Cryptographic Secure Key Transport Protocol using Elliptic Curve over Finite Field with Hash Message Authentication Code

CH. Suneetha
Associate Professor in Mathematics
GITAM University
Visakhapatnam, India
gurukripachs@gmail.com

D Sravana Kumar
Associate Professor in Physics
Dr. VS Krishna Govt. Degree College
Visakhapatnam, India
skdharanikota@gmail.com

P SIRISHA
Faculty in Mathematics
Indian Maritime University
Visakhapatnam, India
sirinivas06@gmail.com

Abstract:* As the internet is the basic means of communication nowadays confirming the integrity and authenticity of the received data is a prime necessity in communication networks. The present paper explores the secure key transport from one entity to the other using elliptic curve over finite field. To assure the authenticity one Message Authentication Code (MAC) is appended to the key to be transported using cartographic hash function. The receiving entity verifies the MAC and ascertains the received message. The secure key transport protocol designed here overcomes the drawback of Diffie-Hellman key exchange protocol. So, this is more suitable for communicating high secured pass words and PIN numbers in monetary transactions.

***Key Words- Encryption, Decryption, Elliptic curve over finite field, Hash Message Authentication Code (HMAC*)

I .INTRODUCTION

Sometimes light weight communications offer high level of security,confidentiality and authentication such as e commerce and digital currency. In communications involving passwords, PIN numbers given by corporate and banks to the customers the input is very small in size. But high level of security is expected even at the cost of computational risk and communication risk. Many password authenticationprotocolswere designed by several cryptographers closely related to Diffie-Hellman key exchange protocol which is a revolutionary in the history of public key cryptography. But, Diffie-Hellman key exchange protocol is insecure against man -in- middle attack. Here both the uses have no control over the secret key for their communication. Later, many authors published modifications to Diffie- Hellman key exchange algorithm over the years as Authenticated Key Agreement (AK) protocol and Authenticated key Agreement protocol with key confirmation (AKC)

II.AUTHENTICATED KEY AGREEMENT (AK) PROTOCOL

In Authenticated key Agreement (AK) protocol between legitimate pair of entities in digital communication system each entity has static or long term public keys and short term public key called session keys.

Consider two entities A and B in a group each having their own public keys, private keys and the global parameters. Suppose the entity A has the long term public keys A_1,A_2 and the entity B has long term public keys B_1,B_2. The Authenticated Key Agreement protocol comprises the steps [2,3]

- ➢ Entity A selects a random private session key, computes short termPublic key or session key and forwards to B
- ➢ Entity B computes a long term shared key K derived from A's public keys and B's private keys with some first function F_1
- ➢ B utilizing this long term shared key K, A's public session key, B'sPublic session key computes an authenticated message and forwards to A
- ➢ A verifies the received authenticated message.
- ➢ The same procedure is repeated from the other end.
- ➢ The key agreement protocol that provides the confirmation of the key between two entities of the group then it is called Authenticated Key agreement Protocol with key confirmation (AKC). i.e., oneentity convinces that the other party actually has the possession of the key

A. Desirable Features of Authenticated Key Agreement Protocols

- Known Key Security: The participating entities should produce unique secret key or session key which is unaffected even when the adversary learns something about the other keys [4].

- Forward Secrecy: Previously constructed session keys should not be influenced by the disclosure of long term private keys of one or more entities [5].

- Key Compromise Impersonation Resilience: An adversary can impersonate if the long term private key of the entity A (say) is revealed. But, this should not give the scope for adversary to impersonate other entities to A.

- Unknown Key-Share Resilience: The entities should be legitimate. i.e., say A should not involve in malpractice in sharing the key with unauthorized entities.

- Key Control: Both the entities sharing the secret key have no control over the secret that is being established.

