Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 7, 2016

A. Kato
NTT Software Corporation
M. Scott
CertiVox
T. Kobayashi
Y. Kawahara
NTT

July 6, 2015

Optimal Ate Pairing
draft-kato-optimal-ate-pairings-00

Abstract

Pairing is a special map from two elliptic curve that called Pairing-friend curves to a finite field and is useful mathematical tools for constructing cryptographic primitives. It allows us to construct powerful primitives. (e.g. [3] and [4])

There are some types of pairing and its choice has an impact on the performance of the primitive. For example, Tate Pairing [3] and Ate Pairing [4] are specified in IETF. This memo focuses on Optimal Ate Pairing [2] which is an improvement of Ate Pairing.

This memo defines Optimal Ate Pairing for any pairing-friendly curve. We can obtain concrete algorithm by deciding parameters and building blocks based on the form of a curve and the description in this memo. It enables us to reduce the cost for specifying Optimal Ate Pairing over additional curves. Furthermore, this memo provides concrete algorithm for Optimal Ate Pairing over BN-curves [7] and its test vectors.

This Internet-Draft will expire on January 7, 2016.

Copyright Notice

Table of Contents

1. Introduction 3
2. Requirements Terminology 3
3. Preliminaries 3
3.1. Elliptic Curve 3
3.2. Bilinear Map 4
4. Optimal Ate Pairing 4
4.1. Guide for Decision on Parameters for Optimal Ate Pairing 5
4.2. Miller Loop 6
4.3. Straight Line Function 7
5. Optimal Ate Pairing over BN-curves 7
5.1. Straight Line Function over BN-curves 8
5.2. Doubling Step of Miller Loop over BN-Curves 9
5.3. Addition Step of Miller Loop over BN-Curves 10
6. Algorithm Identifiers 11
7. Security Considerations 11
8. Acknowledgements 11
9. Change log 11
10. References 11
10.1. Normative References 11
10.2. Informative References 11
Appendix A. Test Vectors of Optimal Ate Pairing over BN-curves . 13
A.1. 254-Bit-Curves by Beuchat et al. 13
A.2. 254-Bit-Curves by Nogami et al. / Aranha et al. 14
A.3. 254-Bit-Curves by Scott 15

A.4. 254-Bit-Curves by BCMNPZ 16
Authors' Addresses 16

1. Introduction

Pairing is a special map from two elliptic curve that called Pairing-friend curves (PFCs) to a finite field and is useful mathematical tools for constructing cryptographic primitives. It allows us to construct powerful primitives like Identity-Based Encryption (IBE) [5] and Functional Encryption (FE) [6]. The IBE and FE provide a rich decryption condition. Some Pairing-Based Cryptography is specified in IETF. (e.g. [3] and [4])

There are some types of pairing and its choice has an impact on the performance of the primitive. For example, primitives by using Tate Pairing [3] and Ate Pairing [4] are specified in IETF. This memo focuses on Optimal Ate Pairing which is an improvement of Ate Pairing. Optimal Ate Pairing allows us to construct Pairing-Based Cryptography with high performance and is implemented in some open source softwares. ([8], [9], and [10])

This memo defines Optimal Ate Pairing [2] for any PFC. We can obtain concrete algorithm by deciding parameters and two building blocks based on the form of a curve. It enables us to reduce the cost for describing the body of Optimal Ate Pairing when Optimal Ate Pairing is specified over additional curves in IETF. Furthermore, this memo provides concrete algorithm for Optimal Ate Pairing over BN-curves [7] and its test vectors. This memo is expected to use by combining Optimal Ate Pairing with a suitable PFC for a primitive in order to realize same functional structure of ECDSA and ECDH. (i.e. DSA over elliptic curve and DH over elliptic curve)

2. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this memo are to be interpreted as described in [1].

3. Preliminaries

In this section, we introduce the definition of elliptic curve and bilinear map, notation used in this memo.

3.1. Elliptic Curve

Throughout this memo, let p > 3 be a prime, q = p^n, and n be a natural number. Also, let F_q be a finite field. The curve defined by the following equation E is called an elliptic curve.

```
E : y^2 = x^3 + A * x + B such that A, B are in F_q,
          4 * A^3 + 27 * B^2 != 0 mod F_q
```


Solutions (x, y) for an elliptic curve E, as well as the point at infinity, are called F_q-rational points. The additive group is constructed by a well-defined operation in the set of F_q-rational points. Typically, the cyclic additive group with prime order r and the base point G in its group is used for the cryptographic applications. Furthermore, we define terminology used in this memo as follows.

