Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **7**, Number **3**, August **2022**

AuthPFS: A Method to Verify Perfect Forward Secrecy in Authentication Protocols

Mingchang Ge

College of Computer Science and Engineering
Shandong University of Science and Technology
266590, Qingdao, Shandong, China
1849371528@qq.com

Saru Kumari

Department of Mathematics
Ch. Charan Singh University
250004, Meerut, Uttar Pradesh, India
saryusiirohi@gmail.com

Chien-Ming Chen*

College of Computer Science and Engineering
Shandong University of Science and Technology
266590, Qingdao, Shandong, China
chienmingchen@ieee.org

*Corresponding author: Chien-Ming Chen

Received April 28, 2022, revised June 3, 2022, accepted July 28, 2022.

Abstract. *Perfect forward secrecy (PFS) is a property of authentication protocols by which the exposure of long-term key material that is used in the protocol to authenticate and negotiate the session key does not compromise the secrecy of the session key established before. In this paper, we first review two recently published protocols and examine why they do not provide perfect forward secrecy. We try to conclude some general principles to design a protocol that fulfills perfect forward secrecy. Then we propose a method named AuthPFS. It can verify if an authentication protocol can provide PFS. We also utilize AuthPFS to show that another authentication protocol does provide PFS.*
Keywords: Authentication protocol, Perfect forward secrecy, network security

1. **Introduction.** Thanks to the rapid development of computers and communication, networks have become essential in our daily lives. People use the network to exchange and obtain information. Due to the continuous promotion and expansion of the network, there is more and more security issues happened. For example, the increasingly popular e-commerce involves online payment [1–3], blockchain [4–7] portal websites, and e-mail, which will convey sensitive information such as users' privacy [8–12], which is likely to be maliciously hijacked and tampered with by attackers. Once a security leak occurs, it will cause tremendous or even unlimited losses to users. Therefore, information security [13–17] is of great significance.

An authentication Protocol (or Authentication and Key Agreement Protocol) is the core method to ensure information security. An authentication protocol provides two functionalities. First, it provides mutual authentication. That is, all participants in the

network can be authenticated. Second, it generates a shared session key for later use. All messages transmitted through a public channel can be encrypted with this session key. In recent years, a series of authentication protocols have been proposed for different kinds of environments and applications, such as IoT [18–23], WSN [24–28], VANET [29–34], digital right management [8, 35], smart grid [36–38], healthcare [39–41], 5G [42, 43], cloud/fog computing [44–50], and drones [51, 52], etc.

In 2013, Xue et al. [53] proposed a temporal-credential-based authentication protocol for WSN (wireless sensor networks). In 2016, Chang et al. [54] proposed another flexible authentication protocol for WSN. In the same year, Farash [55] proposed an improved authentication protocol for session initiation protocol. In 2017, in order to secure LoRaWAN, Kim et al. [56] described a dual key-based activation protocol. Later, Wang et al. [57] proposed an enhanced user authentication protocol for WSN. Also in 2017, Wazid et al. [58] proposed another authentication protocol for smart home environments. In 2018, Mahmood et al. [46] proposed a lightweight authentication protocol for smart gird communication. This protocol is based on elliptic curve cryptography. Also in 2018, Wu et al. proposed an authentication protocol for distributed cloud computing. In 2019, Aghili et al. [59] described a three-factor authentication and access control protocol for E-Health systems in IoT (Internet of Things). In 2020, Altaf et al. [60] gave a novel authentication protocol for satellite communication network. Yang et al. [61] proposed a faster authentication protocol for Industrial IoT. Also in 2020, Ali et al. [62] described an improved symmetric key based authentication protocol for multi-server environments. Shashidhara et al. [63] proposed a robust authentication protocol for framing service in mobility environments. In 2021, Sadri et al. [64] proposed an anonymous two-factor authentication protocol for IoT. Wu et al. [65] described a three-factor authentication protocol for WSN with IoT nation. Also, Rara et al. [66] proposed a lightweight authentication protocol for WBAN (wireless body area networks).

