E P O C-2 評 価 報 告 書

2001 年 12 月

EPOC-2 評価報告書

2001 年 12 月

1　はじめに

本報告書では、本年度（平成 13 年度) において、CRYPTREC に応募された公開鍵暗号方式 EPOC-2 [3][4] についての安全性評価について報告する。

EPOC-2 は、守秘を目的とする暗号化方式 (Encryption Scheme) として提案されている。現在、暗号のコミュニティにおいて、公開鍵暗号化方式に対する最強の安全性として知られている概念は、適応的選択暗号文攻撃 (CCA2: Adaptive Chosen-Ciphertext Attack) に対する Non-Malleability であり [10][11][8]、これはまた適応的選択暗号文攻撃 (CCA2) に対する識別不可能性 (Indistinguishability) と等価であることが示されている [8]。以下、適応的選択暗号文攻撃に対する識別不可能性を IND-CCA2 と略記する。

現在、暗号提案に際しての、1 つの代表的なアプローチとしては、その暗号の安全性がある仮定のもとで証明されることが挙げられる。本年度（平成 13 年度) 応募された EPOC-2 もそのような形で提案されている [4]。より正確に言えば、EPOC-2 は $n = p^2q$ 型の素因数分解問題の困難性を仮定すれば、ランダムオラクルモデル上 [6] で IND-CCA2 をみたすことが主張されている [4]。

本報告書では、EPOC-2 の安全性の評価として、その主張が妥当であるかの検証を行うことを目的とする。

2　CRYPTREC 応募暗号 EPOC-2

本節では、本年度（平成 13 年度) において、CRYPTREC に応募された公開鍵暗号方式である EPOC-2 [3][4] について記述する。公開鍵暗号方式 EPOC は昨年度 (平成 12 年度) においても CRYPTREC に応募されている [1][2]。昨年度応募された暗号 EPOC は 3 つのバージョンをもち、これらは昨年度応募時点では、EPOC-1, EPOC-2, EPOC-3 という名称であった。ただし、ここでいう EPOC-2 は本年度応募されている EPOC-2 とは同一のものではなく、若干の差異が見受けられることに注意する。

2.1 プリミティブ (Cryptographic Primitives)

ここでは、EPOC-2 仕様書 [3] に記載されている暗号プリミティブに関しての記述を行う。

定義 1 (EPOC-2 in [3])

- **鍵生成** (KGP-OU):
  入力: k (セキュリティパラメータ)
  出力: 公開鍵 $PK = (n, g, h, pLen)$, 秘密鍵 $SK = (p, q, pLen, w)$
  1) 異なる 2 つの素数 p, q $(2^{k-1} \leq p, q < 2^k$, i.e. $|p| = |q| = k)$ を選び、$n := p^2 q$ を計算する。
  2) $g_p := g^{p-1} \bmod p^2$ の位数が p となるような $g \in (\boldsymbol{Z}/n\boldsymbol{Z})^*$ をランダムに選ぶ。
  3) $h_0 \in (\boldsymbol{Z}/n\boldsymbol{Z})^*$ をランダムに選び、$h := h_0^n \bmod n$ とする。
  4) $pLen := k$ とする。
  5) $w := L(g_p)$ とする。
  6) $PK = (n, g, h, pLen)$, $SK = (p, q, pLen, w)$ を出力する。

- **暗号化** (EP-OU(PK, σ, r)):
  入力: 公開鍵 $PK = (n, g, h, pLen)$,
  $\sigma \in \boldsymbol{Z}$ $(0 \leq \sigma < 2^{pLen-1}$, i.e. $\sigma \in \{0,1\}^{k-1})$
  乱数 $r \in \boldsymbol{Z}$ $(0 \leq r < n)$
  出力: 暗号文 $c \in \boldsymbol{Z}$ $(0 \leq c < n)$、あるいはエラーメッセージ "invalid"
  1) $0 \leq \sigma < 2^{pLen-1}$ をみたさない場合は、"invalid" を出力して処理を終了する。
  2) $c := g^\sigma h^r \bmod n$ を計算する。
  3) c を出力する。

