Network Working Group
Internet-Draft
Intended status: Informational
Expires: August 21, 2008

M. Lochter
Bundesamt fuer Sicherheit in der
Informationstechnik (BSI)
J. Merkle
secunet Security Networks

February 18, 2008

ECC Brainpool Standard Curves and Curve Generation
draft-lochter-pkix-brainpool-ecc-01

Status of This Memo

Copyright Notice

Abstract

This Memo proposes several elliptic curve domain parameters over finite prime fields for use in cryptographic applications. The domain parameters are consistent with the relevant international

standards, and can be used in X.509 certificates and certificate revocation lists (CRLs), for Internet Key Exchange (IKE), Transport Layer Security (TLS), XML signatures, and all applications or protocols based on the cryptographic message syntax (CMS).

Table of Contents

1. Introduction

Although several standards for elliptic curves and domain parameters exist (e.g. [ANSI1], [FIPS] or [SEC2]), some major issues have still not been addressed:

- The generation of the prime p and the seed from which the curve parameters were derived is irreproducible, leaving out an essential part of the security analysis.

- No proofs are provided that the proposed parameters do not belong to those classes of parameters which are susceptible to cryptanalytic attacks with sub-exponential complexity.

- Recent research results seem to indicate a potential for new attacks on elliptic curve cryptosystems. At least for applications with highest security demands or under circumstances which complicate a change of parameters in response to new attacks, the inclusion of a corresponding security requirement for domain parameters (the class group condition, see Section 2) is justified.

- Some of the proposed subgroups have a non-trivial cofactor, which demands additional checks by cryptographic applications to prevent small subgroup attacks (see [ANSI1] or [SEC1]).

- The domain parameters specified do not cover all bit lengths that correspond to the commonly used key lengths for symmetric cryptographic algorithms. In particular, there is no 512 bit curve defined but only one with 521 bit length, which may be disadvantageous for some implementations.

Furthermore, many of the parameters specified by the existing standards are identical (see [SEC2] for a comparison). Thus, there is still a need for additional elliptic curve domain parameters which overcome the above limitations.

1.1. Scope and Relation to Other Specifications

This RFC specifies elliptic curve domain parameters over prime fields GF(p) with p having a length of 160, 192, 224, 256, 320, 384 and 512 bits. These parameters were generated in a pseudo-random yet completely systematic and reproducible way and have been verified to resist current cryptanalytic approaches. The parameters are compliant with ANSI X9.62 [ANSI1] and ANSI X9.63 [ANSI2], ISO/IEC 14888 [ISO1] and ISO/IEC 15946 [ISO2], ETSI TS 102 176-1 [ETSI], as well as with FIPS-186-2 [FIPS], the SECG specifications ([SEC1] and [SEC2]).

Furthermore, this document identifies and explains the requirements for the parameters that have led to the methods for the generation and security validation of the parameters. Complementing information, including the pseudo-random generation methods for the parameters and the security proofs, are given in [EBP].

Finally, this RFC defines ASN.1 object identifiers for all elliptic curve domain parameter sets specified herein, e.g. for use in X.509 certificates.

This document does neither address the cryptographic algorithms to be used with the specified parameters nor their application in other standards. However, it is consistent with the following RFCs and internet drafts which specify the usage of elliptic curve cryptography in protocols and applications:

- [RFC3278] for the cryptographic message syntax (CMS)
- [RFC3279] and [PKIX] for X.509 certificates and CRLs
- [RFC4050] for XML signatures
- [RFC4492] for TLS
- [RFC4754] for IKE

1.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Requirements on the Elliptic Curve Domain Parameters

Throughout this memo let p > 3 be a prime and GF(p) a finite field (sometimes also referred to as Galois Field or F_p) with p elements. For given A and B with non-zero 4*A^3 + 27*B^2 mod p, the set of solutions (x,y) for the equation E: y^2 = x^3 + A*x + B mod p over GF(p) together with a neutral element O and well-defined laws for addition and inversion define a group - the elliptic curve E(GF(p)). Typically, for cryptographic applications, an element G of prime order q is chosen in E.

