

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: October 13, 2013

J. Merkle  
secunet Security Networks  
M. Lochter  
Bundesamt fuer Sicherheit in der  
Informationstechnik (BSI)  
April 11, 2013

Using the ECC Brainpool Curves for IKEv2 Key Exchange  
draft-merkle-ikev2-ke-brainpool-04

## Abstract

This document specifies the use of ECC Brainpool elliptic curve groups for key exchange in the Internet Key Exchange version 2 (IKEv2) protocol.

## 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 <http://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 October 13, 2013.

## Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">1.1.</a>	Requirements Language . . . . .	<a href="#">3</a>
<a href="#">2.</a>	IKEv2 Key Exchange using the ECC Brainpool Curves . . . . .	<a href="#">4</a>
<a href="#">2.1.</a>	Diffie-Hellman Group Transform IDs . . . . .	<a href="#">4</a>
<a href="#">2.2.</a>	Using the Twisted Brainpool Curves Internally . . . . .	<a href="#">4</a>
<a href="#">2.3.</a>	Key Exchange Payload and Shared Secret . . . . .	<a href="#">4</a>
<a href="#">3.</a>	Security Considerations . . . . .	<a href="#">6</a>
<a href="#">4.</a>	IANA Considerations . . . . .	<a href="#">7</a>
<a href="#">5.</a>	References . . . . .	<a href="#">8</a>
<a href="#">5.1.</a>	Normative References . . . . .	<a href="#">8</a>
<a href="#">5.2.</a>	Informative References . . . . .	<a href="#">8</a>
<a href="#">Appendix A.</a>	Test Vectors . . . . .	<a href="#">10</a>
<a href="#">A.1.</a>	224 Bit Curve . . . . .	<a href="#">10</a>
<a href="#">A.2.</a>	256 Bit Curve . . . . .	<a href="#">11</a>
<a href="#">A.3.</a>	384 Bit Curve . . . . .	<a href="#">11</a>
<a href="#">A.4.</a>	512 Bit Curve . . . . .	<a href="#">12</a>

## 1. Introduction

In [[RFC5639](#)], a new set of elliptic curve groups over finite prime fields for use in cryptographic applications was specified. These groups, denoted as ECC Brainpool curves, were generated in a verifiably pseudo-random way and comply with the security requirements of relevant standards from ISO [[IS01](#)] [[IS02](#)], ANSI [[ANSI1](#)], NIST [[FIPS](#)], and SecG [[SEC2](#)].

While the ASN.1 object identifiers defined in [RFC 5639](#) allow usage of the ECC Brainpool curves in certificates and certificate revocation lists, their utilization for key exchange in IKEv2 [[RFC5996](#)] requires the definition and assignment of additional Diffie-Hellman Group Transform IDs in the respective IANA registry. This document specifies tranform IDs for four curves from [RFC 5639](#) as well as the encoding of the key exchange payload and derivation of the shared secret when using one of these curves.

### 1.1. 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 [[RFC2119](#)].

## [2.](#) IKEv2 Key Exchange using the ECC Brainpool Curves

### [2.1.](#) Diffie-Hellman Group Transform IDs

In order to use the ECC Brainpool curves for key exchange within IKEv2, the Diffie-Hellman Group Transform IDs (Transform Type 4) listed in the following table are to be registered with IANA [[IANA-IKE2](#)]. The parameters associated with these curves are defined in [RFC 5639](#) [[RFC5639](#)].

Curve	Transform ID
brainpoolP224r1	27
brainpoolP256r1	28
brainpoolP384r1	29
brainpoolP512r1	30

Table 1

Test vectors for the groups defined by the ECC Brainpool curves are provided in [Appendix A](#)

### [2.2.](#) Using the Twisted Brainpool Curves Internally

In [RFC5639] for each random curve, a "twisted curve" (defined by a quadratic twist, see [HNV]) is defined offering the same level of security but potentially allowing more efficient arithmetic due to the curve parameter  $A = -3$ . The transform IDs listed in Table 1 also allow using the twisted curve corresponding to the specified random curve: points  $(x,y)$  of any curve listed in can be efficiently transformed to the corresponding point  $(x',y')$  on the twisted curve of same bit length - and vice versa - by setting  $(x',y') = (x*Z^2, y*Z^3)$  with the coefficient  $Z$  specified for that curve in [RFC5639].

