A Practical Authentication with Key Agreement Protocol for Satellite Communications on ECC

H.H. Huang, J.H. Chen, B.J. Huang
School of Mathematics and Statistics, Wuhan University, Wuhan, China

ABSTRACT: In order to remove the pitfalls and enhance the subsequent schemes, a practical authentication scheme is introduced by using elliptic curve cryptosystem (ECC) and identity-based cryptography (IBC) with three-round handshake identification system. It supports seven essential advantages and adequate security attributes. The high performance and countering the strong DoS attack are two main superiorities.
KEYWORD: satellite communications; authentication; ECC; DoS attack

1 INSTRUCTIONS

Due to geographic and environment limitations, traditional personal communication systems broadcast week and delay signals [1-2]. In the last decade, considerable attention has been paid to the low-earth-orbit satellite communications system (LSC system) for its numbers of advantages: Short transmission delay, small attenuation signal, large communication area and far broadcasting range [2-9]. The main components in a LSC system are the low-earth-orbit satellites, the network control center (NCC), the gateways and the mobile devices [8]. Recently, we analyze the security aspects of Chang et al's scheme [9] and identify many loopholes that are common of other related scheme [7-12]. Therefore, an enhanced authentication protocol is proposed using elliptic curve cryptosystem (ECC) and identity-based cryptography (IBC) in three-round handshake identification system. The scheme not only provides seven essential requirements but also counter the strong new DoS attack.

The paper is organized as follows. Section 2 gives preliminaries. The scheme is introduced in Section 3. Section 4 discusses the security of the scheme. We draw some concluding remarks in Section 5.

2 PRELIMINARIES

2.1 *Elliptic curve cryptosystem (ECC)*

In this paper, we just give a simple description of the ECC defined over a prime field F_P. A non-singular elliptic curve $F_P(a,b)$ over F_P is defined by an equation $y^2 = x^3 + ax + b(\mathrm{mod}\ p)$, $a,b \in F_P$ with the discriminant is $\Delta = 4a^3 + 27b^2 \neq 0(\mathrm{mod}\ p)$. Then the set $G_P = \{(x,y) \mid x,y \in F_P\ and (x,y) \in E_P(a,b)\} \cup \{O\}$ can form a cyclic additive elliptic curve group, where the point O is identity element of G_P. Let P is a base of G_P with an order n as $nP = o$ for the smallest integer $n > 0$. The scalar multiplication on the group G_P defined as $kP = P + P + \cdots + P(k\ times)$. Details of elliptic curve group properties are given in [15].

2.2 *Hard computational problem*

- Elliptic curve discrete logarithm problem (ECDLP): Assumed two points $Q, P \in G_P$, to find $k \in [1, n-1]$ such that $Q = kP$ is impossible.
- Computational Diffie-Hellman problem (CDHP) Assumed there are three points P, aP, bP for $a, b \in [1, n-1]$, which guarantee the points in G_P, the computation of abP is hard to G_P.

3 THE PTOPOSED PROTOCOL

The protocol consists of four phases. The notations used in the paper are defined as follows:

- U : a mobile user
- ID_U : *U's* unique identity
- pw : *U*'s password
- x : the private key of *NCC*
- ID_{leos} : *LEOS's* identity
- $h(\cdot)$: one-way hash function
- SK : share session key
- *NCC* : a network control center
- $\oplus$: exclusive-or operation
- *LEOS* : a low-earth-orbit satellite

3.1 *Initialization phase*

The *NCC* executes this phase as follows:

1) Choose a secure elliptic curve equation $E_P(a,b)$ and a generator point P of a cyclic additive elliptic curve group G_P with order n, where p is a k-bit prime number; Select a random number $x \in [1, n-1]$ as its private key and computes the corresponding public key $P_N = xP$.
2) Pick a one-way key derivation function $kdf : \{0,1\}^* \times \{0,1\}^* \times G_P \to \{0,1\}^k$ and define an operation $a \circ A = (a \oplus x_A, y_A)$, $a \in Z_P^*, A = (x_A, y_A) \in G_P$. Publish $\{E_P(a,b), n, P, P_N, h(\cdot), kdf\}$ as the system parameters and keeps its private key x secret.

3.2 *Registration phase*

If a mobile user *U* wants to register at the system, this phase is performed only once as follows:

1) *U* freely chooses his valid identity ID_U and password pw to compute $y = h(ID_U, pw)$. Then he sends the message $\{ID_U, y\}$ to *NCC* through a secure channel.
2) When receives the message, *NCC* selects a nonce N_0 as the initial temporary identity for *U*. Then *NCC* computes $S = (x + h(ID_U))^{-1} P$, $L = y \circ S$ and $v = h(y, ID_U)$. Finally, *NCC* stores $\{N_0, ID_U\}$ into the verifier-table and delivers the message $M_0 = \{N_0, L, v\}$ to *U* via a secure channel.

