IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol. 8, Issue.12, December 2020 E-ISSN: 2347-2693

3-D Digital Signature based on SHA-AES-ECC Scheme using Galois Field over GF(2^n)

Mohammad Amjad[1*], Aman Arora[2]

[1]Department of Computer Engineering, Jamia Millia Islamia, New Delhi, India
[2]Senior Software Engineer, Adobe Inc, New Delhi, India

[*]*Corresponding Author: mamjad@jmi.ac.in, Tel.: +91-011-26980281*

DOI: https://doi.org/10.26438/ijcse/v8i12.5561 | Available online at: www.ijcseonline.org

Received: 20/Dec/2020, Accepted: 23/Dec/2020, Published: 31/Dec/2020

***Abstract*—** Approaches of handwritten signatures is no longer adequate for protection with the development of Internet technology, so the modern technique called digital signature has emerged. Digital signature is more typically used as term encompassing only cryptographic signatures. Digital signatures are mainly used in the delivery of financial transfers, certificates and applications, where the prevention of forgery or tampering of data is crucial. But even within digital signature, there are cryptographic techniques like AES, SHA and asymmetric enciphering mechanism such as ECC combined together to make it highly secure, used three steps mechanism of generation and verification called as 3-D signature. This research paper discusses the combination of all three modes of security such as Symmetrical, Hashing and Asymmetrical cryptography to make the digital signature more secure and invulnerable to attack. The simulation results shows that the proposed 3-D digital signature scheme along with Galois Field is suitable for used in real time environment like IoT, WSN, Cloud computing and low memory devices such as smart cards. The proposed technique is based on mathematical model used in SHA-AES-ECC with Galois Field GF(2^n) with irreducible polynomial. Python Programming language is used to grasp the method used.

***Keywords*—**Digital Signatures, AES, ECC, SHA, Galois Field, 3-D, ECDSA, Encryption using symmetrical and asymmetric cryptography, Hash function, Irreducible Polynomial.
Nomenclature: AES (Advanced Encryption Algorithm), ECC (Elliptical Curve Cryptography), ECDSA (Elliptical Curve Digital Signature Algorithm), SHA (Secure Hash Algorithm), GF (Galois Field), 3-D (Three Domain), IP (Irreducible Polynomial)

I. INTRODUCTION

In this online era, most of the documents are transferred through online mode. Imagine a legal text that could include essential facts regarding rights and privileges in such a document we need to verify its validity. We don't want the promises they have written down to be rejected by people. In addition, this text is likely to have to be mailed to, interpreted and preserved by multiple people. The report may be changed at certain stages in the workflow at different occasions, unless it is voluntary, such as insertion an additional signature. In other words, whether anyone tries to make a forgery from the initial text, it could be involuntary, for example because of a transmission mistake, or deliberately. We have attempted to address this issue for years by placing a so-called digital signature. Digital signature is a method of asymmetric cryptography used in a digital rather than written medium to mimic the authentication properties of a signature. Normally, automated signature systems have two algorithms, one for signing involving the secret or private key of the user and one for checking signatures involving the public key of the user. It is an electronic signature that may be used to authenticate the identification of the sender of a communication or the signer of a document and where necessary, to guarantee that the original substance of the message or of the document sent stays unchanged [5]. Digital signatures are conveniently portable, can't be imitated by any individual, and can be time stamped automatically. The right to promise that the initial signed letter has arrived suggests that it will not be readily repudiated later by the sender [3][7].

Properties of a digital signature:

1. Simple signing a document for the signer: There is no sense in making a digital signature system that requires the signer trying to use sluggish and complicated digital signature operations.
2. Simple to validate a message for anyone: We would also like to make the authentication of a digital signature as effective as possible.
3. Hard for anyone to forge a digital signature.
4. Hard to generate the same signature: Anyone who is not the authorized signer should be virtually unable to compute a digital signature on a message that appears to be legitimate. By appearing to be authentic, we say that someone who tries to validate the digital signature is led to expect that a valid digital signature on a message has only been checked successfully.

