Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 19, 2018

A. Kato
NTT TechnoCross Corporation
T. Kobayashi
Y. Kawahara
T. Kim
T. Saito
NTT

March 18, 2018

The threat of Pairing based cryptographic protocols. draft-kato-threat-pairing-00

Abstract

Pairing is a special map from two elliptic curves that called Pairing-friendly curves to a finite field and is useful mathematical tools for constructing cryptographic primitives.

At CRYPTO 2016, Kim and Barbulescu proposed an efficient number field sieve algorithm for the discrete logarithm problem in a finite field. The security of pairing-based cryptography is based on the difficulty in solving the DLP. Hence, it has become necessary to shift the parameters that the DLP is computationally infeasible against the efficient number field sieve algorithms.

This memo introduce Optimal Ate Pairing and two pairing-friendly curves with parameters of pairing against efficient number field sieve algorithms.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 19, 2018.

Table of Contents

1. Introduction

Pairing is a special map from two elliptic curves that called Pairing friendly 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[14] 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.

We need to choose an appropriate type of elliptic curve and parameters for the pairing-based cryptographic schemes, because the choice has great impact on security and efficiency of these schemes. However, an RFC on elliptic curves with pairings has not yet been provided in the IETF.

In some open source softwares ([7], [8], and [9]) are implemented by Optimal Ate Pairing which is an improvement of Ate Pairing with 254-bit prime order Barreto and Naehrig curve.

At CRYPTO 2016, Kim and Barbulescu proposed an efficient number field sieve(NSF) algorithm for the discrete logarithm problem in a finite field[11]. The security of pairing-based cryptography is based on the difficulty in solving the DLP. Hence, it has become necessary to shift the parameters that the DLP is computationally infeasible against the efficient NSF algorithms.

This memo introduce Optimal Ate Pairing [2] for two PFCs to be able to shift the parameters. 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 BLS-curves Section 3 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)

1.1. 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].

2. Preliminaries

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

2.1. 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 an 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

2.2. Bilinear Map

Let G_1 be an additive group of prime order r and let G_2 and G_3 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_3) are asymmetric bilinear map groups if there exists a bilinear map e: (G_1, G_2) -> G_3 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.

3. Pairing-friend curves and Domain Parameter Specification

In this section, this memo specifies the domain parameters for 128-bit and 256-bit secure elliptic curves which allow us to efficiently compute the operation of a pairing at appropriate levels of security.

We introduce 462-bit Barreto Naehrig (BN462)[13] curve as 128-bit secure elliptic curve and 581-bit Barreto Lynn Scott (BLS48)[12]. curve as 256-bit secure elliptic curve.

3.1. Notation for Domain Parameters

Here, we define notations for specifying domain parameters and explain types of pairing friendly curves.

The elliptic curves E over F_p satisfy following equation.

y^2 = x^3 + B for B in F_p

These domain parameters are described in the following way.

For the elliptic curve E(F_p) on BN462

G1-Curve-ID is an identifier of the G_1 curve with which the curve can be referenced.

p_b is a prime specifying a base field F_p.

B is the coefficient of the equation y^2 = x^3 + B mod p defining E.

G = (x, y) is the base point, i.e., a point with x and y being its x- and y-coordinates in E, respectively.

r is the prime order of the group generated by G.

h is the cofactor of G in E(F_p)

For twisted curve E'(F_{p^2}) on BN462

G2-Curve-ID is an identifier of the G_2 curve with which the curve can be referenced.

p_b is a prime specifying a base field.

e2 is the constant of an irreducible polynomial specifying extension field $F_{p^2} = F_p[u]/(u^2 - e2)$.

B' is the coefficient of the equation $y'^2 = x'^3 + B' \bmod F_{p^2}$ defining E'.

