Internet Draft   W. Ladd
<draft-ladd-safecurves-02.txt>   Grad Student
Category: Informational   UC Berkeley
Expires 14 July 2014   10 January 2014

Additional Elliptic Curves for IETF protocols
<draft-ladd-safecurves-02.txt>

Abstract

This Internet draft contains curves whose Jacobians are groups over

which the Decisional Diffie-Hellman problem is hard, and which have implementation advantages.

Table of Contents

1. Introduction

This document contains a set of elliptic curves over prime fields with many security and performance advantages. They are twist-secure, have large prime order subgroups, high embedding degree, endomorphism rings of large discriminant, complete formulas, and primes for fast arithmetic.

These curves have been generated in a rigid manner by computer search. As such there is very little risk that these curves were selected to exhibit weaknesses to attacks not in the open literature. The field is the only free choice, and in all circumstances has been picked to enable highly efficient arithmetic. Proofs of all properties claimed exist in [SAFECURVES]. It is easier to avoid known implementation issues with these curves then short Weierstrauss curves.

2. The Curves

Each curve is given by an equation and a basepoint, together with an order of the point and cofactor. All curves are elliptic. Validation information is given at [SAFECURVES]. The names given in this document indicate the family. The basepoint is given as an (x,y) ordered pair.

Curve25519 is a curve over GF(2^255-19), formula y^2=x^3+486662x^2+x, basepoint (9, 14781619447589544791020593568409986887264606134616475288964881837755586237401), order 2^252 + 27742317777372353535851937790883648493, cofactor 8.

E382 is a curve over GF(2^382-105), formula x^2+y^2=1-67254x^2y^2, basepoint (3914921414754292646847594472454013487047137431784830634731377862923477302047857640522480241298429278603678181725699, 17), order 2^380 - 1030303207694556153926491950732314247062623204330168346855, cofactor 4.

M383 is a curve over GF(2^383-187), formula y^2=x^3+2065150x^2+x, basepoint (12,
4737623401891753997660546300375902576839617167257703725630 38

97915244635657572992031549016554320965586421172429064 94), order 2^380 + 16623627593137351610521979493554215330803923445576161 3271, cofactor 8.

Curve3617 is a curve over GF(2^414-17), formula x^2+y^2=1+3617x^2y^2, basepoint
(17319886477121189177719202498822615443556957307604340815256226
171904769976866975908866528699294134494857887698432266169206165, 34),
order 2^411 -
33364140863755142520810177694098385178984727200411208589594759,
cofactor 8.

M511 is a curve over GF(2^511-187), formula y^2 = x^3+530438x^2+x, basepoint (5,
2500410645565072423368981149139213252211568685173608590070979264 2
48275228603899706950518127817176591878667784247582124505430745177
116625808811349787373477), order 2^508 +
10724754759635747624044531514068121842070756627434833028965540808827675062043, cofactor 8.

E521 is a curve over GF(2^521-1), formula x^2+y^2=1-376014x^2y^2, basepoint
(1571054894184995387535939749894317568645297350402905821437625
181152304994381188529632591196067604100772673927915114267193389 90
500327667374901205114835604132 4, 12), order 2^519 -
337554763258501705789107630418782636071904961214051226618635150085779108655765, cofactor 4.

3. Explicit Formulas

On Montgomery curves, curves of the form y^2=x^3+Ax^2+x, the typical technique is to work over the Kummer curve instead, i.e. drop y coordinates for use in Diffie-Hellman. Let (X_1,Z_1), (X_2,Z_2), (X_3,Z_3) be coordinates such that X_i/Z_i is the x-coordinate of P_i, with P_i=[i]P_1 on the curve. Then

```
      X5 = Z1*((X3-Z3)*(X2+Z2)+(X3+Z3)*(X2-Z2))2
      Z5 = X1*((X3-Z3)*(X2+Z2)-(X3+Z3)*(X2-Z2))2
      X4 = (X2+Z2)2*(X2-Z2)2
      Z4 = (4*X2*Z2)*((X2-Z2)2+a24*(4*X2*Z2))
```

gives X_i/Z_i as the x coordinate of P_i for i in {4,5} where a24*4=A+2

On Edwards curves, curves of the form, x^2+y^2=1+dx^2y^2 a complete addition formula, which works for doubling as well, is given by representing points as x=Z/X, y=Z/Y. The formula for adding (X_1, Y_1, Z_1) to (X_2, Y_2, Z_2) yielding (X_3, Y_3, Z_3) is then

```
      A = Z1*Z2
```

```
B = d*A2
C = X1*X2
D = Y1*Y2
E = C*D
H = C-D
I = (X1+Y1)*(X2+Y2)-C-D
X3 = c*(E+B)*H
Y3 = c*(E-B)*I
Z3 = A*H*I
```

These formulas are from the [EFD].

Using these formulas the standard double-and-add or Montgomery ladder recurrence can be used to compute multiples of points.

The Montgomery curve formulas require only the x coordinate. Protocols based on ECDH should give strong consideration to transmitting only the x coordinate, in which case no validation is required. The above addition formulas cannot be used to add points on Montgomery curves, as they ignore the y coordinate entirely.

It is highly recommended that Edwards curve points are transmitted in compressed form to avoid implementations with missing curve membership checks from working. The canonical compression is the y coordinate, followed by an indicator of the low bit of the x coordinate. Formulas for decompression are left as an exercise to the reader.

4. Point Encoding

Let (x,y) be a GF_p point on M(GF_p), where M is a Montgomery curve. Then let l=8*ceil[log(p)/log(256)]. A point is represented as l-bytes, representing in big-endian radix 256 the minimal representative of [x] modulo p. This representation works for the standard x-coordinate only arithmetic for ECDH, but cannot be used for protocols requiring addition.

Let (x,y) be a GF_p point on E(GF_p), where E is an Edwards Curve. Let l=ceil[log(p)/log(256)]. A point is represented as l bytes, l representing in big-endian radix 256 the minimal representative of [x] modulo p, and the top bit of the top byte set to equal the low bit of x. Note that as the primes of these curves are all slightly lower than a power of two, this top bit is never required for the minimal representative, and so can indicate the parity of x. This representation is injective from points.

Alternative encodings are used by existing software, and protocol designers should be aware of this.

5. Security Considerations

This entire document discusses methods of implementing cryptography securely. The time for an attacker to break the DLP on these curves is the square root of the group order with the best known attacks. These curves are twist-secure, avoiding the need for some checks in some protocols.

It is recommended that implementors use the Montgomery ladder on Montgomery curves with x coordinate only to avoid side-channel attacks when Diffie-Hellman is being used. In this mode, curve checks are not required. Otherwise standard curve (but not group) membership checks are required for ECDH to be secure.

These curves are complete, avoiding certain attacks against naive implementations of ECC protocols. They have cofactor greater than one, occasionally requiring slight adjustments to protocols.

This is not an exhaustive discussion of security considerations relating to the implementation of these curves. Implementors must be familiar with cryptography to safely implement any cryptographic standard, and this standard is no exception.

6. IANA Considerations

IANA should maintain a registry of these curves, calling them chicagocurve-XXXX where XXXX is the curve identifier.

7. References

[SAFECURVES] safecurves.cr.yp.to

[EFD] http://www.hyperelliptic.org/EFD/g1p/index.html

Author's Address
Watson Ladd
watsonbladd@gmail.com
Berkeley, CA