It is a mathematical term that is determined using defined parameters such as the signatory's identification and pre-decided rules that enable the credibility of the data to be checked [8][11]. SHA-512 is used to generate the fixed 512 bits value for message size 2^{128} which is more than millions of the pages. Output of SHA-512 is passed through AES to generate 128 bits fixed cipher. Irreducible polynomials are used because the solution of each and every set of IP gives the unique value and is defined for every degree of polynomials. The specified AES by the National Institute of Standards and Technology supports a fixed 128-bit block size and 128, 192 or 256-bit key size. The bigger the defined bit size, It will become more difficult to break and as well having the symmetric nature of the cryptography it become most popular among the commercials, private and government organizations. Galois Field GF(2^8) with irreducible polynomial $x^8+x^4+x^3+x+1$ is used by AES as the modulus. The Galois Field is one of the modular arithmetic used for computing the integers in the finite field [13]. As an 8 bit binary value, the multiplicative inverse of the byte is included in GF (2^8). The calculation can be described using the GF(2^n) field in the elliptical curve group. The set elements in this area are n-bit terms that can be represented in the Galois field as polynomials with a coefficient of n. $y^2+xy=x^3+ax^2+b$ is defined as Elliptical curve over GF(2^n), where $b\neq0$, the value of x, y, a and b are polynomial representing n-bit words [14]. For finding inverses, If P = (x, y), then $-P = (x, x + y)$. Similarly to find points on the elliptical curve using generators for polynomials Using the irreducible polynomial of $f(x) = x^3 + x + 1$, which implies that $g^3 + g + 1 = 0$ or $g^3 = g + 1$, we select GF(2^3) with elements {0, 1, g, g^2, g^3, g^4, g^5, g^6}. It is possible to quantify other powers of g similarly. For example using the elliptic curve $y^2 + xy = x^3 + g^3x^2 + 1$, with $a = g^3$ and b = 1, points on this curve will be as follows: P{(0,1), (g^2,1), (g^3,g^2), (g^5,1),(g^6,g)} and –P{(0,1), (g^2,g^6), (g^3,g^5), (g^5,g^4),(g^6,g^5)}

The rest of the research paper is organized as follows: Section II is about the related works. Section III highlights the Digital Signature Technology pertaining to this work. Section IV introduces about Secure Hash Algorithm, working Principle of AES is explained in section V. Introduction to Elliptical curve cryptography using GF(2^n) is discussed in VI. Proposed Elliptical Curve cryptography using SHA-AES-ECC using Galois field is explained in section VII. Implementation and result is shown in section VIII, contribution and conclusion is eventually outlined in Section IX.

II. RELATED WORK

Whitfield Diffie and Martin Hellman established the concept of a digital signature method for the first time in 1976, inferring that such systems existed. R.L. back in 1977. Rivest, Shamir, and Adleman [10] have publicly defined the popular RSA algorithm. To produce rudimentary digital signatures, the RSA algorithm is now used and the data being sent is encrypted or decrypted. A new system was developed after RSA, Lamport signatures, Merkle signatures and Rabin signatures. Handwritten signatures are not feasible for the remotely signed and verifying of a document. The only possible way is the digitally sign a document and would be remotely verified. Three measures, namely the digital signature process, are necessary like Key generation, algorithm signature and method of verification. SHA hash function refers to approved algorithm for computing a condensed digital representation known as message digest. The four flavors of SHA is structured named as SHA-0, SHA-1, SHA-2 and SHA-3. SHA-1 was proposed by NIST as a message digests function. SHA-0 is the first version of the secure hash algorithm and is pragmatic to the original version of the 160 bit hash function published in 1993. Now next improvement in this was published under the nick name SHA-2. SHA-2, defined as SHA-256 and SHA-2, is a family of two related hash functions, with separate block sizes. They vary in word scale, such as 32-bit words used by SHA-256, where SHA-512 uses 64-bit words. New encipherment technique was adopted by NIST in the year 2001 as Advanced Encryption Standard -AES. NIST was looking for the new standard of encipherment which should be efficient, flexible and free to implement in hardware as well as software. An algorithm called Rijndael, named after the two Belgian cryptographers who created and sent it, Dr. Joan Daemen and Dr. Vincent Rijmen, was chosen after a lot of debate and investigation in the cryptographic world. AES-128 become standard in December 2001 and was published as FIPS 197 in the Federal Register. In 1985, Victor Miller and Neal Koblitz invented elliptic curve cryptography. Elliptic curve cryptography proposed as an alternative to established public-key system such as RSA [5][9]. This is the type of public key cryptography base on the structure of elliptical curves. This is similar to current public key cryptosystems in which modular arithmetic and Galois field is substituted by operations specified over elliptical curves.

