危険性が指摘されている. 例えば, 暗号装置の物理的な値 (電磁波や熱など) を観測し秘密情報を盗み出す攻撃をサイドチャネル攻撃という. 特に, IC カードなどは攻撃者が処理時間や消費電力を精密に計測できてしまう.

また, サイドチャネル攻撃の一つに, 物理的に機器にエラーを起こさせ, 不正な値を集めて秘密情報を推測・特定するフォールト攻撃がある. このように, 多項式時間で攻撃可能な手法が見つかっていなくても, 実装の脆弱性を狙った

攻撃を考慮する必要がある.

暗号装置の安全な利用のためにはあらゆる攻撃方法を考慮する必要がある. よって, 本研究では楕円曲線暗号方式に対する新しいフォールト攻撃を提案し, その有効性を示す.

第 2 章

準備

2.1 群

集合 G の直積集合 $G \times G$ から集合 G への写像が 1 つ与えられているとする. この写像を G の **2 項演算**と呼ぶ. $G \times G$ の元 (a, b) の, この写像による像を a と b の積といい, 記号 $a \circ b$ で表す.

集合 G $(\neq \emptyset)$ に対して 2 項演算が成り立ち, 以下の 3 つの条件を全てを満たすとき, G はこの演算に関して**群**であるという.

1. 結合法則
   $^\forall a, b, c \in G$ に対して, $(a \circ b) \circ c = a \circ (b \circ c)$ が成り立つ.
2. 単位元の存在
   $^\forall a \in G$ に対して, $a \circ e = e \circ a = a$ となる $e \in G$ が存在する.
3. 逆元の存在
   $^\forall a \in G$ に対して, $a \circ b = b \circ a = e$ となる $b \in G$ が存在する.

また, 2 項演算が 1. のみを満たす G と $\circ$ の組 $(G, \circ)$ をこの 2 項演算に関する **半群** という. そして, 群 G が下記の 4. も同時に満たすとき, G を **可換群**, または **アーベル群**であるという.

4. 交換法則
   $^\forall a, b \in G$ に対して, $a \circ b = b \circ a$ が成立する.

ちなみに, 群 $(G, \circ)$ において 2 項演算が明らかな場合は, 単に群 G ということもある.
また, 群 G に属する元の個数が有限であるとき G を**有限群**, そうでないとき無限群という.

集合 G が 2 項演算 $\circ$ に対して可換群であり, $\circ$ が $+$ で表される場合その群を**加法群**と呼ぶ. そのとき $x + y$ を x と y の和といい, 単位元を 0, x の逆元を $-x$ で表す.

群 G に属する元の個数を G の**位数**といい, 記号 $|G|$ で表す. G が無限群のときは $|G| = \infty$ とする.
また, 単位元を e とする群 G の元 a に対して, $a^n = e$ となるような最小の正整数があるとき, それを a の**位数**といい, 記号 $|a|$ で表す. そのような整数が無いとき, a の位数は無限といい, $|a| = \infty$ と表す.

可換群 G の任意の元が 1 つの元 a のべき乗になっているとき, G を a で生成された**巡回群**といい, $G = \langle a \rangle$ で表す. a を**生成元**という.

2.2 環

2.2.1 環の定義

2 つの 2 項演算 (加法 $+$ と乗法 $\cdot$) の定義された集合 R が以下 4 つの条件を満たすとき, $(R, +, \cdot)$ は**環**であるという.

1. 加法に関して加法群をなす.
2. 乗法に関して結合法則が成り立つ.
   すなわち, $^{\forall}a, b, c \in R$ に対して, $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ が成り立つ.
3. 乗法に関して単位元が存在する.
   すなわち, $^{\exists}e \in R, ^{\forall}a \in R$ に対して, $a \cdot e = e \cdot a = a$
4. 分配法則が成り立つ. すなわち, $^{\forall}a, b, c \in R$ に対して, 以下の 2 つの等式が成り立つ.
   $a \cdot (b + c) = a \cdot b + a \cdot c$
   $(a + b) \cdot c = a \cdot c + b \cdot c$

特に, 環 R が下記の 5. も同時に満たすとき**可換環**であるといい, 満たさないときは非可換環という.

5. 交換法則が成り立つ.
   すなわち, $^{\forall}a, b \in R$ に対して $a \cdot b = b \cdot a$ が成り立つ.

環 R において, **単位元**というときは, 乗法単位元 1_R のことを意味する. 加法の単位元は **零元** 0_R である. また, 乗法の表現 $a \cdot b$ は, 簡単のために ab と表すことが多いため, 以下この記法を用いることにする.

零元 0_R と異なる零因子のない可換環を**整域**という. すなわち, $ab = 0$ $(a, b \in R)$ ならば $a = 0_R$ または $b = 0_R$.

2.2.2 部分環, イデアル, 商環

環 R の単位元 1_R を含んでいる部分集合 S が, R の演算に関して環 (あるいは体) になっているとき, S を R の**部分環** (あるいは**部分体**) という.
環 R の空でない部分集合 I について, 以下の 3 つの条件を考える.

1. $^{\forall}a, b \in I \Rightarrow a + b \in I$
2. $^{\forall}a \in I, ^{\forall}r \in R \Rightarrow ra \in I$
3. $^{\forall}a \in I, ^{\forall}r \in R \Rightarrow ar \in I0$

条件 1. と 2. を満たすとき, I を環 R の**左イデアル**といい, 条件 1. と 3. を満たすとき, I を環 R の**右イデアル**という. そして条件 1. から 3. まで全てを満たすとき, すなわち左イデアルでかつ右イデアルである I は環 R の**両側イデアル**もしくは単に**イデアル**という. R が可換環であれば, 左イデアル, 右イデアル, 両側イデアルは一致する.

環 R の元 a, b がイデアル I を法として合同であるとは, $a + i = b$ となる元 $i \in I$ が存在することであり, このとき $a \equiv b \pmod{I}$ と書く. この関係は同値関係である. 環 R のイデアル I による同値類を $[a]$ と書けば, $[a] = a + I = \{a + i \mid i \in I\}$ となり, 同値類の集合 R/I に加法と乗法が以下のように定義できる.

$$[a] + [b] = [a + b] \ , \quad [a] \cdot [b] = [ab]$$

これらの演算に関して同値類の集合 R/I は環になり, I を法とする R の商環または剰余環という. 整域 R のイデアル I に対し剰余環 R/I が整域であるとき, I を R の**素イデアル**という.

2.2.3 準同型

R と R' を環とし, f を R から R' への写像とする. 任意の $a,\ b \in R$ について以下が満たされているとき, f を R から R' への環の**準同型写像**であるという.

$$f(a+b) = f(a) + f(b),$$

$$f(ab) = f(a)f(b),$$

$$f(R_1) = 1_{R'}$$

さらに, f が単射であるとき f を**環の単準同型写像**, 全射であるとき**環の全準同型写像**, 全単射であるとき**環の同型写像**という. また, R から R' への環の同型写像が存在するとき R と R' は**環として同型**であるといい, $R \simeq R'$ と表す. R から R 自身への環の同型写像 f を**環 R の自己同型写像**という.

2.2.4 多項式環

可換環 R において, R とは関係ない文字 X を R 上の変数という. R 上の X の多項式とは,

$$f(X) = a_n X^n + a_{n-1}X^{n-1} + \cdots + a_1 X + a_0 (a_i \ \in \ R)$$

の形の式のことである. $a_n \neq 0$ のとき, $f(X)$ は n 次の多項式であり, n を $f(X)$ の**次数**といい, $n = \deg f(X)$ と表す. ただし, すべての i について $a_i = 0$ である $f(X)$ の次数は定めない. また,

$$f(X) = b_m X^m + b_{m-1}X^{m-1} + \cdots + b_1 X + b_0 (b_i \in R)$$

を X の多項式として $m \leq n, a_0 = b_0, \cdots, a_m = b_m, a_{m+1} = \cdots = 0$ であるときに限り, $f(X) = g(X)$ であると定義する.
R 上の多項式全体の集合を $R[X]$ と表し, $R[X]$ に和と積を以下のように定義する.

$$f(X) + g(X) = \sum_i (a_i + b_i)X^i,$$

$$f(X) \cdot g(X) = \sum_k (\sum_{i+j=k} a_i b_j) X^k$$

これにより, $R[X]$ は, 上の 2 つの演算に関して可換環になる. ただし, $x^0 = 1, 0 \cdot x = 0$ $(0, 1 \in R)$ と定める. こうして得られた環 $R[X]$ を R 上の**多項式環**という.