A comprehensive introduction to elliptic curves and their cryptographic applications can be found in [BSS].

Note 1: We choose {0,...,p-1} as a set of representatives for the elements of GF(p). This choice induces a natural ordering on GF(p).

2.1. Security Requirements

The following security requirements are either motivated by known cryptographic analysis or aim to enhance trust in the recommended curves.

1. Immunity to attacks using the Weil- or Tate-Pairing. These attacks allow the embedding of the cyclic subgroup generated by G into the group of units of a degree-l extension GF(p^l) of GF(p), where sub-exponential attacks on the discrete logarithm problem (DLP) exist. Here we have l = min{t | q divides p^t-1}, i.e. l is the order of p mod q. By Fermat's little theorem, l divides q-1. We require (q-1)/l < 100, which means that l is close to the maximum possible value. This requirement is considerably stronger than those of [SEC2] and [ANSI2] and also excludes supersingular curves, as those are the curves of order p+1. Detailed information on this requirement can be found in [BSS].

2. The trace is not equal to one. Trace one curves (or anomalous curves) are curves with #E(GF(p)) = p. Satoh and Araki [SA], Semaev [Sem] and Smart [Sma] independently proposed efficient solutions to the elliptic curve discrete logarithm problem (ECDLP) on trace one curves. Note that these curves are also excluded by requirement 5 of section 2.2.

3. Large class number. The class number of the maximal order of the endomorphism ring End(E) of E is larger than 10^7. Generally, E cannot be "lifted" to a curve E' over an algebraic number field L with End(E) = End(E') unless the degree of L over the rationals is larger than the class number of End(E). Although there are no efficient attacks exploiting a small class number, recent work ([JMV] and [HR]) also may be seen as argument for the class number condition. (See [EBP] for more details on class group computations.) This condition excludes curves that are generated by the well-known CM-method.

4. Prime group order. The group order #E(GF(p)) shall be a prime number in order to counter small-subgroup attacks ([HMV]). Therefore, all groups proposed in this RFC have cofactor 1. Note that curves with prime order have no point of order 2 and therefore no point with y-coordinate 0.

5. Verifiably pseudo-random. The elliptic curve domain parameters shall be generated in a pseudo-random manner using seeds that are generated in a systematic and comprehensive way. Our method of construction is explained in [EBP].

6. Proof of security. For all curves a proof should be given that all security requirements are met. These proofs are provided in [EBP].

In [BG], attacks are described which apply to elliptic curve domain parameters where q-1 has a factor u in the order of q^(1/3). However, the circumstances under which these attacks are applicable can be avoided in most applications. Therefore, no corresponding security requirement is stated here. However, it is highly recommended that developers verify the security of their implementations against this kind of attack.

2.2. Technical Requirements

Commercial demands and experience with existing implementations lead to the following technical requirements for the elliptic curve domain parameters.

1. For each of the bit lengths 160, 192, 224, 256, 320, 384 and 512 one curve shall be proposed. This requirement follows from the need for curves providing different levels of security which are appropriate for the underlying symmetric algorithms. The existing standards specify a 521-bit curve instead of a 512-bit curve.

2. The prime number p shall be congruent 3 mod 4. This requirement allows efficient point compression: One method for the transmission of curve points P=(x,y) is to transmit only x and the least significant bit LSB(y) of y. For p = 3 mod 4 we get (y^2)^(p+1)/4 = y*y^(p-1)/2 which is either y or -y by Fermat's Little Theorem, and hence y can be computed very efficiently using the curve equation.
   This requirement is not always met by the parameters defined in existing standards.

