An Unbalanced Security Protocol for IoT devices

Huinan Wang and Jianmin Zhang $^{1,2}$

$^1$ State Grid Shanxi Marketing Service Center, Taiyuan, Shanxi, 030021, China
$^2$ State Grid Shanxi Marketing Service Center, Taiyuan, Shanxi, 030021, China

Corresponding author’s email: yxfwzx@sx.sgcc.com.cn; dtwhn@163.com

Abstract. Authenticated key exchange (AKE) protocol is a crucial process in Internet of Things. This paper aims to customize the Elliptic Curve Diffie-Hellman (ECDH)-based AKE according to the computational capability of devices. We propose a method to unbalance the computations in the ECDH key exchange scheme by configuring the more powerful device undertake one scalar multiplication on behalf of the computationally limited one. The analysis results show that the computationally unbalanced protocols are more friendly to computationally limited devices than the original one.

1. Introduction

key exchange (AKE) [1] is a significant procedure in protecting wireless communications. It establishes a shared secret for two devices communicating over insecure channels. The shared secret can be used directly or to derive session keys to protect communications between the devices.

A typical solution is combining the Elliptic Curve Diffie-Hellman (ECDH) key exchange scheme [6] with authentication measures such as pre-shared secret with message authentication code (MAC)[2], trusted third parties or authenticated Out-of-Band (OOB) channels[3][4][5]. A large number of ECDH-based AKE protocols have been proposed. Some have been adopted into wireless communication specifications and standards including the Bluetooth specification 5.0 [7] and IEEE standard 802.15.6[8].

However, the majority of the available ECDH-based AKE protocols are “computationally balanced”, i.e., the two parties execute equivalent computations. This is not friendly when devices may have different computational capabilities. Meanwhile, this is a quite common scenario[9][10], e.g., a powerful smart phone will have to establish secure links with a computationally limited headset. Exemplary use cases regarding AKE in practice can be found in[11]-[16],[18].

This paper thus proposes an unbalanced AKE protocol for devices with different computational capabilities. The more powerful device undertake one scalar multiplication on behalf of the computationally limited one. The analysis results show that the computationally unbalanced protocol is more friendly to computationally limited devices than the original one.
The rest of this paper is organized as follows. In Section II, we briefly introduce the system model. Section III presents the protocol. The security of the unbalanced protocol is studied in Section IV. Finally, we conclude the paper in Section V.

2. System Overview
An AKE protocol establishes an authenticated shared secret for the communicating devices. The communication model, attack model and security goals are presented as follows.

2.1. Communication Model
- Participants. An AKE protocol is executed by an initiator, denoted by A, and a responder, denoted by B. In our scenario, A and B are assumed to have significantly different computational capabilities.
- Channels. The channels between A and B include normal wireless channels (e.g., enabled by Bluetooth, Wi-Fi, etc.) and authenticated OOB channels. For example, the display OOB channels require users to compare the numbers shown on the displays of communicating devices.

2.2. Attack Model
We specify what an attacker is able and unable to do to an AKE protocol as follows.
- Assumption 1. The attacker is unable to alter, insert, delay or delete messages transmitted via the authenticated OOB channels.
- Assumption 2. The attacker is able to observe, alter, insert, delay or delete messages transmitted via the normal wireless channels.
- Assumption 3. The attacker is able to obtain any previous session key.
- Assumption 4. The attacker is able to compromise the long-term secret keys of A or B.

2.3. Security Goals
Under the above attack model, an AKE protocol aims to achieve the following security goals:
- Key authentication.
- Key confidentiality.
- Known-key security (key freshness).
- Forward secrecy.
- Forward secrecy.

3. Authenticated Digital Evidence Generation Protocol
This section presents Authenticated Digital Evidence Generation (ADEC) Protocol. The protocol transfers one scalar multiplication from the computationally limited initiator A to the more powerful responder B in ECDH. The protocol is described below and illustrated in Figure 1.
3.1. Authenticated Key Exchange Process

Unbalanced authenticated key exchange process is described below.

