American Journal of Applied Sciences 2 (8): 1232-1235, 2005
ISSN 1546-9239

Key Exchange Protocol in Elliptic Curve Cryptography with No Public Point

[1]K. Kaabneh and [2]H. Al-Bdour
[1]Department of Computer Science,
Amman Arab University for Graduate Studies, Amman, Jordan
[2]Departmeny of Information Technology, Mutah University, Karak, Jordan

Abstract: In recent years some cryptographic algorithms have gained popularity due to properties that make them suitable for use in constrained environment like mobile information appliances, where computing resources and power availability are limited. One of these cryptosystems is Elliptic curve which requires less computational power, memory and communication bandwidth compared to other cryptosystem. This makes the elliptic curve cryptography to gain wide acceptance as an alternative to conventional cryptosystems (DSA, RSA, AES, etc.). All existing protocols for elliptic curve cryptosystems that are used for either key exchange or for ciphering, assume that the curve E, the field F_q and a point P on the curve are all public. In this research we propose a modified protocol for elliptic curve key exchange based on elliptic curve over rings, assuming that only the curve E and F_q are public, keeping the base point P secret, which make attacking the cryptosystem harder by the eavesdropper. Also we provide imbedded authentication, so our protocol does not suffer from the man in the middle attack.

Key words: Elliptic curve, key exchange, man-in-middle, finite field

INTRODUCTION

With the proliferation of the handheld wireless information appliances, the ability to perform security functions with limited computing resources has become increasingly important. In mobile devices such as personal digital assistants (PDAs) and multimedia cell phones, the processing resources, memory and power are all very limited, but he needs for secure transmission of information may increase due to the vulnerability to attackers of the publicly accessible wireless transmission channel[1].

New smaller and faster security algorithms provide part of the solution, the elliptic curve cryptography ECC provides a faster alternative for public key cryptography. Much smaller key lengths are required with ECC to provide a desired level of security, which means faster key exchange, user authentication, signature generation and verification, in addition to smaller key storage needs. The terms elliptic curve cipher and elliptic curve cryptography refers to an existing generic cryptosystem which use numbers generated from an elliptic curve. Empirical evidence suggests that cryptosystems that utilize number derived from elliptic curve can be more secure[2]. As with all cryptosystems and especially with public-key cryptosystems, it takes years of public evaluation before a reasonable level of confidence in a new system is established. ECC seems to have reached that level now. In the last couple of years, the first commercial implementations have appeared, as toolkits but also in real-world applications, such as email security, web security, smart cards, etc. The security of ECC has not been proven but it is based on the difficulty of computing the elliptic curve discrete logarithm in the elliptic curve group[3].

The elliptic curve: An elliptic curve is the set of solutions of an equation of the form:

$$y^2 [+ xy] = x^3 + ax^2 + b \quad (1)$$

where, x and y are variables, a and b are constants. However, these quantities are not necessarily real numbers; instead they may be valued from any field. For cryptographic purposes we always use a "finite" field - that is x, y, a and b are chosen from a finite set of distinct values[4]. If $x^3 + ax + b$ contains no repeated factors, or equivalently if $4a^3 + 27b^2$ is not 0, then the elliptic curve: $y^2 = x^3 + ax + b$[2]. An elliptic curve over real numbers may be defined as the set of points (x, y) which satisfy an elliptic curve equation 1, where x, y, a and b are real numbers.

Elliptic curve over a finite field F_p: Using the real numbers for cryptography will cause a problem because it is very hard to store them precisely in computer memory and to predict how much storage we will need for them. This problem can be solved by using finite fields, i.e., Fields with a finite number of elements. Since the number of elements is finite, we can find a

Corresponding Author: K. Kaabneh, Assistant Professor, Department of Computer Science, Amman Arab University for Graduate Studies, Amman, Jordan

unique representation for each of them, which allows us to store and handle the elements in a manageable way. The number of elements of a finite field (or Galois field) is always a positive prime power p^n, the corresponding field is denoted GF(p^n). Two special cases are popular for use in ECPKCs: fields of the form GF (p) (or n = 1) and fields of the form GF (2n), or p = 2. For GF (p), the formulas are the same as for the reals; for GF (2n) they are slightly different[2].
Given a, b satisfying:

$$4a^3 + 27b^2 \bmod p <> 0$$

Then the elliptic curve over a finite field F_p with parameters a and b is defined as the set of points (x,y) satisfying the equation $Y^2 = x^3 + ax + b$ together with a special point O which is the point at infinity , such a curve will be denoted $E_p(a,b)$[5].

Let P1 and P2 be two points on E , it is possible to find a closed formula that gives the coordinates (x_S,y_S) of the sum P_S of two points P1 and P2 as a function of their coordinates (x1, y1) and (x2, y2):

$$x_S = \lambda^2 - x_1 - x_2$$
$$y_S = \lambda(x_1 - x_S) - y_1$$
$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1} \text{if } P_1 \neq P_2 \\ \frac{3x_1^2 - \alpha}{2y_1} \text{if } P_1 \neq P_2 \end{cases}$$

Recall that the field F_p uses the numbers from 0 to p - 1 and computations end by taking the remainder on division by p. For example, in F_{23} the field is composed of integers from 0 to 22 and any operation within this field will result in an integer also between 0 and 22.

