暗号アルゴリズム

「ECMQVS (Elliptic Curve MQV Scheme) in SEC1」

詳細評価(攻撃評価)レポート

２００１年1月１２日

目次

1 概要 3
2 ECMQVS の安全性評価 4
2.1 安全性考察のフレームワーク 4
2.2 鍵共有プロトコルに必要な特性 4
2.3 プリミティブの安全性 6
2.3.1 楕円曲線離散対数問題 6
2.3.2 楕円曲線 Diffie-Hellman 問題 8
2.4 スキームの安全性 9
3 他の標準化団体への採用実績 10
参考文献 11
付録 A ECMQVS の仕様 12
付録 B 記号の定義 15

１．概要

楕円曲線 MQV スキーム(ECMQVS)は、鍵共有(Key Agreement)の一方式である。そのプリミティブは、エンティティ Uが所有する 2 組の鍵ペアとエンティティ Vが所有する 2 つの公開鍵から、有限体の要素を 1 つ生成し、それを共有された鍵とする。

ECMQV プリミティブは、楕円曲線離散対数問題または楕円曲線 Diffie Hellman 問題が解かれれば破られるが、ECMQVS の安全性は、楕円曲線 Diffie-Hellman 問題の安全性と等価である、と応募者は予測している。ただしその根拠は応募書類中では詳細に評価されておらず、文献[LMQSV98]を参照しなければならない。楕円曲線離散対数問題と楕円曲線 Diffie-Hellman 問題に対する攻撃に対して安全な楕円曲線パラメータを用いれば、ECMQV プリミティブは安全であると思われる。

過去のいくつかの鍵共有法の提案は、安全性に欠陥が見つかっているが、その一因は、安全性と脅威に関する定義が、適切かつフォーマルになされていなかったことである。ECMQVS は、安全性と脅威に関して適切かつフォーマルに定義されている。その定義は、文献[LMQSV98]に述べられている。この定義の上で、ECMQVS は安全であると思われる。

ECMQVS は、ANSI X9.42、ANSI X9.63 および IEEE P1363 において標準化されている。また、応募者自身が中心メンバーである SECG (Standards for Efficient Cryptography Group)でも採用されている。SECG においても外部評価者による安全性評価を実施しているが、欠陥は見つかっておらず、評価は高い。

応募書類に書かれた仕様は、数学的解説から実装に必要な様々な関数まで網羅し、自己完結したわかりやすい記述になっている。

２．ECMQVS の安全性評価

２．１ 安全性考察のフレームワーク

安全なプロトコルを構成するためには、

- 脅威モデル
- 安全性の目標

を適切かつフォーマルに定義し、このモデルの元でプロトコルを設計しなければならない。本応募が安全性に関する議論として参照している文献[LMQSV98]には、プロトコルは、Blake-Wilson、Johnson、Menezes [BJM97]と Bellare、Rogaway [BR94]の定義に沿って構成されている、と記述されている。

過去のいくつかの鍵共有プロトコルの提案は、破られているが、その一因は、脅威モデルと安全性の目標の定義が不適切であったことである。ECMQVS は、十分適切に定義され、安全性に問題は無いと思われる。

２．２ 鍵共有プロトコルに必要な特性

安全な鍵共有プロトコルを構成するために考慮すべき項目として、次の 5 つが挙げられる。ECMQVS は、いずれの項目に対しても安全であると考えれらる。

(1) Known-key security

鍵共有プロトコルにおいては、唯一の秘密鍵が生成されなければならない。このような鍵は“セッション鍵”と呼ばれる。ある他のセッション鍵を第 3 者が入手した場合でも、別なセッション鍵が計算できないようにしなければならない。

(2) Forward security

１つあるいはそれ以上のエンティティの long-term 秘密鍵が破られても、信用できるエンティティによって以前に生成されたセッション鍵の安全性に影響を与えてはならない。

(3) Key-compromise impersonation

エンティティ A の long-term 秘密鍵が公開されたとする。この鍵を知った第 3 者は、A になりすますことができる。しかし、第 3 者が A 以外のエンティティになりすますことができてはならない。