In the authentication protocols mentioned above, the authors have tried their best to show that their protocol is secure. They use several methods such as BAN logic, formal proof, Proverif, etc. However, most of these authentication protocols have been demonstrated do not provide perfect forward secrecy (PFS). PFS means that the leakage of a long-used master key does not lead to the leakage of a past session key. In order to show that an authentication protocol does not provide PFS, an attacker E would have the following abilities. 1) E can obtain a server's long-term key (master key). 2) E has limited/completed control over the messages transmitted over a public/insecure channel, such as intercepting, modifying, and deleting the transmitted message. 3) E can extract the security parameters stored in the smart card. The assumptions of PFS seems a little bit too strong, but various of authentication protocols can provide the PFS [67–74]. It means that PFS is vital for an authentication protocol.

In fact, there is still not an excellent way to verify if an authentication protocol provides PFS. In this paper, we propose a method named AuthPFS. It can verify if an authentication protocol can provide PFS. We construct a directed graph by constructing a dependency graph amount of the session key and other variables/long-term secrets. We further utilize AuthPFS to verify three authentication protocols [71, 75, 76].

The rest of this paper is organized as follows. In Section 2 and 3, we briefly review Radhakrishnan et al.'s protocol and Karuppiah et al.'s protocol. We also explain why these two protocols do not provide PFS. Section 4 proposes a method to verify if an authentication protocol satisfies PFS. In Section 5, we introduce another protocol that does provide PFS.

2. **Review of Radhakrishnan et al.'s protocol.** This section briefly reviews and analyzes Radhakrishnan et al.'s protocol. This protocol contains five phases: initialization phase, registration phase, login and authentication phase, password change phase, and revocation and re-registration phase. Because the security weakness lies in the first three phases, we only describe these three phases. Symbols and notations used in this protocol are presented in Table 1.

Table 1. Notations in Radhakrishnan et al.'s protocol

Notation	Description
U_i	i^{th} mobile user
PW_i	Password of U_i
ID_i	Identity of U_i
S	Server
E	Adversary
SC	Smart card
d	Secret key of S
p, q	Large prime integers
r, b	random number
T_u, T_s	Present time stamp of U_i and S
SK	Session Key
ΔT	Permissible transmission delay
$h(\cdot)$	Cryptographic one-way hash function
$\|$	Concatenation
$\otimes$	Bitwise NOR operation
$\oplus$	Bitwise XOR operation

2.1. **Steps of this protocol.**

2.1.1. *Initialization Phase.*

i First, Server S selects two large prime integers p and q, then S computes $n = p \times q$, $\Phi(n) = (p-1) \times (q-1)$.
ii Next, S selects e satisfying that $gcd(e, \Phi(n)) = 1$ and $1 < e < \Phi(n)$, e is an integer. Besides, S computes an integer d satisfying that $d \equiv e^{-1} \ mod \ \Phi(n)$.
iii Finally, S publishes e and n, while keeps the p and q secret.

2.1.2. *Registration phase.*

When a user U_i wants to login the server S legally, he needs to register in server S by the following steps:

i User U_i first selects his identity ID_i, password PW_i and a random number r to compute $h(PW_i||r)$. Then U_i sends $\{h(PW_i||r), ID_i\}$ to server S via secure channel.
ii After receiving the message from the U_i, S generates a random number b and recovers x by $x = b^e \ mod \ n$. Then S computes $A_u = h(d||b) \oplus h(ID_i)$, $B_u = h(d||ID_i) \oplus h(PW_i||r)$ and stores $\{A_u, B_u, x, e, n, h(\cdot)\}$ into smart card SC. After that, S sends it back to U_i.
iii U_i computes $C_u = h(ID_i||PW_i) \oplus r$, $C_t = h(ID_i \otimes PW_i) \otimes r$ and then injects $\{C_u, C_t\}$ into smart card SC.

