Internet Engineering Task Force S. Shen, Ed.
Internet-Draft X. Lee, Ed.
Intended status: Standards Track Chinese Academy of Science
Expires: April 26, 2012 October 24, 2011

SM2 Digital Signature Algorithm draft-shen-sm2-ecdsa-00

Abstract

This document discribles an Digital Signature Algorithm based on elliptic curves which is invented by Xiaoyun Wang et al. This digital signature algorithm is published by Chinese Commercial Cryptography Administration Office for the use of electronic authentication service system.

The document *** published by Chinese Commercial Cryptography Administration Office includes four parts: general introdocution, Digital Signature Algorithm, Key Exchange Protocol and Public Key Encryption Algorithm. This document only gives the general introduction and digital signature algorithm.

Status of This Memo

This Internet-Draft is submitted to IETF 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 April 26, 2012.

Table of Contents

1. Introduction

This document is mainly the tranlation of the algorithm published by Chinese Commercial Cryptography Administration Office for the convenience of IETF and IRTF community. The credit of inventing this algorithm goes to the authors of the algorithm.

2. Conventions Used in this Document

The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY" in this document are to be interpreted as defined in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

3. Symbols and Terms

3.1. Symbols

a, b	Elements in finite field Fq and they defines a Elliptic Curve E over Fq
B	The MOV threshold. This is a positive integer B such that taking discrete logarithms over GF (q^B) is judged to be at least as difficult as taking elliptic discrete logarithms over GF (q).
deg(f)	The degree of a polynomial f(x)
E	The elliptic curve defined by a and b over a finite field Fq
E(Fq)	The set of all the rational points of E
#E(Fq)	The number of elements in E(Fq), the degree of elliptic curve E(Fq)
ECDLP	Elliptic Curve Discrete Logarithm Problem
Fp	A prime field with p elements
Fq	A prime field with q elements
F*q	The multiplicative group composed of all non-zero elememnts in Fq
F2^m	The binary field extension with 2^m elements
G	A base point on the elliptic curve E, with prime order
gcd(x;y)	The greatest common devisor of x and y
h	The cofactor h= #E(Fp)/n, where n is the degree of a base point G
LeftRotate( )	The operation of Rotation to left
lmax	The upper limit of the largest prime factor of the cofactor h
m	The extention degree of the field F2^m over the binary field F2
modf(x)	The operation module the polynomial f(x). All the coefficients mod 2 when f(x) is a polynomial over F2.
modn	The operation of modulo n, for example, 23 mod7 = 2
n	The degree of a base point G (n is a prime factor of #E(Fq)
O	The point of infinity (or zero) on the elliptic curre E.
P	A point P on the elliptic curre E which is not O. The coordinates xP and yP satisfies the elliptic curve equation
P1+P2	The summation of the two points P1 and P2 on elliptic curve E

p A prime number greater than 3
q The number of elements in the finite field Fq
rmin The lower limit of the degree n of a base point G
Tr( ) The trace function
xP The x-coordinate of the point P
yP The y-coordinate of the point P
x^(-1) The only y such that x*y=1 (modn), 1 < = y < = n, gcd(x, n)=1
x||y The concatenation of x and y, where x and y are bit string or byte string
x == y (modn) x modn = y modn
** y~P The point compression expression of yP
Zp The ring of integers modulo p
< G > The cyclic group generated by base point G
[k]P The k multiple of a point P over elliptic curve, where k is a positive integer
[x;y] The set of integers which greater than or equal to x and less than or equal to y
/x\ The smallest integer greater than or equal to x, for example GBP[not]/7\=7, /8.3\=9
\x/ The largest integer less than or equal to x, for example GBP[not]\7/ =7, \8.3/=8
XOR The exclusive-or operation of two bit strings or byte strings of same length
****
A,B The two users using the public key system
a, b Elements in finite field Fq and they defines a Elliptic Curve E over Fq
dA The private key of the user A
E(Fq) The set of all the rational points of E
e The hash of message M
e' The hash of message M'
Fq A prime field with q elements
G A base point on the elliptic curve E, with prime order
Hv( ) The hash function with output of legnth v bits
IDA The identifier of user A
M The message for signature
M!ae The message for verification
modn The operation of modulo n, for example, 23 mod7 = 2
n The degree of base point G (n is a prime factor of #E(Fq))
O The point of infinity (or zero) on the elliptic curre E
PA The public key of user A
q The number of elements in the finite field Fq
x||y The concatenation of x and y, where x and y are bit string or byte string
ZA The identifier of user A, part of parameters of elliptic curve and hash value of PA
(r,s) The sent signature
(r',s') The received signature