R を整域とする. 多項式環 $R[X]$ の元 $f(X), g(X)$ について, 積 $f(X)g(X)$ の次数は $f(X)$ の次数と $g(X)$ の次数の和である. すなわち以下の式が成り立つ.

$$\deg f(X)g(X) = \deg f(X) + \deg g(X)$$

多項式環 $K[X]$ において, K が体であれば $K[X_1]$ は整域となり, 同様に $K[X_1, \ldots, X_n]$ も整域である. 体 K 上の n 変数の多項式環 $K[X_1, \ldots, X_n]$ の商体を K 上の有理関数体といい, $K(X_1, \ldots, X_n)$ で表し, $K(X_1, \ldots, K_n)$ の元を K 上**有理式**という. ${X_1}^{i_1}{X_2}^{i_2} \cdots {X_n}^{i_n}$ の形の元を**単項式**という. $K(X_1, \ldots, X_n)$ の任意の元は $K[X_1, \cdots, X_n]$ の元 $f(X_1, \cdots, X_n),\ g(X_1, \cdots, X_n)$ により,

$$\frac{f(X_1, \cdots, X_n)}{g(X_1, \cdots, X_n)}$$

と表される.

2.3 体

環 R の 0_R 以外の元がすべて R において可逆元であるとき, R を**斜体**という. R の乗法が可換であれば R を**可換体**または**体**という.

2.3.1 有限体

有限個の元からなる体を**有限体**といい, 元の個数を体の**位数**という. 体 K を有限体とし, K の単位元を 1 で表す. このとき, $\mathbb{Z}$ から K への写像 τ を $\tau\ (n) = n1$ によって定める.

$$\tau : \mathbb{Z} \longrightarrow K$$

$$n \longmapsto n1$$

写像 τ は環準同型写像になり, $\mathbb{Z}$ から K への任意の環準同型写像は $\mathbb{Z}$ の単位元 1 を K の単位元 1 に写像するため τ と一致する. よって, $\mathbb{Z}$ から K への環準同型写像は τ 唯一つだけである. $\ker\tau$ は $\mathbb{Z}$ のイデアルであり,

$$\mathbb{Z}/\mathrm{ker}\,\tau \simeq \tau\ (\mathbb{Z})$$

となる. $\tau\ (\mathbb{Z})$ は K 部分環のため整域となり, よって $\ker\tau$ は $\mathbb{Z}$ の素イデアルである. $\ker\tau = (0)$ とすると $\mathbb{Z} \simeq \tau\ (\mathbb{Z})$ となり, これは $\tau\ (\mathbb{Z})$ が有限集合 K の部分集合であることに矛盾する. したがって $\ker\tau = (p)$(p は素数) である必要がある. よって,

$$\mathbb{Z}/(p) \simeq \tau\ (\mathbb{Z})$$

となり, この p を体 K の**標数**という. 体 K の標数 p は, K の任意の元 a に対して $pa = 0$ である.

有限体 K の 0 以外の元からなる乗法群 K^* は巡回群である. また, K^* の生成元を有限体 K の**原始根**という.

2.3.2 拡大体

体 K の単位元 1_R を含む部分集合 F が K の演算に関して体になっているとき, F を K の**部分体**という. また, 有限体 K の部分体 F が与えられたとき, $F \subset K$ と書き, 体 K は体 F の**拡大体**であるという.

体 L が体 K の拡大体であり, α は L の元とする. α が K 上**代数的**であるとは, α が K の元を係数とするゼロではない多項式の根になっていることと定義する.

L の元 α が K 上代数的でないとき, α は K 上**超越的**であるという.
K 上の 1 変数多項式環 $K[X]$ から L への準同型写像

$$\sigma : K[X] \longrightarrow L$$

$$f(X) \longmapsto f(\alpha)$$

によって, α が K 上代数的であるか超越的であるか判別することができる.

$K \subset L$ であり $\alpha \in L$ であるとき, L の部分体であって K と α を含むものすべての共通部分を記号 $K(\alpha)$ で表し, K に α を**添加**した体という. これは K と α を含む L の部分体のうちで最小のものである. α が K 上代数的であっても超越的であっても $K(\alpha)$ のようにただ 1 個の元を添加して得られる K の拡大体を K の**単純拡大**という.

K に係数を持つ全ての多項式を 1 次因子に完全に分解するという性質を体 K が持つなら, K は代数的に閉じてい

るという．これは, K に係数を持つ全ての多項式が, K に根を持つことと同値である．代数的に閉じている最小の K の拡大体は, K の**代数的閉包**と呼ばれ, $\overline{K}$ で表される．

整数環 $\mathbb{Z}$, 素数 p に対して, p の倍数の集合を $p\mathbb{Z}$ とすると, $p\mathbb{Z}$ は $\mathbb{Z}$ のイデアルである．また, 商環 $\mathbb{Z}_p = \mathbb{Z}/p\mathbb{Z}$ は p 個の元からなる体になる．これと同型な体を $\mathbb{F}_p$ で表す．

2.4 中国の剰余定理

$m_1, \ldots, m_r$ を 1 より大きい整数とし, $(m_i, m_j) = 1 (i \neq j)$ とする．このとき任意の整数の組 $a_1, \ldots, a_r$ に対して以下の連立合同式は, $m =_1 \ldots m_r$ を法として唯一つの解を持つ．

$$\begin{cases} Z \equiv z_1 \pmod{m_1} \\ \quad \vdots \\ Z \equiv z_r \pmod{m_r} \end{cases} \tag{2.1}$$

2.5 離散対数問題

群 G における $g \in G$ に対する離散対数問題とは, $y \in G$ が与えられるとき, $g^x = y$ (演算を加法的に書くと $xg = y$) である整数 x が存在するとしたとき, それを求めるという問題のことである．この x を y の離散対数という．

第 3 章

楕円曲線と楕円曲線暗号

近年, 有限体上で定義される楕円曲線論が暗号理論へ応用されている. その理由は有限体上の楕円曲線が沢山の種類のアーベル群を提供し, 有限体に比べて高い柔軟性を持つためである.

3.1 楕円曲線の定義

体 K 上で定義される楕円曲線とは, 一般的に

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \quad (a_1, a_2, a_3, a_4, a_6 \in K) \tag{3.1}$$

で与えられる (x, y) に関する方程式のことである. K 上の楕円曲線とは, 無限遠点と呼ばれる要素 $\mathcal{O}$ を加えた $x, y \in K$ である点 (x, y) の集合を表す. このとき, 係数 a_n が属する体 K を**係数体**, 変数 x, y が属する体を**定義体**と呼ぶ.

もし, K が標数 2 の体であるとき, K 上の楕円曲線とは, 無限遠点 $\mathcal{O}$ を含む方程式

$$y^2 + xy = x^3 + ax^2 + b \quad (a, b \in K) \tag{3.2}$$

または,

$$y^2 + cy = x^3 + ax + b \quad (a, b, c \in K) \tag{3.3}$$

のどちらかを満たす点の集合である.

また, K が標数 3 の体であるとき, K 上の楕円曲線とは, 無限遠点 $\mathcal{O}$ を含む方程式

$$y^2 = x^3 + ax^2 + bx + c \quad (a, b, c \in K) \tag{3.4}$$

を満たす点の集合である.

K の標数が 3 より大きい場合は,

$$E : y^2 = x^3 + ax + b \quad (a, b \in K) \tag{3.5}$$

を満たす点の集合である.

$F(x, y) = 0$ を, 式 (3.5)(または式 (3.2), 式 (3.3), 式 (3.4)) における y に対する x の陰関数方程式とする. すなわち, $F(x, y) = y^2 - x^3 - ax - b$ とする. このとき, 2 つの偏微分 $\partial F/\partial x$, $\partial F/\partial y$ のうち少なくともひとつがゼロでない曲線上の点 (x, y) は, **非特異** (または滑らかな点) であるという. また, 式 (3.5) や式 (3.4) の右辺の 3 次多項式が多重根を持たないということと, 曲線上のすべての点が非特異でなければならない, ということは同値である.