### 2.3. Key Exchange Payload and Shared Secret

For the encoding of the key exchange payload and the derivation of the shared secret, the methods specified in [RFC5903] are adopted.

In an ECP key exchange in IKEv2, the Diffie-Hellman public value passed in a KE payload consists of two components,  $x$  and  $y$ , corresponding to the coordinates of an elliptic curve point. Each component MUST be computed from the corresponding coordinate using

the FieldElement-to-OctetString conversion method specified in [SEC1] and MUST have bit length as indicated in Table 2. This length is enforced by the FieldElement-to-OctetString conversion method, if necessary, by prepending the value with zeros.

Note: The FieldElement-to-OctetString conversion method specified in [SEC1] is equivalent to applying the conversion between integers and octet strings of Section 6 of [RFC6090] after representing the field element as integer in the interval  $[0, p-1]$ .

Curves	Bit length of each component (x or y)	Bit length of key exchange payload
brainpoolP224r1	224	448
brainpoolP256r1	256	512
brainpoolP384r1	384	768
brainpoolP512r1	512	1024

Table 2

From these components, the key exchange payload MUST be computed as the concatenation of the x and y coordinates. Hence, the key exchange payload has the bit length indicated in Table 2.

The Diffie-Hellman shared secret value consists only of the x value. In particular, the shared secret value MUST be computed from the x coordinate of the Diffie-Hellman common value using the FieldElement-to-OctetString conversion method specified in [\[SEC1\]](#) and MUST have bit length as indicated in the Table 2.

### [3.](#) Security Considerations

The security considerations of [\[RFC5996\]](#) apply accordingly.

In order to thwart certain active attacks, the validity of the other peer's public Diffie-Hellmann key recovered from the received key exchange payload needs to be verified. In particular, the recommendations and requirements of [\[IKE\\_DH\\_Req\]](#) MUST be observed. For the curves listed in Table 1, Section 2.3 of [\[IKE\\_DH\\_Req\]](#) applies.

The confidentiality, authenticity and integrity of a secure communication based on IKEv2 is limited by the weakest cryptographic primitive applied. In order to achieve a maximum security level when using one of the elliptic curves from Table 1 for key exchange, the

key derivation function, the algorithms and key lengths of symmetric encryption and message authentication as well as the algorithm, bit length and hash function used for signature generation should be chosen according to the recommendations of [\[NIST800-57\]](#) and [\[RFC5639\]](#). Furthermore, the private Diffie-Hellman keys should be selected with the same bit length as the order of the group generated by the base point G and with approximately maximum entropy.

Implementations of elliptic curve cryptography for IKEv2 could be susceptible to side-channel attacks. Particular care should be taken for implementations that internally use the corresponding twisted curve to take advantage of an efficient arithmetic for the special parameters ( $A = -3$ ): although the twisted curve itself offers the same level of security as the corresponding random curve (through mathematical equivalence), an arithmetic based on small curve parameters could be harder to protect against side-channel attacks. General guidance on resistance of elliptic curve cryptography implementations against side-channel-attacks is given in [\[BSI1\]](#) and [\[HNV\]](#).

#### [4.](#) IANA Considerations

IANA has updated its Transform Type 4 (Diffie-Hellman Group Transform) registry in [\[IANA-IKE2\]](#) to include the groups listed in Table 1. For these new groups, a reference to Section 2.3 of [\[IKE\\_DH\\_Req\]](#) shall be included in the column named "Recipient Tests" of the Transform Type 4 registry (requested by [\[IKE\\_DH\\_Req\]](#)) indicating the required checks for the other peer's Diffie-Hellman

public keys.



### 5.1. Normative References

- [IANA-IKE2] Internet Assigned Numbers Authority, "Internet Key Exchange Version 2 (IKEv2) Parameters",  
<<http://www.iana.org/assignments/ikev2-parameters>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC5996] Kaufman, C., Hoffman, P., Nir, Y., and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)",  
[RFC 5996](#), September 2010.
- [RFC5639] Lochter, M. and J. Merkle, "Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation",  
[RFC 5639](#), March 2010.
- [IKE\_DH\_Req] Sheffer, Y. and S. Fluhrer, "Additional Diffie-Hellman Tests for IKEv2 (work in progress)",  
[draft-ietf-ipsecme-dh-checks-01](#) (work in progress), January 2013.
- [SEC1] Certicom Research, "Elliptic Curve Cryptography", Standards for Efficient Cryptography (SEC) 1, September 2000.