Key exchange: Key exchange protocols allow two parties to agree on a secret shared secret key that they can use to do further encryption for a long message. One of these protocols is the Diffie-Hellman, which is the most used one. The Elliptic curve Diffie-Helman is considered as an extension to the standard Diffie-Hellman.

Elliptic curve Diffie-Helman: Elliptic curve Diffie-Helman protocol (ECDH) is one of the key exchange protocols used to establish a shared key between two parties. ECDH protocol is based on the additive elliptic curve group. ECDH begin by selecting the underlying field GF (P) or GF (2^k) , the curve E with parameters a, b and the base point P. The order of the base point P is equal to n. The standards often suggest that we select an elliptic curve with prime order and therefore any element of the group would be selected and their order will be the prime number n. At the end of the protocol, the communicating parties end up with the same value K which is a point on the curve.

Alice	Communication	Bob
Choose a random number a ϵ F_q		Choose a random number b ϵ F_q
Compute aP		Compute bP
Retrieve bP	aP → ← bP	Retrieve aP
Compute abP		Compute abP

Fig. 1: Elliptic curve diffie-helman

As shown in Fig. 1 the Elliptic curve Diffie-Helman protocol will work as follows.

Setup: Alice and Bob agree on a common group G and a common group element g. Then Alice chooses a secret number a, which serves as her secret key and Bob chooses a secret key b.

Communication: Alice computes ga and sends it to Bob over a public channel; Bob sends gb to Alice. Although ga and gb is closely related to a and b respectively, the hardness of the DLP ensures that the secret keys cannot be computed from them in a practical situation. Therefore, ga and gb can serve as public keys, corresponding to the private keys of Alice and Bob respectively.

Final step: Alice takes Bob's public key and computes (gb)a = gab; Bob computes (ga)b = gab. As we see, Alice and Bob obtain the same result and this result could not be computed by an adversary who only knows the public keys. Therefore Alice and Bob have agreed on a shared secret key.

Note that the hardness of the DLP does not guarantee the security of the Diffie-Hellman protocol: computing g^{ab} from g^a and g^b may be easier than computing a from ga or b from gb.

Necessity of authenticated key exchange protocol: In the standard Elliptic Curve Diffie-Helman key exchange Fig. 1, a shared secret key is established by multiplying the public point by both the secret key generated by Alice and Bob Ted has now exchanged keys with Bob and Alice. Bob. This scheme has one major problem, it's not authenticated. This means that Alice has no way of knowing if bP actually was sent from Bob. A third party Ted could have intercepted the transmission from Bob and substituted his own value (say cP) as shown in Fig. 2. Ted can exchange keys with Bob and Alice. Bob and Alice think they've exchanged keys with each other.

Alice	Ted (Middle man)	Bob
Choose a random number a ∈ F_q	Choose a random number c ∈ F_q	Choose a random number b ∈ F_q
Compute aP	Compute cP	Compute bP
Retrieve cP	Retrieve aP, bP from both Alice and Bob	Retrieve cP
Compute acP	Compute acP , bcP	Compute bcP

Fig. 2: Man-in-the-middle attack

All Ted has to do now decrypts the message from Bob re-encrypt it with Alice's key and he can monitor the communication without detection. One of solutions to this dilemma is described in Viega[6]. It involves the use of a trusted "certificate authority" or CA. When queried the CA and returns a digitally signed "certificate" that can be compared to one that has been transmitted by another means. In 7] an authenticated key exchange based on the difficulty of the q^{th} root problem was described. In[8] a new three pass key agreement protocol with key confirmation is proposed.

Proposed Protocol: All existing protocols for elliptic curve cryptosystems assume that the curve E, the field F_q and a point P on the curve are all public. This proposed protocol assumes that only the curve E and F_q are public, keeping the base point P secret, which make attacking the cryptosystem harder by the eavesdropper. The proposed protocol is also secure against the "man-in-the-middle" attack allows an eavesdropper to monitor communication between two parties.

Protocol: Key exchange based on E_n (a, b) can be set up as follows.

The two parties agreed on the Elliptic curve equation, that is select prime number n and two parameters a and b , E_n (a,b): $Y^2 = x^3 + ax + b$, satisfying gcd $(4a^3+27b^2, n) = 1$.

Select $n = pq$ as in RSA, p and q are two prime numbers. Ø (n)= (p-1) (q-1). Select an integer e where gcd (Ø (n), e) = 1, where $1< e < Ø (n)$, d then can be calculated by this formula:

$$d \equiv e^{-1} \bmod Ø (n)^{[9]}$$

Now A will select the following:

X_a A's first ephemeral key, random number in F_n
R_a A's second ephemeral key, random number in F_n
P_a Elliptic curve point chosen by A

Similarly B has X_b , R_b , P_b.