III. DIGITAL SIGNATURE TECHNOLOGY

Digital signature is a type of public key cryptography used in digital, rather than written form for the authentication purpose of a signature. Digital signing mechanisms typically have two mechanism, one for signing involving the secret or private key of the user and one for checking signatures involving the public key of the user. The sender first selects a random or pseudo-random confidential integer x to sign a message m, with $0 < x < q-1$. The sender would then evaluate $r = (a^x \bmod p) \bmod q$ and $s = x^{-1}(m + axe) \bmod q$. Then, the digital signature is (m, r, s). Receiver R1 measures $u = s^{-1}(m) \bmod q$ and $v = s^{-1}(r) \bmod q$, x^{-1} and s^{-1} as multiplicative inverses of x and s respectively, in order for the receiver to validate this signature. R1 computes $w = (t^u y^v \pmod p) \bmod q$ and accepts the signature if and only if w = r. It follows from the description of s to see why this works: $sk = (m + ar) \bmod q$, so $k = s^{-1}m + s^{-1} ar = u + av \bmod q$. Consequently,

$t^k = t^{(u+av)} = t^u y^v$ mod p. Therefore, taking the values of mod q, we have r= w. t: Primitive root mod p and y= t^a mod p, p and q are two large prime numbers of the order of 100.

A. Digital signature function:

1. Data integrity: Minor change by unauthorized user in the message leads to mismatch in the verification and digital signature becomes useless.
2. Non repudiation: Any signature done by the authorized user may not deny further. The signatory is prohibited by public key cryptography algorithms from wrongfully alleging that he has not signed and forwarded the received text.
3. Anti-fabrication: Some spurious message may not be added by the authorize user and It prohibits the user from forging or modifying the original message that the signee appears to have received.

B. Attacks on Digital Signature

Interruption: It is a type of attack in which the attacker tries to make the resources of the system unavailable or unusable. Hackers may attain it by:

a. Destruction of the hardware.
b. Unavailability of the communication media.

Modification: In this type of attack the integrity of the message is changed. Hackers may change the assets of the system causes loss of integrity.

Fabrication: The attacker is attempting to apply a spurious message to the initial message in this form of attack. This type of attack breaches the authenticity of the original message. Hackers may attain it by:

a. Addition of records to a file.
b. Forgeries of the message.
c. Counterfeit on a computing system.

Denial of Service: An intruder threatens to prohibit legal users from receiving any or more of the resources applying for them. For example the attacker may create so many logins and send it to the server using random user ids so that it causes deny of the accessing of the actual user to the computing system. DOS attack may cause the system out of service.

C. Applications of Digital Signature

Digital signature are particularly useful for:

1. Education: Educational institutes can reduce paperwork in many areas like student applications, issuance of documents like transfer certificates, report cards etc. This will reduce the time and efforts involved in these processes
2. Business: Sales papers, plans, schedules with suppliers, sales requests, customer deals, rentals, documentation and permits for repayment.
3. E-commerce: Electronic tendering networks facilities and web portals may be used for signing online contracts with external modules.
4. Human Resources: Employee documentation, ration-card data protection and usage of wireless communication
5. Government: Exchange of electronic data between high officials. In India, the government uses digital signatures such as AADHAAR, online train ticket booking and reimbursement for personal recognition.
6. Medicine: Hospital consent documents, patient history, prescription examinations, medicines and other effective analysis.

IV. SECURE HASH ALGORITHM-512

SHA-512 is the family of SHA with 512 bit message digest. We have chosen this particular version in our research paper because it is the latest version, it has more complex structure
and its message digest is the longest digest of practical used.
The digest is initialized to a pre-determined value of 512 bits. The algorithm mixes this previous value.

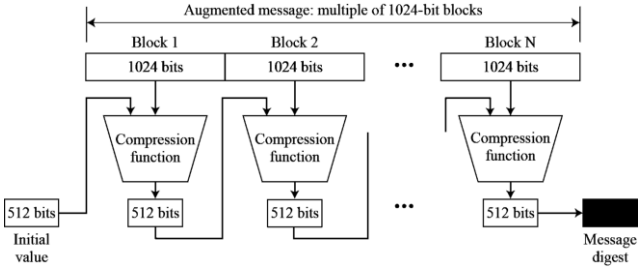

Figure 1: Process of Hash generation using SHA-512

With the first message block to create the first 512-bit intermediate message digest. To create the second intermediate digest, the digest is then mixed with the second block. Finally, in order to produce the nth digest, the (N-1)th digest is combined with the nth block. The resulting digest, when the last block is processed, is the message digest for the entire message to SHA-512.

The algorithm's heart is the compression function. It processes messages in 1024-bit blocks and consists of 80 rounds per block. A 512-bit buffer is initially used to hold the original message. A 64-bit value extracted from the current message block is used to produce the round key. For each and every round, round constants are generated depending on the cube root of the first 80 prime numbers.

V. ADVANCED ENCRYPTION STANDARD

The AES algorithm is a private key encipherment in which a secret key for encryption and decryption is used by both sender and recipient. The size of the text block is set at 128 bits, while the length of the cypher can be 128,192 or 256 bits. In addition, AES algorithm is Feistal structure algorithm. There are total number of rounds as 10, 12 or 14, for key length is 128,192 or 256 respectively, 128 bit data block is divided in 16 bytes. These bytes are arranged into (16*8=128 bits) 4*4 array called States, and all rounds of AES algorithm are performed on these State.