### 5.2. Informative 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.
- [BSI1] Bundesamt fuer Sicherheit in der Informationstechnik, "Minimum Requirements for Evaluating Side-Channel Attack Resistance of Elliptic Curve Implementations", July 2011.
- [FIPS] National Institute of Standards and Technology, "Digital Signature Standard (DSS)", FIPS PUB 186-2, December 1998.
- [BP\_IKE] Harkins, D., "Brainpool Elliptic Curves for the IKE Group Description Registry (work in progress)",  
[draft-harkins-brainpool-ike-groups-04](#) (work in progress), August 2012.

- 
- [HMV] Hankerson, D., Menezes, A., and S. Vanstone, "Guide to Elliptic Curve Cryptography", Springer Verlag, 2004.
- [IS01] International Organization for Standardization, "Information Technology - Security Techniques - Digital Signatures with Appendix - Part 3: Discrete Logarithm Based Mechanisms", ISO/IEC 14888-3, 2006.
- [IS02] International Organization for Standardization, "Information Technology - Security Techniques - Cryptographic Techniques Based on Elliptic Curves - Part 2: Digital signatures", ISO/IEC 15946-2, 2002.
- [NIST800-57] National Institute of Standards and Technology, "Recommendation for Key Management - Part 1: General (Revised)", NIST Special Publication 800-57, March 2007.
- [RFC5903] Fu, D. and J. Solinas, "Elliptic Curve Groups modulo a Prime (ECP Groups) for IKE and IKEv2", [RFC 5903](#), June 2010.
- [RFC6090] McGrew, D., Igoe, K., and M. Salter, "Fundamental Elliptic Curve Cryptography Algorithms", [RFC 6090](#), February 2011.
- [SEC2] Certicom Research, "Recommended Elliptic Curve Domain Parameters", Standards for Efficient Cryptography (SEC) 2, September 2000.

## [Appendix A](#). Test Vectors

This section provides some test vectors for example Diffie-Hellman key exchanges using each of the curves defined in [Section 2](#) . In all of the following sections the following notation is used:

d\_A: the secret key of party A

x\_qA: the x-coordinate of the public key of party A

y\_qA: the y-coordinate of the public key of party A

d\_B: the secret key of party B

x\_qB: the x-coordinate of the public key of party B

y\_qB: the y-coordinate of the public key of party B

x\_Z: the x-coordinate of the shared secret that results from completion of the Diffie-Hellman computation

y\_Z: the y-coordinate of the shared secret that results from completion of the Diffie-Hellman computation

The field elements x\_qA, y\_qA, x\_qB, y\_qB, x\_Z, y\_Z are represented as hexadecimal values using the FieldElement-to-OctetString conversion method specified in [[SEC1](#)].

### [A.1](#). 224 Bit Curve

Curve brainpoolP224r1

dA = 39F155483CEE191FBECFE9C81D8AB1A03CDA6790E7184ACE44BCA161

x\_qA = A9C21A569759DA95E0387041184261440327AFE33141CA04B82DC92E

y\_qA = 98A0F75FBBF61D8E58AE5511B2BCDBE8E549B31E37069A2825F590C1

dB = 6060552303899E2140715816C45B57D9B42204FB6A5BF5BEAC10DB00

x\_qB = 034A56C550FF88056144E6DD56070F54B0135976B5BF77827313F36B  
y\_qB = 75165AD99347DC86CAAB1CBB579E198EAF88DC35F927B358AA683681  
x\_Z = 1A4BFE705445120C8E3E026699054104510D119757B74D5FE2462C66  
y\_Z = BB6802AC01F8B7E91B1A1ACFB9830A95C079CEC48E52805DFD7D2AFE