[k]P The k multiple of a point P over elliptic curve, where k is a positive integer
[x;y] The set of integers which greater than or equal to x and less than or equal to y
/x\ The smallest integer greater than or equal to x, for example GBP[not]/7\=7, /8.3\=9
\x/ The largest integer less than or equal to x, for example GBP[not]\7/=7, \8.3/=8
#E(Fq) The number of elements in E(Fq), the degree of elliptic curve E(Fq)
****

3.2. Terms

The following terms are used in this document.

digital signature
: The metadata over some data. It should provide authentication, integrity protection and non repudiation. [ANSI X9.63-2001]

message
: The bits string of arbitary length. [ISO/IEC 15946-4 3.7]

signed message
: The data composed of a message and its digital signature. [ISO/IEC 15946-4 3.14]

key
: A parameter for cryptographic calculation. It was used for encryption or decryption, shared sectet and verification of digital signature. [ANSI X9.63-2001]

4. General Introdoction to ECC

TBD

5. Digital Signature Algorithm

5.1. Digital Signature System

5.1.1. General Rules

In the digital signtature algorithm, one signer generate digital signature over given data and one verifier verifies the validation of the signature. Each signer ownes one public key and one private key. The private key was used for signing and verifier verfifies the signature using the public key. Before generation of the digital signature, the message M and ZA need to be compressed via a hash function; before the verification of the digital signature, the message M' and ZA need to be compressed via a hash function.

5.1.2. Parameters of Elliptic Curve System

The parameters of an elliptic curve systme include the size q of a finite field Fq (when q=2^m, also include basis representation and irreducible polynomial); the two elements a and b (in Fq) which defines the elliptic curve equation; the base point G=(xG, yG) (G not

euqals O), where xG and yG are ellements in Fq; the degree n of G and other optional parameter such as cofactor h.

5.1.3. Key pairs

The user A's key pair include his private key dA and public key PA=[dA]G=(xA, yA).

5.1.4. Auxilary Functions

5.1.4.1. Introduction

The auxilary functions in the elliptic curve digital signature algorithm in this document include hash algorithm and random number generator.

5.1.4.2. Hash Functions

The sm2 digital signature algorithm requires the hash functions approved by Chinese Commercial Cryptography Administration Office, such as sm3.

5.1.4.3. Random Number Generator

The sm2 digital signature algorithm requires random number generators approved by by Chinese Commercial Cryptography Administration Office.

5.1.4.4. Other User Information

As teh signer, User A has the identifier IDA of length entlenA bits, denote ENTLA as the two bytes transformed from the integer entlenA. In the digital signature algorithms in this document, both signer and verifier need to obtain ZA by calculating the hash value of ZA.

ZA=H256(ENTLA || IDA || a || b || xG || yG || xA || yA)