G' = (x', y') is the base point, i.e., a point with x' and y' being its x'- and y'-coordinates in E', respectively.

r' is the prime order of the group generated by G'.

h' is the cofactor of r' in $\#E'(F_{p^2})$

For $F_{p^{12}}^*$ on BN462

G3-Field-ID is an identifier of the $F_{p^{12}}^*$.

p_b is a prime specifying base field.

r'' is the prime order of the group.

e2 is the constant of the irreducible polynomial of $F_{p^2} = F_p[u]/(u^2 - e2)$.

e6 is the constant of the irreducible polynomial of $F_{p^6} = F_{p^2}[v]/(v^3 - e6)$.

e12 is the constant of the irreducible polynomial of $F_{p^{12}} = F_{p^6}[w]/(w^2 - e12)$.

h'' is the cofactor of r in $F_{p^{12}}^*$ s.t. $h'' = h''_1 * h''_2$

h''_1 is the part of cofactor of r in $F_{p^{12}}^*$ s.t. $h''_1 = (p^4 - p^2 + 1)/r$

h''_2 is the part of cofactor of r in $F_{p^{12}}^*$ s.t. $h''_2 = (p^6 - 1) * (p^2 + 1)$

For the elliptic curve $E(F_p)$ on BLS48

G1-Curve-ID is an identifier of the G_1 curve with which the curve can be referenced.

p_b is a prime specifying a base field F_p.

B is the coefficient of the equation $y^2 = x^3 + B \bmod p$ defining E.

G = (x, y) is the base point, i.e., a point with x and y being its x- and y-coordinates in E, respectively.

r is the prime order of the group generated by G.

h is the cofactor of G in E(F_p)

For twisted curve E'(F_{p^8}) on BLS48

G2-Curve-ID is an identifier of the G_2 curve with which the curve can be referenced.

p_b' is a prime specifying a base field.

B' is the coefficient of the equation y'^2 = x'^3 + B' mod F_p^2 defining E'.

G' = (x', y') is the base point, i.e., a point with x' and y' being its x'- and y'-coordinates in E', respectively.

r' is the prime order of the group generated by G'.

h' is the cofactor of r' in #E'(F_{p^8}).

For F_{p^48}^* on BLS48

G3-Field-ID is an identifier of the F_{p^48}^*.

p_b is a prime specifying a base field.

r'' is the prime order of the group generated by G'.

h'' is the cofactor of r' in #E'(F_{p^48}).

For the definition of the pairing parameter

Pairing-Param-ID is the set of the identifiers G1-Curve-ID, G2-Curve-ID and G3-Field-ID.

3.2. Efficient Domain Parameters for BN462

This section specifies the domain parameters for four 128-bit secure elliptic curves BN462.

3.2.1. Domain Parameters by Beuchat et al.

The domain parameters described in this subsection are defined by elliptic curve E(F_p) : y^2 = x^3 + 5 and sextic twist E'(F_{p^2}) : x'^3 + 5/s = x'^3 + 2 - u, where F_{p^2} = F_p[u]/(u^2 + 1), F_{p^6} = F_{p^2}[v]/(v^3 - u), F_{p^12} = F_{p^6}[w]/(w^2 - v), s = - 5/u. We describe domain parameters of elliptic curves E and E'. The parameter p_b is 1 mod 8.

G1-Curve-ID: Fp462nBN

p_b = 0x240480360120023fffffffff6ff0cf6b7d9bfca000000
0000d812908f41c8020ffffffffff6ff66fc6ff687f640000
000002401b00840138013

x = 0x21a6d67ef250191fadba34a0a30160b9ac9264b6f95f
63b3edbec3cf4b2e689db1bbb4e69a416a0b1e79239c037
2e5cd70113c98d91f36b6980d

y = 0x0118ea0460f7f7abb82b33676a7432a490eeda842ccc
fa7d788c659650426e6af77df11b8ae40eb80f475432c666
00622ecaa8a5734d36fb03de;

