symmetry

MDPI

Article

[3]Revised Paper: A Modified Wei-Hua-He Digital Signature Scheme Based on Factoring and Discrete Logarithm

Elumalai R † and G. S. G. N. Anjaneyulu *,†

Vellore Institute of Technology, Vellore 632014, India
* Correspondence: anjaneyulu.gsgn@vit.ac.in
† These authors contributed equally to this work.

Abstract: A symmetric cipher such as AES in cryptography is much faster than an asymmetric cipher but digital signatures often use asymmetric key ciphers because they provides the sender's identity and data integrity. In this paper, a modified-He digital signature scheme is proposed using a one-way hash function. The proposed scheme, unlike the He signature technique, employs Euclid's Division Lemma with large prime moduli p. Its security is built on large integer factoring, discrete logarithms and expanded root problems. The time complexity of the proposed scheme is $\mathcal{O}(log^3 p)$. The proposed modified-He scheme is efficient, as evidenced by the analytical results with key lengths greater than 512 bits.

Keywords: cryptography; digital signature; discrete logarithm; Euclid's Division Lemma

check for updates

Citation: R, E.; Anjaneyulu, G.S.G.N. [3]Revised Paper: A Modified Wei-Hua-He Digital Signature Scheme Based on Factoring and Discrete Logarithm. *Symmetry* **2022**, *14*, 2443. https://doi.org/10.3390/sym14112443

Academic Editor: Evgeny Nikulchev

Received: 29 September 2022
Accepted: 1 November 2022
Published: 17 November 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

The purpose of cryptography is to design and analyze protocols that prohibit third parties or the general public from reading private communications. The three prevalent categories of cryptography are symmetric cryptography, asymmetric cryptography and protocols. In this paper, a modified-He digital signature scheme is proposed using a one-way hash function. The paper focuses mainly on asymmetric key ciphers because they satisfy most of the security service requirements. The cryptography classification is shown in Figure 1. No further algorithmic requirements are required for design of protocols as symmetric and asymmetric cryptosystems are sufficient. In general, a stream cipher is faster than a block cipher. A5/2 is a stream cipher that replaces A5/1, voice used to protect the privacy of voice calls in the GSM cellular telephone protocol.

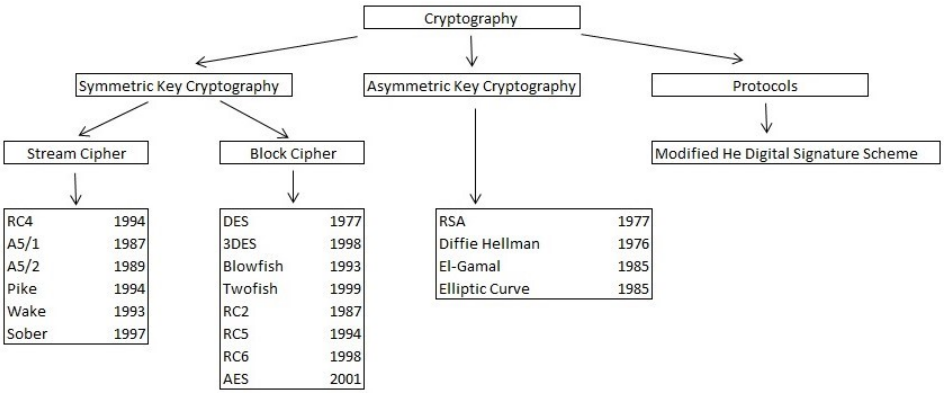

Figure 1. Cryptography classification.

In 1976, Whitfield Diffie and Martin Hellman were the first to explain the concept of a digital signature scheme. In 1977, RSA developed the first asymmetric key family-based digital signature scheme. Its security was based on factoring a large composite integer. In

1985, Elgamal proposed an alternative digital signature scheme, which was based on the discrete logarithm problem. In 1994, Harn presented a new digital signature scheme to improve digital signature security. One must break both the RSA and the Elgamal scheme simultaneously to break the Harn signature scheme. In 1995, Lee showed that if hackers solve the DL problem, they will be able to forge signatures with a high probability.

The suggested bit lengths of the private key and public key algorithms for the security levels are shown in Table 1. RSA and discrete-logarithm techniques need longer keys. The key length of an elliptic curve scheme is considerably shorter, but it is still twice as long as the key length of a symmetric cipher with the same cryptographic strength [1].