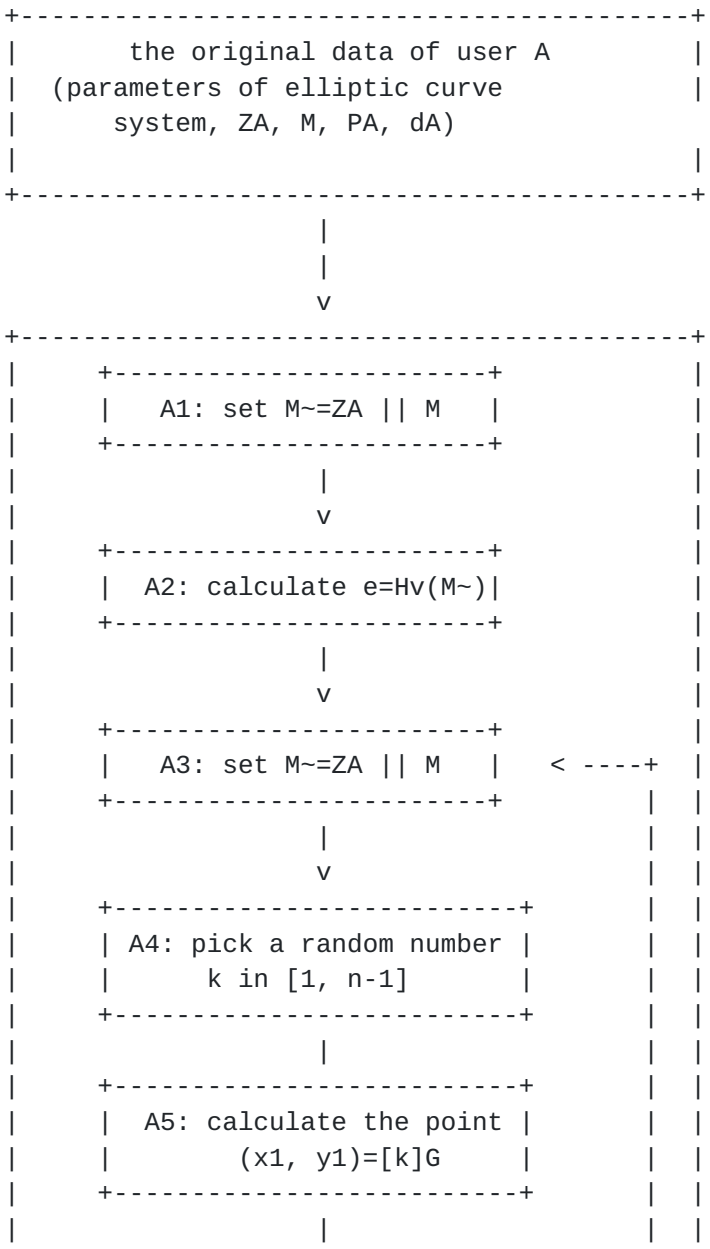
5.2. Generation of Signature

5.2.1. Digital Signature Generation Algorithm

Let M be the message for signing, in order to obtain the signature (r, s), the signer A need to perform the following:

A1: set M~=ZA || M
A2: calculate e=Hv(M~)
A3: pick a random number k in [1, n-1] via a random number generator
A4: calculate the elliptic curve point (x1, y1)=[k]G
A5: calculate r=(e+x1) modn, return to A3 if r=0 or r+k=n
A6: calculate s=((1+dA)^(-1)*(k-r*dA)) modn, return to A3 if s=0
A7: the digital signature of M is (r, s)

