Efficient Anonymous Password-Authenticated Key Exchange Scheme Using Smart Cards

Tsu-Yang Wu[1,2(✉)], Weicheng Fang[3], Chien-Ming Chen[3], and Eric Ke Wang[3]

[1] Fujian Provincial Key Laboratory of Big Data Mining and Applications, Fujian University of Technology, Fuzhou 350118, China
wutsuyang@gmail.com

[2] National Demonstration Center for Experimental Electronic Information and Electrical Technology Education, Fujian University of Technology, Fuzhou 350118, China

[3] Harbin Institute of Technology Shenzhen Graduate School, Shenzhen, China
626558837@qq.com, chienming.taiwan@gmail.com, wk_hit@hit.edu.cn

Abstract. Anonymous authentication is designed to hide the user's identity from any verifiers during an authentication session. Since passwords prevail in many authentication systems, anonymous password-authenticated key exchange (APAKE) has become a candidate technique for privacy-enhancing applications. Recently, Shin and Kobara proposed an improved APAKE protocol using general devices such as public directories. However, we find that their scheme is vulnerable to a credential forgery attack. Then, we propose an efficient protocol using tamper resistant smart cards. The security and efficiency analysis shows that our protocol obtains high security and efficiency.

Keywords: Password authentication · Anonymity · Security · Smart card

1 Introduction

Anonymous authentication allows registered users to authenticate themselves without revealing their identities. It is desirable when users concern about their privacy. For example, users may be reluctant to vote or comment if the privacy is compromised. In fact, as the growth and development of the Internet becomes faster, many privacy-enhancing technologies have been invented to protect user's anonymity.

Recently, many researchers focus on the design of anonymous password-authenticated key exchange (APAKE) protocol. It not only achieves authentication and key exchange based on a low-entropy password, but also preserves the client's privacy. In 2006, Chai et al. [1] found the previous APAKE protocol

P. Krömer et al. (eds.), *Proceedings of the Fourth Euro-China Conference on Intelligent Data Analysis and Applications*, Advances in Intelligent Systems and Computing 682, DOI 10.1007/978-3-319-68527-4_9

proposed by Viet et al. [2] not efficient enough, and thus proposed a new protocol using smart cards. Since then, related schemes [3–5] have been proposed to additionally support traceable client anonymity, but they all failed to resist known attacks. In 2009, a protocol using general devices such as public directories was first proposed by Yang et al. [6] and later improved in [7]. In 2012, to obtain better performance, Qian et al. [8] also utilized general devices in the design of a new APAKE protocol. However, in 2016, Yang et al. [9] pointed out that their scheme suffers from the credential sharing problem. Also, Shin and Kobara [10] observed that their protocol is vulnerable to an active attack, and proposed an improved protocol (S2APA). Despite their improvement, we find that the S2APA protocol cannot resist a credential forgery attack.

Our contribution is twofold. Firstly, we find that S2APA is vulnerable to a credential forgery attack due to the use of general devices. Secondly, to overcome such attacks, we propose an efficient APAKE protocol using tamper-resistant smart cards. Although our scheme resorts to the dedicated devices such as smart cards, according to the analysis, it achieves higher efficiency and resists various known attacks. The rest of the paper is organized as follows: Sect. 2 shows the weakness of Shin and Kobara's protocols. Our proposed APAKE scheme using smart cards is presented in Sect. 3. In Sect. 4, we perform the security and efficiency analysis on the proposed protocol. Finally, we conclude the paper in Sect. 5.

2 Security Weakness in the S2APA Protocols

In this section, we briefly review Shin and Kobara's S2APA protocol, and show that their protocol is vulnerable to a credential forgery attack. S2APA protocol aims to provide unconditional anonymity for users during the authentication, where the server does not care the users' real identities as long as they have registered and provided correct passwords. It utilizes general devices that only guarantee integrity protection, such as external hard drives, software smart cards, and public directories. Also, the protocol employs a homomorphic public key encryption scheme (Gen, E, D) instantiated by ElGamal encryption system:

- $Gen(1^k)$ generates the public key and private key for the scheme.
- $E_{pk}(\cdot)$ returns the ciphertext according to the message m and a random coin r. Also, for messages m_1 and m_2, the homomorphic property states that

$$E_{pk}(m_1 \cdot m_2; r') = E_{pk}(m_1; r_1) \odot E_{pk}(m_2; r_2), \tag{1}$$

  where $\cdot$ and $\odot$ represents the group operations in plaintext and ciphertext, and r', r_1, r_2 are some random coins.
- $D_{sk}(\cdot)$ returns the plaintext according to the input ciphertext.