B.Weaknesses of Authenticated Key Agreement protocol

The Authentic Key Agreement (AK)protocol and Authentic Key Agreement protocol with key confirmation (AKC) so far established are vulnerable against the above listed attacks and security implications. There is every possibility for the intruder to eavesdrops the send data and modifies it. Even the legal entities have no control over what will be the key that is being established. Since each entity of the group contributes only the part of the key it has no knowledge about the rest of the part. The adversary can reflect the message to the same entity where the entity feels as if the message comes from the other side. In addition the intruder can exhaust the resources by sending large number ofsimilar protocols and even stops further establishing the key. So, Authentic Key Agreement protocols (AK) and AKC protocols have not attained the required level of security.

In the present paper we propose an innovative secure key transport protocol using elliptic curve over finite which achieves all the desirable security attributes andovercomes the weaknesses of the key agreement protocols. Besides a MessageAuthentication Code (MAC) is appended to the message using cryptographic hash function to provide additional integrity and authenticity. This tag is recomputed and verified at the other end.

III.Elliptic Curve Cryptography

A. Elliptic Curve Arithmetic

Elliptic curves over finite fields are set of points satisfying the Weirestrass equation

$Y^2 + axy + by = x^3 + cx^2 + dx + e$

where the variables x,y and the constants a,b,c,d,e are field elements with the point 0 called the point at infinity or zero point. The points on the elliptic curve with 0 as the identity element forms an abelian group under addition operation.

Point addition Let $P(x_1,y_1),Q(x_2,y_2)$ E(K) where P,Q Then $P+Q=(x_3,y_3)$

$X_3=\left(\frac{y_2-y_1}{x_2-x_1}\right)^2 - x_1-x_2$

$Y_3=\left(\frac{y_2-y_1}{x_2-x_1}\right) -(x_1 -x_2)-y_1$

Point Doubling: Let $P=(x_1,y_1)$ be a point on the elliptic curve $P\neq-P, 2P=(x_3,y_3)$ where

$X_3=\left(\frac{3x_1^2+a}{2y_1}\right)^2 -2x_1$

$Y_3=\left(\frac{3x_1^2+a}{2y_1}\right) (x_1-x_3)-y_1m$

B.Elliptic Curves over finite field

For Elliptic Curve Cryptography the curves are defined over some finite field Fp (p = 2,3) where p is a large prime number. For the purpose of encryption and decryption using elliptic curves it is sufficient to consider the third order equation of the form

$Y^2 = x^3 + ax + b$ over the prime field Fp , where all the coefficients and the variables take all values in the set of integers from 0 through p-1 and the elliptic field arithmetic is done in this field Fp . With respect to modulo p.

C. Elliptic Curve Discrete logarithmic Problem (ECDLP)

Elliptic curve cryptographydepends onthe strength of hard problem called elliptic curve discrete logarithm problem (ECDLP).For any two points Q,P Ep (a, b). Consider an equation Q = kP where $Q,P\in$ Ep (a, b) and kp. It is relatively easy to calculate Q given k and P. But it is relatively hard todetermine k given Q and P [10,11].

IV.Cryptographic HashFunction

A function that maps variable-length input into a fixed output is called Hash Function. The output is called hash value or messages digest. A hash function used for security applications is called cryptographic hash function

h = H (M).

H: $D_v \rightarrow R_F$ where D_v is the domain of variable length and R_F is the range of fixed length.

A. Properties of Cryptographic Hash Functions

A good cryptographic hash function should have the following properties to resist all types of active and passive attacks

· Pre-Image Resistant: For the given hash value it is difficult to find the message m such that h = hash (m).i.e., the function should be a one-way function.

· Second Pre-Image Resistant: For given hash value h it is in feasible tofind a function y such that h = H(y) computationally.

· Collision Resistant: For two different messages m_1,m_2 it is impossible to find H (m_1) = H(m_2) [6,7]. In present days most widely used cryptographic hash functions are MD5 and SHA - 1.

B. Applications of cryptographic hash functions

Cryptographic hash function is most adaptable algorithm having many applications. Generally cryptographic hash are divided into two types keyed- hash functions and Un keyed hash functions [8,9].