3.3 *Authentication phase*

When *U* intends to login the system, he inputs his ID_U and pw into his device. Then, this phase is performed as follows:

1) The device computes $y = h(ID_U, pw)$ and compares whether $h(y, ID_U) = ?v$ or not. If they are unequal which means *U* inputs incorrect ID_U or pw, it stops and prompts user input again; or else , it selects a variable $r_1 \in [1, n-1]$ to compute $R = r_1 P$, $Q = r_1(y \circ L)$ and $\alpha = h(N_0, ID_U, R, Q)$. Then, the device sends the request message $\{N_0, R, \alpha\}$ to *LEOS*.
2) Upon receiving the message, *LEOS* shifts it and its identity ID_{leos} to *NCC*.
3) On receiving $\{N_0, R, \alpha, ID_{leos}\}$, *NCC* uses N_0 to find the matching ID_U in the verifier-table. If cannot find, *NCC* refuses the request; or else, it computes $Q' = (x + h(ID_U))^{-1} R$, $\alpha' = h(N_0, ID_U, R, Q')$ and checks the condition $\alpha' = ?\alpha$. If the condition fails, *NCC* refuses this session; otherwise, it generates a nonce N_1 and computes the session key $SK = kdf(N_0, ID_U, Q')$, $\beta = h(N_1, SK)$. Then *NCC* sends $\{N_1, \beta, ID_{leos}\}$ to *LEOS*.
4) *LEOS* shifts the message $\{N_1, \beta\}$ to *U*. when *U* receives the response message, he computes $SK' = kdf(N_0, ID_U, Q)$, $\beta' = h(N_1, SK')$ and checks the condition $\beta' = ?\beta$. If it fails, *U* refuses *NCC*; otherwise, *U* stores N_1 next to N_0 in the device. Finally, *U* computes $\delta = h(N_0, N_1, ID_U, SK')$ and sends $\{\delta\}$ to *LEOS*.
5) *LEOS* shifts $\{\delta, ID_{leos}\}$ to *NCC*. When *NCC* gets the message, it computes $\delta' = h(N_0, N_1, ID_U, SK)$ and checks the condition $\delta' = ?\delta$. If the condition holds, it means *U* has received the next temporary identity N_1 and shares the session key SK. Hence, *NCC* can update N_0 as N_1 and apply SK to encrypt the exchange data with *U*. Otherwise, this session is terminated.

U stores N_1 next to N_0 rather than updates N_0 as N_1 in step 4. It means *U* will keep N_0 until receive his next temporary identity N_2. Then, *U* updates N_0 as N_2. As Fig. 2, this technique is used to overcome the 'De-synchronization challenge'.

3.4 *Password change phase*

This phase is activated whenever *U* needs to replace his password. Firstly, *U* inputs ID_U and pw to the device and asks for changing the password. Then the device will automatically perform as follows:

1) The device computes $y = h(ID_U, pw)$ and compares whether $h(y, ID_U) = ?v$ or not. If they are not equal, it refuses the request. Otherwise, *U* is asked to input a new password pw_{new}.
2) The device computes $y_{new} = h(ID_U, pw_{new})$, $L_{new} = y_{new} \circ (y \circ L)$ and $v_{new} = h(y_{new}, ID_U)$. Finally, it replaces the original v, L using v_{new}, L_{new} separately, which completes this phase.

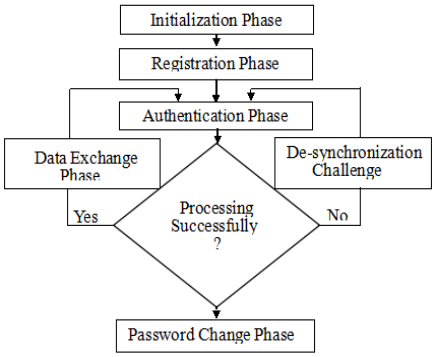

Fig.1 Flow diagram of the protocol

4 SECURITY ANALYSIS OF THE PROTOCOL

4.1 *Valid mutual Authentication*

Through the first and third handshake, *NCC* authenticates *U* because just the legal *U* who holds a real identity ID_U and corresponding password pw can compute valid α and δ for verification; through the second handshake, *U* authenticates *NCC* because only the true *NCC* can generate β with the master key x and *U*'s identity ID_U. Meanwhile, the messages M_1, M_2 and M_3 are authenticated to come from the original sender. For example, the values N_0 and R in M_1 are sure to come from *U* since they are bound to *U*'s identity ID_U and password pw via the value α. In the first handshake *NCC* also authenticated the verifier-table values $\{N_0, ID_U\}$ belong to *U* because these values are verified with the message M_1 via the value α.