3.2 フロベニウス写像

p を素数, n を自然数とし, $q = p^n$ とする. 有限体 $\mathbb{F}_q$ 上で, 曲線上の有理点は有限である. 有理点の個数を $\#E(\mathbb{F}_q)$ で表し, t を以下のように定義する.

$$\#E(\mathbb{F}_q) = q + 1 - t$$

これにより定義された t を, q におけるフロベニウスのトレースと呼ぶ.

$\mathbb{F}_q$ 上の楕円曲線 E の q 上フロベニウス写像 φ は以下の様に定義される.

$$\begin{array}{rcl} E(\overline{\mathbb{F}}_q) & \to & E(\overline{\mathbb{F}}_q) \\ (x, y) & \to & (x^q, y^q) \\ \infty & \to & \infty \end{array}$$

この写像 φ は $E(\overline{\mathbb{F}}_q)$ の群自己同型であり, フロベニウス自己同型写像と呼ばれる. また, ハッセの定理よりフロベニウスのトレース t は

$$|t| \leq 2\sqrt{q}$$

をみたす.

楕円曲線 E には, supersingular 曲線, anomalous 曲線と呼ばれる 2 つの特別なクラスがある.

それぞれ, 標数 p が曲線 $E(\mathbb{F}_q)$ のフロベニウスのトレース t を割る場合, 曲線 $E(\mathbb{F}_q)$ のトレースが 1 となる場合である.

3.3 Hasse の定理

N を体 $\mathbb{F}_q$ 上で定義された楕円曲線の $\mathbb{F}_q$ の点の個数としたとき,

$$|N - (q+1)| \leq 2\sqrt{q} \tag{3.6}$$

が成り立つ.

3.4 楕円曲線上の点の加算

楕円曲線上の有理点における点の加算を以下のように定義すると, 有理点全体と $\mathcal{O}$ の和集合は可換群を成すことが知られている.

楕円曲線上の点 P と Q の加算を考える. ます, 点 P, Q を通る直線を描き, 直線 PQ と曲線の第三の交点 R を見つけ, R の x 軸について対称な点 $P + Q$ とする. $P = Q$ の場合は, P から曲線の接線を描き, 接線と曲線の第二の交点の x 軸について対称な点を $2P$ とする.

この加算で楕円曲線は群構造をなす. 例えば, 点 $P = (x, y)$ と $Q = (x, -y)$ の場合, 第三の交点は無限遠点となる. よって, $P + Q = \mathcal{O}$ となり点 Q が点 P の逆元, $-P$ となる. すなわち単位元が無限遠点, 逆元は x 軸と対称な点になる. 結合法則は加算の定義により自明である.

この定義を実数上で描かれた楕円曲線のグラフを用いて示す. ただし, $P, Q \in E(\mathbb{F}_q)$ について $P = (x_1, y_1)$, $Q = (x_2, y_2)$ とする.

[Case 1] $P \neq Q$, $(x_1 \neq x_2) \longrightarrow$ 図 3.1
[Case 2] $P = Q \longrightarrow$ 図 3.2

[Case 3] $P \neq Q,\ (x_1 = x_2) \longrightarrow$ 図 3.3

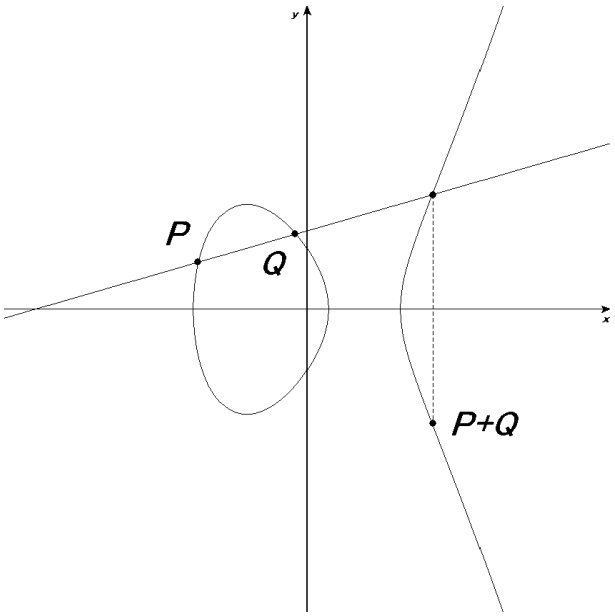

図 3.1 $P + Q$

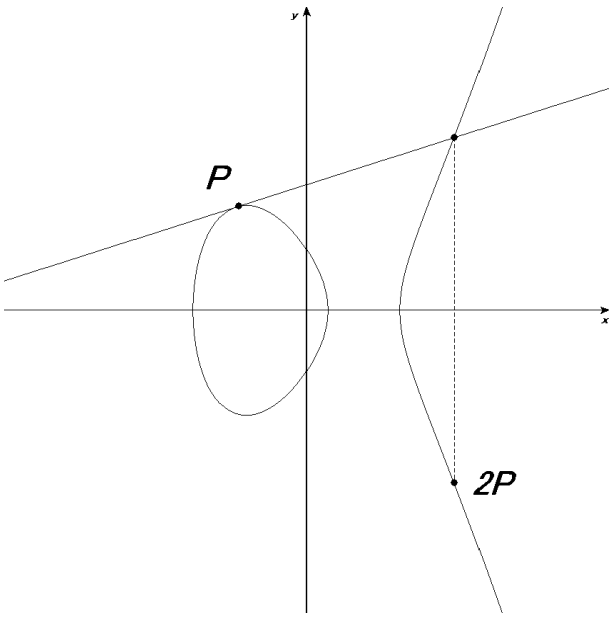

図 3.2 $2P$

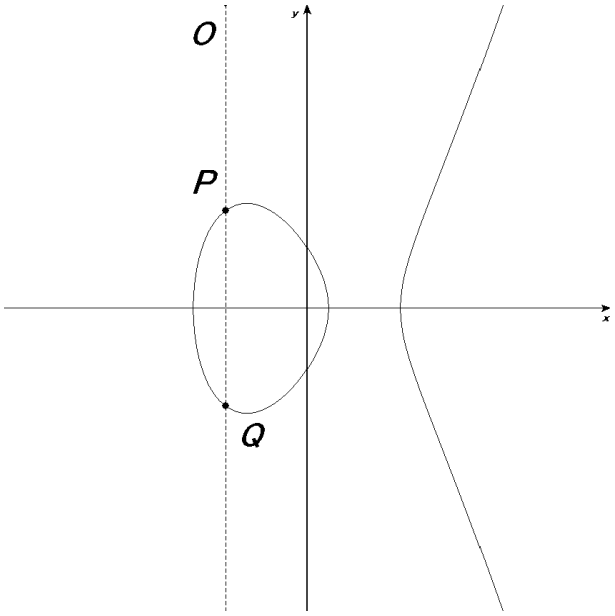

図 3.3 $P + Q = \mathcal{O}$

ここで, $P+Q$ を効率的に計算する為の公式を与える. また, 計算を簡略化する為に, 楕円曲線は標数が 3 より大きいものとする.

$(x_1, y_1), (x_2, y_2), (x_3, y_3)$ を, それぞれ P, Q, $P+Q$ の座標表示とし, 2 点 P, Q を通る直線の方程式を, $y = \alpha x + \beta$ とする. x_3, y_3 を, x_1, y_1, x_2, y_2 を用いて表すことを考える.

$P \neq Q$ のとき, $\alpha = (y_2 - y_1)/(x_2 - x_1)$, $\beta = y_1 - \alpha x_1$ は明らかである.

弦 PQ 上の点を $(x, \alpha x + \beta)$ とすると, この点と曲線の交点は,

$$y^2 = (\alpha x + \beta)^2 = x^3 + ax + b \tag{3.7}$$

を満たす点である. このことから, $x^3 - (\alpha x + \beta)^2 + ax + b = 0$ の根の数は, 交点の数に等しいことが分かる. $x^3 - (\alpha x + \beta)^2 + ax + b$ はモニック多項式なので, モニック多項式の解の和が多項式の 2 番目に高次の項の係数の負の数になることを用いて, 以下が得られる.

$$x_3 = \alpha^2 - x_1 - x_2 = \Big(\frac{y_2 - y_1}{x_2 - x_1}\Big)^2 - x_1 - x_2 \tag{3.8}$$

$$y_3 = \alpha(x_1 - x_3) - y_1 = -y_1 + \Big(\frac{y_2 - y_1}{x_2 - x_1}\Big)(x_1 - x_3) \tag{3.9}$$

$P = Q$ のとき, α は, $\alpha = \frac{\partial_y}{\partial_x} = \frac{\partial_y}{\partial_E} \cdot \frac{\partial_E}{\partial_x}$ を用いて,

$$\alpha = \frac{3x_1^2 - a}{2y_1} \tag{3.10}$$

となり、以下が得られる.

$$x_3 = \Big(\frac{3x_1^2 + a}{2y_1}\Big)^2 - 2x_1 \tag{3.11}$$

$$y_3 = -y_1 + \Big(\frac{3x_1^2 + a}{2y_1}\Big)(x_1 - x_3) \tag{3.12}$$

3.5 Torsion group

楕円曲線上の点 P の位数とは, $NP = \mathcal{O}$ となる最も小さな正整数 N である.