5.2.2. Flow Chart of Digital Signature Generation



| v | |

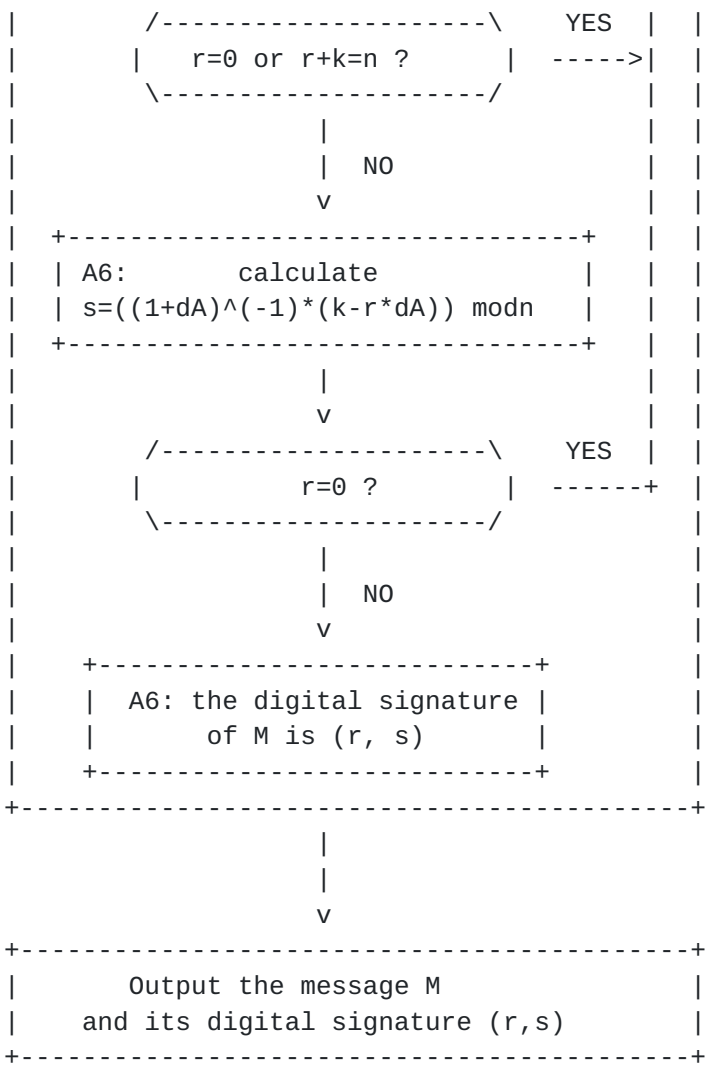


Figure 1: Flow Chart of Digital Signature Generation

5.3. Verification of Signature

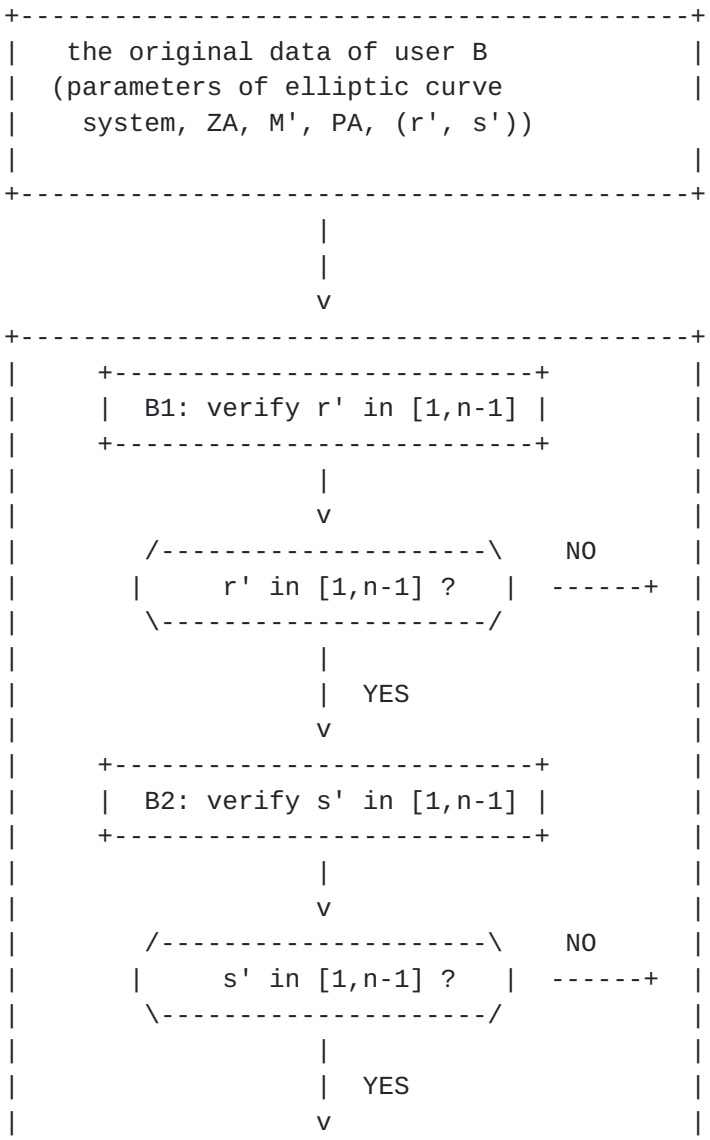
5.3.1. Digital Signagure Vefification Algorithm

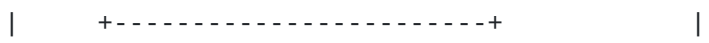
To verfify the received message M' and it's digital signature, the verifier need to perform the following:

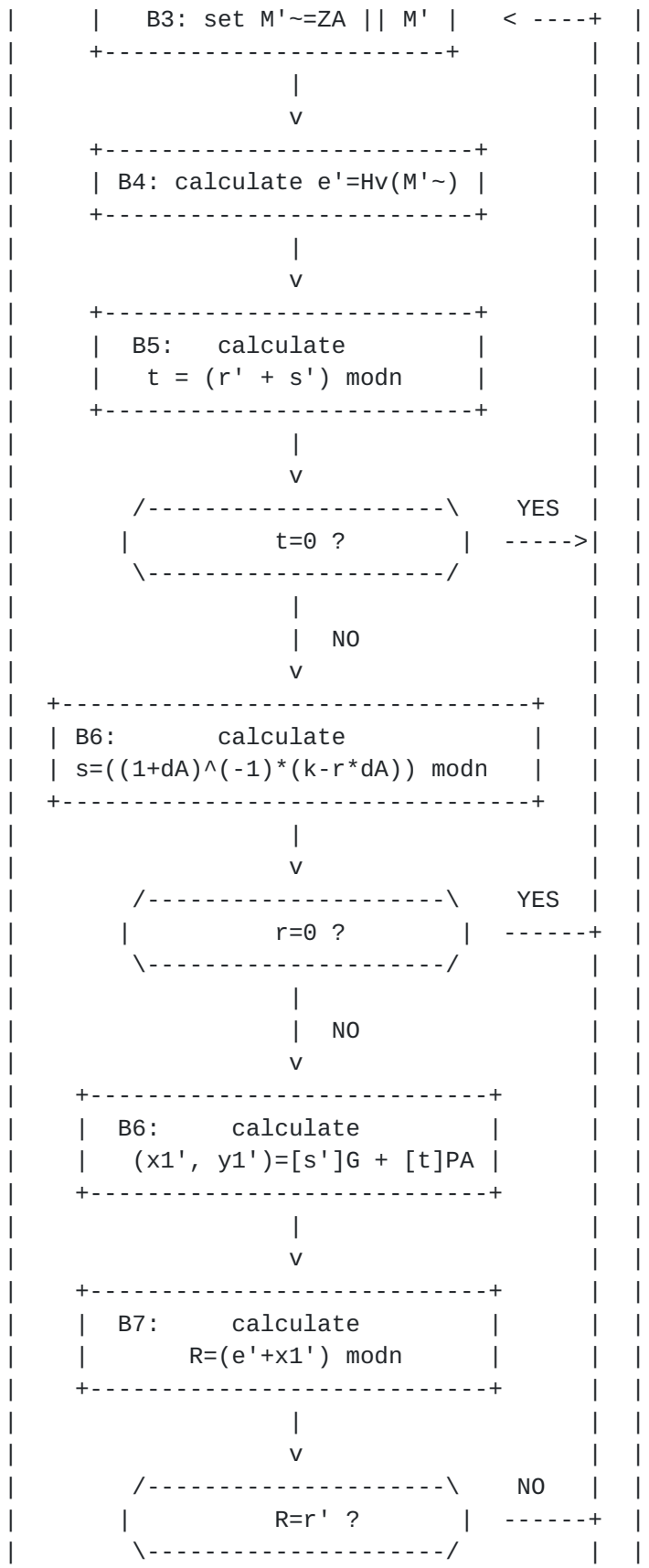
B1: verify whether r' in [1,n-1], verification failed if not
B2: vefify whether s' in [1,n-1], verification failed if not
B3: set M'~=ZA || M'
B4: calculate e'=Hv(M'~)
B5: calculate t = (r' + s') modn, verification failed if t=0
B6: calculate the point (x1', y1')=[s']G + [t]PA
B7: calculate R=(e'+x1') modn, verfication pass if yes, otherwise failed

Note: The verification will certainly fail if ZA does not correspond to
teh hash value of A.