O_E: the point at infinity over elliptic curve E.

#E(F_q): number of points on an elliptic curve E over F_q.

cofactor h: h = #E(F_p)/r.

embedding degree k: minimum integer k such that r is a divisor of q^k - 1

3.2. Bilinear Map

Let G_1 be an additive group of prime order r and let G_2 and G_T be additive and multiplicative groups, respectively, of the same order. Let P, Q be generators of G_1, G_2 respectively. We say that (G_1, G_2, G_T) are asymmetric bilinear map groups if there exists a bilinear map e: (G_1, G_2) -> G_T satisfying the following properties:

1. Bilinearity: for any S in G_1, for any T in G_2, for any a, b in Z_r, we have the relation e([a]S, [b]T) = e(S, T)^{a * b}.

2. Non-degeneracy: for any T in G_2, e(S, T) = 1 if and only if S = O_E. Similarly, for any S in G_1, e(S, T) = 1 if and only if T = O_E.

3. Computability: for any S in G_1, for any T in G_2, the bilinear map is efficiently computable.

4. Optimal Ate Pairing

This section specifies Optimal Ate Pairing e for c_0, ..., c_l and s_i = sum_{j=i}^l c_j * q^j with following conditions

1. c_l is not 0

2. r is a divisor of s_0

3. r^2 is not a divisor of s_0

4. r does not divide s_0 * k * q^{k-1} - (q^k - 1)/r * sum_{i=0}^l i * c_i * q^{i - 1}

Section 4.1 shows a guide to decide these parameters c_0, ..., c_l. Optimal Ate Pairing is specified below and Miller Loop f which are its building blocks are introduced in Section 4.2. Straight Line Function l which is building blocks of Optimal Ate Pairing and Miller Loop are defined in Section 4.3. Section 4.3 only show the definitions because its descriptions are based on the form (of the PFC?). Practically, concrete algorithms need to be specified for a form of PFC.

Input:

- A point P in G_1

- A point Q in G_2

Output:

- The value e(P, Q) in G_T

Method:

1. f = 1

2. ln = 1

3. for i = 0 to l

   (a) f = f * f_{c_i, Q}^{q^i}(P)

   end for

4. for i = 0 to l - 1

   (a) ln = ln * l_{[s_i + 1]Q, [c_i * q^i]Q}(P)

   end for

5. return (f * ln)^{(q^k - 1)/r}

4.1. Guide for Decision on Parameters for Optimal Ate Pairing

This subsection shows a guide for decision on parameters c_0, ..., c_l for Optimal Ate Pairing. According to [2], a way is to choice coefficients of short vector of the following lattice L with a minimal number of coefficients as parameters c_0, ..., c_l.

L = (v_1, ..., v_phi(k)) where

- v_1 is column vector t(r, -q, -q^2, ..., -q^{phi(k) - 1})
- v_i is column vector whose i component is 1 and other components is 0 for i = 2, ..., phi(k)

4.2. Miller Loop

In this subsection, we specify Miller Loop f which is building block of Optimal Ate Pairing.

Input:

- A point P in G_1

o A point Q in G_2

o An integer s

Output:

o f_{s, Q}(P)

Method:

1. compute s_0, ..., s_L such that |s| = sum_{j=0}^L s_j * 2^j with s_j is in {0, 1} and s_L = 1

2. T = Q

3. f = 1

4. for j = L - 1 down to 0

   (A) Doubling Step

      (a) ln = l_{T, T}(P)

      (b) T = 2 * T

   (B) f = f^2 * ln

   (C) if s_j = 1

      (a) Addition Step

         (i) ln = l_{T, Q}(P)

         (ii) T = T + Q

      (b) f = f' * ln

   end if

end for

5. if s < 0, then f = f^{-1}

6. return f

4.3. Straight Line Function

Straight Line Function l_{Q, Q'}(P) is calculated by a point P for linear equation defined as a line l though points Q, Q'. Note that Straight Line Function l_{Q, Q'}(P) is calculated by a point P for linear equation defined as a tangent line to an elliptic curve E at a point Q of E on condition that Q = Q'. The function is used for Optimal Ate Pairing in Section 4 and Miller Loop in Section 4.2

5. Optimal Ate Pairing over BN-curves

In this section, we specify Optimal Ate Pairing over BN-curves [7]. BN-curves define over a finite field F_p, and have embedding degree k = 12, r(t) = 36 * t^4 + 36 * t^3 + 18 * t^2 + 6 * t + 1, and p(t) = 36 * t^4 + 36 * t^3 + 24 * t^2 + 6 * t + 1, where t is the specific integer in [7].