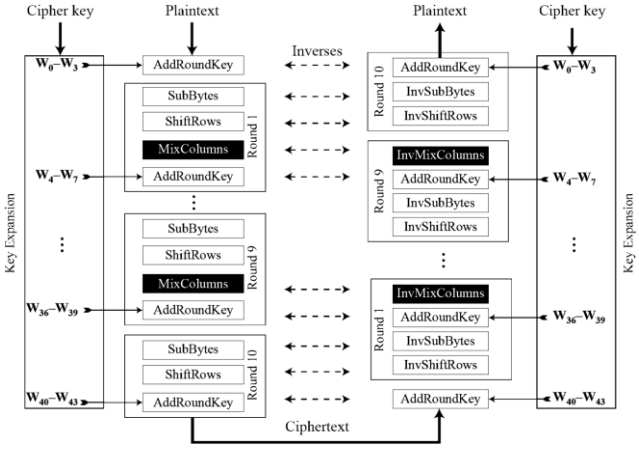

Figure 2: Encipherment process in AES

The above figure is used for 128 bits bock size, it may be extended to the variable block size proposed by Rijndael [13][14]. The four main constituents of the AES encryption and decryption is as follows.

1. Substitute bytes: It is used for byte by byte substitution of the blocks using S- boxes.
2. Shift rows: It is a type of transformation to change the positions of the rows.
3. Mix columns: Intermixing of the columns by using the numerical arithmetic of cryptography.
4. Add round key: Since for each round a set of key of fixed size is required, it is the value of the round key generated by expansion of original key.

In general AES uses the irreducible polynomial $x^8+x^4+x^3+x+1$ as the modulus in GF(2^8). Mix column and add round keys process are the heart of AES encipherment technique. The process of mix column is described as follows:

$$\begin{bmatrix} s'ij \\ s'ij \\ s'ij \\ s'ij \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} sij \\ sij \\ sij \\ sij \end{bmatrix}$$

Where S_{ij} is the element in the i_{th} column and j_{th} row of the input state and s'_{ij} is the corresponding states in the output states. The value of i and j varies from 0 to 3 for the matrix entries. So the first row of S'_{ij} will look like [S'_{00}, S'_{01}, S'_{02}, S_{03}], similarly the values of the matrix of S_{ij} is to be displayed. Each component of the product matrix is the sum of one row and one column of the multiplication function. For example the S'_{ij}= multiplication of first row and first column will give the first value of S'_{ij}.

$S'_{0j} = (2\cdot S_{0j})$ xor $(3\cdot S_{1j})$ xor S_{2j} xor S_{3j}

$S'_{1j} = S_{0j}$ xor $(2\cdot S_{1j})$ xor $(3\cdot S_{2j}$ xor S_{3j}

$S'_{2j} = S_{0j}$ xor S_{1j} xor $(2\cdot S_{2j})$ xor $(3\cdot S_{3j})$

Output of mix column block of states of 128 bits is passed to Add round key. Round key is generated for each and every round by using key expansion process and then states of the mix column are added with round key using Galois Field [15].

VI. ELLIPTIC CURVE CRYPTOGRAPHY GF(2^n)

ECC is created on algebraic structure of elliptic curves over a Galois or finite field. The ECC computational difficulty states that it is hard to calculate the discrete logarithm of an elliptic curve variable given the publicly defined base value. General equation of ECC over GF(2^n) is $y^2+xy=x^3+ax^2+b$, where $b\neq 0$, the value of x, y, a and b are polynomials expressing 3 bit words, in general n bit.

A finite field or Galois field refers to a set of elements finite elements. Two basic binary operations namely multiplicative and additive inverses are defined in this field that satisfy defined arithmetic properties and rules. The number of elements in the field is called its order. A finite field of order g and denoted by F_g exists if x is a prime power x^k, where x is a prime and $k\in(0,\infty)$. Also in a field F_g with prime power x^k adding an element p times results in zero.

The basic operations are described below:

1. Point addition: It is the addition of two points P and Q to obtain a third point R i.e. R= P + Q.

If $P\neq -Q$. The rules for adding points in GF(2^n)is as follows:

a. If first point P = (x_1, y_1), and another point is Q= (x_2, y_2), $P\neq(-Q, Q)$, then R=(x_3,y_3)= P+ Q can be found as: $\lambda= (y2+y1)/(x2+x1)$, $x3= \lambda^2+ \lambda+x1+x2+a$ and $y3= \lambda(x1+x3) +x1+y1$, where R(x3,y3) is the third point.

b. If P=Q, then R= P+Q (or R=2P) can be found as: $\lambda =x1+y1/x1$, $x3= \lambda^2+ \lambda+a$, $y3= x1+(\lambda+1)*x3$,where R is (x3, y3). A negative of a point is reflected with respect to the X-axis of that point.