4.2 *Data confidentiality and integrity*

The value r_1 is protected by solving the ECDLP as knowing P, R to find an integer r_1 such that $R = r_1 P$ is hard. The session key SK is safeguarded by the secure one-way hash function and mixed with the nonce N_1 in $\beta = h(N_1, SK)$. The exchange datum can be encrypted with the shared session key SK. If an adversary *Z* eavesdrops and manipulates the messages M_1, M_2 and M_3, this attack will be quickly detected by checking α, β and δ.

4.3 *Perfect session key secrecy*

Supposed an adversary *Z* captures all the messages M_1, M_2 *and* M_3, even extracts the values $\{N_0, L, v\}$ stored in *U*'s device and steals the values $\{N_0, ID_U\}$ in the verifier-table. At first, he may try to compute the value $S = (x + h(ID_U))^{-1} P$ as it's important to the system. In one way, *Z* may apply several users' identity $\{ID_k\}$ to compute values $\{P, P_N = xP, h(ID_U), (h(ID_1), (x + h(ID_1))^{-1} P), \ldots, (h(ID_k), (x + h(ID_k))^{-1} P)\}$ for getting S. But, this way is meaningless under the basis of k-CAA1 [17]. Put that problem aside, *Z* obtains S someway. Then, he tries to compute the session key SK using $SK = kdf(N_0, ID_U, Q')$ with $Q' = (x + h(ID_U))^{-1} R = r_1 (x + h(ID_U))^{-1} P$. But, knowing the pair $(S, R) = \left((x + h(ID_U))^{-1} P, r_1 P\right)$ to compute the value Q' is equivalent to solve the CDHP. On the other hand, he may apply $SK' = kdf(N_0, ID_U, Q)$ with $Q = r_1 (y \circ L) = r_1 S$. But, r_1 is a secret value. In addition, if the current session key is compromised, *Z* cannot obtain the forward/backward session keys as each session key is independent depending on the random number r_1 and the nonce N_0.

4.4 *User's privacy*

U submits $y = h(ID_U, pw)$ instead of the original password to keep his password from *NCC*. Then a temporary identity N_0 is issued and is refreshed every session to protect *U*'s anonymity. Furthermore, there is not any relevant information about *U* in the messages M_1, M_2 *and* M_3, which can be checked whether the captured messages belong to the specific *U*. A vicious adversary may extract the value v from *U*'s device to launch the off-line password guessing attack, as it's in the form $v = h(y, ID_U) = h(h(ID_U, pw), ID_U)$ with *U*'s ID_U and pw. For this issue, we can strengthen security by asking *U* to choose the high entropy identity on the basis of IBC. Likewise, for the on-line password guessing attack, we set the wrong-input login threshold value h (e.g. h=3) at each start of the device.

4.5 *No sensitive data maintained by NCC and users*

There are only verifier-tables $\{N_0, ID_U\}$ stored in *NCC* side and datum $\{N_0, L, v\}$ stored in *U* side. Any of these values are deadly to the proposed scheme. Because the value N_0 is just a nonce identity of *U* and is refreshed every session; for *U*'s identity ID_U, *U* holds the master password pw; the value v is derived from $v = h(y, ID_U)$ to protect the fatal value $y = h(ID_U, pw)$, and the value L just masks the value S in the form $L = y \circ S$.

4.6 *Low storage, fast computation and update cost*

Comparing the number of parameters stored at *NCC* side and *U* side in Chang et al's scheme and ours: the verifier-tables $\{N_0, ID_U, r, s\}$ to $\{N_0, ID_U\}$, the devices $\{N_0, l\}$ to $\{N_0, N_1, L, v\}$, there are equal parameters in all. However, 160-bit ECC and 1024-bit discrete logarithm problem (DLP) have the same security level [15]. Hence, in the same security level, our parameter storage cost is less than 84% of in Chang et al's. The operations comparison is shown in Table1. The performance of operations is ranked from high to low under the rules in [14]. The elliptic curve scalar point multiplication (PM) is much less than the modular exponentiation (ME) [15]. From Table 1 and considering the computational bits, the

whole computation cost of our scheme is much less than Chang's and Li's schemes, even additional the password change phase which is rarely executed in practice. From the storage requirements and computation cost at *NCC* side, the update of our system is still faster than Chang's scheme.

4.7 *A new denial of service (DoS) attack*

This attack highlights the weakness of the LSC system by jamming a message in application layer [8]. As outlined in Fig.1, we address the problem particularly, which we called 'De-synchronization Challenge'. Generally, *U* uses his temporary identity N_0 to login the system and goes to the exchange data phase. However, if an attacker jams the communicated messages, *U* can still use N_0 to login. Because the scheme applies three-round handshakes identification technique to ensure the synchronization of *U*'s temporary identity N_0 between the *NCC* and *U*. *NCC* updates N_0 only if it confirms *U* received the next temporary identity N_1 in the third handshake. Therefore, as long as the interference is not continuous, *U* will login the system at last.