3. The curves shall be GF(p)-isomorphic to a "cryptographically good curve" (i.e. a curve that meets all security requirements defined in Section 2.1) with A = -3 mod p. This property permits the use of the arithmetical advantages of curves with A = -3 mod p as shown by Brier and Joyce [BJ]. The requirement is fulfilled by a quadratic twist E' of the given curve E with a square in GF(p): If -3 = A*Z^4 mod p is solvable, then E and E': y^2 = x^3 + Z^4*A*x + Z^6*B mod p are GF(p)-isomorphic via the isomorphism F(x,y) := (x*Z^2, y*Z^3). Especially, #E(GF(p)) = #E'(GF(p)) and, most importantly, E and E' have the same algebraic structure, and hence offer the same level of security. Approximately half of the isomorphism classes of elliptic curves over GF(p) with p = 3 mod 4 contain a curve with A = -3 mod p.

This constraint has also been used by [SEC2] and [FIPS].

4. The prime p must not be of any special form; this requirement is met by a verifiably pseudo-random generation of the parameters (see requirement 5 in section 2.1). Although parameters specified by existing standards do not meet this requirement, the need for such curves over (pseudo-)randomly chosen fields has already been foreseen by the Standards for Efficient Cryptography Group (SECG), see [SEC2].

5. #E(GF(p)) < p. As a consequence of the Hasse-Weil-Theorem the number of points #E(GF(p)) may be greater than the characteristic p of the prime field GF(p). In some cases even the bit-length of #E(GF(p)) can exceed the bit-length of p. To avoid overruns in implementations we require that #E(GF(p)) < p. In order to thwart attacks on digital signature schemes, some authors propose to use q > p, but the attacks described e.g. in [BRS] appear infeasible in a well-designed PKI.

6. B shall be a non-square mod p. Otherwise, the compressed representations of the curve-points (0,0) and (0,X) with X being the square root of B with a least significant bit of 0 would be identical. As there are implementations of elliptic curves that encode the point at infinity as (0,0) we try to avoid ambiguities. Note that this condition is stable under quadratic twists as described in condition 3 above. Condition 6 makes the attack described in [G] impossible. It can therefore also be seen as a security requirement.
This constraint has not been specified by existing standards.

3. Parameter Specification

In this section the elliptic curve domain parameters proposed are specified in the following way.

For all curves an ID is given by which it can be referenced.

p is the prime specifying the base field.

A and B are the coefficients of the equation y^2 = x^3 + A*x + B mod p defining the elliptic curve.

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

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

h is the cofactor of G in E, i.e. #E(GF(p))/q.

For the twisted curve, we also give the coefficient Z that defines the isomorphism F (see requirement 3 in section 2.2).

The methods for the generation of the parameters and complete security proofs regarding the security requirements specified in section 2.1 are given in [EBP].