Step 1. A generates a random number $R_A$ and compute $U_A$. Then A generates a nonce $N_A$ and computes a commitment $W_A$. A sends message $M_1$ including its identity $A$, public key $PK_A$, $U_A$ and commitment $W_A$.

$$U_A = R_A + SK_A$$

$$W_A = HMAC_{128}(N_A \oplus PK_A \oplus U_A)$$

$$A \rightarrow B : M_1 = \{A, PK_A, U_A, W_A\}$$

Step 2. After receiving $M_1$, B generates a random number $R_B$ and computes $U_B$. Then B computes $T_B$. B generates a nonce $N_B$, and sends message $M_2$ including its identity $B$, public key $PK_B$, $N_B$ and $T_B$.

$$U_B = R_B + SK_B$$

$$T_B = U_B \times G = (R_B + SK_B) \times G$$

$$B \rightarrow A : M_2 = \{B, PK_B, N_B, T_B\}$$

Step 3. A sends message $M_3$ that reveals $N_A$. Then A computes $D$, and sends $M_4$ that includes $D$ via a NSB channel.

$$A \rightarrow B : M_3 = \{N_A\}$$

$$A \Rightarrow B : M_4 = \{D\}$$

where
Step 4. B verifies \( W_A \) received in Step 1 and \( D \); if the verification succeeds, it sends a confirmation signal via the NSB channel; otherwise, it aborts the protocol, and sends an error signal via the NSB channel.

\[ B \Rightarrow A : M_5 = \{ \sqrt{\text{or } \perp} \} \]

3.2. Advantages
As mentioned before, this protocol transfers one scalar multiplication from the computationally limited device to the one with better computational capabilities. They significantly reduce the computational requirements on the computationally limited device.

The advantages are summarized in Table 1, where \( S_A \) and \( S_B \) represent the time of scalar multiplications in \( A \) and \( B \), respectively. In ECDH, both \( A \) and \( B \) undertake 2 scalar multiplications. However, in our protocol UAKE, \( B \) computes both \( T_B = U_B \times G \) on behalf of \( A \); thus, \( A \) undertakes 1 scalar multiplication in total; and \( B \) undertakes 3 scalar multiplications. Therefore, the scheme has less computational requirement on \( A \) than on \( B \).

| Protocol | \( A \) | \( B \) |
|----------|--------|--------|
| ECDH     | 2      | 2      |
| UAKE     | 1      | 3      |

3.3. Use Case: Evidence Uploading
It can be assumed that a blockchain-based digital evidence preservation system works as in Figure 2. Sensing devices (A) for example mobile phones collect digital evidences; and uploaded evidences to a blockchain node (B) (directly or via a coordinator). In order to achieve this, they could use the pre-mentioned protocol. They will agree on a temporary shared secret \( SHK \), and transmit the significant data \( DE \) that is encrypted using \( SHK \). Step 1 to step 4 were presented previously. Another two steps are listed below.

Step 5. If \( A \) received \( \sqrt{\text{or } \perp} \), it computes the temporary shared secret \( SHK \). Then \( A \) sends \( M_6 \).

\[
SHK = R_A \times (T_B - PK_B) \\
= R_A \times ((R_A + SK_B) \times G - PK_B) \\
= R_A \times (R_B \times G + SK_B \times G - PK_B) \\
= R_A \times (R_B \times G + PK_B - PK_B) \\
= R_A \times (R_B \times G) \\
= R_A \times R_B \times G \\
A \Rightarrow B : M_6 = \{ A, Enc_{SHK}(DE), \}
\]

\[ HMAC_{256}(SHK, A \oplus DE) \]

Step 6. B computes \( SHK \), verifies \( A, DE \). If verification succeeds, B will record \( DE \) in the blockchain.
4. Security Analysis
This section studies the security of the protocol. In particular, we analyze that the protocol achieves the security goals under the attack model mentioned in the section of system overview.