## [A.2.](#) 256 Bit Curve

Curve brainpoolP256r1

dA =  
81DB1EE100150FF2EA338D708271BE38300CB54241D79950F77B063039804F1D  
  
x\_qA =  
44106E913F92BC02A1705D9953A8414DB95E1AAA49E81D9E85F929A8E3100BE5  
  
y\_qA =  
8AB4846F11CACCB73CE49CBDD120F5A900A69FD32C272223F789EF10EB089BDC  
  
dB =  
55E40BC41E37E3E2AD25C3C6654511FFA8474A91A0032087593852D3E7D76BD3  
  
x\_qB =  
8D2D688C6CF93E1160AD04CC4429117DC2C41825E1E9FCA0ADDD34E6F1B39F7B  
  
y\_qB =  
990C57520812BE512641E47034832106BC7D3E8DD0E4C7F1136D7006547CEC6A  
  
x\_Z =  
89AFC39D41D3B327814B80940B042590F96556EC91E6AE7939BCE31F3A18BF2B  
  
y\_Z =  
49C27868F4ECA2179BFD7D59B1E3BF34C1DBDE61AE12931648F43E59632504DE

## [A.3.](#) 384 Bit Curve

Curve brainpoolP384r1

dA = 1E20F5E048A5886F1F157C74E91BDE2B98C8B52D58E5003D57053FC4B0BD6  
5D6F15EB5D1EE1610DF870795143627D042

x\_qA = 68B665DD91C195800650CDD363C625F4E742E8134667B767B1B47679358  
8F885AB698C852D4A6E77A252D6380FCAF068

y\_qA = 55BC91A39C9EC01DEE36017B7D673A931236D2F1F5C83942D049E3FA206  
07493E0D038FF2FD30C2AB67D15C85F7FAA59

dB = 032640BC6003C59260F7250C3DB58CE647F98E1260ACCE4ACDA3DD869F74E  
01F8BA5E0324309DB6A9831497ABAC96670

x\_qB = 4D44326F269A597A5B58BBA565DA5556ED7FD9A8A9EB76C25F46DB69D19  
DC8CE6AD18E404B15738B2086DF37E71D1EB4

y\_qB = 62D692136DE56CBE93BF5FA3188EF58BC8A3A0EC6C1E151A21038A42E91  
85329B5B275903D192F8D4E1F32FE9CC78C48

x\_Z = 0BD9D3A7EA0B3D519D09D8E48D0785FB744A6B355E6304BC51C229FBBCE2  
39BBADF6403715C35D4FB2A5444F575D4F42

y\_Z = 0DF213417EBE4D8E40A5F76F66C56470C489A3478D146DECF6DF0D94BAE9  
E598157290F8756066975F1DB34B2324B7BD

#### [A.4.](#) 512 Bit Curve

Curve brainpoolP512r1

dA = 16302FF0DBBB5A8D733DAB7141C1B45ACBC8715939677F6A56850A38BD87B  
D59B09E80279609FF333EB9D4C061231FB26F92EEB04982A5F1D1764CAD5766542  
2

x\_qA = 0A420517E406AAC0ACDCE90FCD71487718D3B953EFD7FBEC5F7F27E28C6  
149999397E91E029E06457DB2D3E640668B392C2A7E737A7F0BF04436D11640FD0  
9FD

y\_qA = 72E6882E8DB28AAD36237CD25D580DB23783961C8DC52DFA2EC138AD472  
A0FCEF3887CF62B623B2A87DE5C588301EA3E5FC269B373B60724F5E82A6AD147F  
DE7

dB = 230E18E1BCC88A362FA54E4EA3902009292F7F8033624FD471B5D8ACE49D1  
2CFABBC19963DAB8E2F1EBA00BFFB29E4D72D13F2224562F405CB80503666B2542  
9

x\_qB = 9D45F66DE5D67E2E6DB6E93A59CE0BB48106097FF78A081DE781CDB31FC  
E8CCBAAEA8DD4320C4119F1E9CD437A2EAB3731FA9668AB268D871DEDA55A54731  
99F

y\_qB = 2FDC313095BCDD5FB3A91636F07A959C8E86B5636A1E930E8396049CB48  
1961D365CC11453A06C719835475B12CB52FC3C383BCE35E27EF194512B7187628  
5FA

x\_Z = A7927098655F1F9976FA50A9D566865DC530331846381C87256BAF322624  
4B76D36403C024D7BBF0AA0803EAF405D3D24F11A9B5C0BEF679FE1454B21C4CD  
1F

y\_Z = 7DB71C3DEF63212841C463E881BDCF055523BD368240E6C3143BD8DEF8B3  
B3223B95E0F53082FF5E412F4222537A43DF1C6D25729DDB51620A832BE6A26680  
A2

#### Authors' Addresses

Johannes Merkle  
secunet Security Networks  
Mergenthaler Allee 77  
65760 Eschborn  
Germany

Phone: +49 201 5454 3091  
EMail: [johannes.merkle@secunet.com](mailto:johannes.merkle@secunet.com)

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