5.3.2. Flow Chart of Digital Signature Verification






B3: set M'~=ZA || M'
B4: calculate e'=Hv(M'~)
B5: calculate t = (r' + s') modn
t=0 ?
YES
NO
B6: calculate s=((1+dA)^(-1)*(k-r*dA)) modn
r=0 ?
YES
NO
B6: calculate (x1', y1')=[s']G + [t]PA
B7: calculate R=(e'+x1') modn
R=r' ?
NO

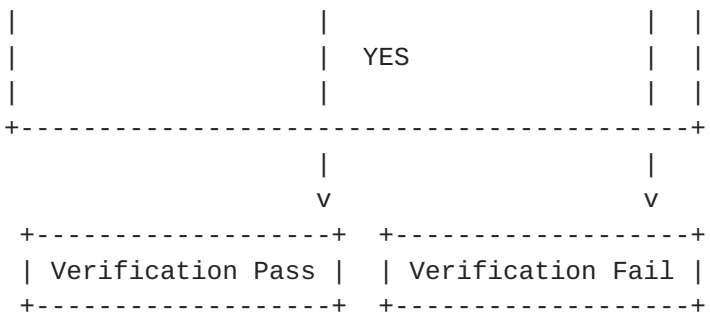


Figure 2: Flow Chart of Digital Signature Verification

6. References

6.1. Normative References

[RFC1341] Borenstein, N. and N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies", RFC 1341, June 1992.

6.2. Informative References

[RFC2049] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples", RFC 2049, November 1996.

Appendix A. Example

A.1. General Introduction

This appendix uses the hash algorithm described in draft-shen-sm3-hash-00, which applies on a bit string of length less than 2^54 and output a hash value of size 256, denotes as H256( ).

In this appendix, all the hexadecimal number has high digits on the left and low digits on teh right.

In this appendix, all the messages are in ASCII code.

Let the user A's identity be: ALICE123@YAHOO.COM. Denoted in ASCII code IDA:

414C 49434531 32334059 41484F4F 2E434F4

ENTLA=0090.

A.2. Digital Signature of over E(Fp)

The elliptic curve equationi is:

y^2 = x^3 + ax + b

Example 1: Fp-256
A Prime p:
8542D69E 4C044F18 E8B92435 BF6FF7DE 45728391 5C45517D 722EDB8B 08F1DFC3

The coefficient a:
787968B4 FA32C3FD 2417842E 73BBFEFF 2F3C848B 6831D7E0 EC65228B 3937E498

The coefficient b:
63E4C6D3 B23B0C84 9CF84241 484BFE48 F61D59A5 B16BA06E 6E12D1DA 27C5249A

The base point G=(xG,yG)GBP[not]whose degree is n:
x-coordinate xG:
421DEBD6 1B62EAB6 746434EB C3CC315E 32220B3B ADD50BDC 4C4E6C14 7FEDD43D
y-coordinate yG:
0680512B CBB42C07 D47349D2 153B70C4 E5D7FDFC BFA36EA1 A85841B9 E46E09A2
degree n:
8542D69E 4C044F18 E8B92435 BF6FF7DD 29772063 0485628D 5AE74EE7 C32E79B7

The message M to be signed:message digest

The private key dA:
128B2FA8 BD433C6C 068C8D80 3DFF7979 2A519A55 171B1B65 0C23661D 15897263

The public key PA=(xA,yA):
x-coordinate xA:
0AE4C779 8AA0F119 471BEE11 825BE462 02BB79E2 A5844495 E97C04FF 4DF2548A
y-coordinate yA:
7C0240F8 8F1CD4E1 6352A73C 17B7F16F 07353E53 A176D684 A9FE0C6B B798E857

Hash value ZA=H256(ENTLA || IDA || a || b || xG || yG || xA || yA)