4.1. Key Authentication
We mainly focus on the question: have A and B agreed on SHK. Before we discuss the question, related definitions are introduced based on paper[17].

Definition. (Secure authentication protocol) Protocol is a secure authentication protocol if for any polynomial time adversary,

- (Matching conversations ⇨ acceptance) If A and B have matching conversations, then all parties accept.
- (Acceptance ⇨ matching conversations) The probability of No – Matching^A is negligible. No – Matching^A is the following events: (1) A is accepted, and there is no B which engaged in a matching conversation; or (2) B is accepted, and there is no A which engaged in a matching conversation.

It is easy to see that matching conversations ⇨ acceptance is fulfilled. We will focus on the probability of No – Matching^A. Assume info = (A, PK_A, U_A, B, PK_B, T_B), we firstly discuss when A is accepted, No – Matching^A is negligible. The analysis is as follows.

- Because messages transmitted via NSB channels cannot be changed or blocked by A. Thus, if there is (t_4, M_5, M_6) in conversation_A, there exists (t_3, (M_4), M_5) in conversation_B.
- If A wants to modify M_5, A must pass the verification in step 4 (B verifies W_5 received in Step 1 and D). Thus, A has the following strategies: (a) A guesses a N'_A without knowing the value of N_A; (b) A finds a N'_A with the knowledge of the value of N_A, which breaks the hiding property of the commitment scheme. In both cases, the probability is negligible. Thus, if there is (t_4, M_5, M_6) in conversation_A, there exists (t_3, (M_3, M_4), M_5) in conversation_B.
If \((t_3, (M_3, M_4), M_5)\) is in conversation_B, there exists \((t_1, M_1, M_2)\) in conversation_B. Let us say that \(A\) can modify or send a fake \(M_1\). (a) \(A\) knows \(N_A\) before \(t_3\), which breaks the hiding property of the commitment scheme. (b) \(A\) does not know \(N_A\) before \(t_3\), but it can successfully find \(PK'_A, U'_A, W'_A\) that can pass the verification in step 4 (B verifies \(W_A\) received in Step 1 and D), which breaks the binding property of the commitment scheme. Thus, there exists \((t_1, M_1, M_2)\) in conversation_B.

Similarly, when B is accepted, \(No – Matching^d\) is negligible. From the analysis, we can conclude that the probability of \(No – Matching^d\) is negligible. A and B agrees on SHK.

4.2. Key Confidentiality

After a completed run of the protocol, \(A\) is unable to derive the shared secret of A and B.

4.2.1 The shared secret can be computed from any of the following equations:

\[
K_A = R_A \times (T_B - PK_B), \\
K_B = R_B \times (U_A \times G - PK_A), \\
SHK = R_A \times R_B \times G.
\]

Therefore, \(R_A\) or \(R_B\) is required to compute the shared secret.

4.2.2 Since \(R_A\) is hidden by the following equation:

\[U_A = R_A + SK_A,\]

\(SK_A\) is required to compute \(R_A\).

Since \(R_B\) is hidden by the following equation:

\[T_B = (R_B + SK_B) \times G\]

\(SK_B\) is required to compute \(R_B\).

4.2.3 According to the attack model, \(A\) has neither \(SK_A\) nor \(SK_B\). \(A\) is unable to compute \(R_A\) or \(R_B\). Therefore, \(A\) is unable to compute \(K_A = K_B = SHK\).

4.3. Known-key Security

After a completed run of the protocol, \(A\) is unable to derive the shared secret from the previous session keys.

In the protocol, the computation of the shared secret takes the \(R_A\) and \(R_B\) as the inputs. Since \(R_A\) and \(R_B\) are random values generated by A and B respectively, in each run of the protocol the values are unique. Therefore, the secret is fresh in each run of the protocol.