- **復号** (DP-OU(SK, c)):
  入力: 秘密鍵 $SK = (p, q, pLen, w)$, 暗号文 $c \in \boldsymbol{Z}$
  出力: $\sigma \in \boldsymbol{Z}$ $(0 \leq \sigma < 2^{pLen-1}$, i.e. $\sigma \in \{0,1\}^{k-1})$ あるいは、エラーメッセージ "invalid"
  1) 受け取った暗号文 c が $0 \leq c < n$ をみたさない場合は、"invalid" を出力して処理を終了する。
  2) $c_p := c^{p-1} \bmod p^2$ を計算する。
  3) $\sigma := \frac{L(c_p)}{w} \bmod p$ を計算する。
  4) $0 \leq m < 2^{pLen-1}$ の場合、σ を平文として出力する。それ以外の場合は、"invalid" を出力して処理を終了する。

2.2 スキーム (Encryption Schemes) - EPOC-2 -

ここでは、EPOC-2 仕様書 [3] に記載されているスキームに関しての記述を行う。ただし、ここで用いている記号は、仕様書における記号とは若干異なることに注意されたい。

定義 2 (EPOC-2 in [3])

- **暗号化** (ES-EPOC-2-ENCRYPT(PK, m, P)):
  入力: 公開鍵 $PK = (n, g, h, pLen)$,
  平文 $m \in \boldsymbol{Z}$ $(0 \leq m < 2^{pLen-1})$
  乱数 $r \in \boldsymbol{Z}$ $(0 \leq r < n)$
  エンコーディングパラメータ P
  出力: 暗号文 (c_1, c_2)、あるいはエラーメッセージ "invalid"
  1) (σ, r, c_2) =EME3-ENCODE$(m, pLen, P)$ を計算する。(別紙の図 1 を参照。)
  2) c_1 =EP-OU(PK, σ, r) $(= g^{\sigma} h^{r}$ if $\sigma \in \{0,1\}^{pLen-1})$ を計算する。
  3) 暗号文 (c_1, c_2) を出力する。

- **復号** (ES-EPOC-2-DECRYPT$(PK, SK, (c_1, c_2), P)$):
  入力: 公開鍵 $PK = (n, g, h, pLen)$,
  秘密鍵 $SK = (p, q, pLen, w)$
  暗号文 (c_1, c_2)
  エンコーディングパラメータ P
  出力: 平文 $m \in \boldsymbol{Z}$ $(0 \leq m < 2^{pLen-1})$、あるいはエラーメッセージ "invalid"
  1) σ =DP-OU(PK, SK, c_1) を計算する。
  2) (m, r') =EME3-DECODE$(c_2, \sigma, pLen, P)$ を計算する。(別紙の図 2 を参照。)
  3) $PK' = (q, g \bmod q, h \bmod q, pLen)$ とする。
  3) $c_1 \bmod q$ =EP-OU$(PK', \sigma, r' \bmod (q-1))$ が成り立つ場合には、m を平文として出力する。それ以外の場合には、"invalid" を出力して処理を終了する。

3 安全性評価

3.1 EPOC-2 の安全性評価

本節では、理論的立場から、EPOC-2 に関しての安全性評価を行う。具体的に言えば、自己評価書において、EPOC-2 が選択暗号文攻撃に対してランダムオラクルモデル上で素因数分解の困難性を仮定すれば安全であることが示されているが、その主張の妥当性を本節では検証する。

定理 1 ([4]) ある攻撃者 A^{epoc} が存在して、EPOC-2 暗号化方式 Π^{epoc} を選択暗号文攻撃のシナリオで、ランダムオラクルモデル内で $(t, q_G + q_H, q_D, \epsilon)$-ブレーク出来たと仮定する。このとき、以下の時間 t_B と確率 ϵ_B で $n = p^2 q$ 型の素因数分解問題を解く、一様なアルゴリズム B が存在する。

$$t_B = t(k) + (q_G + q_H)T_{gcd,n} + q_D q_H(T_{\mathcal{E},q} + T_{gcd,n})$$
$$e_B = \frac{\epsilon(k)}{2}(1 - 2^{-k+1})(1 - 2^{-\rho_h})^{q_D}$$