2.1 Review of the S2APA Protocol

The S2APA protocol consists of three phases: the setup phase, the registration phase, and the authentication phase. In the setup phase, the server S initializes the system with some public parameters $\{G, p, g, h, H_1, H_2, H_3, Gen, E, D\}$,

where G is a cyclic group generated by g of prime order p, h is another generator of G, H_1, H_2 and H_3 are one-way hash functions, and (Gen, E, D) is a public key encryption scheme described as above. S also invokes Gen to generate keys (pk, sk), and randomly selects a master secret element $N \in G$. Figure 1 shows the registration phase and the authentication phase in details.

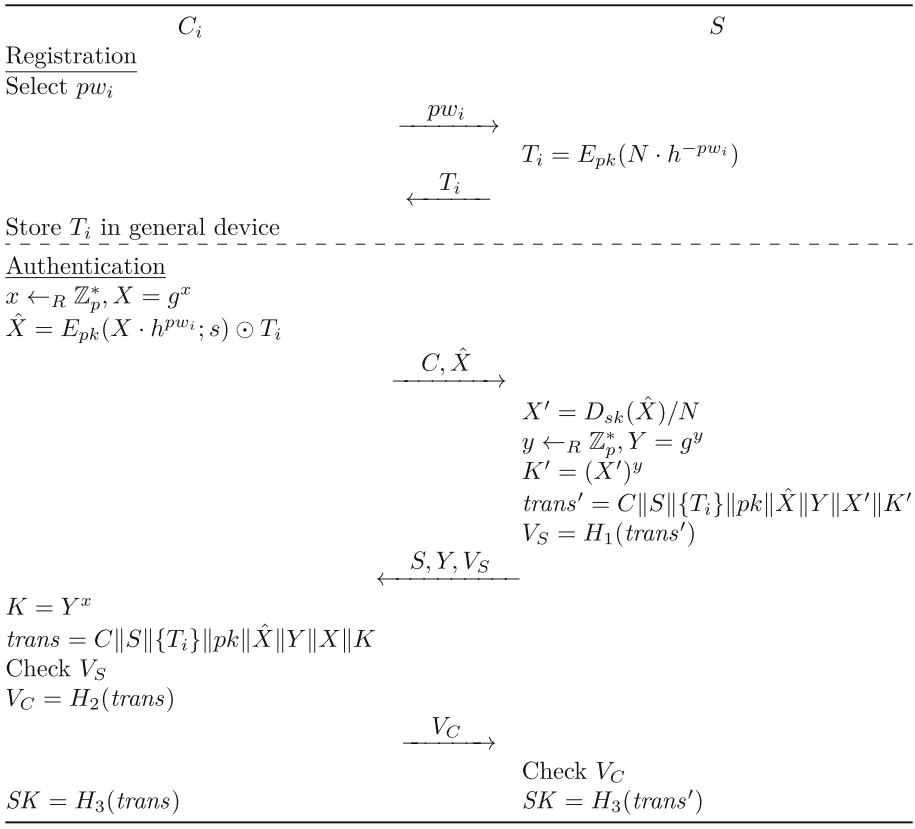

Fig. 1. The registration phase and the authentication phase of the S2APA protocol

2.2 A Credential Forgery Attack

Although the S2APA protocol was proved to be secure against various known attacks, we find it vulnerable to a credential forgery attack. Note that a client C_i's credential T_i is issued by the Server S, and then stored in a general device. In our attack, we assume that an adversary $\mathcal{A}$ has registered with S, and thus obtained a credential $T_{\mathcal{A}} = E_{pk}(N \cdot h^{-pw_{\mathcal{A}}})$ after the registration phase, where $pw_{\mathcal{A}}$ is the password chosen by the adversary. Since the credential is stored in a general device, $\mathcal{A}$ can extract T_i easily. Then, to forge a credential, $\mathcal{A}$ computes

$$T_{\mathcal{A}}^* = E_{pk}(h^{-pw_{\mathcal{A}}}) \odot T_{\mathcal{A}} \equiv E_{pk}(N). \tag{2}$$