2.1.3. *Login and authentication phase.* When U_i wants to communicate with server S, he needs to login by the following steps:

i U_i first inserts his smart card, enters his ID_i, PW_i, next the device will compute $r = C_u \oplus h(ID_i||PW_i)$ and verify that if $h(ID_i \otimes PW_i \otimes r) = C_t$. If this holds, device selects N_u and computes $V_1 = A_u \oplus h(ID_i)$, $V_2 = V_1 \oplus N_u$, $V_3 = h(V_1||N_u) \oplus ID_i$, $V_4 = B_u \oplus h(PW_i||r)$, $V_5 = h(V_2||V_3||V_4||T_u)$, and $z = x \oplus T_u$. After that, U_i sends $M_1 = \{z, V_2, V_3, V_5, T_u\}$ to S.

ii After receiving the message from U_i, S first checks the validity of T_u by comparing the difference of $T_s - T_u \leq \triangle T$. If so, S computes $x = z \oplus T_u$, $b = x^d \ mod \ n$, $N_u^* = V_2 \oplus h(d||b)$, $ID_i^* = V_3 \oplus h(h(d||b)||N_u^*)$, $V_4^* = h(d||ID_i^*)$, and $V_5^* = h(V_2||V_3||V_4^*||T_u)$. S then verifies that if $V_5^* = V_5$. If the equation holds, it means that S authenticates U_i is legal. S generates a random number r_s and further computes $SK_s = h(V_4^*||N_u^*||r_s)$, $V_6 = r_s \oplus h(N_u^* \oplus V_4^*)$, $W = h(SK_s||N_u^*||r_s||T_s)$ then sends $M_2 = \{V_6, W, T_s\}$ back to U_i.

iii U_i checks the validity of the T_s and then computes $r_s^* = V_6 \oplus h(N_u \oplus V_4)$, $SK_u = h(V_4||N_u||r_s^*)$, $W^* = h(SK_u||N_u||r_s^*||T_s)$. Then he verifies that if $W^* = W$. If this trues, it means that U_i authenticates S, U_i and S can securely communicate with each other by using shared session key $SK_u = SK_s = h(V_4||N_u||r_s)$. Fig. 1 shows the details of the Radhakrishnan et al.'s protocol.

2.2. **Cryptanalysis of Radhakrishnan et al.'s protocol.** According to the definition of PFS, when attacker E obtains the transmitted messages, the private key d of S and the security information in the smart card, it's easy for E to compute the session key by the following steps.

i E can compute $b = x^d \ mod \ n$, because x is the information from the smart card and n is public parameter that published by server.

ii Because E knows the parameters $\{V_2, d, b\}$, so E can obtain N_u^* where $N_u^* = V_2 \oplus h(d||b)$.

iii With the parameters $\{V_3, d, b, N_u^*\}$, E can compute ID_i^* where $ID_i^* = V_3 \oplus h(h(d||b)||N_u^*)$.

iv Now E can obtain $V_4^* = h(d||ID_i^*)$, $r_s^* = V_6 \oplus h(N_u^* \oplus V_4)$.

v Finally, E can obtain session key by calculating $SK_u = SK_s = h(V_4||N_u||r_s)$.

Obviously, Radhakrishnan et al.'s protocol does not provide perfect forward secrecy.

3. **Review of Karuppiah et al.'s protocol.** In this section, we review karuppiah et al.'s protocol and then point out that their protocol does not provide PFS. Their protocol includes five phases: initialization phase, registration phase, login phase, authentication phase, and password change phase. Similarly, we focus on the first four phases because the attack occurs in these phases. Table 2 presents notations used in this protocol.

3.1. **The Steps.**

3.1.1. *Initialization phase.* The cloud server S selects the generator g lying in finite filed Z_p^*, a long-term private key x_s. Then S computes public key $y = g^{x_s} (mod \, p)$ and publishes the parameters$\{g, y, p\}$.

3.1.2. *Registration phase.*

i U_i freely selects his identity ID_i, password PWD_i and random number k and computes $rpwd_i = h(PWD_i||k)$. U_i then sends $\{rpwd_i, ID_i\}$ to S via secure channel.

ii S computes $A_i = h(ID_i \oplus x_s)$ and $B_i = A_i \oplus h(ID_i||rpwd_i)$. Then S stores $\{B_i, g, y, p, h(\cdot)\}$ into a smart card SC and sends the smart card back to U_i.