The security comparison among related schemes is summarized in Table 2. Distinctly, our protocol provides more security attributes.

Table 1 Operations comparison among related schemes

	Registration Phase	Authentication Phase	
		U	NCC
Chang et al.[9]	1ME+2MO+1I+2M+1A+2H+1X	3H+5X	1MO+3M+1S+8H+1X
Li et al.[10]	2ME+2MO+1A+2H	4ME+4MO+1I+5H	3ME+4H+3MO
Proposed	1PM+1I+1A+3H+1X	2PM+5H+1X	1PM+1I+1A+5H

ME: modular exponentiation; PM: elliptic curve scalar point multiplication; MO: modular operation
A, S, M, I, H, X: addition, subtraction, multiplication, inverse, hash, XOR operation respectively.

Table 2 Security comparison among related schemes

Security attributes	Chen et al.[7]	Lasc et al.[8]	Chang et al.[9]	Proposed
Impersonation attack	NO	NO	NO	YES
Modification attack	NO	YES	NO	YES
Stolen-verifier	NO	NO	NO	YES
A lost smartcard attack	NO	NO	NO	YES
Session key problem	NA	NA	NO	YES
On-line and off-line guessing	NA	NA	NA	YES
A new denial of service attack	NO	YES	NO	YES
Reply attack	YES	YES	YES	YES

YES: prevents the attack; NO: does not prevent; NA: not address the attack.

5 CONCLUSIONS

In this paper, we introduced an enhanced authentication protocol using ECC and IBC with three-round challenge-response identification system. As a result, combining with the superiority of the LSC system, the scheme can be used in multifarious applications, such as Big Data environment.

REFERENCES

[1] Yiltas D.H, Zaim A. Evaluation of call blocking probabilities in LSC satellite networks. Int J Satell Commun Netw 2009; 27(2):103–115.

[2] Rendon-Morales E, Mata-D áz J. Performance evaluation of transmission control protocol variants over a satellite multimedia system with QoS. International Journal of Communication Systems 2013. DOI: 10.1002/dac.2333

[3] Shahriar A, Atiquzzaman M, Ivancic W. Network mobility in satellite networks: architecture and the protocol. International Journal of Communication Systems 2014; 26(2):177–197.

[4] Cruickshank H.S. A security system for satellite networks. IEEE Satellite System Mobile Communication Navigation. UK 1996; 187–190.

[5] Hwang M.S, Yang C.C. An authentication scheme for mobile satellite communication systems. ACM SIGOPS Operating Systems Review October 2003; 37(4):42–47.

[6] Chang Y.F, Chang C.C. An efficient authentication protocol for mobile satellite communication systems. ACM SIGOPS Operating Systems. 2005; 39(1):70–84.

[7] Chen T.H, Lee W.B. A self-verification authentication mechanism for mobile satellite communication systems. Computers & Electrical Engineering 2009; 35(1):41–48.

[8] Lasc L, Dojen R, Coffey T. Countering jamming attacks against an authentication and key agreement protocol for mobile satellite communications. Computers & Electrical Engineering. 2011; 37(2):160–168.

[9] Chang CC, Cheng TF. An authentication and key agreement protocol for satellite communications. International Journal of Communication Systems 2012. DOI: 10.1002/dac.2448

[10] Li X, Niu J.W. An enhanced smart card based remote user password authentication scheme. Journal of Network and Computer Applications. 2013 (36) 1365–1371

[11] Chen C.L, Cheng K.W. An improvement on the self-verification authentication mechanism for a mobile satellite communication system. Applied Mathematics & Information Sciences.2013; 7(1):365–374.

[12] Zhang Y.Y, Chen J.H, H B.J. Security analysis of an authentication and key agreement protocol for satellite communications. International Journal of Communication Systems 2013. DOI: 10.1002/dac.2612

[13] Zheng G, Ma H.T. Design and logical analysis on the access authentication scheme for satellite communication network. ET Information Security 2012; 6(1):6–13.

[14] Menezes AJ, Oorschot PC, Vanstone SA. Handbook of Applied Cryptography. CRC Press, USA, October 1996.

[15] H. D, M. A. Guide to elliptic curve cryptography. New York, USA: LNCS, Springer-Verlag 2004.

[16] He D.B, Chen J.H, Hu J. An ID-based client authentication with key agreement protocol for mobile client-server environment on ECC with provable security. Information Fusion, 13(2012), 223–230.