In fact, according to the homomorphic property shown in Eq. (1), the newly forged credential becomes S's master secret N encrypted under the public key pk. To use it in the authentication phase, $\mathcal{A}$ follows the steps as shown in Fig. 1 except that the value $\hat{X}$ is computed as follows:

$$\hat{X} = E_{pk}(X; s) \odot T^*_{\mathcal{A}}. \tag{3}$$

The modification on $\hat{X}$ with the forged credential will have no influence on S's computation of X'. Therefore, S will accept the authentication request. Note that the authentication phase no longer involves the input of passwords when using the forged credential. So, $\mathcal{A}$ can secretly shares or publishes $T^*_{\mathcal{A}}$ to those who have not registered without leaking the adversary's own password $pw_{\mathcal{A}}$. In a long run, such a credential forgery attack launched by many other registered adversaries will undermine the system's registration.

The credential forgery attack against the S2APA protocol can be applied to similar protocols that merely depend on a general device to store its secrets. The SAPAKE protocol proposed by Qian et al. [8], the SAP protocol proposed by Shin and Kobara [11], and the protocol proposed by Son et al. [12] also suffer from such an attack if they insist on using general devices. As a simple but effective countermeasure, one replaces the general devices by tamper resistant dedicated devices such as smart cards, from which the securely stored secrets cannot be extracted by an adversary any more. The replacement trades usability for security. However, we observe that once using tamper proof smart cards, their schemes can be further improved in terms of efficiency.

3 The Proposed APAKE Protocol

In this section, we propose a new APAKE using smart cards. We assume our scheme uses such a tamper resistant smart card that no information can be extracted by an adversary. The scheme includes three phases as well. In the setup phase, the Server S chooses a random secret n and a secure one-way hash function h, computes $N = h(n)$ sets $N, G, g, p, h, H_1, H_2, H_3, H_4\}$ as public parameters, G is a cyclic group generated by g of order p, and H_1, H_2, H_3 and H_4 are secure one-way hash functions.

Figure 2 shows the rest of the scheme. We describe the registration phase and the authentication phase in details.

In the registration phase, a client C_i select a password pw_i and a random nonce r, computes hashed password hpw_i and sends it to the server S. The server computes the client's credential T_i and issues a smart card. After that, the smart card computes an off-line password verifier R, and appends r and R to itself. Then, C_i hold a tamper resistant smart card $\{T_i, r, R\}$.

In the authentication phase, C_i inputs the password to complete an off-line verification. If it is not verified, the smart card rejects the login request immediately. Otherwise, it sends an authenticator A and a group element X to the server S. S checks the validity of A, selects a group element Y, forms the Diffie-Hellman key K, and computes another authenticator V_S. Then, it sends