5. Conclusion

This paper has presented a method to unbalance the scalar multiplications in the ECDH key exchange scheme. Based on the method, we have proposed a protocol for updating digital evidence to the blockchain. The analysis results show that the computationally unbalanced protocols are more friendly to computationally limited devices than the original one.

Acknowledgment

This paper is supported by science and technology program of State Grid “Research on the cipher application key technologies of client-side terminal based on unified cryptographic infrastructure” (5700-202055171A-0-0-00).
References

[1] R. Canetti and H. Krawczyk. (2001) Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels. Advances in Cryptology-EUROCRYPT 2001: 453-474.

[2] M. Bellare, R. Canetti and H. Krawczyk. (1996) Keying Hash Functions for Message Authentication. Annual International Cryptology Conference: 11-15.

[3] X. Huang, (2014) Multi-Channel Security Protocols in Personal Networks. Doctoral dissertation, University of Oxford.

[4] M. Cagalj, S. Capkun and J-P. (2006) Hubaux, Key Agreement in Peer-to-Peer Wireless Networks. Proceedings of the IEEE 94(2): 467-478.

[5] D. Balfanz, D. K. Smetters, P. Stewart and H. C. Wong. (2002) Talking to Strangers: Authentication in Ad-Hoc Wireless Networks. NDSS.

[6] N. Koblitz. (1987) Elliptic Curve Cryptosystems. Mathematics of Computation, 48(177): 203-209.

[7] Bluetooth SIG Proprietary. (2016) Bluetooth Core Specification Version 5.0.

[8] IEEE Computer Society. (2012) IEEE Standard 802.15.6: Wireless Body Area Networks.

[9] O. Jo, Y. K. Kim and J. Kim. (2018) Internet of Things for Smart Railway: Feasibility and Applications. IEEE Internet of Things Journal, 5(2): 482-490.

[10] A. Ometov, S. V. Bezzateev, J. Kannisto, J. Harju, S. Andreev and Y. Koucheryavy. (2017) Facilitating the Delegation of Use for Private Devices in the Era of the Internet of Wearable Things. IEEE Internet of Things Journal, 4(4): 843-854.

[11] N. Xue, L. Liang, J. Zhang and X. Huang. (2016) An access control system for intelligent buildings. The 9th EAI International Conference on Mobile Multimedia Communications. Xi’an, China. 11–17.

[12] N. Xue, C. Jiang, X. Huang and D. Liu. (2017) A Role-Based Access Control System for Intelligent Buildings. 11th International Conference on Network and System Security. Springer, Helsinki, Finland. 710–72.

[13] J. Zhang, X. Huang, P. Craig, A. Marshall and D. Liu. (2016) An Improved Protocol for the Password Authenticated Association of IEEE 802.15.6 Standard that Alleviates Computational Burden on the Node. Symmetry 8(11): 131.

[14] N. Xue, L. Liang, X. Huang and J. Zhang. (2016) POSTER: A framework for IoT reprogramming.16th EAI International Conference on Security and Privacy in Communication Systems. Springer, Guangzhou, China. 751–754.

[15] J. Zhang, N. Xue and X. Huang. (2016) A Secure System for Pervasive Social Network-Based Healthcare. IEEE Access, 4: 9239-9250.

[16] J. Cai, X. Huang, J. Zhang, J. Zhao, Y. Lei, D. Liu and X. Ma. (2018) A Handshake Protocol with Unbalanced Cost for Wireless Updating. IEEE Access, 6: 18570-18581.

[17] M. Bellare and P. Rogaway. (1993) Entity Authentication and Key Distribution. In Proceedings of the 13th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO ’93). Springer-Verlag, Berlin, Heidelberg. 232–249.

[18] N. Xue, X. Huang and J. Zhang. (2016) S²Net: A security framework for software defined intelligent building networks. IEEE Trustcom/BigDataSE/ISPA. Tianjin, China. 654–661.