(4) Unknown key-share

エンティティ A が知らないうちに、エンティティ B と鍵共有されてはならない。つまり、

エンティティ A がエンティティ C≠B と鍵共有した時、B が正しく A と鍵共有できてはならない。

(5) Key control

どのエンティティも、あらかじめ用意された値とセッション鍵が等しくなるように、鍵生成できてはならない。

２．３ プリミティブの安全性

ECMQV プリミティブの安全性に関して、

- 2 つのエンティティの公開鍵のみから、共有された有限体の要素を、第 3 者が計算することができない

ということについては、楕円曲線 Diffie-Hellman の安全性と等価である、と応募者は予測している。ただし応募書類中で詳細な安全性評価はなされておらず、“ 文献[LMQSV98]を参照 ” と記載されているだけである。

この等価性が破られない限りは、ECMQV プリミティブは安全であると思われる。

本節では、楕円曲線離散対数問題と、楕円曲線 Diffie-Hellman 問題の既知解法に対する安全条件を記述する。

２．３．１ 楕円曲線離散対数問題

ECMQV プリミティブは、楕円曲線離散対数問題と楕円曲線 Diffie-Hellman 問題を解くことの困難性に安全性の根拠をおいている。もしこれらの問題が解ければ、攻撃者は、共有された秘密情報 z を、$Q_{1,U}$、$Q_{2,U}$、$Q_{1,V}$、$Q_{2,V}$から計算することができる。

本節では、楕円曲線離散対数問題の既知解法の概要を記述する。

(1) Pollard-rho 法と Pohlig-Hellman 法に対する安全性

Pollard-rho 法と Pohlig-Hellman 法による楕円曲線離散対数問題(ECDLP)の解読を避けるために、楕円曲線の $\mathbf{F}_q$ 有理点の個数$\#E(\mathbf{F}_q)$は、十分大きな素数 n で割り切れなければならない。n の大きさは、通常 $n>2^{160}$ であることが推奨される。

(2) MOV 還元法と FR 還元法に対する安全性

Menezes、岡本、Vanstone による還元法および Frey、Ruck による還元法を避けるためには、非超特異な楕円曲線を選ぶべきである。つまり p が $q+1-\#E(\mathbf{F}_q)$を割らないような楕円曲線を選ぶ。さらに、n が q^k-1 を、十分大きな C に対して $1 \leqq k \leqq C$ の範囲で割らないことを検査する必要がある。実用上は、C は 20 で十分とされている[LMQS98]。

(3) SSSA 法に対する安全性

Semaev、Smart、佐藤、荒木による解読法(SSSA 法)を避けるために、$\mathbf{F}_q$ -anomalous な楕円曲線は避けなければならない。つまり、$\#E(\mathbf{F}_q) \neq q$ となる楕円曲線パラメータを選択する。

(4) Weil Decent 法に対する安全性

ある種の楕円曲線に対して、Weil Decent を用いた解読法の研究が、近年進展している。この解読法に対しては、今後の研究動向を注意深く見守る必要がある。

離散対数問題解法に対する安全条件のまとめ：

これらの既知解読法を避けるためには、$\#E(\mathbf{F}_q)$が十分大きな素数で割り切れるという条件のもとで、“ ランダム ” に楕円曲線を選択する、という方法がある。現在発見されていないある種の曲線に対して、将来、解読法が見つかった場合に備えるためにも有効である。

曲線がランダムに選択されたことが検査可能な、曲線生成法もある。楕円曲線の方程式の係数を、SHA-1 などの一方向性関数の出力を用いて、ある種の方法によって決める方法である。この方法は ANSI X9.62 に記述されている。

２．３．２ 楕円曲線 Diffie-Hellman 問題

ECMQV プリミティブは、楕円曲線離散対数問題と楕円曲線 Diffie-Hellman 問題を解くことの困難性に安全性の根拠をおいている。もしこれらの問題が解ければ、攻撃者は、共有された秘密情報 z を、$Q_{1,U}$、$Q_{2,U}$、$Q_{1,V}$、$Q_{2,V}$から計算することができる。