Table 1. Public-key and private-key algorithm bit lengths.

Cryptosystems	Cipher	Security Level (bit) 256
Asymmetric key cryptosystem	RSA	15,360 bit
	DH, DSA, Elgamal	15,360 bit
	ECDH, ECDSA	512 bit
Symmetric key cryptosystem	AES, 3DES	256 bit

The authors of the papers [2–9] presented a digital signature scheme based on IFP and DLP. In 2007, Yu-Fang Chung et al. [10] suggested a technique depending on the difficulty of solving the ECDLP. In 2011, Pin-Chang Su [11] introduced the enhanced short signature method, which is based on knapsack and Gap Diffie–Hellman (GDH) groups and whose security is strongly connected to the discrete logarithm assumption. The authors of the papers [12–14] outlined a new digital signature scheme based on two difficult problems. In 2018, the authors of the papers [15–17] developed and presented new digital signature techniques based on the IFP and DLRP over Z_n. In 2020, Xuan et al. [18] proposed a new digital signature scheme based on the difficulty of solving expanded root problems over Z_p. In 2021–2022, the authors of the papers [19–21] presented a digital signature over HMAC entangled chains, the hidden logarithm problem and visually meaningful image encryption algorithm.

The purpose of this study is to get beyond the He-digital signature method's limitations. The He digital signature scheme is based on the group $Z_{(4p_1q_1+1)}$, where p_1 and q_1 are safe prime numbers [8]. Because $p-1=4p_1q_1$ has two larger prime factors, p_1 and q_1, factoring $p-1$ is extremely difficult in practice. As n increases, the likelihood of the prime p_n being a safe prime decreases dramatically. Furthermore, finding an element $g \in Z_p$ such that $|g| = R$, where $R = p_1q_1$ and the primes of the form $p = 4p_1q_1 + 1$ are difficult. As a result, a modified-He digital signature scheme using a one-way hash function is proposed to increase efficiency. The proposed scheme comes under the category of asymmetric key digital signature.

2. Motivation and Outline of the Paper

The digital signature is now the most important part of cryptography because it is used so often. Digital signatures can be used to sign contracts legally, update software in a safer way and make online transactions safe by using digital certificates. It provides integrity, message authentication and nonrepudiation. This paper is organized as follows: Section 3 details the mathematical background: FACT, DLP and expanded root problems; Section 4 presents the proposed algorithm; Section 5 provides a concrete example of the proposed scheme; Section 6 depicts the situation of security attacks; and the last section concludes the contribution of the paper.

3. Security Model

3.1. Factoring Large Integer

Consider the expression

$$x_1 x_2 = a y_1 y_2 + b. \tag{1}$$

where the variable x_1, x_2, y_1 and y_2 are all unknown. Factoring the solution $x_1 x_2$ into x_1, x_2 and $y_1 y_2$ into y_1, y_2 is extremely difficult for large integers. Equation (1) is known as another form of Euclid's Division Lemma [22].

3.2. Discrete Logarithm Problem

It is difficult to find the unique exponent x ($0 \leq x \leq (p-1)$) of g for given integers $y, g \in Z_p$, i.e.,)

$$y \cong g^x (mod\ p). \tag{2}$$

where g is a primitive root and p is a prime number.

3.3. Expanded Root Problem

The following problem is known as a discrete logarithm since a is a constant, b and x are variables [23]

$$x = a^b\ mod\ p.$$

The above problem is called an expanded root problem [18], if both a and b are variables.

4. Proposed a Modified-He Digital Signature Scheme

4.1. Modified-He Digital Signature Scheme

Figure 2 demonstrates that Alice signed her message m using her private key x and Alice's signature (r_1, r_2, s, k) along with the message m is sent through an insecure channel. Later, this signature is validated by Bob.