The extension fields are defined by following:

F_{p^2} is set to F_p[u]/(u^2 - e2)

F_{p^6} is set to F_{p^2}[v]/(u^3 - e6)

F_{p^12} is set to F_{p^6}[w]/(w^2 - e12)

The constants e3, e6 and e6 which are varied by G_T are defined in [7].

Hence parameters for Optimal Ate Pairing over D-Type twisted curve are following by the method in Section 4.1:

1. l = 3

2. c_0 = 6 * t + 2

3. c_1 = 1

4. c_2 = -1

5. c_3 = 1

These short vectors are specified in section 4. A of [2].

Algorithm of Optimal Ate Pairing by Miller Loop in Section 4.2 based on building blocks specified in Section 5.2 and Section 5.3 and Straight Line Function f in Section 5.1 over BN-curves is as following:

Input:

- A point P in G_1
- A point Q in G_2

Output:

- The value e(P, Q) in G_T

Method:

1. f_1 = f_{c_0, Q}(P)
2. l_1 = l_{[p^3]Q}, - [p^2]Q}(P)
3. l_2 = l_{[p^3]Q - [p^2]Q, [p]Q}(P)
4. l_3 = l_{[p]Q - [p^2]Q + [p^3]Q, [6 * t + 2]Q}
5. return (f_1 * l_1 * l_2 * l_3)^{(p^k - 1)/r}

5.1. Straight Line Function over BN-curves

This subsection shows an operation of Straight Line Function over BN-curves for Optimal Ate Pairing.

Input:

- A point Q = (x_1, y_1) in G_2
- A point Q' = (x_2, y_2) in G_2
- A point P = (x, y) in G_1

Output:

o l_{Q, Q'}(P)

Method:

1. If Q != +- Q'

   (A) lambda = (y_2 - y_1)/(x_2 - x_1)

   (B) t0 = -lambda * x

   (C) t1 = lambda * x_1 - y_1

   (D) ln = y + t0 * w + t1 w^3

2. If Q = Q'

   (A) lambda = (3 * x_1^2)/(2 * y_1)

   (B) t0 = -lambda * x

   (C) t1 = lambda * x_1 - y_1

   (D) ln = y + t0 w + t1 w^3

   (E) return ln

3. If Q = -Q'

   (A) ln = x - x_1 w^3

4. return ln

5.2. Doubling Step of Miller Loop over BN-Curves

This subsection shows an operation of Doubling Step of Miller Loop over BN-curves. (i.e. operation of method 4-(A) in Section 4.2 over BN-curves)

Input:

o A point P = (x, y) in G_1

o A point Q = (x_1, y_1) in G_2

Output:

- ln such that l_{Q, Q}(P)

- A point T = (x_3, y_3) such that [2]Q

Method:

1. lambda = (3 * x_1^2)/(2 * y_1)
2. x_3 = lambda^2 - 2 * x_1
3. y_3 = lambda * (x_1 - x_3) - y_1
4. t0 = -lambda * x
5. t1 = lambda * x_1 - y_1
6. ln = y + t0 w + t1 w^3
7. return ln and T

5.3. Addition Step of Miller Loop over BN-Curves

This subsection shows an operation of Addition Step of Miller Loop over BN-curves. (i.e. operation of method 4-(C)-(a) in Section 4.2 over BN-curves)

Input:

- A point Q = (x_1, y_1) in G_2
- A point Q' = (x_2, y_2) in G_2
- A point P = (x, y) in G_1

Output:

- ln such that l_{Q, Q'}(P)
- A point T = (x_3, y_3) such that Q + Q'

Method:

1. lambda = (y_2 - y_1)/(x_2 - x_1)

2. x_3 = lambda^2 - x_1 - x_2

3. y_3 = lambda * (x_1 - x_3) - y_1

4. t0 = -lambda * x

5. t1 = lambda * x_1 - y_1

6. ln = y + t0 w + t1 w^3

7. return ln and T

6. Algorithm Identifiers

TBD

7. Security Considerations

The security of cryptographic primitive which is constructed by pairing depends on pairing-friendly curves (PFC). PFC must satisfy computational assumption which the primitive requires at the level of security strength in system when the primitive is constructed by using Optimal Ate Pairing.