3.1. Parameters for 160 Bit Curves

Curve-ID: brainpoolP160r1

p = E95E4A5F737059DC60DFC7AD95B3D8139515620F

A = 340E7BE2A280EB74E2BE61BADA745D97E8F7C300

B = 1E589A8595423412134FAA2DBDEC95C8D8675E58

x = BED5AF16EA3F6A4F62938C4631EB5AF7BDBCDBC3

y = 1667CB477A1A8EC338F94741669C976316DA6321

q = E95E4A5F737059DC60DF5991D45029409E60FC09

h = 1

#Twisted curve

Curve-ID: brainpoolP160t1

Z = 24DBFF5DEC9B986BBFE5295A29BFBAE45E0F5D0B

A = E95E4A5F737059DC60DFC7AD95B3D8139515620C

B = 7A556B6DAE535B7B51ED2C4D7DAA7A0B5C55F380

x = B199B13B9B34EFC1397E64BAEB05ACC265FF2378

y = ADD6718B7C7C1961F0991B842443772152C9E0AD

q = E95E4A5F737059DC60DF5991D45029409E60FC09

h = 1

3.2. Parameters for 192 Bit Curves

Curve-ID: brainpoolP192r1

p = C302F41D932A36CDA7A3463093D18DB78FCE476DE1A86297

A = 6A91174076B1E0E19C39C031FE8685C1CAE040E5C69A28EF

B = 469A28EF7C28CCA3DC721D044F4496BCCA7EF4146FBF25C9

x = C0A0647EAAB6A48753B033C56CB0F0900A2F5C4853375FD6

y = 14B690866ABD5BB88B5F4828C1490002E6773FA2FA299B8F

q = C302F41D932A36CDA7A3462F9E9E916B5BE8F1029AC4ACC1

h = 1

#Twisted curve

Curve-ID: brainpoolP192t1

Z = 1B6F5CC8DB4DC7AF19458A9CB80DC2295E5EB9C3732104CB

A = C302F41D932A36CDA7A3463093D18DB78FCE476DE1A86294

B = 13D56FFAEC78681E68F9DEB43B35BEC2FB68542E27897B79

x = 3AE9E58C82F63C30282E1FE7BBF43FA72C446AF6F4618129

y = 97E2C5667C2223A902AB5CA449D0084B7E5B3DE7CCC01C9

q = C302F41D932A36CDA7A3462F9E9E916B5BE8F1029AC4ACC1

h = 1

3.3. Parameters for 224 Bit Curves

Curve-ID: brainpoolP224r1

p = D7C134AA264366862A18302575D1D787B09F075797DA89F57EC8C0FF

A = 68A5E62CA9CE6C1C299803A6C1530B514E182AD8B0042A59CAD29F43

B = 2580F63CCFE44138870713B1A92369E33E2135D266DBB372386C400B

x = D9029AD2C7E5CF4340823B2A87DC68C9E4CE3174C1E6EFDEE12C07D

y = 58AA56F772C0726F24C6B89E4ECDAC24354B9E99CAA3F6D3761402CD

q = D7C134AA264366862A18302575D0FB98D116BC4B6DDEBCA3A5A7939F

h = 1

#Twisted curve

Curve-ID: brainpoolP224t1

Z = 2DF271E14427A346910CF7A2E6CFA7B3F484E5C2CCE1C8B730E28B3F

A = D7C134AA264366862A18302575D1D787B09F075797DA89F57EC8C0FC

B = 4B337D934104CD7BEF271BF60CED1ED20DA14C08B3BB64F18A60888D

x = 6AB1E344CE25FF3896424E7FFE14762ECB49F8928AC0C76029B4D580

y = 374E9F5143E568CD23F3F4D7C0D4B1E41C8CC0D1C6ABD5F1A46DB4C

q = D7C134AA264366862A18302575D0FB98D116BC4B6DDEBCA3A5A7939F

h = 1

3.4. Parameters for 256 Bit Curves

Curve-ID: brainpoolP256r1

p =
A9FB57DBA1EEA9BC3E660A909D838D726E3BF623D52620282013481D1F6E5377

A =
7D5A0975FC2C3057EEF67530417AFFE7FB8055C126DC5C6CE94A4B44F330B5D9

B =
26DC5C6CE94A4B44F330B5D9BBD77CBF958416295CF7E1CE6BCCDC18FF8C07B6

x =