ZA:
F4A38489 E32B45B6 F876E3AC 2168CA39 2362DC8F 23459C1D 1146FC3D BFB7BC9A

The intermediate value during signing processing:
M~=ZA || M:
F4A38489 E32B45B6 F876E3AC 2168CA39 2362DC8F 23459C1D 1146FC3D BFB7BC9A
6D657373 61676520 64696765 7374
hash value e=H256(M):
B524F552 CD82B8B0 28476E00 5C377FB1 9A87E6FC 682D48BB 5D42E3D9 B9EFFE76

```
random number k:
6CB28D99 385C175C 94F94E93 4817663F C176D925 DD72B727 260DBAAE 1FB2F96F
point (x1,y1)=[k]G:
x-coordinate x1:
110FCDA5 7615705D 5E7B9324 AC4B856D 23E6D918 8B2AE477 59514657 CE25D112
y-coordinate y1:
1C65D68A 4A08601D F24B431E 0CAB4EBE 084772B3 817E8581 1A8510B2 DF7ECA1A
r=(e+x1) modn:
40F1EC59 F793D9F4 9E09DCEF 49130D41 94F79FB1 EED2CAA5 5BACDB49 C4E755D1
(1 + dA)^(-1)
79BFCF30 52C80DA7 B939E0C6 914A18CB B2D96D85 55256E83 122743A7 D4F5F956
s = ((1 + dA)^(-1)1 * (k - r * dA)) modn:
6FC6DAC3 2C5D5CF1 0C77DFB2 0F7C2EB6 67A45787 2FB09EC5 6327A67E C7DEEBE7

Digital Signature of the message M: (r,s)
r:
40F1EC59 F793D9F4 9E09DCEF 49130D41 94F79FB1 EED2CAA5 5BACDB49 C4E755D1
s:
6FC6DAC3 2C5D5CF1 0C77DFB2 0F7C2EB6 67A45787 2FB09EC5 6327A67E C7DEEBE7

The intermediate value during verification processing:
hash value e' = H256(M'~):
B524F552 CD82B8B0 28476E00 5C377FB1 9A87E6FC 682D48BB 5D42E3D9 B9EFFE76
t=(r!ae+s!ae) modn:
2B75F07E D7ECE7CC C1C8986B 991F441A D324D6D6 19FE06DD 63ED32E0 C997C801
point (x0!ae, y0')=[s']G:
x-coordinate x0':
7DEACE5F D121BC38 5A3C6317 249F413D 28C17291 A60DFD83 B835A453 92D22B0A
y-coordinate y0':
2E49D5E5 279E5FA9 1E71FD8F 693A64A3 C4A94611 15A4FC9D 79F34EDC 8BDDEBD0
point (x00', y00')=[t]PA:
x-coordinate x00':
1657FA75 BF2ADCDC 3C1F6CF0 5AB7B45E 04D3ACBE 8E4085CF A669CB25 64F17A9F
y-coordinate y00':
19F0115F 21E16D2F 5C3A485F 8575A128 BBCDDF80 296A62F6 AC2EB842 DD058E50
point (x1', y1')=[s']G + [t]PA:
x-coordinate x1':
110FCDA5 7615705D 5E7B9324 AC4B856D 23E6D918 8B2AE477 59514657 CE25D112
y-coordinate y1':
1C65D68A 4A08601D F24B431E 0CAB4EBE 084772B3 817E8581 1A8510B2 DF7ECA1A
R = (e' + x1') modn:
40F1EC59 F793D9F4 9E09DCEF 49130D41 94F79FB1 EED2CAA5 5BACDB49 C4E755D1
```

A.3. Digital Signature of over E(F2^m)

The elliptic curve equationi is:

y^2 + xy = x^3 + ax + b

Example 1: F2^m -257
The polynomial to generate base field is: x^257 + x^12 + 1

The coefficient a:
0

The coefficient b:
00 E78BCD09 746C2023 78A7E72B 12BCE002 66B9627E CB0B5A25 367AD1AD 4CC6242B

The base point G=(xG,yG)GBP[not]whose degree is n:
x-coordinate xG:
00 CDB9CA7F 1E6B0441 F658343F 4B10297C 0EF9B649 1082400A 62E7A748 5735FADD

y-coordinate yG:
01 3DE74DA6 5951C4D7 6DC89220 D5F7777A 611B1C38 BAE260B1 75951DC8 060C2B3E

degree n:
7FFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF BC972CF7 E6B6F900 945B3C6A 0CF6161D

The message M to be signed:message digest

The private key dA:
771EF3DB FF5F1CDC 32B9C572 93047619 1998B2BF 7CB981D7 F5B39202 645F0931

The public key PA=(xA,yA):
x-coordinate xA:
01 65961645 281A8626 607B917F 657D7E93 82F1EA5C D931F40F 6627F357 542653B2
y-coordinate yA:
01 68652213 0D590FB8 DE635D8F CA715CC6 BF3D05BE F3F75DA5 D5434544 48166612