2. Multiplying a point with a constant:

To multiply a point by a constant, the points must be added iteratively with rule R=2P. To obtain a new point Q is the addition of a point P to itself, i.e. P = 2Q.

3. Finding multiplicative inverse: It's also quite easy to find the multiplicative inverse of each element. For example, we can find the multiplicative inverse of 3- bit values in binary with irreducible polynomial $f(x)= x^3+x+1$of g^3 as $inv(g^3)=g^{(15-3)}=g^{12}=g^3+g+1=$ 110. For instance, using the irreducible polynomial $f(x) = x^3 + x + 1$, we choose GF(2^3) with elements {0, 1, g, g^2, g^3, g^4, g^5, g^6}, meaning that $g^3 + g + 1 = 0$ or $g^3 = g + 1$. It is possible to quantify other powers of g accordingly. The following table shows the values of the generator polynomial showing the values of g.

Table 1: Generator polynomial for f(x)

Generator function (g)	Binary value	Generator function (g)	Binary value
0	000	$g+1= g^3$	011
1	001	$g^2+g =g^4$	110
g	010	$g^2+g+1= g^5$	111
g^2	100	$g^2+1= g^6$	101

VII. PRPOSED 3-D DIGITAL SIGNATURE TECHNOLOGY

Our proposed method mainly focuses on framework of complex mechanism of Digital signature used for

authentication as well as verification of message or entity. Any length of paragraph should be digitally signed by 3-D digital signature technique. The paragraph or length of the message is passed though SHA-512 which will be able to generate the fixed size of 512 bit length output irrespective of size of message. The fixed 512 bit output bit cipher text. The hashed message is encrypted by AES128 to make the digital envelope. Finally 128 bit ciphered message is converted to decimal value and it is chosen as secret key used in ECC. The working flow of 3-D digital signature is described as follows:

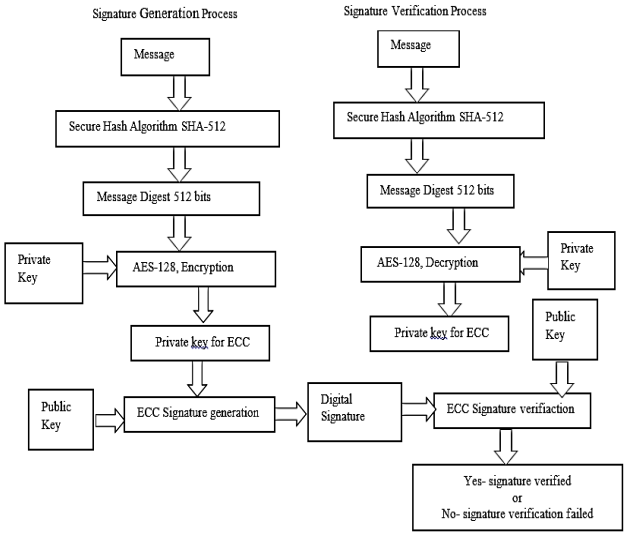

Figure 3: Working procedure of 3-D signature

1. Original message of sender is passed through SHA-512 hashing technique.
2. Bit length of 512 bits is encrypted using AES-128 bit symmetrical key encipherment.
3. Two sets of public and private key is generated using ECC.
4. Output of AES-128 bit cipher length will be assumed as private key of ECC.
5. Public key for ECC in GF(2^n) is by choosing a point on ECC.
6. Signature is generated using private key of ECC.
7. Verification of signature is done at the receiver end using the public key of ECC.

A. Key Generation:

Suppose sender A to sign the message M. With an elliptic curve E, for some prime q and Galois field GF(2^3) as Fg.

Choose a point P of order k so that P∈E(Fg). Now A performs the following steps to create the public, private key pair: Choose a unique and unpredictable integer i such that i∈[1, k- 1].

1. Convert 128 bit output of AES cipher into decimal, let us say c.
2. Calculate k1=i*c for the private key.
3. Find the multiplication of the decimal value of step 2 with point of the ECC as k2 = i*c*P.

A's private key is the above calculated k1 and public key is k2.