E を体 K 上で定義された楕円曲線とする. 自然数 n について楕円曲線 E/K 上の点の位数が n となる場合,

$E[n] = \{P \in E(\overline{K}) \mid nP = \mathcal{O}\}$ (ただし, $\overline{K}$ は K の代数的閉包)

を n-torsion group といい, $E[n]$ のことを n-torsion point という.

3.6 楕円曲線上の点のスカラー倍算

楕円曲線上の点のスカラー倍算とは, 楕円曲線上の任意の点 $P \in E(\mathbb{F}_q)$ と, 任意の整数 $n \in \mathbb{Z}$ に対して

$$nP = \begin{cases} P + P + \cdots + P \ (n \text{ 回}) & (n > 0) \\ -P - P - \cdots - P \ (-n \text{ 回}) & (n < 0) \end{cases}$$

と定義される. この演算は楕円曲線暗号の根幹を成す部分であり, 暗号演算の大半の時間を占めている. この演算をより高速に計算でき, かつ安全性を兼ね備えたスカラー倍算アルゴリズムが提案されている.

3.6.1 Algorithm 1 : Double and add

以下は秘密鍵 k を 2 進展開 $k = (k_{n-1}, ..., k_0)_2$ として表し, スカラー倍算 kP を計算するアルゴリズムである [1].

表 3.1 Algorithm 1:Double and add

input: $k = (k_{n-1}, ..., k_0)_2, P \in E(K)$ **output**: kP
1 : $Q \leftarrow \mathcal{O}$
2 : **for** $i = n - 1$ **down to 0 do**
2.1 : $Q \leftarrow 2Q$
2.2 : **if** $k_i = 1$
2.2.1 : $Q \leftarrow Q + P$
3 : **end for**
4 : **return** Q

Algorithm 1 の計算時間は, k_i の値に依存するため, 暗号装置の電力消費を観測するサイドチャネル攻撃に対して脆弱性を持つ.

3.6.2 Algorithm 2 : Double and add always

Double and add always は, Algorithm 1 を改良した, 各演算ステップの計算量が等しいスカラー倍算アルゴリズムである [2].

表 3.2 Algorithm 2:Double and add always

input: $k = (k_{n-1}, ..., k_0)_2, P \in E(K)$ **output**: kP
1 : $Q[0] \leftarrow \mathcal{O}$
2 : **for** $i = n - 1$ **down to 0 do**
2.1 : $Q[0] \leftarrow 2Q$
2.2 : $Q[1] \leftarrow Q[0] + P$
2.3 : $Q[0] \leftarrow Q[k_i]$
3 : **end for**
4 : **return** $Q[0]$

3.6.3 Algorithm 3 : Montgomery ladder

Montgomery ladder[3] はさらに Algorithm 1, Algorithm 2 を改良した, スカラー倍算アルゴリズムである. Montgomery ladder は, 点の加算や倍算において y 座標を使用せずに x 座標を計算可能だということが示されている.

具体的には, $P = (x_1, y_1), Q = (x_2, y_2), Q - P = (x_3, y_3)$ を, 楕円曲線 $E : y^2 = x^3 + ax + b$ 上の点に移し, $P + Q = (x_4, y_4), 2P = (x_5, y_5)$ の x 座標を以下の方程式で計算できる.

$$x_4 = \frac{2(x_1 + x_2)(x_1 x_2 + a) + 4b}{(x_1 + x_2)^2} - x_3 \tag{3.13}$$

$$x_5 = \frac{(x_1^2 - a)^2 - 8bx_1}{4(x_1^3 + ax_1 + b)} \tag{3.14}$$

このことを利用した高速なスカラー倍算アルゴリズムである Montgomery ladder を以下に記す.

表 3.3　Algorithm 3: Montgomery ladder

input: $k = (k_{n-1}, ..., k_0)_2, P \in E(K)$
output: The x-coordinate of kP
$1: Q[0] \leftarrow \mathcal{O}, Q[1] \leftarrow P$
$2:$ **for** $i = n-1$ **down to 0 do**
$2.1: Q[1-k_i] \leftarrow Q[0] + Q[1]$
$2.2: Q[k_i] \leftarrow 2Q[k_i]$
$3:$ **end for**
$4:$ **return** $Q[0]$

3.7　楕円曲線暗号

楕円曲線暗号の安全性は, **楕円曲線離散対数問題 (ECDLP)** の求解の困難性に基づいている. ECDLP を解くには, 一般に完全指数時間かかる. しかし, supersingular な曲線などの特別な曲線は準指数関数時間で解けることも示されている. そのため, 楕円曲線暗号を実装する際には暗号的に脆弱な曲線を避けた, 強固な曲線を選ぶ必要がある.

3.7.1　楕円曲線上の離散対数問題

有限体 $K = \mathbb{F}_q$ 上の楕円曲線を E とし, その曲線上の位数 n の点を $P \in E(K)$ とする. 点 P に対して, $\langle P \rangle = \{\mathcal{O}, P, 2P, \ldots\}$ は有限巡回群となる. この巡回群の位数を n とすると, 任意の $Q \in \langle P \rangle$ に対して,

$$xP = Q \ (x \in (\ \mathbb{Z}\ /\ \mathbb{Z}_n))$$

となる x が唯一つ存在する.

P と Q が与えられたとき, x を求める問題を楕円曲線離散対数問題という. これは, x と P から $xP = Q$ となる Q を求めることは容易だが, P と Q から x を求めるのは非常に困難であることに基づいている.

第 4 章

ペアリング暗号

ペアリング暗号とは, 楕円曲線上で定義される双線形写像である「ペアリング」を用いた暗号方式である. ペアリング暗号を応用した「関数型暗号」や「検索可能暗号」は, 従来の公開鍵暗号では実現困難であったクラウドに適した新しい暗号として研究が盛んに行われている.

4.1 ペアリング

ペアリングとは, 楕円曲線上の torsion group の直積から有限体の乗法群への写像である. ペアリングの例としては，有限体上の楕円曲線上の Weil ペアリングと Tate ペアリングがある.

4.1.1 双線形ペアリング

G_1, G_2 を単位元 0 の加法に関するアーベル群とし, G_3 を単位元 1 の乗法に関する位数 n の巡回群とする. このとき,

$$e : G_1 \times\ G_2 \rightarrow G_3 \tag{4.1}$$

が, 以下の 2 つの性質を満たす時, e をペアリング写像と呼ぶ.

1. 双線形性
   $^{\forall}P, P' \in G_1$ と $^{\forall}Q, Q' \in G_2$ に対して, 以下の 2 つが成り立つ.

$$e(P + P', Q) = e(P, Q) + e(P', Q) \tag{4.2}$$

$$e(P, Q + Q') = e(P, Q) + e(P, Q') \tag{4.3}$$

2. 非退化

   - $^{\forall}P \in G_1 (P \neq\ 0)$ に対して $e(P, Q) \neq\ 1$ となる $Q \in G_2$ が存在する.
   - $^{\forall}P \in G_2 (P \neq\ 0)$ に対して $e(P, Q) \neq\ 1$ となる $Q \in G_1$ が存在する.

また, e を双線形ペアリングとし, $P \in G_1,\ Q \in G_2$ とすると, 以下が成り立つ.