r = 0x240480360120023FFFFFFFFF6FF0CF6B7D9BFCA0000000000
D812908F41C8020FFFFFFFFFF6FF66FC6FF687F64000000000
2401B00840138013

h = 1

G2-Curve-ID: Fp462n2BN

p_b = 0x2370fb049d410fbe4e761a9886e502417d023f40180000017e80600000
000001

e2 = -5 in F_p

B' = 2 - u

x' = 0x1d2e4343e8599102af8edca849566ba3c98e2a354730cb
ed9176884058b18134dd86bae555b783718f50af8b59bf7e
850e9b73108ba6aa8cd283*u +
0x257ccc85b58dda0dfb38e3a8cbdc5482e0337e7c1cd9
6ed61c913820408208f9ad2699bad92e0032ae1f0aa6a8b4
8807695468e3d934ae1e4df

y' = 0x73ef0cbd438cbe0172c8ae37306324d44d5e6b0c69ac5
7b393f1ab370fd725cc647692444a04ef87387aa68d53743
493b9eba14cc552ca2a93a*u +

```
0xa0650439da22c1979517427a20809eca035634706e23c3f
a7a6bb42fe810f1399a1f41c9ddae32e03695a140e7b11d
7c3376e5b68df0db7154e

r' = r

h' = 0x240480360120023fffffffff6ff0cf6b7d9bfca0000000000
d812908fa1ce0227fffffffff6ff66fc63f5f7f4c000000000
2401b008a0168019

G3-Field-ID: Fp462n12

p_b = 0x2370fb049d410fbe4e761a9886e502417d023f40180000017e80600000
000001

r'' = r

e2 = -5 in F_p

e6 = u in F_{p^2}

e12 = v in F_{p^6}

h'' = (TBD)

h''_1 = (TBD)

h''_2 = (TBD)

Pairing-Param-ID: Fp462BN = {
    G1-Curve-ID: Fp462nBN
    G2-Curve-ID: Fp462n2BN
    G3-Field-ID: Fp462n12BN
}
```

3.3. Efficient Domain Parameters for BLS48

This section specifies the domain parameters for four 256-bit secure elliptic curves BLS48.

3.3.1. Domain Parameters by Kiyomura et al.

The domain parameters described in this subsection are defined by elliptic curve E(F_p) : y^2 = x^3 + 1 and following tower structures E'(F_{p^8}) : y^2 = x^3 -1/w, where F_{p^2} = F_p[u]/(u^2 + 1), F_{p^4} = F_{p^2}[v]/(v^2 + u + 1), F_{p^8} = F_{p^4}[w]/(w^2 + v), F_{p^24} = F_{p^8}[z]/(z^3 + w), f_{p^48} = f_{p^24}[s]/(s^2 + z).

The polynomials p(x), r(x) and t(x) satisfies CM equation D = 3, n(x) = p(x) + 1 - t(X), 4p(x) - t(x)^2 = Df(x)^2.

p(x) = (x - 1)^2(x^16 - x^8 + 1)/3 + x.

r(x) = x^16 - x^8 + 1.

t(x) = x + 1.

x_0 is specific parameter in CM equation.

x_0 = -1 + 2^7 - 2^20 - 2^30 - 2^32.

G1-Curve-ID: Fp581nBLS48

p_b =
0x1280f73ff3476f313824e31d47012a0056e84f8d122131bb3be6c0f1f3975444
a48ae43af6e082acd9cd30394f4736daf68367a5513170ee0a578fdf721a4a48ac
3ed c154e6565912b

x = 0x05

y = 0x491acfa2307425af23c3444bb9f7c38b86fe62a4105f1a06bac418fb4244
afb7b6b932b9a4a3c048637613a50e88b86e9e37a154f077398b0d26f51ce737e
2e1e768d5b0dc461d83a