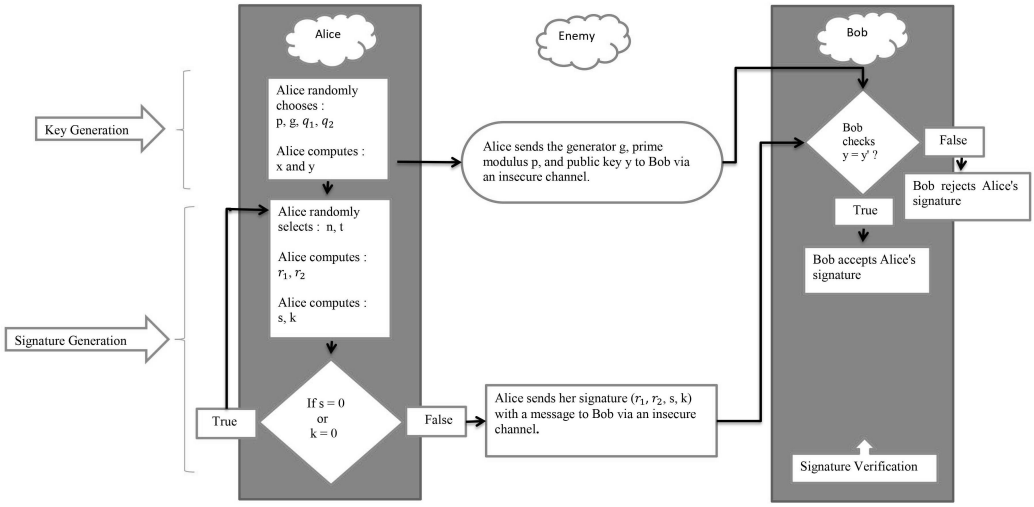

Figure 2. Modified-He Digital Signature Scheme.

4.1.1. Public Parameter

Alice randomly chooses the prime p and she chooses another element $g \in Z_p$ such that $\gcd(g, p) = 1$. Then, the public parameter is the pair (p, g). The three main parts of the digital signature are key generation, signature generation and signature verification, which are as follows.

4.1.2. Key Generation

Alice randomly chooses two distinct primes, q_1 and q_2 ($q_1 < q_2 < p$); she computes

$$x = (q_1 + q_2)^{q_1 q_2} \, mod \, p \tag{3}$$

and

$$y = g^{(q_1 x)^2} mod \, p. \tag{4}$$

Then, Alice's private key is x and the public key is y. The key generation algorithm can be described on Algorithm 1 as:

Algorithm 1 Key Generation Algorithm

Require: (p, g)
Ensure: (p, q_1, q_2, x, y)
1: **generate** p
2: **select** $g : g \in Z_p$
3: **generate** q_1,$q_2 : q_1 < q_2 < p$
4: $x \leftarrow (q_1 + q_2)^{q_1 q_2} \bmod p$
5: $y \leftarrow g^{(q_1 x)^2} \bmod p$
6: **return** (q_1, q_2, x, y)

Wherein:

- p,g: Global parameters.
- x: Alice's Private key.
- y: Alice's Public key.

4.1.3. Signature Generation

To sign a message, m, the following steps are taken:
Alice chooses the random integers $n, t \in Z_P$ for each message m and she computes

$$r_1 = g^{nt} \, mod \, p \tag{5}$$

and

$$r_2 = g^{(nt)^2} \, mod \, p. \tag{6}$$

Then, Alice finds positive integer s and k values satisfying

$$q_1 x = h(m)nts + k. \tag{7}$$

where hashing the message m is denoted by $h(m)$. Alice chooses different n and t values, if either $s = 0$ or $k = 0$. Alice sends the quadruple (r_1, r_2, s, k) associated with m to the Bob. So, the signature algorithm is described in Algorithm 2 as:

Algorithm 2 Signature Generation Algorithm

Require: (p, g, q_1, x)
Ensure: (r_1, r_2, s, k, m)
1: **select** $n, t \in Z_p$
2: $r_1 \leftarrow g^{nt} \bmod p$
3: $r_2 \leftarrow g^{(nt)^2} \bmod p$
4: **choose** the message m
5: $q_1 x \leftarrow h(m)nts + k$
6: **if** (s == 0) or (k == 0) **then**
7: **go to** 1
8: **end if**
9: **return** (r_1, r_2, s, k, m)

Wherein:

- m: The message must be signed.
- (r_1, r_2, s, k): Alice's signature on m.