ここで、

- ρ_h については、[4] を参照。
- $k := pLen$
- q_G, q_H はそれぞれ A^{epoc} がランダムオラクル G 及び H をそれぞれ呼び出す回数。
- q_D は A^{epoc} が復号オラクルを呼び出す回数。
- $T_{gcd,n}$ はサイズ $|n|$ の 2 つの整数の最大公約数を計算する計算時間。
- $T_{\mathcal{E},q}$ は、次の等式が成立するか否かを検査する計算時間：

$$c_1 = g^{\sigma} h^H \bmod q, \qquad (\sigma \in \{0,1\}^{k-1},\ H \in \{0,1\}^{2k+const})。$$

以下では、自己評価書 [4] における定理 1 の証明に関しての検証を行った結果を記述する。

(1) 自己評価書 [4] において ρ_h は次の定義により与えられている。

定義 3 (定義 B.4 in [4]) ρ_h は、以下をみたす最小の整数を表すこととする。

$$(\frac{1}{2})^{\rho_h} \leq \frac{1}{\#\langle h \rangle} + (\frac{1}{2})^{2k+const}.$$

しかしながら、評価者は、定理 1 における ρ_h の役割を考えると次のような形で ρ_h を定義する方が妥当であると考える

定義 4 ρ_h は、以下により定義される。

- $\#\langle h \rangle = 1$ のとき、$\rho_h := 0$
- $\#\langle h \rangle \geq 2$ のとき、ρ_h は次式をみたす最大の正整数とする。

$$\frac{1}{\#\langle h \rangle} + (\frac{1}{2})^{2k+const} \leq (\frac{1}{2})^{\rho_h}.$$

実際、ρ_h は、以下の補題の証明の中で、確率 $\Pr[\text{Fail} \mid 0]$ の上限を与える目的：

$$\Pr[\text{Fail} \mid 0] \leq 2^{-\rho_h}$$

のために使用されている。しかし、自己評価書にある定義 3 ではこの式は必ずしも従わないと思われる。この式を導くのであれば、上記の定義 4 を用いて

- $\#\langle h \rangle = 1$ のとき、

$$\Pr[\text{Fail} \mid 0] \leq \max_{c,x}\{p_{c,x}\} = (\frac{1}{2})^0,$$

- $\#\langle h \rangle \geq 2$ のとき,

$$\Pr[\text{Fail} \mid 0] \leq \max_{c,x}\{p_{c,x}\} \leq \frac{1}{\#\langle h \rangle} + (\frac{1}{2})^{2k+const} \leq (\frac{1}{2})^{\rho_h}$$

とすべきであると考える。

補題 1 (補題 F.3 in [4]) 事象 AskG∨H∨D がおきていないとき、B は、復号オラクルの動作を少なくとも確率 $1 - 2^{-\rho_h}$ で模擬できる。

(2) EPOC-2 における公開鍵として、g, h $(\in(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ が選ばれるが、このとき正当な暗号文は $\langle g \rangle \times \langle h \rangle$ $(\subset(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ の元であり、正当な暗号文全体は実際はこの部分群においての確率分布をもつ。ここで、自己評価書 [4] での EPOC-2 の安全性の証明では g^z $(z \in [0, n-1])$ により暗号文を模擬（シミュレート）しようとしているが、この方法で暗号文を正確に模擬できているのかをより厳密に示す必要があると考える。その再考察の結果として、評価者は次のいずれかが生ずる可能性が高いと推測する。