本節では、楕円曲線 Diffie-Hellman 問題に関して記述する。

Diffie-Hellman のプリミティブは、エンティティ Uが所有する秘密鍵とエンティティ V が所有する公開鍵とから、有限体の要素を生成する。2 つのエンティティが同様の演算を行えば、両者とも共通の有限体要素を生成することになる。Diffie-Hellman 問題の安全条件の基本的事項は、Uと Vの公開鍵を知る第 3 者が、共有される有限体要素を計算することができないことである。

楕円曲線離散対数問題が解ければ、楕円曲線 Diffie-Hellman 問題も解けることは明らかである。楕円曲線離散対数問題を解く最適なアルゴリズムの計算量が fully exponential であれば、楕円曲線離散対数問題と楕円曲線 Diffie-Hellman 問題は等価であるとの研究もある。

２．４ スキームの安全性

過去に提案されたいくつかの鍵共有法は、安全性に欠陥が見つかっている。欠陥が発見された一因は、安全性と脅威に関する定義が、適切かつフォーマルになされていなかったことである。ECMQVS は、安全性と脅威に関して適切かつフォーマルに定義されている。その定義は、文献[LMQSV98]に述べられている。この定義の上で、ECMQVS は安全であると思われる。

本節では、ECMQV スキームに関する既知の攻撃法と、それらに対する安全条件を記述する。

(1) 楕円曲線離散対数問題と楕円曲線 Diffie-Hellman 問題に対する攻撃

前節で述べたように、楕円曲線離散対数問題または楕円曲線 Diffie-Hellman 問題が解ければ、ECMQVS は破られる。

(2) 鍵生成に対する攻撃

ECMQVS には鍵生成が含まれる。そこでは乱数または擬似乱数生成関数を必要とする。乱数または擬似乱数生成関数は、他の多くの暗号系でも必要とする関数である。あるエンティティが、意図的に特殊な鍵を生成することを防ぐために、安全な乱数または擬似乱数生成関数が必要である。

(3) 鍵派生関数(Key Derivation Function)に対する攻撃

共有される鍵のビットの一部分が予測できたり、ビット間の相関性が見つかったりした場合、攻撃者は共有される鍵に関するある種の情報を学習することができる。

(4) 不正な楕円曲線パラメータの使用に対する攻撃

使用される楕円曲線パラメータは、正しい値であること(例えば曲線上の点であるべきデータは、その曲線上の点であること)が必要である。例えば、曲線の位数が不正であると、Pohlig-Hellman 法による攻撃が成立することがある。

スキームの安全性のまとめ：

楕円曲線パラメータを適切に選択し、適切な乱数または擬似乱数生成関数を使用すれば、ECMQVS は安全であると考えられる。

３．他の標準化団体での採用実績

ECMQVS は、ANSI X9.63(のドラフト)と IEEE P1363 において標準化されている。また、応募者自身が中心メンバーである SECG (Standards for Efficient Cryptography Group)でも採用されている。

SECG 自身でも、安全性に関する外部評価者による評価を実施している。その評価レポートは、SECG の Web サイト http://www.secg.org から得ることができる。4 つの評価レポートが現在アップロードされているが、これらのレポートによると、ECMQVS は十分な安全性があるとされており、評価者は高い評価を与えている。

参考文献

[BR94] M. Bellare and P. Rogaway. Entity authentication and key distribution. Advances in Cryptology - Crypto '93, Lecture Notes in Computer Science, vol. 773, Springer-Verlag, pages 232-249, 1994.

[BJM97] S. Blake-Wilson, D. Johnson, and A.J. Menezes. Key agreement protocols and their security analysis. In *Proceedings of the sixth IMA International Conference on Cryptography and Codin*g, pages 30–45, 1997.

[Ka98] B. Kaliski. MQV vulnerability. Posting to ANSI X9F1 and IEEE P1363 newsgroups. 1998.