A. Signature Generation

User A performs the following steps to sign a message M with the private key k1:

1. Select a random or pseudo-random integer n, such as n∈ [1,k-1]
2. Calculate n*P as the key pair (x1, y1) and t= x1 (mod k), where x1 is an integer between 0 and g-1, respectively. Repeat step 1 for t = 0 (g is a generator polynomial in GF(2^n).
3. Now the value of u= n^{-1}(mod k) is computed.
4. Hash of the message m is: H=h(m) where H is the Secure Hash Algorithm (SHA-512)
5. Calculate s= n^{-1} {H+ k1*n} mod k, then go back to step 1 for s=0. (n^{-1} is multiplicative inverse in Galois Field).

The pair (t,s) is the signature for the message M.

B. Signature Verification

User B first obtains an authenticated copy of A's public key k2 to check A's signature (t, s) on the message M. B perform the following now:

1. Verify that t and s are part of the [1, k-1] interval.
2. Calculate w = s^{-1} mod k and the Hash H=h(M).
3. Determine part of the signature as: s_1 = H*w*(mod k) and s_2 = n*w *(mod k).
4. Evaluate $s_1P + s_2k2 = (x_0, y_0)$ and v= x_0 (mod k). If v=t,

Message M is confirmed as being authentic.

C. Comparing 3-D Signature with ECDSA

In comparison to 512 bit operations in 3-D Signature, ECDSA uses 160 bit arithmetic to provide the same degree of security.

Our signature is more secure and unsusceptible to attack.

1. 3-D signature is the combination of symmetric and asymmetric algorithm, whereas ECDSA is variant of RSA signature scheme which is only asymmetrical.
2. 3-D have fast signing and verification characteristics whereas ECDSA have slower process of signature generation and verification.
3. In both algorithms, it is time consuming to produce typical domain parameters.

D. Advantages of 3-D Signature

1. More secure than other similar algorithm, because it is based on irreducible polynomial and generator polynomial as used in GF(2^n).
2. Hardware as well as software implementation is easy.
3. Time complexity is less and required to generate and validate signature when compared with its counterparts.
4. Implemented over low memory devices as it requires very lesser size of public and private key sizes.

VIII. MPLEMENTATION AND RESULTS

Comparison between our proposed 3-D algorithm and Elliptical Curve Digital Signature Algorithm is done with respect to their key sizes. In the final result we can observe that proposed 3-D signature scheme is more secure as it has longer size of the set of public and private key pair.

Table 2: Comparison of key sizes in bits

3-D Signature	ECDSA
128 (using AES128 and IP)	128 (using DSA 128 bits)
160 (using AES192 and IP)	512 (using DSA 512 bits)
192 (using AES256 and IP)	1024(using ElGamal 256 bits)
224(using AES128 and IP with SHA512)	1280 (using RSA 512 bits)

This research work is divided into well-defined modules and procedures e.g. AES ciphering process for the purpose of private key generation k1 and elliptical curve generation using GF(2^n). The process of Private and public key generation is explained. Signature generational and verification process is carried out in a separate module of the programming set and then it is compared for the final signature verification at the receiver end. Python language is used to realize the result. The simple Elliptical curve based on GF(2^n) is considered for the analysis of the result.

$y^2+xy=x^3+g^3*x^2+1$, where g is a generator polynomial defined with irreducible polynomial $f(x) = x^3+x+1$. In this figure 3. P is the point lies on this curve and –P is the additive inverse of this point P. The calculated points on this Elliptical curve will be as follows:

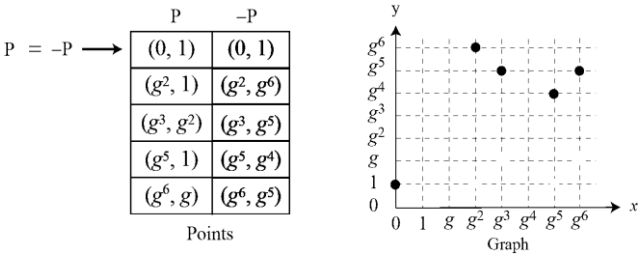

Figure 4: Points on the ECC over GF(2^n)

A. Results of 3-D Digital Signature:

- Key generation for 128 bits:

Input message M= "hello digital signature", Secret key = 2020654321987654, AES output: 297234785248526685862451117931277793 in decimal. Private key k1= i*c=2* 297234785248526685862451117931277793 = 594469570497053371724902235862555586 (where i=2). Public key will be calculated by using the method in the algorithm as k2 = i*c*P= 2*297234785248526685862451117931277793*5= 297234785248526685862451117931277930 where Point P on the curve is (g^2,1)=(101,001)=(5,1). Public and private key pair again will be calculated by the procedure as in the steps of algorithm and this will be (594469570497053371724902235862555586, 297234785248526685862451117931277930)

- Signature generation:

Suppose a random number is n=13, then the value of n*P= 13* (594469570497053371724902235862555586, 297234785248526685862451117931277930)= 772810441646169383242372906621322618 and t= x1(mod k) = 594469570497053371724902235862555586 mod 119 = 46. (k is the order of the element in the Group),

Now u= n^{-1}mod k= 13^{-1}mod 119=55.