- g^z で暗号文の模擬を正確に行うには、g, h の選び方にもう少し条件が必要。
- g^z 以外の形で正当な暗号文を模擬する
- g, h 及び g^z は現状のままであるとするが、模擬に失敗する確率があり、これを考慮すると定理 1 の主張する reduction の効率性を修正する必要がある。

(3) $h = h_0^n$ $(\in(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ の位数に関しての記述は仕様書には特に見当たらないが、評価者はこれに関しての記述が必要であると考える。実際例えば、$h = 1 \bmod n$ となっているとき、$\rho_h = 0$ となり、$\epsilon_B = 0$ となってしまうため、自己評価書において安全性を示す定理 1 の意図することがあまり意味があるようには思えない。

3.2 ランダムオラクルモデルについて

EPOC-2 の安全性は、理論的には真にランダムな関数を仮定して（つまり、ランダムオラクルモデルのもとで）証明されている。しかし、方式の実現に際しては、実際にはそのランダム関数の部分を実用的なハッシュ関数（例えば、SHA-1）で置き換えて構成されている。したがって、このように実現された方式に対しては、厳密には理論どおりの安全性がそのまま保証される訳ではない。しかしながら、このようなアプローチで安全性に対して一つの指標を与えることは現在暗号のコミュニティにおいて標準的な方法とされており、現在までこのアプローチに対する大きな問題は特に報告されていない。ただし、これに関連した結果として、ランダムオラクルモデルのもとでは安全であるが、それを実際的な関数で実現した場合、安全でなくなるような特別な例が示されている [9]。彼らの結果は、ランダムオラクルモデルとその実際上の実現方式において安全性のギャップが生じる例が現実にあるという意味では大変興味深い結果である。しかしながら、応募暗号 EPOC-2 のようなランダムオラクルモデルのもとで安全性が証明された方式を、実用的なハッシュ関数で置き換えた場合に、直ちに安全でないという結果を導くものではない。実際、RSA-OAEP [7][15][13] をはじめとするような、ランダムオラクルモデルのもとで安全性が証明され、広く議論されている暗号方式に対して実用的なハッシュ関数で置き換えた場合、安全性を損なうことが証明された方式は未だ報告されていない。

3.3 共通鍵暗号としてブロック暗号を用いる場合の注意点

EPOC-2 で用いられる共通鍵暗号は受動的攻撃のもとで安全（平文を識別することが困難）である必要があると思われるが、共通鍵暗号としてブロック暗号を用いた場合、利用方法によってはこの条件が満たされない場合がある。例えば、ブロック暗号を ECB モードで（2 ブロック以上）暗号化する場合や、CBC モードで（2^{32} ブロック以上）暗号化する場合などがそうである。前者の場合、攻撃者は探索ステージで $m_0 = (a_0||a_1)$、$m_0 = (a_0||a_0)$ を選択することにより推測ステージにおいてアドバンテージ 1 で $b \in \{0, 1\}$ を推測することができる。ここで、$a_0 \neq a_1$、a_0、a_1 をブロック暗号の平文 1 ブロック分とする。後者の場合、探索ステージで $m_0 = (a_0||a_0||\cdots||a_0)$ となるような長さ 2^{32} ブロック以上の平文と同じ長さのランダムに選択した m_1 を選択する。ブロック数が 2^{32} 以上であるため暗号文一致攻撃が 1/2 以上の確率で成功し、攻撃者は $a_i \oplus a_j$ を知ることができる。m_0 を暗号化した場合必ず $a_i \oplus a_j = a_0 \oplus a_0$ となるが、m_0 を暗号化した場合ランダムな値となるため平文を識別することができる。この場合のアドバンテージは約 1/2 である。