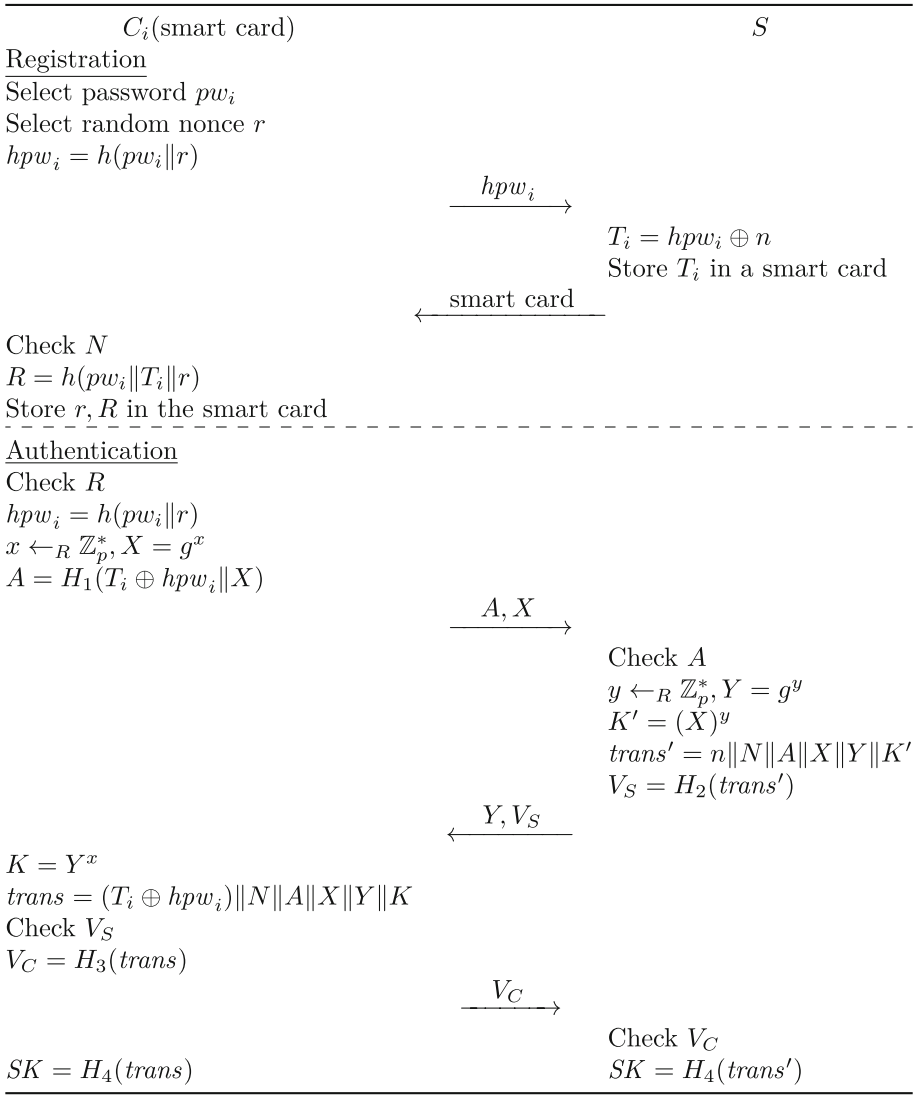

Fig. 2. The proposed APAKE protocol

$\{Y, V_S\}$ back to C_i. After that, C_i checks V_S and sends a respond V_C to S to achieve mutual authentication. Finally, S checks V_C and accepts the session. Both will compute the correct session key SK given successful verifications.

4 Security and Efficiency Analysis

For the past decade, various authenticated key exchange protocols have been proposed buy many of them have been proven insecure [13–17]. Therefore, in this

section, we first demonstrate that our protocol satisfies security requirements and withstands known attack. Then, we evaluate computation and communication cost of our scheme. Trough analysis and comparison, though less user-friendly, our scheme proves to be more secure and efficient.

Mutual Authentication. In the authentication phase, both client C_i and the server S verifies each other to complete mutual authentication. If an adversary $\mathcal{A}$ attempts to impersonate either side, $\mathcal{A}$ has to compute the correct authenticators that consists of the master secret x and a Diffie-Hellman key K. However, it is impossible for $\mathcal{A}$ to obtain x. Thus, the adversary cannot generate a correct authenticator to achieve impersonation.

Unconditional Anonymity. The proposed scheme provides unconditional anonymity for the clients. During each protocol run, a client C_i sends a randomized authenticator A to the server S. So, S cannot track the user by checking A since it differs among protocol runs. Note that a malicious server trying to break the user's anonymity can modifies the credential in the registration phase. Specifically, it picks a unique random nonce n_i for each client C_i:

$$T_i' = hpw_i \oplus n_i \tag{4}$$

Then, in the authentication phase, S may check the authenticator A against different master secrets until it finds a match, i.e., $A =?H_1(n_i \| X)$. However, the proposed scheme can detect malicious servers. This is because S have publicized $N = h(n)$ in the setup phase, and the smart card will check the master secret in the registration phase. Therefore, our scheme guarantees user anonymity.

Forward Secrecy. The security requirement of forward secrecy guarantees that even if the secrets are compromised, previous session keys will not break. In our scheme, the session key consists of a Diffie-Hellman key $K = X^y = Y^x$. Owing to the hardness of computational Diffie-Hellman problem, as long as the ephemeral exponents are not compromised, the session keys will stay secure.

Resistance to Man-in-the-Middle Attack. Suppose an adversary $\mathcal{A}$ tries to stand in the middle of a client C_i and the server S, $\mathcal{A}$ has to modifies some transmitted messages. If the value X in A, X sent by C_i is modified, $\mathcal{A}$ must compute a correct authenticator A. However, this is impossible since the adversary cannot compute the master secret. Therefore, our scheme can resist man-in-the-middle attacks.