- *Message Authentication Code (MAC)*
  Keyed hash function is referred to as Message Aauthentication Code (MAC). Message Authentication Code (MAC) is a technique of verifying the integrity of the message [1]. This assures the received data isexactly as sent. In thismechanism the hash code of the secret key is encrypted to achieve the required authentication. The message and the concatenated hash code of the secret key is encrypted to attain the authentication as well as confidentiality the required authentication.

Hash Message Authenticated Code (HMAC):

Most commonly used MAC construction is HMAC. A Message Authenticated Code that uses cryptographic hash functionssuch as MD 5, SHA-1and secret key is Hash Message Authentication Code (HMAC).

HMAC (K,M) = H[(K+ $\oplus$ opad) ||H [(K+ $\oplus$ ipad) ||M]]
Where K+ is the derived key by padding K to the right extra zeros to the input block size ipad is inner padding of the bits and opad is outer padding of thebits. $\oplus$ represents exclusive OR operation.

Nested construction of Message Authentication Code (NMAC)

It is a hash function that uses two keys. The message is concatenated with the one key and again hash function is applied on the output with the second key. If K_1, K_2 are two keys then the Message Authentication Code is
NMAC (K_1, K_2, x) = hK_1 [HK_2(x)]
NMAC (K_1,K_2,M) = H[FK_1<G K_2>|| M)]

1. Digital Signature:

In case of digital signature the message digest is encrypted with the entity's private key. The security of the digital signature using hash functions is based on the rigidity of finding the collision. To attain both confidentiality and digital signature it is more advisable to encrypt the message plus private-key encrypted hash code. Besides these applications cryptographic hash functions are used to learn intrusion detection and virus detection.

V. PROPOSED METHOD

In key agreement protocol all participating entities contribute information which is used to establish the shared secret key. The weaknesses of these protocols are discussed in section IIB. If two parties want to communicate the messages through public channel with absolute security, one of the means of achieving the security is using a one-time key. Here we propose a new technique secure key transport protocol where the secret key is constructed by one entity of the group of entities and transports to the other entity in a public channel network using the elliptic curve over finite field.If two legitimate entities Alice and Bob want to communicate with each other they concur with each other to use the elliptic curve Ep (a, b), and a point C on it. Alice selects a large random number λ which is less than the order of the generator of the elliptic curve Ep (a, b) and a point A on the elliptic curve. She computesA_1 = λ(C + A) and A_2 =λC

She keeps the random number λ and the point A as her secret keys and publishes A_1 and A_2 as her long term public keys or static public keys.

Similarly Bob selects a large number μ and a point B on the ellipticcurve. He computesB_1 = μ(C + B) and B_2 = μC. He keeps the random number μ and the pointB ashis secret keys and publishes B_1 and B_2 as his longterm public keys or static public keys.

Alice reconstructs A_B = λ B_2 and publishes it as her ephemeral public key or short term publickey that is specified for Bob only to communicate with him.

In the same way Bob reconstructs B_A = μ A_2 and publishes it as his ephemeral public key or short term public key that is specified for Alice only to communicate with her.

Alice's private key 1 = λ, a largerandom number less than the order of Ep (a, b) less than the order of the generator.

Alice's private key 2 = a point Aon the elliptic curve Ep (a, b)

Alice's long term public key 1 = a point A_1 on the elliptic curve Ep (a, b)

Alice's long term public key 2 = a point A_2 on the elliptic curve Ep (a, b)

Alice'sephemeral public key or short term public key

specific for Bob = a point A_B on the elliptic curve Ep (a, b)
Bobs private key 1 = µ, a large random number lessthan the order ofthe generator Ep (a, b) less than the order of the generator
Bobs private key 2 = B, a point on the elliptic curve Ep (a, b)
Bobs long term public key 1 = B_1, a point on the elliptic curve Ep (a, b)
Bobs long term public key 2 = B_2, a point on the elliptic curve Ep (a, b)
Bobs ephemeral public key or short term public key specific for Alice = B_A, a point on the ellipticcurve

A. Encryption of the secret key

Suppose Bob wants to use a point S on the elliptic curve as the secret key in communication with Alice. The secret key S is a pair of numbers. Bob encrypts the secret key S as follows.