EPOC-2 の共通鍵暗号としてブロック暗号を用いる場合、（それがどのような使われ方をしても）十分な安全性が保証されるとの誤解を利用者に受けないように

注意する必要があるであろう。

4 まとめ

以上に述べた EPOC-2 に対する安全性の考察により、最終的には以下のような安全性に関しての結論を得る。

1. EPOC-2 における公開鍵として、g, h $(\in(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ が選ばれるが、このとき正当な暗号文は $\langle g\rangle \times \langle h\rangle$ $(\subset(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ の元であり、正当な暗号文全体は実際はこの部分群においての確率分布をもつ。ここで、自己評価書 [4] での EPOC-2 の安全性の証明では g^z $(z \in [0, n-1])$ により暗号文を模擬（シミュレート）しようとしているが、評価者は、この方法で暗号文を正確に模擬できているのかをより厳密に示す必要があると考える。
2. $h = h_0^n$ $(\in(\boldsymbol{Z}/n\boldsymbol{Z})^*)$ の位数に関しての記述は仕様書には特に見当たらないが、評価者はこれに関しての記述が必要であると考える。
3. EPOC-2 の安全性の証明は提案者による自己評価書 [4] において、用いる共通鍵暗号が OTP（one-time pad）の場合にのみ与えられている。しかしながら、提案者による仕様書 [3] において推奨されているのは、OTP(one-time pad) 及び Camellia [5] である。したがって、後者のようなブロック暗号を用いる場合、EPOC-2 が安全であるためにブロック暗号 (Camellia) に要求される条件を明確にし、また利用法（mode of operation）のレベルでも要求される条件を明確にする必要があると思われる。その際、OTP におけるような効率的な reduction を用いた安全性の証明があれば、より望ましいと考える。

以上をまとめると、本質的に上記いずれの項目にしても EPOC-2 の安全性を否定するものではなく、第三者が仕様書に基づいて安全に実装できるよう、EPOC-2 の仕様に関してのより厳密な記述を要求するものである。

参考文献

[1] “EPOC 仕様書” submitted to CRYPTREC, 2000.

[2] “EPOC 自己評価書” submitted to CRYPTREC, 2000.

[3] “EPOC-2 仕様書” submitted to CRYPTREC, 2001.

[4] “EPOC-2 自己評価書” submitted to CRYPTREC, 2001.

[5] “Camellia”, submitted to CRYPTREC, 2001.

[6] M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols", Proc. of the First ACM Conference on Computer and Communications Security, 62-73, 1994.

[7] M. Bellare and P. Rogaway, "Optimal Asymmetric Encryption", Advances in Cryptology - Eurocrypt'94, Lecture Notes in Computer Science 950, Springer-Verlag, 92-111, 1994.

[8] M. Bellare, A. Desai, D. Pointcheval and P. Rogaway, "Relations Among Notions of Security for Public-Key Encryption Scheme", Advances in Cryptology - Crypto'98, Lecture Notes in Computer Science 1462, Springer-Verlag, 26-45, 1998.

[9] R. Canetti, O. Goldreigh and S. Halevi, "The Random Oracle Methodology, Revisited" (preliminary version), Proc. of STOC, ACM Press, 209-218, 1998.

[10] D. Dolev, C. Dwork and M. Naor, "Non-malleable cryptography", In 23rd Annual ACM Symposium on Theory of Computing, 542-552, 1991.

[11] D. Dolev, C. Dwork and M. Naor, "Non-malleable cryptography", SIAM J. Comput., 30 (2), 391-437, 2000.

[12] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes", Advances in Cryptology - Crypto '99, Lecture Notes in Computer Science 1666, 537-554, Springer-Verlag, 1999.

[13] E. Fujisaki, T. Okamoto, D. Pointcheval and J. Stern, "RSA-OAEP is secure under the RSA Assumption", Advances in Cryptology - Crypto 2001, Lecture Notes in Computer Science 2139, 260-274, 2001.

[14] T. Okamoto and S. Uchiyama, "A new public-key cryptosystem as secure as factoring", Advances in Cryptology - Eurocrypt '98, Lecture Notes in Computer Science 1403, 308-318, 1998.

[15] V. Shoup, "OAEP Reconsidered", Advances in Cryptology - Crypto 2001, Lecture Notes in Computer Science 2139, 239-259, 2001.