4.1.4. Signature Verification

Alice verifies $r_1, r_2 \in [1, p-1]$. Then, she computes

$$y' \cong r_1^{2h(m)sk} r_2^{(h(m)s)^2} g^{k^2} \; mod \; p \tag{8}$$

By checking the equality of y and y', Bob can validate the validity of the signature. The signature verification algorithm can be described in Algorithm 3 as:

Algorithm 3 Signature Verification Algorithm

Require: p, g, r_1, r_2, s, k, m
Ensure: $True/False$
1: $y' \leftarrow r_1^{2h(m)sk} \; r_2^{(h(m)s)^2} \; g^{k^2} \bmod p$
2: **if** $(y == y')$ **then**
3: **return** $True$
4: **else**
5: **return** $False$
6: **end if**

If Alice adheres to the guidelines outlined above, the signature is always accepted by Bob. Taking squares on Equation (7) and exponentiating the generator g with the resultant over modulo p proves that $y = y'$.

The time complexity for modular multiplication is $\mathcal{O}(\log^2 p)$, modular exponentiation is $\mathcal{O}(\log^3 p)$. In addition, finding $H(m)$ and k require $\mathcal{O}(\log^3 p)$ [23]. Therefore the time complexity of the proposed scheme for key generation, signature generation and signature verification is $\mathcal{O}(\log^3 p)$.

5. Concrete Example

5.1. Case No.1

For a length greater than 512 bits, choose the positive integer values q_1, q_2, p, x and g.
-Key Generation:
q_1 value:
24204868133185954948052711126594325904950644150970719475783883549526026684120585565950513865485930072746045479777366698672918299160381477023509089521582173311372610467900279664561595681832113078528139
q_2 value:
46173725832913847121436400620212043936761099218517385905639143767656777138039469879991523917154111781240677870551683249265592165846507448502443411532538863488166829223210826165970651547223649717795467
p value:
97609316808472870454405302102049090631926400165007064779331430655320976676509628237857272091298564595016008517939469595141604772331116744670145968787226236724839270110107504464205570942192609743912751
x value:
96927043459506992604160311113571440553541919877615452252875327060463577853683601233201782815133749826662760029498030584804703017548549339102193245959130740336143062341792496995862986391669440371955440
g value:
60013100423475503376399617651347989516045909050522869961906226199920595075171660893963793787893012012968840554284018252356769557437724220116706787567813657613554342599141372868197388513025892460290169

y value:
85357427054426942199227586846504813044343555691588811131491758591820003607591087796535054464276499368983800094634708787100885187797117994310277076638012713825790478365924942604617693838464217803924069
-Signature Generation:
n value:
66060543862913573648226582032349286914035619085069029586786315260850680542030346493608332665962190203
t value:
80530636648297567886476358999055044920161884190114108080522424811596009350204715077423152163199190 49
r_1 value:
70067210427643294166222138993027798549657752697978469744048422500317294658740486555926205285400445471012016076449158573561242986921045773455811824689759844689682710067019646496820350622771577003668315
r_2 value:
63286413639814488044951356532672917048602261731658337043906390063897091166780570790757474238526364226010764551385536115912885996253027630186549611517935457368517603605898077282586571429762806196993171
$h(m)$ value:
43623344949994787127752234921460092267633038253725859161875319030652406683751930302664421092019081 73
s value:
10109401522387889362354893052037631219529810827236397051544803016293074663620418259531567409456774642
k value:
89881079201538338220393406015238195358929126275967524834232024829552517663753517265223614378459978284022849122332354878981301520340512477020707075692653202807205629702940580761107577790339645950841917584875586333453715730999863979292629122054004168288828032880046018817432607968266238388909227544658
-Signature Verification:
r_1 value:
70067210427643294166222138993027798549657752697978469744048422500317294658740486555926205285400445471012016076449158573561242986921045773455811824689759844689682710067019646496820350622771577003668315
r_2 value:
63286413639814488044951356532672917048602261731658337043906390063897091166780570790757474238526364226010764551385536115912885996253027630186549611517935457368517603605898077282586571429762806196993171
$h(m)$ value:
43623344949994787127752234921460092267633038253725859161875319030652406683751930302664421092019081 73
s value:
10109401522387889362354893052037631219529810827236397051544803016293074663620418259531567409456774642
k value:
89881079201538338220393406015238195358929126275967524834232024829552517663753517265223614378459978284022849122332354878981301520340512477020707075692653202807205629702940580761107577790339645950841917584875586333453715730999863979292629122054004168288828032880046018817432607968266238388909227544658
y value:
85357427054426942199227586846504813044343555691588811131491758591820003607591087796535054464276499368983800094634708787100885187797117994310277076638012713825790478365924942604617693838464217803924069

y' value:
85357427054426942199227586846504813044343555691588811131491758591820003607591087796535054464276499368983800094634708787100885187797117994310277076638012713825790478365924942604617693838464217803924069
In this scenario, Bob accepts Alice's signature (Because Alice's public key y = Bob's computed public key y').