1. $e(P, 0) = e(0, Q) = 1$
2. $e(-P, Q) = e(P, Q)^{-1} = e(P, -Q)$
3. すべての $j \in \mathbb{Z}$ に対して, $e([j]P, Q) = e(P, Q)^j = e(P, [j]Q)$

4.1.2 Weil ペアリング

K を体とする．E を K 上で定義された楕円曲線とし，n を K の標数と互いに素な整数とする．$E[n]$ を n-torsion group とし，μ を $\overline{K}$ の中の 1 の n 乗根の集合，すなわち

$$\mu_n = \{x \in \overline{K} \mid x^n = 1\}$$

とする．Weil ペアリングとは

$$e_n : E[n] \times E[n] \to \mu_n \subseteq \overline{K}^*$$

となる写像のことである．Weil ペアリングには以下に示すような性質がある．
$P, P', Q, Q' \in E[n]$ とする．

1. 任意の P, P', Q, Q' に対し，

   $$e_n(P + P', Q) = e_n(P, Q)e_n(P', Q)$$

   かつ

   $$e_n(P, Q + Q') = e_n(P, Q)e_n(P, Q')$$

   である．
2. $e_n(P, P) = 1$
3. $e_n(P, Q) = e_n(Q, P)^{-1}$
4. 任意の Q において $e_n(P, Q) = 1$ ならば，$P = \mathcal{O}$ である．

4.1.3 Tate ペアリング

E を K_0 上の楕円曲線とする．n を体 K_0 の標数と互いに素な正整数とする．1 の n 乗根の集合を $\mu_n = \{x \in \overline{K_0}^* \mid x^n = 1\}$ とする．1 の n 乗根から生成された K_0 の拡大体を $K = K_0(\mu_n)$ と定義する．また，

$$E(K)[n] = \{P \in E(K) \mid nP = \mathcal{O}\}$$
$$nE(K) = \{nP \mid P \in E(K)\}$$

と定義する．Tate ペアリングとは

$$\langle \cdot, \cdot \rangle_n : E(K)[n] \times E(K)/nE(K) \to K^*/(K^*)^n$$

となる写像のことである．

Tate ペアリングには以下のような性質がある．
E は K_0 上の楕円曲線とし，n は K_0 の標数と互いに素であるとする．$K = K_0(\mu_n)$ とする．このとき，Tate ペアリングは以下を満たす．

1. 任意の $P, P_1, P_2 \in E(K)[n]$ と $Q, Q_1, Q_2 \in E(K)/nE(K)$ に対し，

   $$\langle P_1 + P_2, Q \rangle_n = \langle P_1, Q \rangle_n \langle P_2, Q \rangle_n$$

   かつ，

   $$\langle P, Q_1 + Q_2 \rangle_n = \langle P, Q_1 \rangle_n \langle P, Q_2 \rangle_n$$

である．

2. K を有限体とする．任意の $P \in E(K)[n]$ $(P \neq \mathcal{O})$ に対して，$\langle P, Q \rangle_n \neq 1$ となるような $Q \in E(K)/nE(K)$ が存在する．同様にして，任意の $Q \in E(K)/nE(K)$ $(Q \notin nE(K))$ に対して，$\langle P, Q \rangle_n \neq 1$ となるような $P \in E(K)[n]$ が存在する．

4.2 ペアリング暗号

ペアリング暗号とは, ペアリングを用いた暗号方式である．ペアリングの処理は, RSA 暗号の暗号化・復号処理に比べて, 安全性強度が同程度と仮定した上で数倍の計算コストを要すると言われている．そのため, ペアリング計算の高速化はペアリング暗号の普及のための課題の一つとなっている．

4.2.1 Miller algorithm

Miller algorithm は, ペアリングを計算するアルゴリズムとして広く用いられており, Weil ペアリングや Tate ペアリングを計算する際の最も重要なステップである．

Miller algorithm は, $G_1 \subset E(\mathbb{F}_q)$ の生成元 P に関する有理関数 F_p で構成されており, 同時に, 点 $Q \in G_2 \subset E(\mathbb{F}_q^2)$ に対する $F_P(Q)$ を評価する．

表 4.1　Algorithm 4:Miller algorithm

input: $r = (r_n, ..., r_0)_2, P \in G_1(\subset E(\mathbb{F}_q)), Q \in G_2(\subset E(\mathbb{F}_{q^k}))$ **output**: $F_P(Q) \in G_3(\subset \mathbb{F}_{q^k}^*))$
$1: T \leftarrow P$
$2: f_1 \leftarrow 1$
$3: f_2 \leftarrow 1$
$4:$ **for** $i = n-1$ **down to 0 do**
$4.1: T \leftarrow [2]T, where\ T = (X, Y, Z), [2]T = (X_2, Y_2, Z_2)$
$4.2: f_1 \leftarrow {f_1}^2 \times\ h_1(Q)$
$4.2.1:$ **if** $r_i = 1$ **then**
$4.2.2: T \leftarrow T + P$
$4.2.3: f_1 \leftarrow f_1 \times\ h_2(Q)$
$5:$ **end for**
$6:$ **return** f_1

4.2.2 ペアリング暗号の課題

ペアリングの処理は, RSA 暗号の暗号化・復号に比べると, 安全性強度が同程度と仮定した上で, 数倍の計算コストを要すると言われている．そのため, ペアリング暗号の普及のためにはペアリング計算の高速化が大きな課題となっている．

第 5 章

楕円曲線暗号に対する攻撃

5.1　攻撃の種類

楕円曲線暗号の安全性の根拠となっている離散対数問題を解く攻撃手法は主に 2 つに分類される [4].

1. 任意有限可換群に対する方法
   任意のパラメータを利用する楕円曲線暗号に適用可能である．主な攻撃方法として, Pohlig-Hellman 法 [5], BSGS 法 [6], Pollard's rho 法 [7] 等がある．これらの攻撃は, 計算量が鍵長に対して指数関数時間であるため, 鍵長を大きくとることで回避できる.
2. 特殊な曲線に対する方法
   楕円曲線の離散対数問題をより簡単な問題に変換して解く方法．全数探索型の攻撃に比べ適用できるパラメータに制約があるが, 効率が良くなる傾向があり, 計算量が準指数時間の攻撃手法もある．主な攻撃方法として, MOV 帰着法 [8], GHS 法 [9] 等がある．これらの攻撃手法はそれぞれ適用条件が決まっており, パラメータが適用条件を満たさないことを確認することで回避できる.

また, 一部の特殊な曲線を除いて楕円曲線暗号に対する準指数時間で攻撃可能な攻撃手法は未だに発見されていないため, 攻撃は数学的な弱点よりも, 暗号システムの実装部分を標的とした攻撃手法が提案されている.

5.2　サイドチャネル攻撃

サイドチャネル攻撃とは, 暗号装置が発する電磁波や熱などを外部から観測し, 暗号解読の手がかりを得ようとする手法である．今まで提案されてきた攻撃手法の一部を説明する.

5.2.1　タイミング攻撃

タイミング攻撃は, 暗号化の処理時間が秘密鍵のデータに依存して異なるとき, その差を解析して秘密場情報を推測する攻撃手法である.

タイミング攻撃を行う際に, 次を仮定する.

- 攻撃中, 秘密情報は不変である
- 攻撃者は実装されているアルゴリズムを既に知っている
- 攻撃者は暗号化 1 回における処理時間を正確に測定可能である

上の仮定を満たすとき, 攻撃は以下のように行われる.

1. 実装アルゴリズム中で処理時間に差が生じる部分のうち, 1 ヶ所に注目する.
2. 測定できるのは暗号化全体の時間だけなので, 注目した部分以外の 時間差が無視できるまで平文を入力し, 暗

号化の処理時間を測定する.

3. 測定データから秘密情報を推測する.

この攻撃の対策として, 次の 3 つが挙げられる.

- データに依らず, すべての演算が一定時間で終わるようにする
- 正確に時間計測を行えないようにする
- ブラインド署名のアルゴリズムの応用 (内部処理において, 攻撃者が入力したメッセージとは異なるメッセージを用いる方法)

5.2.2 電力解析攻撃

電力解析攻撃は, 暗号装置が処理する演算や内部情報と消費電力に相関があることを利用し, 秘密鍵を推測する攻撃手法である. 主に, SPA(Simple Power Analysis) 方式と, DPA(Differential Power Analysis) 方式の 2 つの方式に分類される.

SPA 攻撃は, 暗号処理一回分の消費電力波形を用いる. 例えば, 秘密鍵である k による条件分岐が存在するアルゴリズムで, 処理される命令が異なる場合, 秘密鍵を復元することが可能である. 対策法としては, 条件分岐を使用しない, または条件分岐後に行われる命令を同様にすることが挙げられる.