Hash of the message "hello digital signature" using SHA-12 will be:SHA-12 result is converted to decimal H= h(m)=h("hello digital signature") = 68667595114752400408338147449761536712996356605218992076310304930174838057636629860871660232459737216920665156073283209127314076424048229081241962592022321899207631030493017483805763662986087166023245973721692066515607328320912731407642404822908124196259202232 in decimal.

Signature will be pair (t,s) where s= n^{-1} {H+ k1*n} mod k= 13^{-1}{68667595114752400408338147449761536712996356605218992076310304930174838057636629860871660232459737216920665156073283209127314076424048229081241962592022321899207631030493017483805763662986087166023245973721692066515607328320912731407642404822908124196259202232 + 297234785248526685862451117931277793*13) mod119 = 13^{-1}(36)=25. t is calculated earlier as 62, so the digital signature pair will be (46,25).

- Signature Verification:

It is also verified that t and s belongs to the range 1 to k-1. Now w= s^{-1} mod k = 25^{-1} mod119=100 and H=h(m)=86675951147524004083381474497615367129963566052189920763103049301748380576366298608716602324597372169206651560732832091273140764240482290812419625920223208124 in decimal. s_1 = H*w*(mod k)= 50 and s_2 = n*w (mod k)=13*100 mod 119=110. Now s_1P + s_2k2 =(x_0, y_0), 50*5 + 110*297234785248526685862451117931277930, 50*1+110*297234785248526685862451117931277930= (x_0,y_0) and v= x_0 mod k=3269582637733793544486962297244047 0 mod 119=46, now the value of v=t, so the signature is verified. It is also be proved that if the message is changed by some unauthorized person, then all the values will become changed and signature will not be verified.

B. Comparison of 3-D signature with ECDSA based signatures.

Table 3: Signature Generation

Key length		Time (seconds)	
3-D	ECDSA	3-D	ECDSA
128	128	0.08	0.10
160	512	0.18	1.35
192	1024	0.27	2.80
224	1280	0.64	3.33
256	1546	1.44	5.75

Table 4: Signature verification

Key length		Time (seconds)	
3-D	ECDSA	3-D	ECDSA
128	128	0.15	0.07
160	512	0.34	1.15
192	1024	0.59	2.20
224	1280	1.18	3.08
256	1546	3.07	5.02

Performance of two algorithm doesn't differ until the large key size. It can be observed from the table that 3-D signature generation and verification outperforms over the ECDSA as we increased the message size and key length. One important consideration is that some of the time is consumed by secure hash algorithm by both of the methods for generation as well verification of signature.

IX. CONCLUSION AND FUTURE SCOPE

Combining the strength of AES, SHA and ECC altogether become a very secure and entity authentication mechanism. ECC has transformed from being an exciting theoretical concept to an appealing and cutting edge technology adopted by multiple organizations for commercial purpose. We have presented practical applications of the generation and verification algorithm of 3-D digital signatures in this paper. The proposed method of digital signature is elaborated and implemented in programming language Python. This mechanism uses the three domain of signature generation and verification and is tightly coupled with new advances and requirements of security goals. It will be useful to build a stronger and faster mechanism for security. The key storage required for the 3-D signature technique is minimal compared to similar techniques such as ECDSA, even a smaller key length provides greater security.

Changing the single character of the message, the signature verification process shows a high avalanche effect and the signature becomes redundant. Time analysis of 3-D signature shows that it is a fast, efficient and secure. Because of its faster signature generation and smaller size of key, it is suitable for low memory devices such as IoT, WSN smart cards and ZigBee. The proposed signature generation is demonstrated only for AES-128 bits with SHA-512. Other versions for the signature generation could be used with AES-192 and AES-256 using the same procedure.

REFERENCES

[1] Roy A., Banik S., Karforma S., "Object Oriented Modelling of RSA Digital Signature in E-Governance Security", International Journal of Computer Engineering and Information Technology (IJCEIT), Vol. **26**, Issue No. **01**, pp. **24-33, 2011**.

[2] Kain K, Smith SW, Asokan R., "Digital signatures and electronic documents: a cautionary tale". In the proceeding of 2002 International Conferences in advanced communications and multimedia security, Springer **USA**, 2002, pp. **293-307**, **2002**.

[3] D. Boneh, H. Shacham, "Group signatures with verifier-local revocation", 11th ACM Conference on Computer and Communications Security CCS, Washington DC **USA**, pp.**168–177, 2004**

[4] Sergei G. Chernyi, Aslamin A. Ali, Vycheslav V. Veselkov, Ivan L. Titov and Vlad Yu. Budnik, "Security of Electronic Digital Signature in Maritime Industry", IEEE International Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus*)*, **Russia**, pp. **29-32, 2017**.

