Internet Engineering Task Force (IETF)
Request for Comments: 7027
Updates: 4492
Category: Informational
ISSN: 2070-1721

J. Merkle
secunet Security Networks
M. Lochter
BSI

October 2013

Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS)

Abstract

This document specifies the use of several Elliptic Curve Cryptography (ECC) Brainpool curves for authentication and key exchange in the Transport Layer Security (TLS) protocol.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Not all documents approved by the IESG are a candidate for any level of Internet Standard; see Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc7027.

Table of Contents

1. Introduction

[RFC5639] specifies a new set of elliptic curve groups over finite prime fields for use in cryptographic applications. 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 [ISO1] [ISO2], ANSI [ANSI1], NIST [FIPS], and SecG [SEC2].

[RFC4492] defines the usage of elliptic curves for authentication and key agreement in TLS 1.0 and TLS 1.1; these mechanisms may also be used with TLS 1.2 [RFC5246]. While the ASN.1 object identifiers defined in [RFC5639] already allow usage of the ECC Brainpool curves for TLS (client or server) authentication through reference in X.509 certificates according to [RFC3279] and [RFC5480], their negotiation for key exchange according to [RFC4492] requires the definition and assignment of additional NamedCurve IDs. This document specifies such values for three curves from [RFC5639].

2. Brainpool NamedCurve Types

According to [RFC4492], the name space NamedCurve is used for the negotiation of elliptic curve groups for key exchange during a handshake starting a new TLS session. This document adds new NamedCurve types to three elliptic curves defined in [RFC5639] as follows:

```
enum {
     brainpoolP256r1(26),
     brainpoolP384r1(27),
     brainpoolP512r1(28)
} NamedCurve;
```

These curves are suitable for use with Datagram TLS [RFC6347].

Test vectors for a Diffie-Hellman key exchange using these elliptic curves are provided in Appendix A.

3. IANA Considerations

IANA has assigned numbers for the ECC Brainpool curves listed in Section 2 in the "EC Named Curve" [IANA-TLS] registry of the "Transport Layer Security (TLS) Parameters" registry as follows:

Value	Description	DTLS-OK	Reference
26	brainpoolP256r1	Y	RFC 7027
27	brainpoolP384r1	Y	RFC 7027
28	brainpoolP512r1	Y	RFC 7027

Table 1

4. Security Considerations

The security considerations of [RFC5246] apply to the ECC Brainpool curves described in this document.

The confidentiality, authenticity, and integrity of the TLS communication 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 authentication and/or key exchange in TLS, 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 TLS may be susceptible to side-channel attacks. Particular care should be taken for implementations that internally transform curve points to points on the corresponding "twisted curve", using the map (x',y') = (x*Z^2, y*Z^3) with the coefficient Z specified for that curve in [RFC5639], in order to take advantage of an efficient arithmetic based on the twisted curve's 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 may 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 [HMV].

5. References

5.1. Normative References

[IANA-TLS] Internet Assigned Numbers Authority, "Transport Layer Security (TLS) Parameters", <http://www.iana.org/assignments/tls-parameters>.

[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.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC5639] Lochter, M. and J. Merkle, "Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation", RFC 5639, March 2010.

[RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", RFC 6347, January 2012.

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.

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

[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.

[NIST800-57] National Institute of Standards and Technology, "Recommendation for Key Management - Part 1: General (Revised)", NIST Special Publication 800-57, March 2007.

[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.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, March 2009.

[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.

Appendix A. Test Vectors

This section provides some test vectors for example Diffie-Hellman key exchanges using each of the curves defined in Table 1. The following notation is used in the subsequent sections:

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, i.e., the hex representation of the pre-master secret

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, and y_Z are represented as hexadecimal values using the FieldElement-to-OctetString conversion method specified in [SEC1].

A.1. 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.2. 384-Bit Curve

Curve brainpoolP384r1

dA = 1E20F5E048A5886F1F157C74E91BDE2B98C8B52D58E5003D57053FC4B0BD65D6F15EB5D1EE1610DF870795143627D042

x_qA = 68B665DD91C195800650CDD363C625F4E742E8134667B767B1B476793588F885AB698C852D4A6E77A252D6380FCAF068

y_qA = 55BC91A39C9EC01DEE36017B7D673A931236D2F1F5C83942D049E3FA20607493E0D038FF2FD30C2AB67D15C85F7FAA59

dB = 032640BC6003C59260F7250C3DB58CE647F98E1260ACCE4ACDA3DD869F74E01F8BA5E0324309DB6A9831497ABAC96670

x_qB = 4D44326F269A597A5B58BBA565DA5556ED7FD9A8A9EB76C25F46DB69D19DC8CE6AD18E404B15738B2086DF37E71D1EB4

y_qB = 62D692136DE56CBE93BF5FA3188EF58BC8A3A0EC6C1E151A21038A42E9185329B5B275903D192F8D4E1F32FE9CC78C48

x_Z = 0BD9D3A7EA0B3D519D09D8E48D0785FB744A6B355E6304BC51C229FBBCE239BBADF6403715C35D4FB2A5444F575D4F42

y_Z = 0DF213417EBE4D8E40A5F76F66C56470C489A3478D146DECF6DF0D94BAE9E598157290F8756066975F1DB34B2324B7BD

A.3. 512-Bit Curve

Curve brainpoolP512r1

dA = 16302FF0DBBB5A8D733DAB7141C1B45ACBC8715939677F6A56850A38BD87BD59B09E80279609FF333EB9D4C061231FB26F92EEB04982A5F1D1764CAD57665422

x_qA = 0A420517E406AAC0ACDCE90FCD71487718D3B953EFD7FBEC5F7F27E28C6149999397E91E029E06457DB2D3E640668B392C2A7E737A7F0BF04436D11640FD09FD

y_qA = 72E6882E8DB28AAD36237CD25D580DB23783961C8DC52DFA2EC138AD472A0FCEF3887CF62B623B2A87DE5C588301EA3E5FC269B373B60724F5E82A6AD147FDE7

dB = 230E18E1BCC88A362FA54E4EA3902009292F7F8033624FD471B5D8ACE49D12CFABBC19963DAB8E2F1EBA00BFFB29E4D72D13F2224562F405CB80503666B25429

x_qB = 9D45F66DE5D67E2E6DB6E93A59CE0BB48106097FF78A081DE781CDB31FCE8CCBAAEA8DD4320C4119F1E9CD437A2EAB3731FA9668AB268D871DEDA55A5473199F

y_qB = 2FDC313095BCDD5FB3A91636F07A959C8E86B5636A1E930E8396049CB481961D365CC11453A06C719835475B12CB52FC3C383BCE35E27EF194512B71876285FA

x_Z = A7927098655F1F9976FA50A9D566865DC530331846381C87256BAF3226244B76D36403C024D7BBF0AA0803EAFF405D3D24F11A9B5C0BEF679FE1454B21C4CD1F

y_Z = 7DB71C3DEF63212841C463E881BDCF055523BD368240E6C3143BD8DEF8B3B3223B95E0F53082FF5E412F4222537A43DF1C6D25729DDB51620A832BE6A26680A2

Authors' Addresses

Johannes Merkle
secunet Security Networks
Mergenthaler Allee 77
65760 Eschborn
Germany

Phone: +49 201 5454 3091
EMail: 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