DPA 攻撃は複数の消費電力波形を用い統計的に解析を行う. この攻撃の対策として, 乱数により演算, 秘密鍵, 平文, 点 P をランダム化することが挙げられる.

5.2.3 故障利用攻撃

物理的操作で暗号装置を誤動作させ, 誤った出力と正しい出力を比較することで秘密情報を推測する攻撃である. フォールト解析攻撃とも呼ばれる. この攻撃に関しては次の章で詳しく述べる.

第 6 章

フォールト攻撃

フォールト攻撃は一般的に, 暗号装置が秘密鍵を用いた演算中に起こる誤りを利用する攻撃である. 誤りは攻撃者がハードウェア上の誤動作, またはソフトウェア上のバグを狙った物理的な干渉をすることにより挿入される. 攻撃者は, 誤りによって得られる不正な出力値をもとに秘密鍵の情報を盗みだす. このように意図的に導入される誤りをフォールトと呼ぶ. 以下にフォールト攻撃の既存手法を紹介する.

6.1 フォールト導入技術

IC チップへの攻撃を例に, 具体的なフォールトの導入について説明する [10].

IC チップへ物理的な刺激を与える手段として, レーザー照射がある. レーザー照射により, 正しくないパスワードで IC カード内のパスワードを照合を通過し, 秘密情報にアクセスする. この攻撃に対しては, 入力されたパスワードに対して照合を 2 回行うという対策を実施している. しかし, 攻撃の高度化も同時に進んでいるため, 常に最新の攻撃への対策を考える必要がある.

6.1.1 マルチレーザー攻撃

2 回のパスワード照合に対して, 2 度レーザー照射を行い対策を無効化する攻撃.

6.1.2 電磁波照射攻撃

動作中の IC チップに, コイルを近接させ, 電流を流し, 電磁波を照射することで IC チップを誤動作させる. この攻撃の特徴は, 非接触で攻撃可能な点である.

6.2 Invalid-Curve Attack

Invalid-Curve Attack は Biehl らによって提案された [11]. Invalid-Curve Attack の狙いは, 暗号的に強固で安全な楕円曲線 E を, より弱い曲線へと移すことである.

標数が 2 以上の有限体 K 上に定義された暗号的に強固な楕円曲線を E とする.

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6. \tag{6.1}$$

この攻撃の前提として, 攻撃対象となる暗号装置が入力点 P とその演算結果である点 $Q = kP$ の 2 点に対して, 楕円曲線 E 上に載るかどうかのチェックはしないものとする. また, ANSI X9.63 や IEEE 1363 で標準化される, 楕円曲線 E のパラメータ a_6 を使用しない点の加算公式を想定している.

暗号装置が入力として点 $P' = (x', y') \in K \times K$ かつ $P' \notin E(K)$ とする. このときスカラー倍算 $Q' = kP'$ の計

算は, 楕円曲線 $E'(a_1, a_2, a_3, a_4, {a_6}')$ 上で行われる.

$$ {a_6}' = {y'}^2 + a_1 x' y' + a_3 y' - a_2 {x'}^2 - a_4 x' \tag{6.2} $$

もし, P' が, $E'(K)$ 上で比較的小さい位数を持つなら, $E'(K)$ 上の部分群 $\langle P' \rangle$ で離散対数問題を解くことで, $k \pmod{ord(P')}$ の位数を見つけることができる. この部分群が比較的小さいサイズならば, BSGS 法 [6], または, Pollard's rho 法 や Pollard's lambda 法 [7] のような一般的に知られている離散対数問題の計算アルゴリズムが使えるようになる.

また, E' 上の点 P' を, 小さい約数を持つ合成数を位数に持つように選択できる場合, Pohlig-Hellman 法 [5] を用いて大きい離散対数問題から, 独立して解くことができる素数べき乗位数の部分群でのより小さい問題にすることができる.

この方法を, k の十分な剰余が集まり, k を復元するために中国の余剰定理が使えるまで繰り返し, 中国の余剰定理を適用することで秘密鍵 k を復元する. この攻撃は多項式時間で実行できる.

6.2.1 防御策

この攻撃を防ぐためには, 出力前に入力点 P とそのスカラー倍算 kP の両方が楕円曲線 E に載るかどうかを曲線パラメータの a_6 を用いてチェックを行う.

6.3 Skipping Attack

Skipping Attack は, Schmidt と Herbst によって提案された [12]. この攻撃は, 低価格の器具を使用しているという点から RSA 暗号に対する実用的なフォールト攻撃であるが, 楕円曲線暗号にも拡張できる.

RSA 暗号で用いられる Square and Multiply algorithm に対する攻撃を例として挙げる.

表 6.1 Algorithm 5:Square and Multiply algorithm

input: $x \in \mathbb{G}, d = (d_{t-1}, \ldots, d_0)_2$ **output**: $y = x^d$
1 : $R_0 \leftarrow 1$
2 : $R_1 \leftarrow x$
3 : **for** $i = t-1$ **down to 0 do**
3.1 : $R_0 \leftarrow {R_0}^2$
3.2.1 : **if** $d_i = 1$ **then**
3.2.2 : $R_0 \leftarrow R_0 \cdot R_1$
4 : **end for**
5 : **return** R_0

例えば, Algorithm5 の for ループの j 回目の処理をスキップするとすると, 出力結果 $\hat{y_j}$ が結果として得られる.

$$ \hat{y_j} = \prod_{i=j+1}^{t-1} x^{d_i 2^{i-1}} \cdot \prod_{i=0}^{j} x^{d_i 2^i} \tag{6.3} $$

そして, 以下の様にしてべき数 d_j をビット毎に取り出す.

$$ \hat{y_j} = \begin{cases} \hat{y}_{j-1} & (d_j = 0) \\ x^{2^{j-1}} \hat{y}_{j-1} & (d_j = 1) \end{cases} $$

6.3.1 防御策

Skipping Attack に耐性のある Square and Multiply algorithm を, Algorithm6 と Algorithm7 に記す.

表 6.2 Algorithm 6:Square and Multiply protected against skipping attacks(I)

input: $x \in \mathbb{G}, d = (d_{t-1}, \ldots, d_0)_2$ **output**: $y = x^d$
1 : $R_0 \leftarrow 1$
2 : $R_1 \leftarrow x$
3 : $T_0 \leftarrow 0$
4 : $T_1 \leftarrow 1$
5 : **for** $i = t-1$ **down to 0 do**
5.1 : $(R_0, T_0) \leftarrow ({R_0}^2, 2 \cdot T_0)$
5.2.1 : **if** $d_i = 1$ **then**
5.2.2 : $(R_0, T_0) \leftarrow (R_0 \cdot R_1, T_0 + T_1)$
6 : **end for**
7 : **if** $T_0 \neq d$ **then**
7.1 : **return error**
8 : **return** R_0

表 6.3 Algorithm 7:Square and Multiply protected against skipping attacks(II)

input: $x \in \mathbb{G}, d = (d_{t-1}, \ldots, d_0)_2$ **output**: $y = x^d$
1 : $R_0 \leftarrow 1$
2 : $R_1 \leftarrow x$
3 : $T_0 \leftarrow 0$
4 : $T_1 \leftarrow 1$
5 : **for** $i = t-1$ **down to 0 do**
5.1 : $(R_0, T_0) \leftarrow ({R_0}^2, 2 \cdot T_0 \pmod{\Omega})$
5.2.1 : **if** $d_i = 1$ **then**
5.2.2 : $(R_0, T_0) \leftarrow (R_0 \cdot R_1, T_0 + T_1 \pmod{\Omega})$
6 : **end for**
7 : **if** $T_0 \not\equiv d \pmod{\Omega}$ **then**
7.1 : **return error**
8 : **return** R_0

6.4 ペアリング暗号に対するフォールト攻撃

ペアリング暗号に対するフォールト攻撃の一つに, El Mrabet によって提案された手法がある [13]. この攻撃は, Miller algorithm の Miller ループの反復回数を標的としており, その回数を変更した際の出力を得ることで秘密点 P を見つける.

まず, リバースエンジニアリングによって反復回数のカウンターに属しているフリップフロップを見つける. そして, 例えばレーザー等を用いて機器に障害を引き起こし, Miller ループの反復回数を変更する. 反復回数 τ 番目の出力 $F_{\tau,P(Q)}$ と, $\tau+1$ 番目の出力 $F_{\tau+1,P(Q)}$ の商 $\frac{F_{\tau+1,P(Q)}}{F_{\tau,P(Q)}{}^2}$ を考える. そして, $\mathbb{F}_q^k$ の基底を特定することで, 秘密点 P を見つける方程式を導くことができる.

