Public Comments Received on Draft NIST SP 800-186:
Recommendations for Discrete Logarithm-Based Cryptography:
Elliptic Curve Domain Parameters
(January 29, 2020 deadline)

Bernstein, Dan/Lange, Tanja

From: D. J. Bernstein
Sent: Wednesday, January 29, 2020 11:58 PM
To: fips186-comments
Cc: Tanja Lange
Subject: Comment on FIPS 186

Please see attached PDF for comments from Daniel J. Bernstein and Tanja Lange.

Giessmann, Ernst

From: Ernst G Giessmann
Sent: Thursday, January 30, 2020 8:22 AM
To: SP800-186-comments
Subject: Comment on Draft NIST SP 800-186

Dear editors,
thanks for good job. The included remarks are almost editorials and I guess, that they are found already by others. If there are new ones, let me know ;-)
One remark on the "(mod n)" notation. You defined "mod" as the operation reducing an integer "modulo n" to the remainder. Therefore "6 mod 5" is defined 0 mod 4 as well. And if you write "0 (mod 4)", then it can't be the modulo reduction. Quite often it is clear that you had the equivalence relation in GF(n) in mind. But, as it is not defined explicitly, you must use only one symbol for it, either two bar equal sign or three bar equivalence sign (line 506). Therefore many of the remarks can be resolved by using "(mod n)" in brackets with the equivalence symbol or "mod n" without brackets.

Kind regards,
/Ernst.

Received: December 18, 2019
Status: Posted
Posted: January 29, 2020
Tracking No. 1k3-9dy4-dnfu
Comments Due: January 29, 2020
Submission Type: API

Hartog, Kyle

Docket: NIST-2019-0004
Request for Comments on FIPS 186-5 and SP 800-186

Comment On: NIST-2019-0004-0001
Request for Comments on FIPS 186-5 and SP 800-186

Document: NIST-2019-0004-0004
Comment on FR Doc # 2019-23742

Submitter Information

Name: Kyle Den Hartog
Email: kyle.denhartog@mattr.global
Organization: MATTR

General Comment

At MATTR we're concerned about the exclusion of the now popularly used curves secp256k1 and Curve25519 missing from these documents. The prominent use throughout the blockchain space should be an encouraging factor for FIPS 186-5 and SP 800-186 to additionally support these curves. We believe that FIPS 186-5 and SP 800-186 should add these curves to make it more likely that FIPS compliant hardware emerges. The inclusion of these curves would support blockchain solutions which would impact many different industries in a positive manor. As more and more capabilities in finance, Identity and access management, cybersecurity, governments agencies, and other major industries began working with Ethereum and Bitcoin it will be especially advantageous to support the use of these curves. We urge the authors of FIPS 186-5 and SP 800-186 to support the addition of secp256k1 and Curve25519 for key agreements to make compatibility and support of strong software implementations and hardware support more likely.

Ireland, Marc

From: Marc Ireland marc.ireland@nxp.com
Sent: Monday, January 27, 2020 9:08 AM
To: fips186-comments fips186-comments@nist.gov
Subject: Comment on Draft FIPS 186-5

Hello,

Attached please find comments from NXP Semiconductors on draft FIPS 186-5. Please confirm receipt as soon as is convenient.

Thank you,

Marc Ireland
Certifications Expert
NXP Semiconductors

FIPS 186-5

Physical attacks

The draft writes (Section 7.1) “Care must be taken to protect implementations against attacks such as side-channel attacks and fault attacks”.

In order to aid the practitioners who care about e.g. fault resistance why not follow the advice from Section 4.2 of [7]? By having the choice to randomize the signature algorithm many of the presented attacks are prevented or at least getting much harder.

This means including additional random nonce in the hash computation (Step 2, Section 7.6 of the NIST FIPS 186-5 draft). Adding some randomness does not change the proposed verification algorithm, does not weaken security and one can still do unit testing by using a constant value. Moreover, noise from a poor random number generator will not harm the security of the signature scheme. When no such protection is needed this additional random nonce can be constant or omitted.

This does mean, however, that the scheme loses the deterministic signature property.

Same remark holds for the ECDSA deterministic signature .

Cofactorless EdDSA Signature Verification

In [EdDsaCofVer], the authors propose a cofactorless verification of the EdDSA signature. Shouldn't this cofactorless verification be an option proposed in FIPS 186-5?