r = 0x2386f8a925e2885e233a9ccc1615c0d6c635387a3f0b3cbe003fad6bc972
c2e6e741969d34c4c92016a85c7cd0562303c4ccbe599467c24da118a5fe6fcd6
71c01

h = 0x85555841aaaec4ac

G2-Curve-ID: Fp581n8BLS48

p_b' = 0x2386f8a925e2885e233a9ccc1615c0d6c635387a3f0b3cbe00
3fad6bc972c2e6e741969d34c4c92016a85c7cd0562303c4cc
be599467c24da118a5fe6fcd671c01

x' = 0x827d5c22fb2bdec5282624c4f4aaa2b1e5d7a9defaf47b5211cf
741719728a7f9f8cfca93f29cff364a7190b7e2b0d4585479bd6ae
bf9fc44e56af2fc9e97c3f84e19da00fbc6ae34*u*v*w + b9b795
1c6061ee3f0197a498908aee660dea41b39d13852b6db908ba2c0b
7a449cef11f293b13ced0fd0caa5efcf3432aad1cbe4324c22d633
34b5b0e205c3354e41607e60750e057*v*w + c96c7797eb073860
3f1311e4ecda088f7b8f35dcef0977a3d1a58677bb037418181df6
3835d28997eb57b40b9c0b15dd7595a9f177612f097fc7960910fc
e3370f2004d914a3c093a*u*w + 38b91c600b35913a3c598e4caa
9dd63007c675d0b1642b5675ff0e7c5805386699981f9e48199d5a

```
c10b2ef492ae589274fad55fc1889aa80c65b5f746c9d4cbb739c3
a1c53f8cce5*w + be2218c25ceb6185c78d8012954d4bfe8f5985
ac62f3e5821b7b92a393f8be0cc218a95f63e1c776e6ec143b1b27
9b9468c31c5257c200ca52310b8cb4e80bc3f09a7033cbb7feafe *u*v +
1fccc70198f1334e1b2ea1853ad83bc73a8a6ca9ae237ca
7a6d6957ccbab5ab6860161c1dbd19242ffae766f0d2a6d55f028c
bdfbb879d5fea8ef4cded6b3f0b46488156ca55a3e6a*v + 7c497
3ece2258512069b0e86abc07e8b22bb6d980e1623e9526f6da1230
7f4e1c3943a00abfedf16214a76affa62504f0c3c7630d979630ff
d75556a01afa143f1669b36676b47c57*u + 5d615d9a7871e4a38
237fa45a2775debabbefc70344dbccb7de64db3a2ef156c46ff79b
aad1a8c42281a63ca0612f400503004d80491f510317b797663221
54dec34fd0b4ace8bfab

y' = 0x35e2524ff89029d393a5c07e84f981b5e068f1406be8e50c8754
9b6ef8eca9a9533a3f8e69c31e97e1ad0333ec719205417300d8c4
ab33f748e5ac66e84069c55d667ffcb732718b6*u*v*w + 896767
811be65ea25c2d05dfdd17af8a006f364fc0841b064155f14e4c81
9a6df98f425ae3a2864f22c1fab8c74b2618b5bb40fa639f53dccc
9e884017d9aa62b3d41faeafeb23986*v*w + 7d0d03745736b7a5
13d339d5ad537b90421ad66eb16722b589d82e2055ab7504fa8342
0e8c270841f6824f47c180d139e3aafc198caa72b679da59ed8226
cf3a594eedc58cf90bee4*u*w + d209d5a223a9c46916503fa5a8
8325a2554dc541b43dd93b5a959805f1129857ed85c77fa238cdce
8a1e2ca4e512b64f59f430135945d137b08857fdddfcf7a43f4783
1f982e50137*w + aec25a4621edc0688223fbbd478762b1c2cded
3360dcee23dd8b0e710e122d2742c89b224333fa40dced28177427
70ba10d67bda503ee5e578fb3d8b8a1e5337316213da92841589d *u*v +
b36a201dd008523e421efb70367669ef2c2fc5030216d5b
119d3a480d370514475f7d5c99d0e90411515536ca3295e5e2f0c1
d35d51a652269cbc7c46fc3b8fde68332a526a2a8474*v + 284dc
75979e0ff144da6531815fcadc2b75a422ba325e6fba01d7296473
2fcbf3afb096b243b1f192c5c3d1892ab24e1dd212fa097d760e2e
588b423525ffc7b111471db936cd5665*u + eb53356c375b5dfa4
97216452f3024b918b4238059a577e6f3b39ebfc435faab0906235
afa27748d90f7336d8ae5163c1599abf77eea6d659045012ab12c0
ff323edd3fe4d2d7971

r' = r

h' = 0x170e915cb0a6b7406b8d94042317f811d6bc3fc6e211ada42e
58ccfcb3ac076a7e4499d700a0c23dc4b0c078f92def8c87b7
fe63e1eea270db353a4ef4d38b5998ad8f0d042ea24c8f02be
1c0c83992fe5d7725227bb27123a949e0876c0a8ce0a67326d
b0e955dcb791b867f31d6bfa62fbdd5f44a00504df04e186fa
e033f1eb43c1b1a08b6e086eff03c8fee9ebdd1e191a8a4b04
66c90b389987de5637d5dd13dab33196bd2e5afa6cd19cf0fc
3fc7db7ece1f3fac742626b1b02fcee04043b2ea96492f6afa
```