$S^E = S + A_B + \left(\frac{B_1}{\mu} - B\right)$

To achieve the known-key security the active entities in the group of network A and B discards the ephemeral public key or short term public key and reconstruct time to time.

B. Construction of Hash Message Authenticated Code (HMAC)

Bob computes the Hash Message Authentication Code (HMAC) using his long term public keys B_1, $B_2$and the secret key SE to be transported (i.e., the message to be sent to the other entity). Here B_1, $B_2$and S^E are points on elliptic curve Ep(a, b).Suppose$B_1=(b_1^1, b_1^2)$.
$B_2=(b_2^1, b_2^2)$,$S^E=(S_1^E, S_2^E)$
First Bob calculates P_1, P_2 the points on the elliptic curve as $P_1=B_1.S^E = (p_1^1, p_1^2)$
$P_2=B_2.S^E = (p_2^1, p_2^2)$
For conventional encryption Bob generates a single number from the pair of numbers
(x , y) i.e., the point on the elliptic curve using some function F(x,y),say $F(x,y)=x^2+y^2$.Bob computes the K_1,K_2 applying the function F(x,y) on the above computed points P_1,P_2 as
$K_1=F[P_1]=F[p_1^1, p_1^2] = [p_1^1]^2 + [p_1^2]^2$
$K_2=F[P_2]=F[p_2^1, p_2^2] = [p_2^1]^2 + [p_2^2]^2$
$K = K_1^2 + K_2^2$ $M=(s_1^E)^2 + (s_2^E)^2$
Then the Hash message Authentication Code HMAC(K,M) is determined. Bob transports the secret key S^E to Alice along with the Hash Message Authentication Code (HMAC). The scheme of construction of thekey K from the static public keys of the entities public keys, the function F(x, y) used and the message digest algorithm (eg.MD5, SHA1 etc.) are agreement between the active participating entities in the group of network. The hardness of the Hash Message Authentication Code (HMAC) relies on the scheme of construction and the parameters involved. To defend the success probability of the adversary the function F(x,y) may be varied time to time, or the function F(x,y) may be defined asF(x, y) = $x^2 + y^2 + N^2$, the natural number 1,2,3,.......i.e., N takes 1 for the first message, 2 for the second message and soon.

C. Decryption

Verification of Hash Message Authentication Code

Alice after receiving the secret key along with the Hash Message Authentication Code first computes the secret keys K_1, K_2 by using Bob's long term public keys B_1, B_2 and the secret key S^E received. Then she calculates theHash Message Authentication Code according tothe agreed upon scheme, Function F(x,y) and the Hash algorithm. Then she verifies the calculated hash value is same as received hash value. After confirming the Hash Message Authentication Code (HMAC) Alice decrypts the transported secret key as follows
Alice retrieves S as

$$S = S^E - B_A - \left(\frac{A_1}{a} - A\right)$$

The Decryption works out properly:-

$$S^E = S + A_B + \left(\frac{B_1}{«} - B\right)$$

$$S = S^E - B_A - \left(\frac{A_1}{a} - A\right)$$

S + λµ C + (B+C-B)- λµ C - (A+C-A)

= S

VI. Example

Consider an elliptic curve whose equationy$_2$=x$_3$+22x+15 . The graph of elliptic curve is

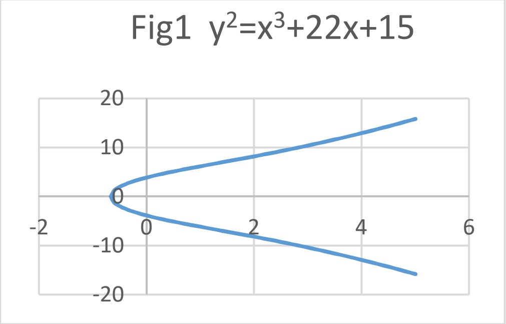
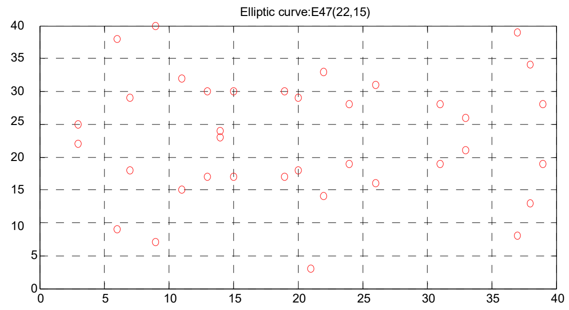