8BD2AEB9CB7E57CB2C4B482FFC81B7AFB9DE27E1E3BD23C23A4453BD9ACE3262

y =
547EF835C3DAC4FD97F8461A14611DC9C27745132DED8E545C1D54C72F046997

q =
A9FB57DBA1EEA9BC3E660A909D838D718C397AA3B561A6F7901E0E82974856A7

h = 1

#Twisted curve

Curve-ID: brainpoolP256t1

Z =
3E2D4BD9597B58639AE7AA669CAB9837CF5CF20A2C852D10F655668DFC150EF0

A =
A9FB57DBA1EEA9BC3E660A909D838D726E3BF623D52620282013481D1F6E5374

B =
662C61C430D84EA4FE66A7733D0B76B7BF93EBC4AF2F49256AE58101FEE92B04

x =
A3E8EB3CC1CFE7B7732213B23A656149AFA142C47AAFBC2B79A191562E1305F4

y =
2D996C823439C56D7F7B22E14644417E69BCB6DE39D027001DABE8F35B25C9BE

q =
A9FB57DBA1EEA9BC3E660A909D838D718C397AA3B561A6F7901E0E82974856A7

h = 1

3.5. Parameters for 320 Bit Curves

Curve-ID: brainpoolP320r1

p = D35E472036BC4FB7E13C785ED201E065F98FCFA6F6F40DEF4F92B9EC7893EC
28FCD412B1F1B32E27

A = 3EE30B568FBAB0F883CCEBD46D3F3BB8A2A73513F5EB79DA66190EB085FFA9
F492F375A97D860EB4

B = 520883949DFDBC42D3AD198640688A6FE13F41349554B49ACC31DCCD884539
816F5EB4AC8FB1F1A6

x = 43BD7E9AFB53D8B85289BCC48EE5BFE6F20137D10A087EB6E7871E2A10A599
C710AF8D0D39E20611

y = 14FDD05545EC1CC8AB4093247F77275E0743FFED117182EAA9C77877AAAC6A
C7D35245D1692E8EE1

q = D35E472036BC4FB7E13C785ED201E065F98FCFA5B68F12A32D482EC7EE8658
E98691555B44C59311

h = 1

#Twisted curve

Curve-ID: brainpoolP320t1

Z = 15F75CAF668077F7E85B42EB01F0A81FF56ECD6191D55CB82B7D861458A18F
EFC3E5AB7496F3C7B1

A = D35E472036BC4FB7E13C785ED201E065F98FCFA6F6F40DEF4F92B9EC7893EC
28FCD412B1F1B32E24

B = A7F561E038EB1ED560B3D147DB782013064C19F27ED27C6780AAF77FB8A547
CEB5B4FEF422340353

x = 925BE9FB01AFC6FB4D3E7D4990010F813408AB106C4F09CB7EE07868CC136F
FF3357F624A21BED52

y = 63BA3A7A27483EBF6671DBEF7ABB30EBEE084E58A0B077AD42A5A0989D1EE7
1B1B9BC0455FB0D2C3

q = D35E472036BC4FB7E13C785ED201E065F98FCFA5B68F12A32D482EC7EE8658
E98691555B44C59311

h = 1

3.6. Parameters for 384 Bit Curves

Curve-ID: brainpoolP384r1

p = 8CB91E82A3386D280F5D6F7E50E641DF152F7109ED5456B412B1DA197FB711
23ACD3A729901D1A71874700133107EC53

A = 7BC382C63D8C150C3C72080ACE05AFA0C2BEA28E4FB22787139165EFBA91F9
0F8AA5814A503AD4EB04A8C7DD22CE2826

B = 4A8C7DD22CE28268B39B55416F0447C2FB77DE107DCD2A62E880EA53EEB62D
57CB4390295DBC9943AB78696FA504C11

x = 1D1C64F068CF45FFA2A63A81B7C13F6B8847A3E77EF14FE3DB7FCAFE0CBD10
E8E826E03436D646AAEF87B2E247D4AF1E

y = 8ABE1D7520F9C2A45CB1EB8E95CFD55262B70B29FEEC5864E19C054FF99129
280E4646217791811142820341263C5315

q = 8CB91E82A3386D280F5D6F7E50E641DF152F7109ED5456B31F166E6CAC0425
A7CF3AB6AF6B7FC3103B883202E9046565

h = 1

#Twisted curve

Curve-ID: brainpoolP384t1

Z = 41DFE8DD399331F7166A66076734A89CD0D2BCDB7D068E44E1F378F41ECBAE