EME3-ENCODE

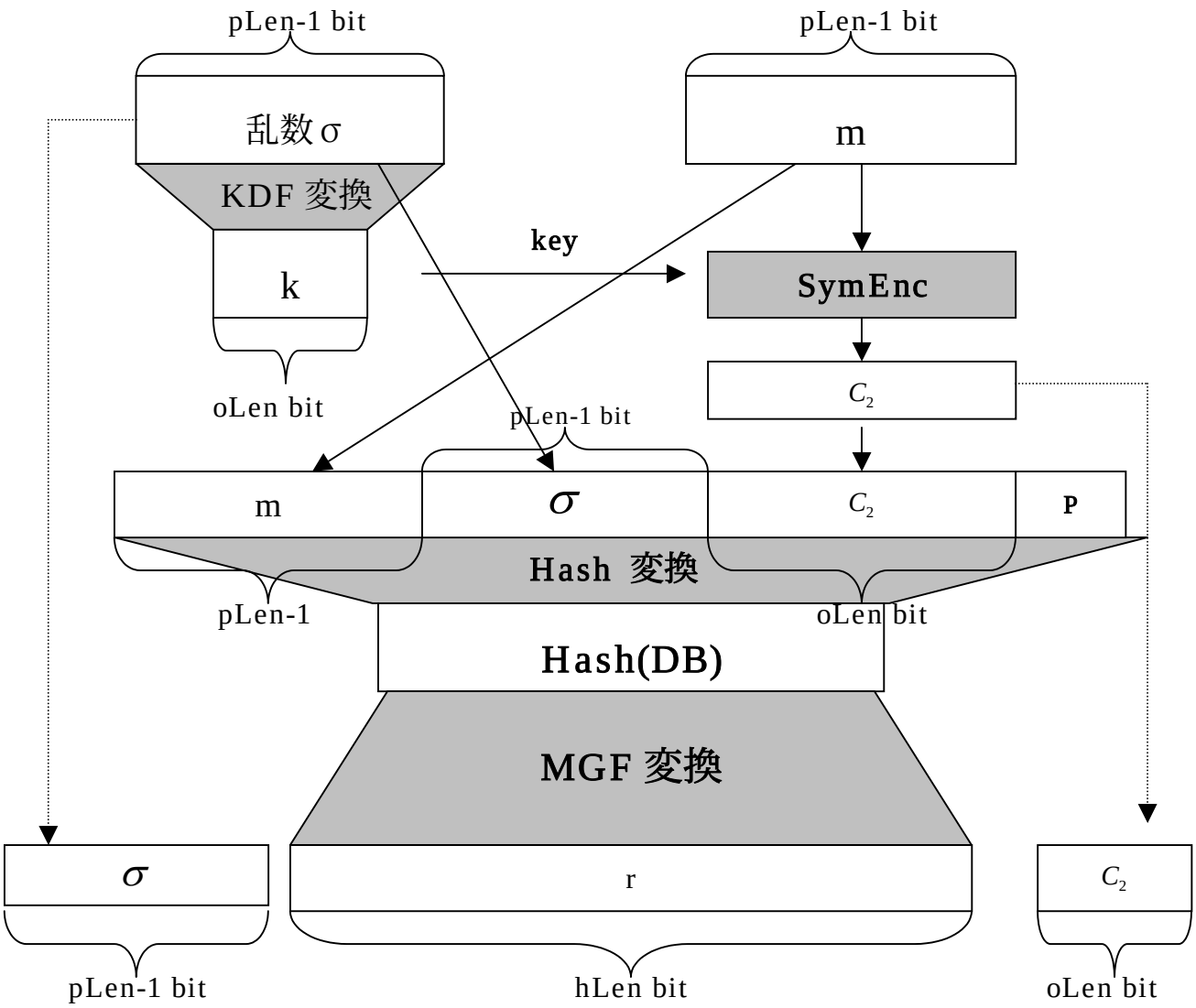

図 1: EME-3-ENCODE(m, pLen, P)