5.2. Case No.2—The Digital Signature s Is a Forgery

-Key Generation:
q_1 value:
24204868133185954948052711126594325904950644150970719475783883549526026684120585565950513865485930072746045479777366698672918299160381477023509089521582173311372610467900279664561595681832113078528139
q_2 value:
46173725832913847121436400620212043936761099218517385905639143767656777138039469879991523917154111781240677870551683249265592165846507448502443411532538863488166829223210826165970651547223649717795467
p value:
97609316808472870454405302102049090631926400165007064779331430655320976676509628237857272091298564595016008517939469595141604772331116744670145968787226236724839270110107504464205570942192609743912751
x value:
96927043459506992604160311113571440553541919877615452252875327060460357853683601233201782815133749826662760029498030584804703017548549339102193245959130740336143062341792496995862986391669440371955440
g value:
60013100423475503376399617651347989516045909050522869961906226199920595075171660893963793787893012012968840554284018252356769557437724220116706787567813657613554342599141372868197388513025892460290169
y value:
85357427054426942199227586846504813044343555691588811131491758591820003607591087796535054464276499368983800094634708787100885187797117994310277076638012713825790478365924942604617693838464217803924069
-Signature Generation:
n value:
66060543862913573648226582032349286914035619085069029586786315260850680542030346493608336659621903
t value:
80530636648297567886476358999055044920161884190114108080522424811596009350204715077423152163199190 49
r_1 value:
70067210427643294166222138993027798549657752697978469744048422500317294658740486555926205285400445471012016076449158573561242986921045773455811824689759844689682710067019646496820350622771577003668315
r_2 value:
63286413639814488044951356532672917048602261731658337043906390063897091166780570790757474238526364226010764551385536115912885996253027630186549611517935457368517603605898077282586571429762806196993171
$h(m)$ value:
43623344949994787127752234921460092267633038253725859161875319030652406683751930302664421092019081 73
s value:
10109401522387889362354893052037631219529810827236397051544803016293074663620418259531567409456774642

k value:
898810792015383382203934060152381953589291262759675248342320248295525176637535172652236143784599782840228491223323548789813015203405124770207070756926532028072056297029405807611075777903396459508419175848755863334537157309998639792926291220540041682888280328800460188174326079682662383889092275446581
-Signature Verification:
r_1 value:
700672104276432941662221389930277985496577526979784697440484225003172946587404865559262052854004454710120160764491585735612429869210457734558118246897598446896827100670196464968203506227715770036683151
r_2 value:
632864136398144880449513565326729170486022617316583370439063900638970911667805707907574742385263642260107645513855361159128859962530276301865496115179354573685176036058980772825865714297628061969931711
$h(m)$ value:
436233449499947871277522349214600922676330382537258591618753190306524066837519303065240668375193030266442109201908173
s value:
452130824144401412289868080947718713659343350172501277060077749733943950160633042575600111855580806191
k value:
898810792015383382203934060152381953589291262759675248342320248295525176637535172652236143784599782840228491223323548789813015203405124770207070756926532028072056297029405807611075777903396459508419175848755863334537157309998639792926291220540041682888280328800460188174326079682662383889092275446581
y value:
853574270544269421992275868465048130443435556915888111314917585918200036075910877796535054464276499368983800094634708787100885187797117994310277076638012713825790478365924942604617693838464217803924069
y' value:
948054047702831191691842379340368493890093298528569404373292025143526786751123849980243933726059009825976798077779373747254148325146243299965626426118226559631136717465372556857093982623998552152806231.

The signature s has been changed in this instance and the result is a denial of the signature and message authentication. As a result ($y \neq y'$), Bob rejects the forged signature.

6. Security Attacks

6.1. Public Key Attack

Assume an enemy tries to launch an assault by exposing the private key deriving it from the public key. First, the enemy has to solve the discrete logarithm problem from (4) to acquire $(q_1 x)^2$. Then, the enemy has to solve the FAC problem to obtain x from $(q_1 x)^2$.