97D2D63DBC87BCCDDCCC5DA39E8589291C

A = 8CB91E82A3386D280F5D6F7E50E641DF152F7109ED5456B412B1DA197FB711
23ACD3A729901D1A71874700133107EC50

B = 7F519EADA7BDA81BD826DBA647910F8C4B9346ED8CCDC64E4B1ABD11756DCE
1D2074AA263B88805CED70355A33B471EE

x = 18DE98B02DB9A306F2AFCD7235F72A819B80AB12EBD653172476FECD462AAB
FFC4FF191B946A5F54D8D0AA2F418808CC

y = 25AB056962D30651A114AFD2755AD336747F93475B7A1FCA3B88F2B6A208CC
FE469408584DC2B2912675BF5B9E582928

q = 8CB91E82A3386D280F5D6F7E50E641DF152F7109ED5456B31F166E6CAC0425
A7CF3AB6AF6B7FC3103B883202E9046565

h = 1

3.7. Parameters for 512 Bit Curves

Curve-ID: brainpoolP512r1

p = AADD9DB8DBE9C48B3FD4E6AE33C9FC07CB308DB3B3C9D20ED6639CCA703308
717D4D9B009BC66842AECDA12AE6A380E62881FF2F2D82C68528AA6056583A48F3

A = 7830A3318B603B89E2327145AC234CC594CBDD8D3DF91610A83441CAEA9863
BC2DED5D5AA8253AA10A2EF1C98B9AC8B57F1117A72BF2C7B9E7C1AC4D77FC94CA

B = 3DF91610A83441CAEA9863BC2DED5D5AA8253AA10A2EF1C98B9AC8B57F1117
A72BF2C7B9E7C1AC4D77FC94CADC083E67984050B75EBAE5DD2809BD638016F723

x = 81AEE4BDD82ED9645A21322E9C4C6A9385ED9F70B5D916C1B43B62EEF4D009
8EFF3B1F78E2D0D48D50D1687B93B97D5F7C6D5047406A5E688B352209BCB9F822

y = 7DDE385D566332ECC0EABFA9CF7822FDF209F70024A57B1AA000C55B881F81
11B2DCDE494A5F485E5BCA4BD88A2763AED1CA2B2FA8F0540678CD1E0F3AD80892

q = AADD9DB8DBE9C48B3FD4E6AE33C9FC07CB308DB3B3C9D20ED6639CCA703308
70553E5C414CA92619418661197FAC10471DB1D381085DDADDB58796829CA90069

h = 1

#Twisted curve

Curve-ID: brainpoolP512t1

Z = 12EE58E6764838B69782136F0F2D3BA06E27695716054092E60A80BEDB212B
64E585D90BCE13761F85C3F1D2A64E3BE8FEA2220F01EBA5EEB0F35DBD29D922AB

A = AADD9DB8DBE9C48B3FD4E6AE33C9FC07CB308DB3B3C9D20ED6639CCA703308
717D4D9B009BC66842AECDA12AE6A380E62881FF2F2D82C68528AA6056583A48F0

B = 7CBBBCF9441CFAB76E1890E46884EAE321F70C0BCB4981527897504BEC3E36
A62BCDFA2304976540F6450085F2DAE145C22553B465763689180EA2571867423E

x = 640ECE5C12788717B9C1BA06CBC2A6FEBA85842458C56DDE9DB1758D39C031
3D82BA51735CDB3EA499AA77A7D6943A64F7A3F25FE26F06B51BAA2696FA9035DA

y = 5B534BD595F5AF0FA2C892376C84ACE1BB4E3019B71634C01131159CAE03CE
E9D9932184BEEF216BD71DF2DADF86A627306ECFF96DBB8BACE198B61E00F8B332

q = AADD9DB8DBE9C48B3FD4E6AE33C9FC07CB308DB3B3C9D20ED6639CCA703308
70553E5C414CA92619418661197FAC10471DB1D381085DDADDB58796829CA90069

h = 1

4. Object Identifiers and ASN.1 Syntax

The root of the tree for the object identifier of the domain parameters defined in this specification is given by

```
ecStdCurvesAndGeneration OBJECT IDENTIFIER::= {iso(1)
identifified-organization(3) teletrust(36) algorithm(3) signature-
algorithm(3) ecSign(2) 8}
```