[EdDsaCofVer]: Daniel J. Bernstein, Simon Josefsson, Tanja Lange, Peter Schwabe and Bo-Yin Yang, “EdDSA for more curves", 2015.07.04 (https://pure.tue.nl/ws/portalfiles/portal/3850274/375386888374129.pdf)

Small subgroup attack

Small subgroup attacks are applicable to curves with a cofactor > 1. Such curves are referenced in SP800-185 (e.g. W-25519), therefore a small sub-group check shall be performed in the ECDSA algorithm described in FIPS 186-5.

E448

It is not clear what the curve E448 specified in SP800-186 shall be used for. If it shall be used in the EdDSA scheme, then additional information needs to be specified (choice of hash function, point encoding mechanism, etc.)

SP800-186

Correspondence between curves (Appendix B)

It should be made clear that the correspondence between twisted Edwards curves and Montgomery curves does not hold for all curves, only in the case a is a square and d a non-square (I1372).

Besides the correspondence between some curves is missing (e.g. correspondence between Curve25519 and W25519, between Edwards448 and Curve448)

Typo

In FIPS 186-5:

- Page 10, paragraph 2: "the public key needs"
- Remove DSA from document (e.g. figure 2)
- Section 5.2 RSA Key Pair management. Point 5: remove domain parameters
- Section 5.4.1, correct "defination"
- Section 6.4.2, ECDSA Signature Verification Algorithm, step 4.
  Compute s−1 = (1/s) (mod n) using the routine in Appendix C.1. To be replaced by Appendix B.1
- Section 7.7: [2c * S]G = [2c]R + (2c * t)Q should be [2^c * S]G = [2^c]R + [2^c * t]Q
- Section 7.2 Encoding: The multiplication of 2^8 and h[1] and 2^248 and h[31] seem to use different notation. The first operator is not defined in Section 2.3.
- Section A.1.2.2 : reference to C.10 should probably be B.10
- We suggest the Section 3 to be reworked. For instance page 10, "For both the signature generation and verification processes, the message (i.e., the signed data) is converted to a fixed-length representation of the message by means of an approved hash function. Both the original message and the digital signature are made available to a verifier."
  It is not completely correct, since for the pure EdDSA there is no hashing of the message.

- Reference [7]:
  Ambrose C, Bos JW, Fay B, Joye M, Lochter M, Murray B (2017) Differential Attacks on Deterministic Signatures. Cryptology ePrint Archive preprint. https://ia.cr/2017/975
  was actually published, better to reference
  Christopher Ambrose, Joppe W. Bos, Björn Fay, Marc Joye, Manfred Lochter and Bruce Murray: Differential Attacks on Deterministic Signatures. RSA Conference Cryptographers' Track - CT-RSA, Lecture Notes in Computer Science 10808, pp. 339–353, Springer, 2018.
- "Section 7.7, first step of "Process": The variable "s" conflicts with the integer "s" in Step 4 of "Process" in Section 7.6. This conflict should be resolved.

Markowitz, Michael

From: Michael Markowitz <markowitz@infoseccorp.com>
Sent: Tuesday, January 28, 2020 1:24 PM
To: SP800-186-comments <sp800-186-comments@nist.gov>
Subject: question on E448 in SP800-186 draft

Folks: I can't find comments on the draft posted online, so please pardon me if this has been previously discussed …

First I want to note the typo in the value for the y-coordinate of the generator of E448 in section 4.2.3.3: there an extraneous 'L' at the end of line 983.

Second, I want to ask whether the order, n, of the subgroup generated by (Gx,Gy) has been validated: when I compute n(Gx,Gy), I get the point (0,-1) of order 2, rather than the identity element (0,1). I find this very strange… Is my code (which works fine with the parameters for Edwards25519 and Edwards448) flawed, or is the value of n in the draft (and in RFC7748) incorrect?

```
n =
181709681073901722637330951972001133588410340171829515070372549795146003961539585716195755291692375963310293709091662304773755859649779
  = 0x3fffffffffffff fffffffffffffff ffffffffffffff ffffffff7cca23e9
c44edb49aed63690 216cc2728dc58f55 2378c292ab5844f3
  = 2^446 - 0x8335dc163bb124b65129c96fde933d8d723a70aadc873d6d54a7bb0d
```

Thanks in advance for your response!

Regards,
Michael

==========
Michael J. Markowitz, Ph.D.
Vice President R&D
Information Security Corporation
1011 Lake Street, Suite 425
Oak Park, IL 60301

Email: markowitz@infoseccorp.com
Office: 708-445-1704
Direct: 708-872-0962
Fax: 708-445-9705
WWW: http://www.infoseccorp.com

Received: November 13, 2019
Status: Posted
Posted: January 29, 2020
Tracking No. 1k3-9dag-qwp5
Comments Due: January 29, 2020
Submission Type: Web

Mattsson, John

Docket: NIST-2019-0004
Request for Comments on FIPS 186-5 and SP 800-186

Comment On: NIST-2019-0004-0001
Request for Comments on FIPS 186-5 and SP 800-186

Document: NIST-2019-0004-0002
Comment on FR Doc # 2019-23742

Submitter Information

Name: John Prue Mattsson
Email: john.mattsson@ericsson.com

General Comment

Dear NIST,

Thanks for your continuous efforts to produce well-written open-access security documents. These are two very important and well-written document. Comments submitted in two parts as there is a limit of 5000 characters.