Hash value ZA=H256(ENTLA || IDA || a || b || xG || yG || xA || yA)
ZA:
26352AF8 2EC19F20 7BBC6F94 74E11E90 CE0F7DDA CE03B27F 801817E8 97A81FD5

The intermediate value during signing processing:
M~=ZA || M:
26352AF8 2EC19F20 7BBC6F94 74E11E90 CE0F7DDA CE03B27F 801817E8 97A81FD5
6D657373 61676520 64696765 7374

hash value e=H256(M~):
AD673CBD A3114171 29A9EAA5 F9AB1AA1 633AD477 18A84DFD 46C17C6F A0AA3B12
random number k:
36CD79FC 8E24B735 7A8A7B4A 46D454C3 97703D64 98158C60 5399B341 ADA186D6
point (x1,y1)=[k]G:
x-coordinate x1:

00 3FD87D69 47A15F94 25B32EDD 39381ADF D5E71CD4 BB357E3C 6A6E0397 EEA7CD66
y-coordinate y1:
00 80771114 6D73951E 9EB373A6 58214054 B7B56D1D 50B4CD6E B32ED387 A65AA6A2
r=(e+x1) modn:
6D3FBA26 EAB2A105 4F5D1983 32E33581 7C8AC453 ED26D339 1CD4439D 825BF25B
(1 + dA)^(-1)
73AF2954 F951A9DF F5B4C8F7 119DAA1C 230C9BAD E60568D0 5BC3F432 1E1F4260
s = ((1 + dA)^(-1)1 * (k - r * dA)) modn:
3124C568 8D95F0A1 0252A9BE D033BEC8 4439DA38 4621B6D6 FAD77F94 B74A9556

Digital Signature of the message M: (r,s)
r:
6D3FBA26 EAB2A105 4F5D1983 32E33581 7C8AC453 ED26D339 1CD4439D 825BF25B
s:
3124C568 8D95F0A1 0252A9BE D033BEC8 4439DA38 4621B6D6 FAD77F94 B74A9556

The intermediate value during verification processing:
hash value e' = H256(M'~):
AD673CBD A3114171 29A9EAA5 F9AB1AA1 633AD477 18A84DFD 46C17C6F A0AA3B12
t=(r!ae+s!ae) modn:
1E647F8F 784891A6 51AFC342 0316F44A 042D7194 4C91910F 835086C8 2CB07194
point (x0!ae, y0')=[s']G:
x-coordinate x0':
00 252CF6B6 3A044FCE 553EAA77 3E1E9264 44E0DAA1 0E4B8873 89D11552 EA6418F7
y-coordinate y0':
00 776F3C5D B3A0D312 9EAE44E0 21C28667 92E4264B E1BEEBCA 3B8159DC A382653A
point (x00', y00')=[t]PA:
x-coordinate x00':
00 07DA3F04 0EFB9C28 1BE107EC C389F56F E76A680B B5FDEE1D D554DC11 EB477C88
y-coordinate y00':
01 7BA2845D C65945C3 D48926C7 0C953A1A F29CE2E1 9A7EEE6B E0269FB4 803CA68B
point (x1', y1')=[s']G + [t]PA:
x-coordinate x1':
00 3FD87D69 47A15F94 25B32EDD 39381ADF D5E71CD4 BB357E3C 6A6E0397 EEA7CD66
y-coordinate y1':
00 80771114 6D73951E 9EB373A6 58214054 B7B56D1D 50B4CD6E B32ED387 A65AA6A2
R = (e' + x1') modn:
6D3FBA26 EAB2A105 4F5D1983 32E33581 7C8AC453 ED26D339 1CD4439D 825BF25B

Authors' Addresses

Sean Shen (editor)
Chinese Academy of Science
No.4 South 4th Zhongguancun Street
Beijing, 100190
China

Phone: +86 10-58813038
EMail: shenshuo@cnnic.cn

Xiaodong Lee (editor)
Chinese Academy of Science
No.4 South 4th Zhongguancun Street
Beijing, 100190
China

Phone: +86 10-58813038
EMail: shenshuo@cnnic.cn