8. Acknowledgements

TBD

9. Change log

NOTE TO RFC EDITOR: Please remove this section in before final RFC publication.

10. References

10.1. Normative References

[1] Bradner, S., "Key words for use in RFCs to Indicate

Requirement Levels", RFC 2119, March 1997.

[2] Vercauteren, F., "Optimal pairings", Proceedings IEEE Transactions on Information Theory 56(1): 455-461 (2010), 2010.

10.2. Informative References

[3] Boyen, X. and l. Martin, "Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems", RFC 5091, December 2007.

[4] Hitt, L., "ZSS Short Signature Scheme for Supersingular and BN Curves", draft-irtf-cfrg-zss-02 (work in progress), 2013.

[5] Boneh, D. and M. Franklin, "Identity-based encryption from the Weil pairing", Proceedings Lecture notes in computer sciences; CRYPTO --CRYPTO2001, 2001.

[6] Okamoto, T. and K. Takashima, "Fully Secure Functional Encryption with General Relations from the Decisional Linear Assumption", Proceedings Lecture notes in computer sciences; CRYPTO --CRYPTO2011, 2010.

[7] Kasamatsu, K., Kanno, S., Kobayashi, T., and Y. Kawahara, "Barreto-Naehrig Curves", draft-kasamatsu-bncurves-01 (work in progress), 2015.

[8] "University of Tsukuba Elliptic Curve and Pairing Library", 2013, <http://www.cipher.risk.tsukuba.ac.jp/tepla/index_e.html>.

[9] Aranha, D. and C. Gouv, "RELIC is an Efficient LIbrary for Cryptography", 2013, <https://code.google.com/p/relic-toolkit/>.

[10] Scott, M., "The MIRACL IoT Multi-Lingual Crypto Library", 2015, <https://github.com/CertiVox/MiotCL.git>.

Appendix A. Test Vectors of Optimal Ate Pairing over BN-curves

In this section, we specify test vectors of optimal ate pairing over BN-curves which are specified by [7] in the following way.

Parameter:

Pairing-Param-ID is an identifier with which the pairing parameter set can be referenced.

Input:

P is a point of E in G_1

Q is a point of E' in G_2

Output:

e(P, Q) is computation of pairing in G_T

A.1. 254-Bit-Curves by Beuchat et al.

This subsection shows test vector of 254-bit curves by Beuchat et al. [7] and reprints its parameters under F_{p^2} = F_p[u]/(u^2 + 5), F_{p^6} = F_{p^2}[v]/(v^3 - u), F_{p^12} = F_{p^6}[w]/(w^2 - v) as a reference.

Parameter:

Pairing-Param-ID: Beuchat

Input:

P = (0x0A971735A70FBDD0F94D7D6EFBBC81BEA78D2D92A8510F3344038A416419AD97, 0x09456E41754237447752A448282C0873785F724447E1299826F53AC556936D3F)

Q = (0x115231D7B49901BA97CB93B5227F7F7F438A346532893DD5FAFD518950924AA9 + 0x0DF12398FB78695A50BB3499B7E23B0D9035989B91A76D13AF7BC64374BFB8A6 u, 0x051D0E087527BC9F41379FB0272EC91E5F28EE011B183EF7D6712EF3FC9A1A66 + 0x0107E6654DC6C36E163B7867AECB98E4046084734524DBB562E73E5A811F678A u)

Output:

e(P,Q) = (0x06A4E0DD1F7FD2F9E5DACAB02CEC9CE8254925C5DC6697E153F05A242CBCA8A8 + 0x22A0E22C097AEC1187087B7632C9B963B0E779BC8D09848C44D3EA95CD1C1F8C u + 0x0751037182B5F93BCAB31B115A2C0A0DCC09C6DB7602E0

551DD44925F3D364B3 v + 0x04B6BFFB9EB68AD6A99ACF52B8AAD1D17D328847C6313201A6B659C9DAA5CDFE uv + 0x13BE65D47487BF6D96C146C18855C1F87BF994F9F1048524568EA0CB9DC402AD v^2 + 0x1202BE31EB2BDCBEF9F3CC00F1B2CC35FADBE1A0D66CCBF40B024ADFA84C77D1 uv^2 + 0x15F9E3D10B580FF1AB2282EF1DC39A88E06F93A18303E9520D99B86D665F5380 w + 0x0A1C6D26A6D683031D95C4369DB90F5FEE36D5008AA498D2CB6F2DDE6258CDA6 uw + 0x1611153BF02F1CF7985B98C3F3CB641D39283DBA55E22D1C614568F84959C6FC vw + 0x10BEF55B7539743CBEAB13E49116A143302F6F28CCD71A69860CEF5208483809 uvw + 0x166BD873D0C65DE66300A168BBDC16F0AB1B57A0809973239F2109A7D25AD349 v^2w + 0x14D4B5014F840144D03C0C6B6010BB246EE6A69BF704D7542FBAA8F2D2A27308 uv^2w)

A.2. 254-Bit-Curves by Nogami et al. / Aranha et al.