Resistance to Replay Attack. In this attack, an adversary $\mathcal{A}$ tries to replay some transmitted messages in different protocol runs to achieve malicious ends. Since all messages does not contain an identity, $\mathcal{A}$ can replay a message from other sessions. However, our scheme employs a challenge-response design pattern so that responses not corresponding to the challenge will be rejected. Therefore, our scheme can resist replay attacks.

Efficiency. Our protocol achieves high efficiency in terms of both computation and communication cost. The computation overhead in the authentication phase includes a single exclusive-or (XOR), several hash and two exponentiation operations. Compared with lightweight operations such as XOR and hash, although exponentiations are expensive, in order to achieve forward secrecy, at least two exponentiations have to be performed. Regarding to communication overhead, the authentication phase is a three-round protocol. The largest size of the messages transmitted in a single round consists of only a group element and a hash digest.

We compare our scheme with other APAKE protocols in Table 1. The computation cost is evaluated by the operations performed by a clients or the server, including paring (P), exponentiation (E), and hash (H) operations. In our analysis, the symmetric encryption/decryption is viewed as a hash operation for simplicity. The communication cost comes from the transmitted messages, including the size of any group elements ($|G|$), nonces, and authenticators. We treat all those other than the group elements as hash digests ($|H|$). In some schemes, the costs involves the number of registered clients, which is denoted by n.

Table 1. Efficiency comparison of APAKE protocols

Protocols	Computation overhead		Comm.	Usability	Revoc.
	Client	Server			
NAPAKE [3]	$4E + 2H$	$(n+3)E + 2H$	$(n+3)\|G\| + \|H\|$	(3)	✓
VEAP [4]	$2E + 5H$	$(n+2)E + nH$	$3\|G\| + (n+2)\|H\|$	(3)	✓
Yang et al. [7]	$2P + 16E + 2H$	$4P + 13E + 3H$	$8\|G\| + 6\|H\|$	(2)	✓
Yang et al. [6]	$7E + 3H$	$5E + 3H$	$3\|G\| + 5\|H\|$	(2)	✓
S2APA [2]	$5E + 3H$	$3E + 3H$	$3\|G\| + 2\|H\|$	(2)*	×
SAPAKE [5]	$6E + 3H$	$3E + 3H$	$3\|G\| + 2\|H\|$	(2)*	×
Chai et al. [1]	$2E + 5H$	$2E + (2n+2)H$	$(n+1)\|G\| + 3\|H\|$	(1)	✓
Ours	$2E + 3H$	$2E + 3H$	$2\|G\| + 3\|H\|$	(1)	×

These protocols varies in computation and communication overhead. On the one hand, they apply different level of usability. As shown in the table, there are mainly three levels related to device usage in APAKE protocols: (1) dedicated device-assisted, e.g. tamper resistant smart cards, (2) general device-assisted, e.g., public directories, and (3) password-only. The higher the level is, the more user-friendly the protocol will be in practice. Note that both the S2APA [10] protocol and the SAPAKE [8] protocol suffer from the credential forgery attack due to the use of general devices. One possible solution is to downgrade its usability to 1. Thus, among those schemes of least usability, our proposed scheme is the most efficient. On the other hand, some schemes emphasize unconditional anonymity, and thus do not take into account the client revocation, while others can be extended to support revocation efficiently. Although our scheme cannot be easily adapted to revoke user's smart card, it achieves high efficiency.

5 Conclusions

In this paper, we show that Shin and Kobara's S2APA protocol is insecure against a credential forgery attack. The attack lies in the use of general devices. A simple but effective solution is to replace with dedicated devices such as tamper proof smart cards. To improve its efficiency, we propose another APAKE protocol using smart cards. Through analysis, we show that although our protocol requires a dedicated device, it is secure against various attacks and efficient at protecting user's anonymity.

Acknowledgement. The work of Chien-Ming Chen was supported in part by the Project NSFC (National Natural Science Foundation of China) under Grant number 61402135 and in part by Shenzhen Technical Project under Grant number JCYJ20170307151750788. The work of Eric Ke Wang was supported in part by National Natural Science Foundation of China (No. 61572157), grant No. 2016A030313660 from Guangdong Province Natural Science Foundation, JCYJ20160608161351559 from Shenzhen Municipal Science and Technology Innovation Project.