EME3-DECODE

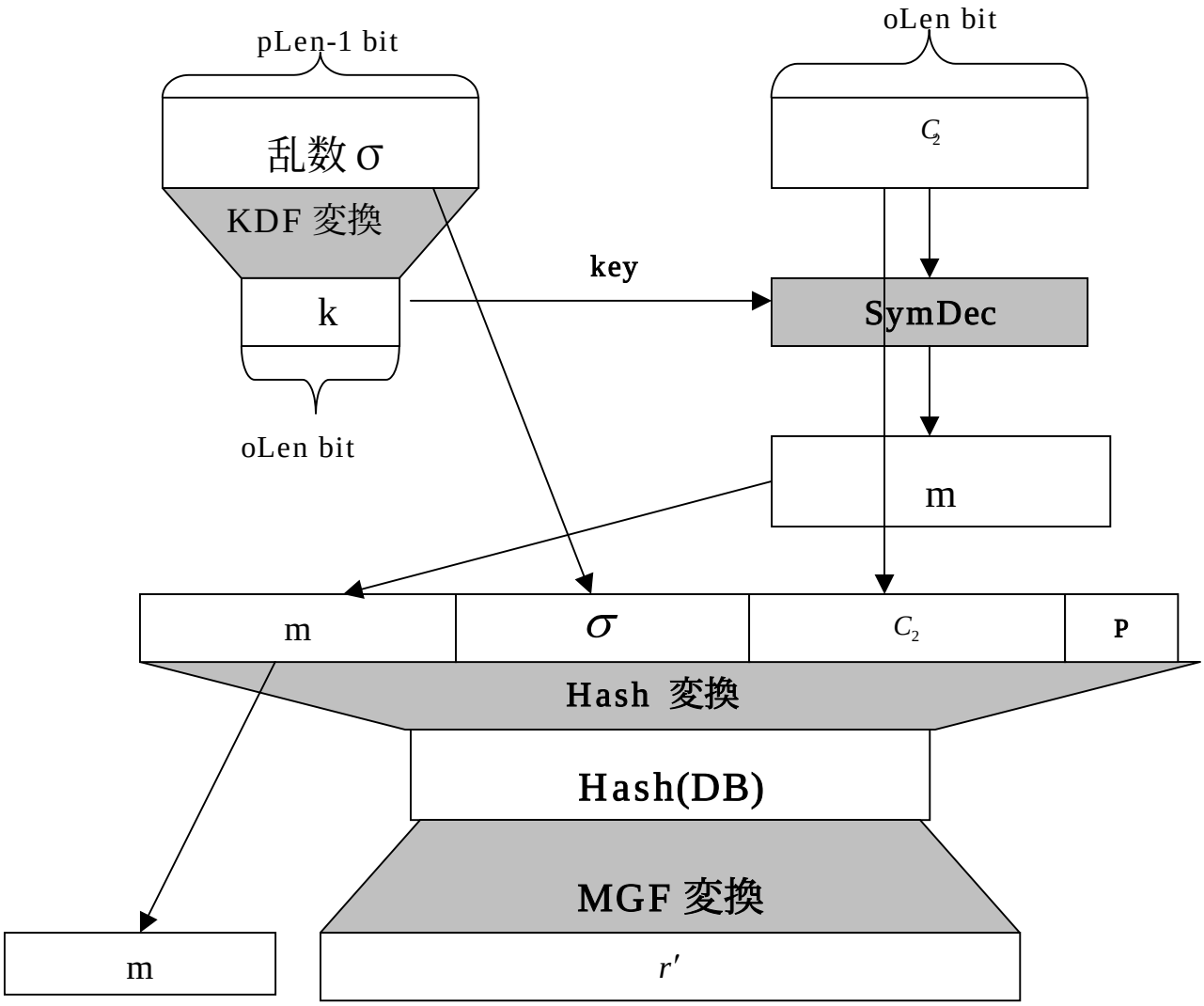

図 2: EME-3-DECODE(C_2, σ , P)