6.2. Valid Signature Attack

In case an enemy tries to launch an assault by exposing the private key from a valid signature (r_1, r_2, s, k) for the known message m. To obtain x from (7), first, the enemy must be aware of q_1, n and t. Nonetheless, given y, g and r_1 or r_2, obtaining nt from (5) or (6) is likewise subject to the FAC and DL assumptions.

6.3. Forging a Valid Signature Attack

In case an enemy tries to launch an assault by faking a legitimate signature (r_1, r_2, s, k) with a certain message m without being aware of any acceptable signatures or Alice's private key. With the message m in mind, if the enemy tries to solve a four-variable r_1, r_2, s and k satisfying (8). Then, three variables are set to fixed integers before determining

the answer to the last variable from (8). Given y, g, m, r_1, s, k or given y, g, m, r_2, s, k, finding r_1 or r_2 in order to satisfy (8) is according to FAC and the DL assumptions.

6.4. Known Message Attack

Suppose the enemy has access to the signatures (r_1, r_2, s_1, k_1), (r_1, r_2, s_2, k_2), ..., (r_1, r_2, s_z, k_z) for a set of messages m_1, m_2, ..., m_z. Then, the enemy tries to find the nt value from the following formula to make a forgery:

$$nt = \frac{1}{z-1} \frac{(z-1)k_z - \sum_{i=1}^{z-1} k_i}{\sum_{i=1}^{z-1} A_i - A_z}, \tag{9}$$

where $A_i = h(m_i)s_i$, but finding x to satisfy (7) is according to FAC.

6.5. Total Break

Since (3) is an expanded root problem, finding the private key x is impossible.

6.6. Existential Unforgeability under Chosen Message Attack

Suppose the information (m_1, s_1, k_1) and (m_2, s_2, k_2) are known to the enemy. With these details, there must be at least one (m', s', k') that meets the (8) condition. The above strategy only works if the nt value is the same for both m_1 and m_2 messages. Alice chooses different n and t values for each message m. As a result, (m', s', k') does not satisfy the requirement of (8).

7. Conclusions

Finding the primes of the form $p = 4p_1q_1 + 1$, where p_1 and q_1 are safe prime numbers, is challenging in the Wei-Hua-He signature scheme. So, the He digital signature scheme is inappropriate for large prime numbers. The contribution of the paper is to get rid of the flaws in the He digital signature scheme. As a result, a modified-He digital signature scheme is proposed based on a one-way hash function with a big prime modulus p. The time complexity of the algorithm is $\mathcal{O}(log^3 p)$. The proposed modified-He digital signature scheme is more efficient and gives a higher level of security.

Author Contributions: Conceptualization, E.R.; methodology, E.R. and G.S.G.N.A.; writing—original draft preparation, E.R.; writing reviews and editing, E.R. and G.S.G.N.A.; supervision, G.S.G.N.A.; All authors have read and agreed to the published version of the manuscript.

Funding: This research work is supported by Vellore Institute of Technology, Vellore.

Data Availability Statement: Not applicable.

Acknowledgments: The authors wish to thank the management of Vellore Institute of Technology (Vellore-632014) for their continuous support and encouragement to carry out this research work.

Conflicts of Interest: The authors declare no conflict of interest.

Notations

Symbol/Acronym	Description
p	Prime number
Z_p	Finite Field
g	Generator of Z_p
$gcd(p, g)$	Greatest Common Divisor of the integers p and g
g mod p	Remainder upon dividing g by p
x	Secret key
y	Public key
m	Message

$h(m)$	Hashing of the message
(r_1, r_2, s, k)	Signature
IFP	Integer Factoring Problem
DLP	Discrete Logarithm Problem
RC	Rivest Cipher
RSA	Rivest Shamir Adleman
DH	Diffie Hellman
DSA	Digital Signature Algorithm
ECDLP	Elliptic Curve Discrete Logarithm Problem
ECDH	Elliptic Curve Diffie Hellman
ECDSA	Elliptic Curve Digital Signature Algorithm
3DES	Trible Data Encryption Standard
AES	Advanced Encryption Standard