51739597c54bb78aa6b0b99319fef9d09f768831018ee6564c
68d054c62f2e0b4549426fec24ab26957a669dba2a2b6945ce
40c9aec6afdeda16c79e15546cd7771fa544d5364236690ea0
6832679562a68731420ae52d0d35a90b8d10b688e31b6aee45
f45b7a5083c71732105852decc888f64839a4de33b99521f09
84a418d20fc7b0609530e454f0696fa2a8075ac01cc8ae3869
e8d0fe1f3788ffac4c01aa2720e431da333c83d9663bfb1fb7
a1a7b90528482c6be7892299030bb51a51dc7e91e915687441
6bf4c26f1ea7ec578058563960ef92bbbb8632d3a1b695f954
af10e9a78e40acffc13b06540aae9da5287fc4429485d44e62
89d8c0d6a3eb2ece35012452751839fb48bc14b515478e2ff4
12d930ac20307561f3a5c998e6bcbfebd97effc6433033a236
1bfcdc4fc74ad379a16c6dea49c209b1

G3-Field-ID: Fp581n48BLS48

p_b = (TBD)

r'' =
0x33325e51b8485cacb1cf7e79521a2c07ed618593bc2b823693827cddd501 412
8f299770734d9658ec3da72c829e2bfdfa5bc0dcac0f0dd385da0b70a1f1800f3
920706ac684379b30abec1422f3428ce3b9ea1d92e0995ded30bb3f127dd47570d
12 1a8200c4091f4c1a039e4dea3f3e733d60d4788b3a2db1954fa31287ef5b2f8
d31c9 b5f5107074dc917ffa3ebef388907b3e2400a0108fb4b983592be1718c4a
206f401d 2fd25126d86f05bd447da88d3240e4ebfd3c06ffacd5a6035b8599108
3e27f7ec56e 001b64a11949e1c61fca24a0634794818600eebf30801a216d1dc7
e2ac05b743e9bd 89e033b09757a9e3f9dd4bcdfd8c7ca6e2b5c39833111583a14
a800b430ae5ea8a3c 6ab3ad627523d1e7dedcf79a56483a81cf6a6deb7505ed45
dc8a3d557237ef0f98ac 7ca9e577d5c7d429fdbdc87a0a0b056dd44b9c8ae1432
ac96dde432512ea1782c476 727732b7ace3a30d90fd4ad586edd8ee2b5b10e3cf
f6cc31e9137f98d3debad7ca51 2af4f876915edb46c3d5d51c4c3c7e268727ab9
14ae89f05c7a7f9fa1df8ee053622 b60033bc7970f902d6a9ebc1b6ff316d5457
cdbd926cf183a6114ae6448650286067 ababbd5d747a5117b691e1e7138f2e4f8
d8025df47f695681f0555463005c211ac9c 52c56b7c96d4dbc30e86bcb3c7013d
e1913fa60e2e58f1877fa6bd690f7f37858d69 9dcc083c27cd1837efb00d0bdda
265e73adca2760f99d911463fa51614aaf308a54f 46a15f08ad24c378210c60aa
64ff1772ec3d6d84fcaadd697aef4f87423b215d4ab9 aee8f260865b1