以下に反復回数 τ 番目の際の具体的な手順を説明する.

$B=\{1, \xi, \sqrt{v}, \xi\sqrt{v}\}$ を $\mathbb{F}_q^k$ の基底とする.

$r_{\tau+1}=0$ の場合, r の τ ビット目を読むことによって j が得られ, 以下の 3 つの式が得られる.

$$F_{\tau+1,P}(Q) = (F_{\tau,P}(Q))^2 \times h_1(Q) \tag{6.4}$$

$$[j]P = (X_j, Y_j, Z_j) \tag{6.5}$$

$$T = [2j]P = (X_2 j, Y_2 j, Z_2 j) \tag{6.6}$$

式 (6.4) の h_1 の方程式を用いて以下の方程式を得る.

$$F_{\tau+1,P}(Q) = (F_{\tau,P}(Q))^2 \times (Z_{2j} Z_j^2 y\sqrt{v} - 3(X_j - Z_j^2)(X_j + (Z_j^2)(xZ_j^2 - X_j)) \tag{6.7}$$

点 P を秘密点とすると, j, τ, Q の座標が分かる. さらに, Miller algorithm により $F_{\tau+1,P}(Q)$ と $F_{\tau,P}(Q)$ が与えられる.

比 $R = \frac{F_{\tau+1,P}(Q)}{(F_{\tau,P}(Q))^2}$ を計算する. R の理論上の形と, 基底 B の分解を用いると, 恒等式によって以下の集合が得られる.

$$\begin{cases} Y_j Z j^2 = \lambda_2, \\ Z_j^2(X_j^2 - Z_j^4) = \lambda_1, \\ 3X_j(X_j^2 - Z_j^4) + 2Y_j^2 = \lambda_0, \end{cases} \tag{6.8}$$

非線形方程式のシステムの分解により, 以下の式が与えられる.

$$(\lambda_0^2 - 9\,\lambda_1^2) Z^1 2 - (4\,\lambda_0 \lambda_2^2 + 9\,\lambda_1^3) Z^6 + 4\,\lambda_1^4 = 0 \tag{6.9}$$

Z_j の方程式を解くと, せいぜい $24 = 12 \times 2 \times 1$ で, 点 $[j]P$ の座標 (X_j, Y_j, Z_j) を見つけられる. $[j]P$ の値が分かると, $j \pmod r$ の逆数 j' を見つけることで, $[j'][j]P = [j'j]P = P$ を見つけることが可能となる.

6.4.1 防御策

この攻撃を防ぐための方法はいくつか存在する [14].

a) Miller ループの回数を変化させる攻撃の標的にならないよう, フォールトに耐性のあるカウンターを用いる.
b) アルゴリズムを, 正しい反復回数よりも大きいランダムな回数実行させる.
c) 計算中に中間値の結果をチェックする. すなわち, 点が楕円曲線上にあるかどうか最後の点と中間値を比較して確認する.
d) $R_1 = e(P, Q),\ R_2 = e(aP, bQ)$ を用いて $R_2 = {R_1}^{ab}$ かどうかをチェックし, 二重で計算する.

第 7 章

提案手法

7.1 提案手法

本研究では, ペアリング暗号の Miller algorithm に対する反復回数をフォールト導入対象とした攻撃手法を, 楕円曲線暗号の高速スカラー倍算アルゴリズムに適用する.

7.1.1 前提

定義体を素体 $K = \mathbb{F}_q$ とし, その体 K 上に定義された楕円曲線を $E : y^2 = x^3 + ax + b\ (a, b \in K)$ とする. 点の高速スカラー倍算 Double and add, Double and add always, Montgomery ladder アルゴリズムに対し, それぞれアルゴリズム中のループの反復回数を変更する.

7.1.2 反復回数の変更

上で挙げた高速スカラー倍算アルゴリズム中のループの回数は, 秘密鍵 k のビット数 n によって定まる. まず, リバースエンジニアリングによって反復回数のカウンターに属しているフリップフロップを見つける. そして, クロック周期をカウントすることでループが何回行われたか, つまり n の値が分かる. その際に, スカラー倍算の結果 kP と反復回数 n の値を記録しておく. そして, 例えばレーザー等を用いて障害を起こし, アルゴリズムの反復回数を変更する.

7.1.3 秘密鍵 k の復元

反復回数を変更して得られた出力を集め, 秘密鍵 k を復元する.

1 回ずつ減らす方法

反復回数を 1 回減らすと, k の値は先頭のビットが抜かされ $k = (k_{n-1}, k_{n-2}, \ldots, k_0)_2$ から $k' = (0, k_{n-2}, \ldots, k_0)_2$ へと変化する. もし kP と $k'P$ が同じ点ならば, k_{n-1} は 0 であることが分かり, 異なる点ならば, k_{n-1} は 1 であることが分かる. これを繰り返し, k を 1 ビットずつ復元していく.

2 回ずつ減らす方法

フォールトの挿入回数はより少ない方が好ましいため, 反復回数を 2 回ずつ減らし, 1 回ずつ減らす方法に比べてフォールト挿入回数を減らす. 2 回減らすと, k の値は $k = (k_{n-1}, k_{n-2}, k_{n-3}, \ldots, k_0)_2$ から, $k' = (0, 0, k_{n-3}, \ldots, k_0)_2$ へと変化する. もし kP と $k'P$ が同じ点ならば, $k_{n-1} = 0$ かつ $k_{n-2} = 0$ であることが分かる. もし異なる点ならば, (k_{n-1}, k_{n-2}) は, $(0, 1), (1, 0), (1, 1)$ のいずれかである.

具体例

例えば, 秘密鍵が $k = 25 = (k_4, k_3, k_2, k_1, k_0)_2 = (1, 1, 0, 0, 1)_2$ であるとし, kP を計算するときのことを考える.

$k = 25 = (1, 1, 0, 0, 1)_2$ のとき, $n - 1 = 4$ である. 初めに, $kP = 25P$ を求める. 次に, $n - 1 = 3$ として計算すると, $k'P = (1, 0, 0, 1)_2 P = 9P$ が得られる. 同様に $n - 1 = 2$ とすると $k''P = (0, 0, 1)_2 P = P$, $n - 1 = 1$ とすると $k'''P = (0, 1)_2 P = P$, $n - 1 = 0$ とすると $k''''P = (1)_2 P = P$ がそれぞれ得られる.

このとき, $k''''P = \mathcal{O}$ ならば $k_0 = 0$, $k''''P = P$ ならば $k_0 = 1$ であることが分かる. そして, $k'''P = k''''P$ ならば $k_1 = 0$, $k'''P \neq k''''P$ ならば $k_1 = 1$ であることが分かる. 今回, $k''''P = P$ なので $k_0 = 1$, $k'''P = k''''P$ なので $k_1 = 0$ であることが分かる. 同様に, $k''P = k'''P$ より $k_2 = 0$, $k'P \neq k''P$ より $k_3 = 1$, $kP \neq k'P$ より $k_4 = 1$ であることが分かる. よって, $k = (k_4, k_3, k_2, k_1, k_0)_2 = (1, 1, 0, 0, 1)_2$ と k を復元することができる.

表 7.1　Algorithm 8:スカラー倍算アルゴリズムの反復回数に対する攻撃

input: $k = (k_{n-1}, ..., k_0)_2, P \in E(K)$ **output**: スカラー数 $find\_k$
1 : $T = n - 1$, $R[T] \leftarrow MontgomeryLadder(k, P, n - 1)$ //反復回数を 1 ずつ減らしたスカラー倍算の結果を保存する 2 : **while** $T \neq 0$ **do** 2.1 : $T = T - 1$ 2.2 : $R[T] \leftarrow MontgomaryLadder(k, P, T)$ 3 : **end while** 4 : $j = 0$ 5 : **if** $R[0] = \mathcal{O}$ **then** 5.1 : $find\_k = (0)_2$ 5.2 : **while** $R[j + 1] = \mathcal{O}$ **do** 5.2.1 : $find\_k$ の $j + 1$ 番目に, 0 を挿入. 5.2.2 : $j = j + 1$ 5.3 : **end while** 6 : **else** 6.1 : $find\_k = (1)_2$ 6.2 : **while** $R[j] = R[j + 1]$ 6.2.1 : $find\_k$ の $j + 2$ 番目に, 0 を挿入. 6.2.2 : $j = j + 1$ 6.3 : **end while** 7 : **end if** 8 : **for** $i = n - 1$ **to** j **do** 8.1 : **if** $R[i] = R[i + 1]$ **then** 8.1.1 : $find\_k$ の $j + 2$ 番目に, 0 を挿入. 8.2 : **else** 8.2.1 : $find\_k$ の $j + 2$ 番目に, 1 を挿入. 8.3 : **end if** 9 : **end for** 10 : **return** $find\_k$

7.2　防御策

この攻撃手法に対する防御策は, ペアリング暗号に対するフォールト攻撃と同様に,

- フォールトに耐性のあるカウンターを用いる.
- 反復をランダムな回数実行する.
- 中間値の点が楕円曲線上にあるかチェックする.