[LMQSV98] L. Law, A. Menezes, M. Qu, J. Solinas, and S. Vanstone. An efficient protocol for authenticated key agreement. Technical report CORR 98-05, Department of Combinatorics & Optimization, University of Waterloo, March, 1998. Available from: http://www.cacr.math.uwaterloo.ca/

[SECG] http://www.secg.org/

付録 A. ECMQVS の仕様

ECMQVS の仕様を記述する(応募書類から抜粋)。仕様は、ECMQV プリミティブと、ECMQV スキームとに分けて記述する。

A.1 プリミティブ

楕円曲線 MQV のプリミティブの仕様を記述する。これは、次節で記述する、楕円曲線 MQV スキームの基本部分である。このプリミティブの基本的アイディアは、エンティティ U が所有する 2 つの鍵対と、エンティティ V が所有する 2 つの公開鍵から、共有される秘密情報を生成することである。U は V と秘密情報を共有するために、次のプロセスを演算する。

入力：

・正当な楕円曲線パラメータ T=(p, a, b, G, n, h)または(m, $f(x)$, a, b, G, n, h)

・U が所有する、T に付随する 2 つの鍵対($d_{1,U}$, $Q_{1,U}$)と($d_{2,U}$, $Q_{2,U}$)

・V が所有すると称される、T に付随する 2 つの公開鍵 $Q_{1,V}$ と $Q_{2,V}$

$Q_{1,V}$ と $Q_{2,V}$ は正当でなければならない。

出力：

・共有された有限体の要素 z または " 不正 "

アルゴリズム：

(1) T=(p, a, b, G, n, h)ならば q=p、(m, $f(x)$, a, b, G, n, h)ならば q=2^m とおく

(2) $Q_{2,U}$ =(x_Q, x_Q)を用いて、整数 $\overline{Q_{2,U}}$ を次のように計算する

(2-1) x_Q を整数 x に変換する(変換法は応募書類に記述された方法を使用する)

(2-2) 次を計算する： $\bar{x} \equiv x \pmod{2^{\lceil (\log_2 n)/2 \rceil}}$

(2-3) 次を計算する： $\overline{Q_{2,U}} = \bar{x} + 2^{\lceil (\log_2 n)/2 \rceil}$

(3) 次の整数を計算する： $s \equiv d_{2,U} + \overline{Q_{2,U}} \cdot d_{1,U} \pmod{n}$

(4) $Q_{2,V}$ =($x_{Q'}$, $x_{Q'}$)を用いて、整数 $\overline{Q_{2,V}}$ を次のように計算する

(4-1) $x_{Q'}$ を整数 x' に変換する(変換法は応募書類に記述された方法を使用する)

(4-2) 次を計算する： $\bar{x}' \equiv x' \pmod{2^{\lceil (\log_2 n)/2 \rceil}}$

(4-3) 次を計算する： $\overline{Q_{2,V}} = \overline{x'} + 2^{\lceil (\log_2 n)/2 \rceil}$

(5) 次の楕円曲線の点を計算する： $P = (x_P, y_P) = hs \times (Q_{2,V} + \overline{Q_{2,V}} Q_{1,V})$

(6) Pが無限遠点かどうかを検査する。もし無限遠点なら“不正”と出力し終了。

(7) $z = x_P$ を共有された秘密の有限体の要素として出力する。

A.2 スキーム

楕円曲線 MQV スキームは、楕円曲線に基づく鍵共有(key agreement)方式である。スキームは、3 つのステップで構成される。第 1 はスキームセットアップ法、第 2 は鍵展開(key deployment procedure)、第 3 は鍵共有(key agreement)である。

(1) スキームセットアップ

Uと Vは、楕円曲線 MQV スキームを用いるための準備として、次を実行する。

1. U と V は、使用する鍵展開関数を決める(注：応募書類には、鍵展開関数として、ANSI-X9.63 に記載の鍵展開関数のみ記載されている)。また、鍵展開関数へのオプションがもしあれば選択する。KDF を選択された鍵展開関数とする。

2. Uと Vは、必要なセキュリティレベルをもつ楕円曲線パラメータ $T = (p, a, b, H, n, h)$または$(m, f(x), a, b, H, n, h)$を選択する(Tの生成法は、応募書類に記述された方法を使用する)。Uと Vは、Tが正当であることを確認する(応募書類に記述された方法を使用する)。