h'' =
0x170e915cb0a6b7406b8d94042317f811d6bc3fc6e211ada42e58ccfcb3ac 076
a7e4499d700a0c23dc4b0c078f92def8c87b7fe63e1eea270db353a4ef4d38b59
98ad8f0d042ea24c8f02be1c0c83992fe5d7725227bb27123a949e0876c0a8ce0a
67 326db0e955dcb791b867f31d6bfa62fbdd5f44a00504df04e186fae033f1eb4
3c1b1 a08b6e086eff03c8fee9ebdd1e191a8a4b0466c90b389987de5637d5dd13
dab33196 bd2e5afa6cd19cf0fc3fc7db7ece1f3fac742626b1b02fcee04043b2e
a96492f6afa 51739597c54bb78aa6b0b99319fef9d09f768831018ee6564c68d0
54c62f2e0b4549 426fec24ab26957a669dba2a2b6945ce40c9aec6afdeda16c79
e15546cd7771fa544 d5364236690ea06832679562a68731420ae52d0d35a90b8d

```
10b688e31b6aee45f45b 7a5083c71732105852decc888f64839a4de33b99521f0
984a418d20fc7b0609530e4 54f0696fa2a8075ac01cc8ae3869e8d0fe1f3788ff
ac4c01aa2720e431da333c83d9 663bfb1fb7a1a7b90528482c6be7892299030bb
51a51dc7e91e9156874416bf4c26f 1ea7ec578058563960ef92bbbb8632d3a1b6
95f954af10e9a78e40acffc13b06540a ae9da5287fc4429485d44e6289d8c0d6a
3eb2ece35012452751839fb48bc14b51547 8e2ff412d930ac20307561f3a5c998
e6bcbfebd97effc6433033a2361bfcdc4fc74a d379a16c6dea49c209b1

Pairing-Param-ID: Fp581BLS48 = {
    G1-Curve-ID: Fp581BLS48n
    G2-Curve-ID: Fp581BLS48n8
    G3-Field-ID: Fp581BLS48n48
}
```

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 the 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_3

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 choose 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-th 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
- A point Q in G_2
- An integer s

Output:

- 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. Algorithm Identifiers

(TBD)

6. Security Considerations

The pairing is a map from two elliptic curves G1 and G2 to a multiplicative subgroup of a finite field.

Typically, G1 (respectively G2) is a cyclic subgroup in E(F_p) (respectively E'(F_{p^{k/d})) of prime order r, where k is the embedding degree and d is the degree of the twist. The group G3 is a set of r-th roots of unity in F_{p^k}^*. In this section, G_1', G_2' and G_3' denote E(F_p), E'(F_{p^{k/d}}) and F_{p^k}^* respectively.

Pairing-based cryptographic primitives are often based on the hardness of the following problems.