This subsection shows test vector of 254-bit curves by Nogami et al. / Aranha et al. [7] and reprints its parameters under F_{p^2} = F_p[u]/(u^2 + 1), F_{p^6} = F_{p^2}[v]/(v^3 - (1 + u)), F_{p^12} = F_{p^6}[w]/(w^2 - v) as a reference.

Parameter:

Pairing-Param-ID: Nogami-Aranha

Input:

P = (0x2074A81D4402A0B63B947335C14B2FC3C28FEA2973860F686114BEC4670E4EB7, 0x06A41108087B20038771FC89FB94A82B2006034A6E8D871B3BC284846631CBEB)

Q = (0x049EEDB108B71A87BFCFC9B65EB5CF1C2F89554E02DF4F8354E4A00F52183C77 + 0x1FB93AB676140E87D97226185BA05BF5EC088A9CC76D966697CFB8FA9AA8845D u, 0x0CD04A1ED14AD3CDF6A1FE4453DA2BB9E686A637FB3FF8E2573644CC1EDF208A + 0x11FF7795CF59D1A1A7D6EE3C3C2DFC765DEF1CAA9F14EA264E71BD7630A43C14 u)

Output:

e(P,Q) = (0x03E1F2693AC6D549898C78897EB158490A4832E296F888D30140500DB7BD3D12 + 0x1EBC54A76E844EB5D352945226FB103DE9EC1A4FC689B87FAA66EF8ABA79D3ED u + 0x0A5A5405542F67384D683A48C281F3676B67554ED5DA1700784169A0B47A57E4 v + 0x048B66DAFCAEE86DB4D46AB71A9FE848443EF81F488D8366A727B39698CF7201 uv + 0x142715D6482BC6FA77377C9CBC2A51C047C16DE88483D5A889C7EF4DF5F03BDB v^2 + 0x11EE0C12164133041C3DCF312CE111C845B60092818F7B72805D4AFF61427934 uv^2 + 0x22371AF975DAE562F686988CDBBD02702C959BBF843A1FB3C7532D07BE3D7A3A w + 0x04052CA960900684A1B26C434B2776AA70736841474C16208CCD1A7C27927E19 uw + 0x05D259DA3F3AAAA54A6AE5FE8272A5B79D7F4E5BDF3B5E3C815AD781113F7548 vw + 0x0843C37BC5BDBF253E3BCE568F5905A63867D8836855B74CBA0C800D5DC41B71 uvw

+ 0x13CA93E1377EF0F6DD38FC2F96DBD3E8B0922F60D1F274EAC63DC1AF2EE9754C v^2w + 0x0D467F3DA4FB329A5CB406D0A7B743A3A2FFCD09BF95EE8A856B94AF191D96AF uv^2w)

A.3. 254-Bit-Curves by Scott

This subsection shows test vector of 254-bit curves by Scott [7] and reprints its parameters under F_{p^2} = F_p[u]/(u^2 + 1), F_{p^6} = F_{p^2}[v]/(v^3 - (1 + u)), F_{p^12} = F_{p^6}[w]/(w^2 - v) as a reference.

Parameter:

Pairing-Param-ID: Scott

Input:

P = (0x8a9143801f541142f89e498a1c06ba0959b8f9713abda0881e5de80d8aff11a + 0x17df54e2be5e8afeb9a42f412825f79c32841307471fb2b6a14e3a0fc6e010f4)

Q = (0x21794a9da7b34b2c1614315d7d90a282c484c8fd49c0c8ba75b079ae3047d566 + 0x1a9b474c4519e6faee5b32c7cb65547d8707137bca00c9c182d10b7e3e305936 u, 0xb00d54bf5a298d0eacdefb0efdb74d1a7e744722f61cc8844884fcce20ff876 + 0x5ecf8bd02e1f5363c8402163c9a235df56b133cc2c8a926c0e65e985d746b7b u)