Fig1 $y^2=x^3+22x+15$

Now consider the elliptic curve E 47 (22, 15). The points on the curve E 47 (22,15) are

{∞, (3,22),(3,25),(6,9),(6,38),(7,18),(7,29),(9,7),(9,40),(11,15),(11,32),(13,17),(13,30),(14,23),(14,24),(15,17),(15,30),(19,17),(19,30),(20,18),(20,29),(21,3),(21,44),(22,14),(22,33),(24,19),(24,28),(26,16),(26,31),(31,19),(31,28)}

The graph of the function



Let C= (13, 30).Alice selects a random number λ=4, any point A= (20, 18) on the elliptic curve. She computes
$A_1 = \lambda(C + A) = 4[(13, 30) + (20, 18)] = (31, 28)$
$A_2 = \lambda C = (3, 22)$
She keeps the random number λ and the point A on the elliptic curve as her secret keys and publishes A1 and A2 as her long term public keys or static public keys.
Similarly Bob selects μ =6,B=(9,7) on the elliptic curve. He computes $B_1 = \mu(C + B) = (20, 29)$$B_2 = \mu C = (24,19)$
He keeps the random number μ and the point B on the elliptic curve as his secret keys and publishes B_1 and B_2 as his long term public keys or static public keys.
Again Alice computes $A_B = \lambda\ B_2 = (7,29)$ and publishes as her short term public key specifically to communicate with Bob.
Bob computes $B_A = \mu A_2 = (7,29)$ and publishes as his short time public key specifically to communicate with Alice.
Encryption of the secret key by Bob: If Bob wants to send the secret key S= (6,9) on the elliptic curve $E_{47}(22,15)$ he encrypts $S^E = S + A_B + \left(\frac{B_1}{\mu} - B\right) = (21,3)$.

Construction of Hash Message Authentication Code (HMAC):

$B_1=(20,29)$,$B_2=(24,19)$,$S_E=(21,3)$ are the points on the elliptic curve E47(22,15)
Bob calculates the points P_1 and P_2 on the elliptic curve as
$P_1=B_1.S_E= (420,87)$$P_2=B_2.S_E= (504, 57)$
For conventional encryption Bob generates a single number from the pair of numbers (x, y) say$F(x, y)=x^2+y^2$.
Bob computes K_1, K_2 as
$K_1=F [P_1] = 183969$
K2=F [P2] =257265

Then the Hash message Authentication code can be calculated as $h=H[F(K_1\ K_2)\|M]$
Where $F(K_1,K_2)=K_1+K_2= 100027873186$
And the message $M=(s_1^E)^2 + (s_2^E)^2 = 450$
Therefore h =
812ae199d894caabe22102fa684cec6404f70c57 (SHA- 1)
Bob sends the encrypted secret key to Alice along with the Hash Message Authentication Code (HMAC) value to Alice over an insecure channel.
Decryption:
At reception end Alice after getting the encrypted secret key along with the tagged HMAC value first confirms the communication by Bob. She calculates the HMAC value by using the agreed upon mechanism and checks the obtained HMAC value tagged to the encrypted secret key equals the calculated HMAC value. If the value matches with received value then she confirms that the communication is from Bob only as it is not altered on the way from Bob to Alice. Then she starts decrypting the received secret key
Decryption of the secret key:

$S^E = S + A_B + \left(\frac{B_1}{\mu} - B\right) = (21,3)$

S as $S = S^E - B_A - \left(\frac{A_1}{\lambda} - A\right) = (6,9)$

Security Analysis and Conclusions: The secure key transport protocol proposed here achieves all the desirable security attributes of Authentic KeyAgreement (AK)protocol as well as non-repudiation of the communication among the entities of the group.