The security of this technique is based on the discrete logarithm problem for both the elliptic curve and RSA because the proposed protocol has the characteristics of both of them.

The agreement between two entities A and B will be proposed in two-pass key agreement, the scheme works as follows:

1. A will compute the point: $G_a = X_aP_a$ and send it to B , on the other hand B will compute the point : $G_b = X_bP_b$ and send it to A
2. A receives G_b from B and compute the point: $S_a = R_aG_b$ and send it to B.
3. B receives G_a and computes the point: $S_b = R_bG_a$ and send it to A
4. A receives S_b from B and compute the session key: $K = e (S_a + S_b)$
5. B receives S_a from A and compute the session key: $K = e (S_a + S_b)$

In the last two steps the resulting point much be checked no to be equal to O. If K= O the scheme will terminate with failure.

Multiplication by e is important for two reasons. First it gives the protocol public key characteristics, so that the public key will K for both parties and the private key will be $d (S_a + S_b)$. Second it increases the security of the protocol, so it will not suffer from the man-in-the-middle attack as discussed in the next two sections.

Protocol characteristics: Here we prove our protocol meets the following desirable security attributes.

Known-Key Security: The protocol provides known-key security. Each run of the protocol between two entities A and B should produce a unique session key. Although an adversary has learned some other session keys, he can't compute K, because he doesn't know private keys d.

Perfect Forward Secrecy: It also possesses forward secrecy. Suppose that shared key is compromised.

However, the secrecy of previous session keys established by honest entities is not affected, because in each time the two parties need to share a session key they select different points on the elliptic curve and different ephemeral key.

Unknown key-share: It also prevents unknown key-share. It is difficult for the adversity to know the private key for any party, or the shared key. This is discussed in the next session.

How the attack is blocked: Now we will show how the protocol doesn't suffer from the man-in-the-middle

attack. The imposer would be able to select the following he elliptic curve

X_i Imposer's first ephemeral key, random number in F_n
R_i Imposer's second ephemeral key, random number in F_n
P_i Elliptic curve point chosen by Imposer

And so still he can compute $G_i = X_iP_i$ and send the value of both A and B, but still the attack is blocked because of four reasons:

1. The public point in the elliptic curve is not known.
2. The protocol is to pass key agreement, so the imposer must catch two values to continue his attack. These values are G_i , S_i where i = A, B . So for each key agreement the imposer must watch the line to get these values.
3. It's difficult for the imposer to calculate $K = e (S_a + S_b)$ because he will not know the value of e.
4. If the imposer known the value K, he will not be able to use this value to communicate with the parties, because the private key (which depend on d) for both parties will be hidden.

CONCLUSION

Using the elliptic curve in cryptography has been gaining a lot of attention lately. In this research a new protocol for exchanging key between two parties was defined. This new protocol has two major advantages over all previous key exchange protocol, first this technique offers no exchange in public point which will increase the security, the second one is that this protocol does not suffer from the man-in-the-middle problem, it has an embedded solution to this problem that has only two-pass key agreement. The proposed protocol is also easy to implement, it's based on the characteristics of both elliptic curve and RSA encryption systems. The protocol has only to pass key agreements, which mean no communication overhead will be added as offered in some key-exchange algorithms.

REFERENCES

1. Murat Fiskiran, A. and B. Ruby Lee, 2002. Workload characterization of elliptic curve cryptography and other network security algorithms for constrained environments. Proc. IEEE Intl. Workshop on Workload Characterization, pp: 127-137
2. De Win, E. and B. Preneel, 1998. Elliptic curve public-key cryptosystems - an introduction. State of the Art in Applied Cryptography, LNCS 1528, pp: 131-141.
3. Aydos, M., E. Savas and C.K. KoV, 1999. Implementing network security protocols based on elliptic curve cryptography. Proc. fourth Symp. Computer Networks, pp: 130-139.
4. Online Elliptic curve cryptography Tutorial-Certicom, http://www.certicom.com/index.php?action=ecc_tutorial,home
5. Koyama, K., U.M. Maurer, T. Okamoto and S.A. Vanstone, 1991. New Public-Key Schemes Based on Elliptic Curves over the Ring Zn. CRYPTO'91, LNCS 576, pp: 252-266.
6. Viega, J., M. Messier and P. Chandra, 2002. Network Security with OpenSSL, O'Reilly. First Edn.
7. Anna M. Johnston and Peter Gemmell, 2002. Authenticated key exchange provably secure against the man-in-the-middle attack. J. Cryptol., 15: 139-148.
8. Al_Sultan, K., M. Saeb, M. Elmessiery and U. El-Raouf, 2003. A new two-pass key agreement protocol. Proc. IEEE Midwest, Symp. Circuits, Systems and Computers.
9. Sallings, W., 2003. Cryptography and Network Security. Prentice Hall, Third Edn.