Output:

e(P,Q) = (0x13d3127ba07feffc8c1a608afc58a33a25148176968ef0ec0a2e09b62344f984 + 0x1774dfc7361e1d4cd2de4bf62cd9b460f0a78487e75994f9e2551fed2f9d2b78 u + 0x2c7888f053123b5a815125b2c409e3f986594f6c35585cfb1ed1a1cbbd2ea65 v + 0xe7e7af51c459f6e0ef489348664bc4277e023a5031bee98658d5b357c07d7e8 uv + 0x8d0f0dd32f31d3624dd9e179233a1f2f2d13cc1869f2eb933cd3cded75efe0d v^2 + 0x63e676f8cc5be53e8718cc9e61a8c5a018ac47e3a66f83f4c403ec8caaa130e v^2u + 0x1643c6ec6cf54a1970bfea19c55e34a312eb5c825f8d31354200d29339d2ca61 w + 0xaae41d356d24b0234dc2b714b595aa297f585bbe9a7c4840d58d62cdfaa1764 wu+ 0x1ea5e2efa342adcbc3ac757254d03bfde32ef6a8445bfa6a7b13aee776430594 wv + 0x3aa5bc92f95887ce42ef03e666dd1455d640a031b062ed7a65fbf0a59d996b8 wvu + 0xf7735a9655207b2fe6e8e73d8f8c3f79f8a08aaeb670e6b9059d8f0739891ec wv^2 + 0x1a501fad47a0406e50b705a544377ee1ad7518adbbb49cbe30ce31770ae9be2e wv^2u)

A.4. 254-Bit-Curves by BCMNPZ

This subsection shows test vector of 254-bit curves by BCMNPZ [7] and reprints its parameters under F_{p^2} = F_p[u]/(u^2 + 1), F_{p^6} = F_{p^2}[v]/(v^3 - (1 + u)), F_{p^12} = F_{p^6}[w]/(w^2 - v) as a reference.

Parameter:

Pairing-Param-ID: BCMNPZ

Input:

P = (0x1bec8eae1f1d3959e394588e49d09f2d3070efda1f836640288cf21af5488765 + 0x2d148d39f9edf5325d9a1f4820774930675669a6fe20284e435f4bfe3d3273c)

Q = (0xd62cf33cd0e46fdc338cfab52ca5cdebf1a9348e4460545441584ff4f8dc275 + 0x22701025e0cd2bfed4518febe8e7fa97a3c7f33f2fdd280e24d651be9d17d7a8, 0x1cc6cbd065535e7f83be0cfc4f39d4687558fc21dcdc6e46aca508c4f6cc1f90 + 86ee46779f9e9922a870137d033e484ec5c5ba979b75bba179064abff0cf2a u)

Output:

e(P,Q) = (0x20f263ae42e42cfd53cf99dc238ed7b61951c1c767af88a72ad3c19ca54cdb2d + 0xa96b263aade3501f7201808028c4ce11793dd84127d80525fa57f892d3043f6 u + 0x3a31ca4864d996d64181d9a0b025e7368d60b1f53a8276a2c39e02a58b6636e v + 0x2301fe7eb607f6dd63b72979753c96d23fdd487f11677644884f86a83c837174 uv + 0xcbe52ab6e1c210cf80215816f38d8964c45347bd3802c66d85e616ca9786dde v^2 + 0x1c039dee75146d8ae6812568e77d11cfa060d11e0224dc6e28606bfb14090650 v^2u + 0x2344fb2b5dd57710d54458383cd33bd8f928babfe6f7d641887a565790b88e24 w + 0x8e48a543c2a73cca42811a2fea2e79eb3e628e27e54a477b5e1652466629608 wu+ 0x96a48564f586e1d59d8a9393730824b885818e93a3ce4bfae057682efc37aeb wv + 0x17260fa31ed89d4e90d7a1a2652379e4329927e61f15b11a2ce2a93c84050245 wvu + 0x5bd893369435b63a10384db8248dab8908f2173e166129d0cccd6d37c89dce6 wv^2 + 0x2a4dec6bbfe98df2c9169b06410c329d4c699747ca649e611d9960416d615b5 wv^2u)

Authors' Addresses

Akihiro Kato
NTT Software Corporation

EMail: kato.akihiro-at-po.ntts.co.jp

Michael Scott
CertiVox

EMail: mike.scott-at-certivox.com

Tetsutaro Kobayashi
NTT

EMail: kobayashi.tetsutaro-at-lab.ntt.co.jp

Yuto Kawahara
NTT

EMail: kawahara.yuto-at-lab.ntt.co.jp