References

1. Paar, C.; Pelzl, J. *Understanding Cryptography: A Textbook for Students and Practitioners*; Springer Science & Business Media: New York, NY, USA, 2009.
2. Harn, L. Enhancing the security of El Gamal's signature scheme. *IEE Proc.-Comput. Digit. Tech.* **1995**, *142*, 376. [CrossRef]
3. Lee, N.Y.; Hwang, T. The security of He and Kiesler's signature schemes. *IEE Proc.-Comput. Digit. Tech.* **1995**, *142*, 370–372. [CrossRef]
4. Tiersma, H. Enhancing the security of El Gamal's signature scheme. *IEE Proc.-Comput. Digit. Tech.* **1997**, *144*, 47–48. [CrossRef]
5. Shao, Z. Signature schemes based on factoring and discrete logarithms. *IEE Proc.-Comput. Digit. Tech.* **1998**, *145*, 33–36. [CrossRef]
6. Li, J.; Xiao, G. Remarks on new signature scheme based on two hard problems. *Electron. Lett.* **1998**, *34*, 2401. [CrossRef]
7. Lee, N. Security of Shao's signature schemes based on factoring and discrete logarithms. *IEE Proc.-Comput. Digit. Tech.* **1999**, *146*, 119–121. [CrossRef]
8. He, W.H. Digital signature scheme based on factoring and discrete logarithms. *Electron. Lett.* **2001**, *37*, 220–222. [CrossRef]
9. Pon, S.; Lu, E.; Jeng, A. Meta-He digital signature schemes based on factoring and discrete logarithms. *Appl. Math. Comput.* **2005**, *165*, 171–176.
10. Chung, Y.F.; Huang, K.H.; Lai, F.; Chen, T.S. ID-based digital signature scheme on the elliptic curve cryptosystem. *Comput. Stand. Interfaces* **2007**, *29*, 601–604. [CrossRef]
11. Su, P.C. Enhanced short signature scheme with hybrid problems. *Comput. Electr. Eng.* **2011**, *37*, 174–179. [CrossRef]
12. Verma, S.; Sharma, B.K. A new digital signature scheme based on two hard problems. *Int. J. Pure Appl. Sci. Technol.* **2011**, *5*, 55–59.
13. Vishnoi, S.; Shrivastava, V. A new digital signature algorithm based on factorization and discrete logarithm problem. *Int. J. Comput. Trends Technol.* **2012**, *3*, 653–657.
14. Berezin, A.; Moldovyan, N.; Shcherbacov, V. Cryptoschemes Based on Dificulty of Simultaneous Solving Two Diferent Dificult Problems. *Comput. Sci. J. Mold.* **2013**, *62*, 280–290.
15. Van Hiep, P.; Mong, N.H.; Dung, L.H. Constructing a digital signature algorithm based on the difficult of co-resolve two hard problems: Integer factorization and discrete logarithm. *J. Sci. Technol. Danang Univ.* **2018**, *7*, 28.
16. Thai, N.V.; Dung, L.H. A public key cryptosystem based on the difficult of co-resolved two hard problems: Discrete logarithm and root finding. *J. Inf. Commun. Minist. Inf. Commun.* **2018**, *12*, 2018.
17. Dung, L.H.; Duc, T.M.; Van, L.X. A new method for constructing digital signature schemes base on difficulty of the integer factorization and discrete logarithm root problems the Zn. In Proceedings of the Fundamental and Applied IT Research Conference, Hanoi, Vietnam, 8–9 October 2018; pp. 1–9.
18. Hong, D.L. A new digital signature scheme based on the hardness of some expanded root problems. *Procedia Comput. Sci.* **2020**, *171*, 541–550.
19. Lizama-Pérez, L.A. Digital signatures over HMAC entangled chains. *Eng. Sci. Technol. Int. J.* **2022**, *32*, 101076. [CrossRef]
20. Moldovyan, D. A practical digital signature scheme based on the hidden logarithm problem. *Comput. Sci. J. Mold.* **2021**, *86*, 206–226.
21. Huang, X.; Dong, Y.; Ye, G.; Yap, W.S.; Goi, B.M. Visually meaningful image encryption algorithm based on digital signature. *Digit. Commun. Netw.* **2022**, *in press*. [CrossRef]
22. Gallian, J.A. *Contemporary Abstract Algebra*; Chapman and Hall: New York, NY, USA, 2021.
23. Koblitz, N. *A Course in Number Theory and Cryptography*; Springer Science & Business Media: New York, NY, USA, 1994; Volume 114.