High level comments:

- Excellent that NIST is adding Montgomery and Edwards curves from RFC 7748. Because of their excellent performance, these curves have already found substantial use in various industries for key exchange, ECIES, and signatures in deployed systems (TLS), new standards (IMSI protection in 5G), and upcoming standards like TLS ESNI, Group OSCORE, and EDHOC.

- We are fine with NIST deprecating binary curves and DSA. While they are quite many libraries that support them, we do not know of any deployment that are actually using them.

Comments on SP 800-186

- Line 160: Elliptic curves over binary field are deprecated, but still included in the document. What does this mean in practice? When is use allowed and when is it not allowed? This should be explained.

- Line 164: Specification of new Montgomery and Edwards curves, which are detailed in Elliptic

Curves for Security [RFC 7748]. These curves are only to be used with the EdDSA

This make it seems like the Weierstrass curve W-25519 can be used for anything, the Edwards curve Edwards25519 can be used for EdDSA and the Montgomery curve cannot really be use for anything. Are Montgomery curves specified only as a way to use a Montgomery code library for EdDSA? For industry use cases, we would like to use the Montgomery curve Curve25519 as much as possible for key exchange (e.g. in TLS) and for hybrid encryption like in ECIES.

- Line 286: have garnered academic interest.

These curves already have substantial deployment in various industries.

- Section 4.1.2: The document lets the reader calculate the security strengths themselves from the curve parameters and SP 800-57. It would be easier for the reader if the document listed the security strengths of the curves instead of forcing the reader to calculate them.

- Section 4.1.2: Following the security strength calculation, W-25519, Curve25519 and Edwards25519 has a security strength of only 112 as n is 255 bits. Algorithms with a 112 bit security strength are only approved to be used beyond 2030 minus the number of year of protection needed. We strongly suggest that NIST changes the security strength calculations and approve W-25519, Curve25519, and Edwards25519 as having 128 bit security strength.

- Line 481: For each curve size range, the following curves are given

This is not true as Edwards and Montgomery curves are not given for all ranges. Also the Weierstrass curves W-25519 and W-448 seem special rather than pseudorandom.

Best Regards,
John Preu Mattsson, Senior Specialist, Ericsson

Received: January 29, 2020
Status: Posted
Posted: January 30, 2020
Tracking No. 1k4-9epy-d584
Comments Due: January 29, 2020
Submission Type: Web

Patil, Harsh

Docket: NIST-2019-0004
Request for Comments on FIPS 186-5 and SP 800-186

Comment On: NIST-2019-0004-0001
Request for Comments on FIPS 186-5 and SP 800-186

Document: NIST-2019-0004-0008
Comment on FR Doc # 2019-23742

Submitter Information

Name: Harsh Kupwade Patil
Email: harsh.patil@lge.com

General Comment

See attached file(s)

Attachments

Smith, David

From: Smith, David E. <David.Smith@cyber.gc.ca>
Sent: Thursday, January 30, 2020 4:52 PM
To: SP800-186-comments <sp800-186-comments@nist.gov>
Subject: Cyber Centre comments on SP 800-186 (Draft)

Please find below our editorial and technical comments on the Draft SP 800-186 issued for comment in October 2019.

David Smith
Canadian Centre for Cyber Security

Page, section, paragraph	Type	Comment
8, 4, 7	Editorial	The formatting of angle brackets on <P> on line 409 should be to be consistent with lines 408 and 413.
10, 4.1.3, *Polynomial Basis*	Editorial	The explanation of how to choose the pentanomial states that "the second term t^a has the lowest degree m". It should read "the second term t^a has the lowest degree among all irreducible pentanomials of degree m". Also the phrase "the third term t^b has the lowest degree among all irreducible pentanomials of degree m and the second term t^a …" should read "the third term t^b has the lowest degree among all irreducible pentanomials of degree m with the second term t^a …".
26, 4.3.1.4, *f*(*z*)	Editorial/ Technical	The definition of *f*(*z*) on line 1103 is incorrect. It should be *f(z) = z^{409} + z^{87} + 1*. See ANSI X9.142 Table B.3 for reference.