References

1. Chai, Z., Cao, Z., Lu, R.: Efficient password-based authentication and key exchange scheme preserving user privacy. In: International Conference on Wireless Algorithms, Systems, and Applications, pp. 467–477. Springer (2006)
2. Viet, D.Q., Yamamura, A., Tanaka, H.: Anonymous password-based authenticated key exchange. In: International Conference on Cryptology in India, pp. 244–257. Springer (2005)
3. Kim, S., Rhee, H.S., Chun, J.Y., Lee, D.H.: Anonymous and traceable authentication scheme using smart cards. In: International Conference on Information Security and Assurance, ISA 2008, pp. 162–165. IEEE (2008)
4. Liu, Y., Zhao, Z., Li, H., Luo, Q., Yang, Y.: An efficient remote user authentication scheme with strong anonymity. In: 2008 International Conference on Cyberworlds, pp. 180–185. IEEE (2008)
5. Shao, S., Li, H., Niu, X., Yang, Y.: A remote user authentication scheme preserving user anonymity and traceability. In: 5th International Conference on Wireless Communications, Networking and Mobile Computing, WiCom 2009, pp. 1–4. IEEE (2009)
6. Yang, Y., Zhou, J., Weng, J., Bao, F.: A new approach for anonymous password authentication. In: Annual Computer Security Applications Conference, ACSAC 2009, pp. 199–208. IEEE (2009)
7. Yang, Y., Zhou, J., Wong, J.W., Bao, F.: Towards practical anonymous password authentication. In: Proceedings of the 26th Annual Computer Security Applications Conference, pp. 59–68. ACM (2010)
8. Qian, H., Gong, J., Zhou, Y.: Anonymous password-based key exchange with low resources consumption and better user-friendliness. Secur. Commun. Netw. **5**(12), 1379–1393 (2012)

9. Yang, Y., Lu, H., Liu, J.K., Weng, J., Zhang, Y., Zhou, J.: Credential wrapping: from anonymous password authentication to anonymous biometric authentication. In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, pp. 141–151. ACM (2016)
10. Shin, S., Kobara, K.: Simple anonymous password-based authenticated key exchange (SAPAKE), reconsidered. IEICE Trans. Fundam. Electron. Commun. Compu. Sci. **100**, 639–652 (2017)
11. Shin, S., Kobara, K.: A secure anonymous password-based authentication protocol with control of authentication numbers. In: 2016 International Symposium on Information Theory and its Applications (ISITA), pp. 325–329. IEEE (2016)
12. Son, K., Han, D.G., Won, D.: Simple and provably secure anonymous authenticated key exchange with a binding property. IEICE Trans. Commun. **98**(1), 160–170 (2015)
13. Chen, C.M., Li, C.T., Liu, S., Wu, T.Y., Pan, J.S.: A provable secure private data delegation scheme for mountaineering events in emergency system. IEEE Access **5**, 3410–3422 (2017)
14. Chen, C.M., Fang, W., Wang, K.H., Wu, T.Y.: Comments on an improved secure and efficient password and chaos-based two-party key agreement protocol. Nonlinear Dyn. **87**(3), 2073–2075 (2017)
15. Chen, C.M., Xu, L., Wu, T.Y., Li, C.R.: On the security of a chaotic maps-based three-party authenticated key agreement protocol. J. Netw. Intell. **2**, 61–65 (2016)
16. Sun, H.M., He, B.Z., Chen, C.M., Wu, T.Y., Lin, C.H., Wang, H.: A provable authenticated group key agreement protocol for mobile environment. Inf. Sci. **321**, 224–237 (2015)
17. Chen, C.M., Wang, K.H., Wu, T.Y., Pan, J.S., Sun, H.M.: A scalable transitive human-verifiable authentication protocol for mobile devices. IEEE Trans. Inf. Forensics Secur. **8**(8), 1318–1330 (2013)
18. Yang, J., Zhang, Z.: A new anonymous password-based authenticated key exchange protocol. In: International Conference on Cryptology in India, pp. 200–212. Springer (2008)
19. Shin, S.H., Kobara, K., Imai, H.: Very-efficient anonymous password-authenticated key exchange and its extensions. In: International Symposium on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes, pp. 149–158. Springer (2009)
20. Zhang, Z., Yang, K., Hu, X., Wang, Y.: Practical anonymous password authentication and TLS with anonymous client authentication. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1179–1191. ACM (2016)