The elliptic curve discrete logarithm problem in G_1' and G_2' (ECDLP)

The finite field discrete logarithm problem in G_3' (FFDLP)

The elliptic curve computational Diffie-Hellman (CDH) problem in G_1' and G_2'

The elliptic curve computational co-Diffie-Hellman problem in G_1' and G_2'

The elliptic curve decisional Diffie-Hellman (DDH) problem in G_1'

The bilinear Diffie-Hellman (BDH) problem

On the side of G1' and G2', the best known algorithm (for instance, Pollard-rho algorithm [10]) to solve ECDLP has a running time of O(sqrt(r)), which is exponential in r, where r is the order of the target group. Thus, for a security parameter l, one should set r so that log_2(r)>2*l.

On the side of G3', one has more efficient algorithm based on Number Field Sieve methods (cite Gordon93 paper here, if you want). The complexity of the NFS (including its variants) is subexponential in the size of the finite field and is independent from the size of the subgroup order r. Recall the classical L-notation, L(q, a, c)= exp((c+o(1))*log(q)^a * log(log(q))^(1-a)). Before 2016, the best known algorithm for FFDLP has had a running time of L(p^k, 1/3, 1.92) and the parameters of currently used pairings are derived from the above value. At CRYPTO2016, however, Kim and Barbulescu proposed a new

variant of the NFS method that drastically reduces the complexity of solving FFDLP from L(p^k, 1/3, 1.92) to L(p^k, 1/3, 1.53) in the best case[11]. For instance, Barbulescu estimates that the security of the pairing over BN curve, which was believed to have 128-bit security, drops to approximately 100-bit security. Hence, it has become necessary to revise the bit length of q=p^k so that the FFDLP over F_{p^k} is computationally infeasible against this new efficient NFS algorithm.

G_3' to be larger than G_1' and G_2', because FFDLP can be computed more efficiently than ECDLP in most cases. Security level of schemes based on pairing depends most weak level for each problems. Thus implementers should necessary to ensure adequate security level for both of problems.

6.1. 128-bit Secure PFC

Barreto and Naehrig showed how to construct pairing-friendly curves[13]. We have chosen 462-bit prime order curve and described in Section 3.

6.2. 256-bit Secure PFC

Five 256-bit secure domain parameters have been proposed by Kiyomura[12]. We have chosen the BLS48 as 256-bit secure PFC and described in Section 3.

7. Acknowledgements

(TBD)

8. Change log

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

9. References

9.1. 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.

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

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

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

[10] Pollard, J., "Monte Carlo Methods for Index Computation (mod p)", Proceedings Mathematics of Computation, Vol.32, 1978.

[11] Kim, T. and R. Barbulescu, "Extended tower number field sieve: a new complexity for the medium prime case.", CRYPTO 2016 LNCS, vol. 9814, pp. 543.571, 2016.

[12] Kiyomura, Y., Inoue, A., Kawahara, Y., Yasuda, M., Takagi, T., and T. Kobayashi, "Secure and Efficient Pairing at 256-Bit Security Level", ACNS 2017 LNCS, vol. 10355, pp. 59.79, 2017, 2017.

[13] Barreto, Paulo. and Michael. Naehrig, "Pairing-Friendly Elliptic Curves of Prime Order", Selected Areas in Cryptography-SAC 2005. volume 3897 of Lecture Notes in Computer Science, pages 319-331, 2006.

[14] "ISO/IEC 14888-3:2016", ISO/IEC Information technology -- Security techniques -- Digital signatures with appendix -- Part 3: Discrete logarithm based mechanisms, 2016.

Appendix A. Test Vectors of Optimal Ate Pairing

In this section, we specify test vectors of Optimal Ate Pairing over BN-curve and BLS-curve which are specified by Section 3 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_3

(TBD)

Authors' Addresses

Akihiro Kato
NTT TechnoCross Corporation

EMail: kato.akihiro-at-po.ntt-tx.co.jp

Tetsutaro Kobayashi
NTT

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

Yuto Kawahara
NTT

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

Teachan Kim
NTT

EMail: taechan.kim-at-lab.ntt.co.jp

Tsunekazu Saito
NTT

EMail: saito.tsunekazu-at-lab.ntt.co.jp