(2) 鍵展開

Uと Vは、楕円曲線 MQV スキームを用いるための準備として、次の鍵展開法を実行する。

1. U は、セットアップで生成された楕円曲線パラメータ Tに附随する、楕円曲線の 2 つの鍵対$(d_{1,U}, Q_{1,U})$と$(d_{2,U}, Q_{2,U})$を生成する。鍵対生成法は、応募書類に記載された方法を用いる。
2. Vは、セットアップで生成された楕円曲線パラメータ Tに附随する、楕円曲線の 2 つの鍵対$(d_{1,V}, Q_{1,V})$と$(d_{2,V}, Q_{2,V})$を生成する。鍵対生成法は、応募書類に記載された

方法を用いる。

3. Uは、Vが選択した第 1 の楕円曲線公開鍵 $Q_{1,V}$を信頼できる方法で得る。Uは $Q_{1,V}$ が正当であることを確認する(確認方法は、応募書類に記述された方法を用いる)。
4. Vは、Uが選択した第 1 の楕円曲線公開鍵 $Q_{1,U}$ を信頼できる方法で得る。V は $Q_{1,U}$ が正当であることを確認する(確認方法は、応募書類に記述された方法を用いる)。
5. U と V は、第 2 の公開鍵 $Q_{2,U}$と $Q_{2,V}$を交換する。
6. U は $Q_{2,V}$が少なくとも部分的に正当であることを確認する(確認方法は、応募書類に記述された方法を用いる)。
7. V は $Q_{2,U}$ が少なくとも部分的に正当であることを確認する(確認方法は、応募書類に記述された方法を用いる)。

(3) 鍵共有

Uと Vは、楕円曲線 MQV スキームを用いて鍵とするデータを生成するために、以下に記述する鍵共有を行う。簡単のために、U が行うオペレーションを記述する。V のオペレーションは Uと同様である。Uは、セットアップと鍵展開で得られた鍵とパラメータを用いて、以下のように Vとの鍵となるデータを生成する。

入力：

1. 鍵となるデータのオクテット数：*keydatalen* (整数)
2. (オプション) Uと Vで共有されたデータのオクテットストリング *SharedInfo*

出力：

長さ *keydatalen* オクテットの、オクテットストリング鍵データ K または“ 不正 ”

アルゴリズム：

(1) 楕円曲線 MQV プリミティブを用いて、鍵展開で得られた Uの鍵対($d_{1,U}$, $Q_{1,U}$)と($d_{2,U}$, $Q_{2,U}$)と、鍵展開で得られた Vの公開鍵 $Q_{1,V}$と $Q_{2,V}$とから、共有される有限体の秘密の要素 $z \in \mathrm{F}q$ を得る。MQV プリミティブが“ 不正 ”と出力したら、“ 不正 ”と出力し終了する。

(2) $z \in \mathrm{F}q$ をオクテットストリング Z に変換する(変換方法は、応募書類に記載されている)。

(3) セットアップで得られた鍵展開関数 *KDF* を用いて、Z と[*SharedInfo*]から長さ *keydatalen* の鍵データ Kを生成する。鍵展開関数が“ 不正 ”と出力したら、“ 不正 ”と出力し、終了する。

(4) Kを出力する。

付録 B 記号の定義

本評価書で使用する主な記号の定義を記述する。(応募書類からの抜粋)

[X] … X を含むことはオプション

$\lceil x \rceil$ … x 以上の最小の整数。

E … 有限体 $\mathbf{F}_q$ 上の楕円曲線。

$E(\mathbf{F}_q)$ … 楕円曲線 E 上の点全体。無限円点 O を含む。

$\#E(\mathbf{F}_q)$ … E が $\mathbf{F}_q$ 上定義されている時、曲線上の点全体(無限円点 O を含む)の数。

$\mathbf{F}_p$ … p 個の要素を含む有限体。p は素数。

$\mathbf{F}_{2^m}$ … 2^m 個の要素を含む有限体。m は正整数。

G … ベースポイント。

h … $\#E(\mathbf{F}_q)/n$。n はベースポイント G の位数。co-factor ともいう。