The present algorithm designed here attains the Known Key – Security because the ephemeral public key or short term public key published by each entity in a group is specific. i.e., the entity A publishes the ephemeral public key A_B specific to B only to communicate with B using B's long term or static public keys.

The present algorithm possesses best forward secrecy. Even if the static private keys of the entities λ , μ are compromised the previous session keys established by legitimate entities are not influenced because the formation of ephemeral keys are protected by hard problem called Elliptic Curve Discrete Logarithm Problem (ECDLP).

Suppose the entity A's long term or static private key is disclosed an adversary impersonates the entity A to B. But the adversary cannot impersonate B to A unless he has the knowledge about B's long term private key. So, the present algorithm is safe against the key compromise impersonation.

In addition it prevents the unknown key – share because the Bob uses his own private key μ , B ; his long term or static public key B_1 and the short term or ephemeral public key A_B which is specific to Bob only.

Besides all these security attributes the secret which is transported is shielded with Hash message Authentication Code (HMAC). HMAC value is calculated by the sender utilizing his static or long term public keys and the secret key which is being transported called the message. In calculating the HMAC value the active entities in the group of network concurto use a random function F(x,Y),a mechanism to construct the Hash message Authentication Code (HMAC) value and available Hash algorithms like MD 5, SHA – 1 etc. To conquer the active attacks on HMAC value the random function F(x,y) is changed periodically. The message to be communicated (i.e., the secret key to be transported) is concatenated with the sender's long term public keys in calculating the Hash Message Authentication Code (HMAC) value. At the receiver's end the receiver first calculates HMAC value and verifies the HMAC value with communicated hash value to authenticate thecommunication. The summary of this algorithm is it provides at most security in key communication that is guarded by Hash Message Authentication Code (HMAC).

References:

[1] B. Preneel and P. Van Oorschot On the security of two MAC algorithms Advanes in Cryptology EUROCRYPT 96, Proceedings Lecture Notes in Computer Science Vol. 1070 U. Maurer ed., Springer Verlag 1996.

[2] A. Menzes, P. Van Oorschot and S. Vanstone Hand book of Applied Cryptography CRC Press 1997.

[3] B. Song and K. Kim Two- pass Authenticated Key agreement protocol with key confirmation Progress in Cryptology INDOCRYPT LNCS 1977 Springer – Verlag pp 237-249, December 2000.

[4] Simon Blake-Wilson, Don Johnson, Alfred Menezes "Key agreement protocols and their security analysis", IMA International Conference on Cryptography and Coding

[5] ANSI X9.63-1997, Elliptic Curve Key Agreement and Key Transport Protocols, October 1997, working draft.

[6] B. den Boer, A. Bosselaeres, "Collosions for the Compression function of MD 5" Advances in Cryptology EUROCRYPT ' 93 proceedings Springer – Verlag 1994.

[7]X.Y. wang, F.D. Guo, X.J. Lai, H.B. Yu "Collisions for Hash functions MD 4, MD 5 HAVEL -128 and RIPEMD, rump session of CRYPTO'04 E-print 2004

[8] "Hash Functions in Cryptography" Master of Science thesis Joseph Sterling Grah submitted to University I Bergen June 2008.

[9] Junko Nakajima and Mitsuramatsui "Performnce Analysis and Parallel Implementation of dedicated hash functions" Proc. Of EUROCRYPT 2002 Lecture Notes in Computer Science 2332 Springer pp165-180 2002

[10] Narn-Yih Lee, Chein – Nan Wu, "Authenticated multiple key exchange protocols based on Elliptic curves and blinear pairings" Computers & Electrical EngineeringVolume 34, Issue 1, January 2008, Pages 12-20

[11] Smart M.P. "The Discrete Logarithm on Problem on Elliptic Curve of Trace one" Journa of Cryptology 1999 Vol. 12, No.No. Page 193-196.