[5] Joppe W. Bos, Craig Costello, Patrick Longa, Michael Naehrig, "Selecting elliptic curves for cryptography: An efficiency and security analysis", Journal of cryptographic Engineering Springer, Volume **6**, issue **4**, pp. **259-286, 2016**.

[6] S. M. C. Vigila and K. Muneeswaran, "Implementation of text based cryptosystem using Elliptic Curve Cryptography," IEEE International Conference on Advanced Computing, **Chennai**, pp. **82-85**, **2009**.

[7] T. Ebanesar, G. Suganthi, "Improving Login Process by Salted Hashing Password Using SHA-256 Algorithm in Web Applications," International Journal of Computer Sciences and Engineering, Vol.**7**, Issue.**3**, pp.**27-32, 2019**.

[8] Laiphrakpam, Dolendro Singh and Khumanthem Manglem Singh, "Image Encryption using Elliptic Curve Cryptography", Elsevier, Procedia Computer Science,11th Int. Multi-Conference on Information Processing-2015 (IMCIP-2015), **Bangalore**, pp. **472-481**, **2015.**

[9] Li, Ahmed A. Abd El-Latif , Xiamu Niu, "Elliptic Curve ElGamal based homomorphic image encryption scheme for sharing secret images", Elsevier Signal Processing, (2012), Vol.**92**, Issue-**4**, pp. **1069-1078, 2012.**

[10] D.S. Kumar, CH. Suneetha A, Chandrasekh A R, "Encryption of Data Using Elliptic Curve Over Finite Fields," International Journal of Distributed and Parallel Systems, vol**. 3**, Issue **01**, pp. **103-108, 2012.**

[11] R. Singh, R. Chauhan, V. K. Gunjan, P. Singh, "Implementation of Elliptic Curve Cryptography for Audio Based Application," International Journal of Engineering Research & Technology (IJERT), vol**. 3**, Issue no**. 01**, pp. **2210-2214, 2014.**

[12] Parthajit Roy, "A Tripartite Zero Knowledge Authentication Protocol based on Elliptic Curve Weil Pairing," International Journal of Computer Sciences and Engineering, Vol.**5**, Issue.**9**, pp.**27-31, 2017.**

[13] P. S. Yadav, P. Sharma, K. P. Yadav, "Implementation of RSA Algorithm Using Elliptic Curve Algorithm for Security and Performance Enhancement," International Journal of Scientific & Technology Research, vol. **1**, Issue no. **4**, pp. **102-105, 2012**.

[14] L. Tawalbeh, M. Mowafi, W. Aljoby, "Use of Elliptic Curve Cryptography for Multimedia Encryption," IET Information Security, Volume-**07**, Issue-**02**, pp. **67-74, 2012.**

[15] William Stallings, "Cryptography and Network Security", Prentice Hall, **India**, 5th Edition, **Pages 285-296, 2010.**

[16] Kaufman, c., Perlman, R., and Speciner, M., "Network Security, Private Communication in a public world", **India**, 3rd Edition. Prentice Hall Print, **pp. 283-290, 2015.**

[17] Behrouz A Forouzan, "Cryptography and Network Security", McGraw Hill, **Inida**, 2nd Edition, **pp. 290-297, 2010.**

[18] Bruce Schneier, "Applied Crypt**ography: Protocols, Algorithms** and source code in C", Wiley publishing Inc., India, 2nd Edition, **pp. 201-235, 2015.**

AUTHOR'S PROFILE

Dr. Mohd. Amjad pursed Bachelor of Engineering in Computer Engineering from A.M.U. Aligarh, M. Tech. Degree in Information Technology from GGSIP University New Delhi and Ph.D. from Jamia Millia Islamia, from the Department of Computer Engineering New Delhi and currently working as Professor in the Department of Computer Engineering, F/o Engineering & Technology, Jamia Millia Islamia (Central University), New Delhi. He has published more than 60 research papers in reputed international journals including Thomson Reuters (SCI & Scopus) and International conferences including IEEE and Springer. His research interests includes Network Security, Internet and mobile computing, Mobile Ad hoc Networks and wireless sensor networks. Dr. Amjad has more than 18 Years of teaching experience at U.G and P.G. Level.

Aman Arora is currently working as Senior Software Engineer at Adobe Inc, New Delhi. He received his B.Tech. in Computer Engineering from Jamia Millia Islamia (Central University), New Delhi. He has published more than 7 US patent office approved international patents and many research papers in international journals.