などが挙げられる.

第 8 章

実験

提案手法の攻撃アルゴリズムの数値実験を行う. 実験プログラムには Magma を用いる.

8.1 実験条件

以下の条件において, 試行回数に対して攻撃成功となる確率を測定する.

- Double and add, Double and add always, Montgomery ladder アルゴリズムのループ回数 n を変更する.
- 復元した k が, 秘密鍵 k と一致した際に攻撃成功とする.
- 試行回数 m は 100 回とする.
- 使用するパラメータは, NIST(米国国立標準技術研究所) が推奨する楕円曲線のシステムパラメータ P-192, P-256[15] を用いる.

8.1.1 NIST 曲線 P-192, P-256

NIST 推奨の楕円曲線 P-192, P-256 は Weierstrass 型の楕円曲線であり, パラメータは次の通りである [15].

P-192 Curve :

$y^2 \equiv x^3 - 3x + b \pmod{q}$

$b = 2455155546008943817740293915197451784769108058161191238065$

$q = 6277101735386680763835789423176059013767194773182842284081$

P-256 Curve :

$y^2 \equiv x^3 - 3x + b \pmod{q}$

$b = 41058363725152142129326129780047268409114441015993725554835256314039467401291$

$q = 115792089210356248762697446949407573529996955224135760342422259061068512044369$

8.2 実験結果

8.2.1 結果の考察

表 8.1　NIST P-192 Curve(フォールト挿入回数 (n-1) 回)

アルゴリズム	成功率 (%)	実行時間 (秒)
Double and add	100	27.190
Double and add always	100	41.435
Montgomery Ladder	100	36.633

表 8.2　NIST P-256 Curve(フォールト挿入回数 (n-1) 回)

アルゴリズム	成功率 (%)	実行時間 (秒)
Double and add	100	72.038
Double and add always	100	104.035
Montgomery Ladder	100	93.511

表 8.3　NIST P-192 Curve(フォールト挿入回数 (n-1)/2 回)

アルゴリズム	成功率 (%)	実行時間 (秒)	n 回との実行時間の差
Double and add	100	31.824	+4.634
Double and add always	100	55.395	+13.96
Montgomery Ladder	100	41.636	+5.003

表 8.4　NIST P-256 Curve(フォールト挿入回数 (n-1)/2 回)

アルゴリズム	成功率 (%)	実行時間 (秒)	n 回との実行時間の差
Double and add	100	82.432	+10.394
Double and add always	100	142.225	+38.19
Montgomery Ladder	100	106.566	+13.055

第 9 章

結論

9.1 結論

本研究では, ペアリング暗号の Miller algorithm に対する反復回数をフォールト導入対象とした攻撃手法を楕円曲線暗号のスカラー倍算へ適用する, という新しい攻撃手法を提案した. また, プログラミング言語 Magma 上で, NIST 標準パラメータ [15] を用いた楕円曲線暗号に対する攻撃の実験を行い, 提案手法の有効性を実際に確かめた. これは, 楕円曲線暗号の実装における, フォールト攻撃への対策の必要性を強調するものである.

9.2 今後の課題

今後の課題としては以下のものがあげられる.

- 本手法と他の手法との組み合わせる (途中から他の離散対数問題の解法を用いるなど).
- 反復回数がランダムである場合の対抗策の付加.
- デバイスを用いた実験.

謝辞

本研究を進めるにあたり，適切な御指導, 御助言, 御検討を頂いた中央大学 理工学部 趙 晋輝 教授に，深く感謝いたします．また, 日ごろの学生生活においてお世話になった中央大学趙研究室の皆様に深く感謝いたします．

参考文献

[1] Francois Morain and Jorge Olivos. *Speeding Up The Computations On An Elliptic Curve Using Addition-Subtraction Chains*, RAIRO Ther. Inform. Appl. 24, pp. 531-543, 1990.

[2] Jean-Sebastien Coron. *Resistance Against Differential Power Analysis For Elliptic Curve Cryptosystems*, Cryptographic Hardware and Embedded Systems, pp. 292-302, Springer Berlin, 1999.

[3] Peter L. Montgomery. *Speeding the Pollard and Elliptic Curve Methods of Factorization*, Mathematics of Computation, Volume 48, pp. 243-264, 1987.

[4] 清藤武暢, 四方順司 : **公開鍵暗号を巡る新しい動き : RSA から楕円曲線暗号へ**, 日本銀行金融研究所/金融研究/2013.7.

[5] Pohlig, Stephen, and Margin Hellman. *An Improved Algorithm for Computing Logarithms over GF(p) and Its cryptographic Significance (Correspondance)*, IEEE Transactions of Information Theory, vol.24 no.1, pp. 106-110, 1978.

[6] Shanks, Daniel. *Class Number, a Theory of Factorization, and Genera,*" *Proceeding of Symposia in Pure Mathematics*, vol.20, pp. 415-440, 1971.

[7] J.M. Pollard. *Monte Carlo methods for index computation (mod p)*, Mathematics of Computation, vol.32 no.143, pp. 918-924, 1978.

[8] Menezes, Alfred, Scott Vanstone, and Tatsuaki Okamoto. *Reducing elliptic curve logarithms to logarithms in a finite field*, Proceedings of Symposium on Theory of Computing (STOC), pp.80-89, 1991.

[9] A.Enge and P.Gaudry. *A general framework for subexponential discrete logarithm algorithms*, Acta Arith, vol.102, pp. 83-103, 2002.

[10] 中田 量子 : **ハードウェア脆弱性評価技術の最新動向**, 情報セキュリティ EXPO/独立行政法人情報処理推進機構技術本部セキュリティセンター情報セキュリティ認証室/2013.5.

[11] I. Biehl, B. Meyer, and V. Müller. *Differential Fault Attacks on Elliptic Curve Cryptosystems*, Springer-Verlag, CRYPTO, pp.131-146, LNCS, 2000.

[12] Jörn-marc Schmidt and Christoph Herbst. *A Practical Fault Attack on Square and Multiply*, Fault Diagnosis and Tolerance in Cryptography, 2008. FDTC '08. 5th Workshop on, pp. 53-58, IEEE, 2008.

[13] Nadia El Mrabet. *What about vulnerability to a fault attack of the Miller algorithm during an Identity Based Protocol ?*, Advances in Information Security and Assurance, pp. 122-134, Springer Berlin Heidelberg, 2009.

[14] Nadia El Mrabet, Jacques J. A. Fournier, Louis Goubin. *What about vulnerability to a fault attack of the Miller algorithm during an Identity Based Protocol ?*, Cryptography and Communications Volume 7, Issue 1 , pp. 185-205, Springer US, 2015.

[15] Federal Information Pro cessing Standards Publication FIPS 186-2. Digital Signature Standard (DSS), appendix 6: *Recommended Elliptic Curves for Federal Government Use*, Technical report, NIST, January 27, 2000.

[16] : **平成 11 年度 スマートカードの安全性に関する調査 調査報告書** 2000.

[17] Neal Koblitz. *A Course in Number Theory and Cryptography*, Springer-Verlag, 1987.

[18] Christophe Doche, Tanja Lange : *Arithmetic of Elliptic Curves*, Handbook of elliptic and hyperelliptic curve cryptography, pp. 267-302, Chapman & Hall/CRC, 2006.
[19] Marc Joye and Michael Tunstall. *Fault Analysis in Cryptography*, Springer-Verlag Berlin Heidelberg, 2012.
[20] 新妻弘, 木村哲三 : **群・環・体　入門**, 共立出版, 1999.