The object identifier ellipticCurve represents the tree containing the object identifiers for each set of domain parameters specified in this RFC. It has the following value:

```
ellipticCurve OBJECT IDENTIFIER ::= {ecStdCurvesAndGeneration 1}
```

The tree for the domain parameters defined in this RFC is

```
versionOne OBJECT IDENTIFIER ::= {ellipticCurve 1}
```

The following object identifiers represent the domain parameters defined in this RFC:

```
brainpoolP160r1 OBJECT IDENTIFIER ::= {versionOne 1}
```

```
brainpoolP160t1 OBJECT IDENTIFIER ::= {versionOne 2}

brainpoolP192r1 OBJECT IDENTIFIER ::= {versionOne 3}

brainpoolP192t1 OBJECT IDENTIFIER ::= {versionOne 4}

brainpoolP224r1 OBJECT IDENTIFIER ::= {versionOne 5}

brainpoolP224t1 OBJECT IDENTIFIER ::= {versionOne 6}

brainpoolP256r1 OBJECT IDENTIFIER ::= {versionOne 7}

brainpoolP256t1 OBJECT IDENTIFIER ::= {versionOne 8}

brainpoolP320r1 OBJECT IDENTIFIER ::= {versionOne 9}

brainpoolP320t1 OBJECT IDENTIFIER ::= {versionOne 10}

brainpoolP384r1 OBJECT IDENTIFIER ::= {versionOne 11}

brainpoolP384t1 OBJECT IDENTIFIER ::= {versionOne 12}

brainpoolP512r1 OBJECT IDENTIFIER ::= {versionOne 13}

brainpoolP512t1 OBJECT IDENTIFIER ::= {versionOne 14}
```

The ASN.1 syntax for elliptic curve domain parameters according to ANSI X9.62 [ANSI1] allows indicating whether a curve and base point have been generated verifiably at random or not. The parameters specified in section 3 have all been generated verifyably at random; however, the algorithms used for their generation deviate from those specified in ANSI X9.62. Consequently, applications following ANSI X9.62 will not be able to verify the randomness of the parameters. In order to avoid rejection of the parameters, the ASN.1 encoding SHOULD NOT specify that the curve or base point has been generated verifiably at random. In particular, CAs SHOULD encode SpecifiedECDomain in the following way:

- The field Version is set to ecdpVer1(1).
- The field curve.seed is absent.
- The field hash is absent.

5. Security Considerations

Security issues are discussed throughout this memo in particular in Section 2.1. Further security discussions specific to elliptoc curve cryptography can be found in [ANSI1] and [SEC1].

6. IANA Considerations

This memo includes no request to IANA.

7. Intellectual Property Rights

The authors have no knowledge about any intellectual property rights which cover the usage of the domain parameters defined herein. However, readers should be aware that implementations based on these domain parameters may require use of inventions covered by patent rights.

8. References

8.1. Normative References

[ANSI1] American National Standards Institute, "Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)", ANSI X9.62, 2005.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

8.2. Informative References

[ANSI2] American National Standards Institute, "Public Key Cryptography For The Financial Services Industry: Key Agreement and Key Transport Using The Elliptic Curve Cryptography", ANSI X9.63, 2001.

[BJ] Brier, E. and M. Joyce, "Fast multiplication on Elliptic Curves through Isogenies", Applied Algebra AAECC-15, Lecture Notes in Computer Science 2643, Springer Verlag, 2003.

[BG] Brown, J. and R. Gallant, "The static Diffie-Hellman Problem", Centre for Applied Cryptographic Research, University of Waterloo, Technical Report CACR 2004-10, 2005.

[BRS] Bohli, J., Roehrich, S., and R. Steinwandt, "Key

substitution attacks revisited: taking into account malicious signers", PreprintInternational Journal of Information Security Volume 5, Issue 1, January 2006.

[BSS] Blake, I., Seroussi, G., and N. Smart, "Elliptic Curves in Cryptography", Cambridge University Press, 1999.

[EBP] ECC Brainpool, "ECC Brainpool Standard Curves and Curve Generation", October 2005, <http://www.ecc-brainpool.org/download/Domain-parameters.pdf>.

[ETSI] European Telecommunications Standards Institute (ETSI), "Algorithms and Parameters for Secure Electronic Signatures, Part 1: Hash functions and asymmetric algorithms", TS 102 176-1, July 2005.

[FIPS] National Institute of Standards and Technology, "Digital Signature Standard (DSS)", FIPS PUB 186-2, December 1998.

[G] Goubin, L., "A refined power-analysis-attack on Elliptic Curve Cryptosystems", Proceedings of Public-Key-Cryptography - PKC 2003, Lecture Notes in Computer Science 2567, Springer Verlag, 2003.

[HMV] Hankerson, D., Menezes, A., and S. Vanstone, "Guide to Elliptic Curve Cryptography", Springer Verlag, 2004.

[HR] Huang, M. and W. Raskind, "Global methods for discrete logarithm problems III", 2006, <http://www-rcf.usc.edu/~mdhuang/mypapers/062806dl3.pdf>.

[ISO1] International Organization for Standardization, "Information technology - Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms", ISO/IEC 14888-3, 2006.

[ISO2] International Organization for Standardization, "Information technology - Security techniques - Cryptographic techniques based on elliptic curves - Part 2: Digital signatures", ISO/IEC 15946-2, 2002.

[JMV] Jao, D., Miller, SD., and R. Venkatesan, "Ramanujan graphs and the random reducibility of discrete log on isogenous elliptic curves", IACR Cryptology ePrint Archive 2004/312, 2004.

[PKIX] Brown, D., "Additional Algorithms and Identifiers for use of Elliptic Curve Cryptography with PKIX",

draft-ietf-pkix-ecc-pkalgs-03 (work in progress), October 2006.

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, November 1998.

[RFC3278] Blake-Wilson, S., Brown, D., and P. Lambert, "Use of Elliptic Curve Cryptography (ECC) Algorithms in Cryptographic Message Syntax (CMS)", RFC 3278, April 2002.

[RFC3279] Bassham, L., Polk, W., and R. Housley, "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3279, April 2002.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.

[RFC4050] Blake-Wilson, S., Karlinger, G., Kobayashi, T., and Y. Wang, "Using the Elliptic Curve Signature Algorithm (ECDSA) for XML Digital Signatures", RFC 4050, April 2005.

[RFC4492] Blake-Wilson, S., Bolyard, N., Gupta, V., Hawk, C., and B. Moeller, "Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS)", RFC 4492, May 2006.

[RFC4754] Solinas, J. and D. Fu, "IKE and IKEv2 Authentication the Elliptic Curve Digital Signature Algorithm (ECDSA)", RFC 4754, January 2007.

[SA] Satoh, T. and K. Araki, "Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves", Commentarii Mathematici Universitatis Sancti Pauli 47, 1998.

[SEC1] Certicom Research, "Elliptic Curve Cryptography", Standards for Efficient Cryptography (SEC) 1, September 2000.

[SEC2] Certicom Research, "Recommended Elliptic Curve Domain Parameters", Standards for Efficient Cryptography (SEC) 2, September 2000.

[Sem] Semaev, I., "Evaluation of discrete logarithms on some elliptic curves", Mathematics of Computation 67, 1998.

[Sma] Smart, N., "The discrete logarithm problem on elliptic

curves of trace one", Journal of Cryptology 12, 1999.

Authors' Addresses

Manfred Lochter
Bundesamt fuer Sicherheit in der Informationstechnik (BSI)
Postfach 200363
53133 Bonn
Germany

Phone: +49 228 9582 5643
EMail: manfred.lochter@bsi.bund.de

Johannes Merkle
secunet Security Networks
Mergenthaler Allee 77
65760 Eschborn
Germany

Phone: +49 6196 95888 55
EMail: johannes.merkle@secunet.com

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).