User U_i	*Server* S
Inputs ID_i,PW_i Computes $r = C_u \oplus h(ID_i \parallel PW_i)$ Verifies $h(ID_i \otimes PW_i \otimes r) \stackrel{?}{=} C_t$ If true, selects N_u and computes $V_1 = A_u \oplus h(ID_i)$, $V_2 = V_1 \oplus N_u$, $V_3 = h(V_1 \parallel N_u) \oplus ID_i$, $V_4 = B_u \oplus h(PW_i \parallel r)$, $V_5 = h(V_2 \parallel V_3 \parallel V_4 \parallel T_u)$, $z = x \oplus T_u$. $\xrightarrow{M1=\{z,V_2,V_3,V_5,T_u\}}$	
	Checks validity of T_u using $T_s - T_u \leq \Delta T$ If so, obtains $x = z \oplus T_u$ and $b = x^d \ mod \ n$ $N_u^* = V_2 \oplus h(d \parallel b)$, $ID_i^* = V_3 \oplus h(h(d \parallel b) \parallel N_u^*)$, $V_4^* = h(d \parallel ID_i^*)$, $V_5^* = h(V_2 \parallel V_3 \parallel V_4^* \parallel T_u)$. Verifies $V_5^* \stackrel{?}{=} V_5$ If true, S authenticates U_i. Generates r_s. Computes $SK_s = h(V_4^* \parallel N_u^* \parallel r_s)$, $V_6 = r_s \oplus h(N_u^* \oplus V_4^*)$, $W = h(SK_s \parallel N_u^* \parallel r_s \parallel T_s)$, $\xleftarrow{M2=\{V_6,W,T_s\}}$
Checks validity T_s using $T_u' - T_s \leq \Delta T$. Computes $r_s^* = V_6 \oplus h(N_u \oplus V_4)$, $SK_u = h(V_4 \parallel N_u \parallel r_s^*)$, $W^* = h(SK_u \parallel N_u \parallel r_s^* \parallel T_s)$. Verifies $W^* \stackrel{?}{=} W$. If true, U_i authenticates S Shared session key is $SK_u = h(V_4 \parallel N_u \parallel r_s) = SK_s$.	

FIGURE 1. Radhakrishnan et al.'s protocol

iii U_i computes $N_i = k \oplus h(ID_i \oplus PWD_i)$ and $N_t = k \otimes ID_i \otimes PWD_i$ and then stores $\{N_i, N_t\}$ into the smart card SC. Now SC stores $\{B_i, g, y, p, h(\cdot), N_i, N_t\}$.

3.1.3. *Login phase.* If U_i desires to login the system, U_i needs to perform the steps as follows:

i U_i first inserts his smart card SC, enters his ID_i and PWD_i. SC retrieves $k = N_i \oplus h(ID_i \oplus PWD_i)$ to compute $N_t^* = k \otimes ID_i \otimes PWD_i$.
Then SC verifies that if $N_t^* = N_t$, if this not holds, SC aborts the session; Otherwise, SC computes $A_i = B_i \oplus h(ID_i||h(PWD_i||k))$, $W_i = h(T_u \oplus A_i) \oplus (r_i \oplus k)$, $C_i = g^{r_i \oplus k} \ (mod \ p)$, $DID = ID_i \oplus h(y^{(r_i \oplus k)} \ (mod \ p))$, and $V_i = h(ID_i||A_i||W_i||(r_i \oplus k)||T_u)$.
ii SC sends login request message $\{C_i, W_i, V_i, DID, T_u\}$ to S.

TABLE 2. Notations in Karuppiah et al.'s protocol

Notations	Descriptions
U_i	User i
PWD_i, ID_i	Password and identity of U_i
k	Random number of U_i
S	Cloud server
E	Adversary
SC	Smart card
x_s	Secret number and key of S
y	Public key of S
SK_u, SK_s	Session key
p	Large prime number
r_i	a random nonce selected by SC
Z_p^*	Finite field of prime order p
g	Generator of Z_p^*
T_u, T_s	Present timestamp of U_i and S
ΔT	Permissible transmission delay
$h(\cdot)$	Cryptographic one-way hash function
$\oplus$	Bitwise XOR operation
$\otimes$	Bitwise NOR operaion
$\|\|$	Concatenation

3.1.4. *Authentication phase.*

i Upon receiving the login request, S first verifies the received T_u. If T_u is valid, S calculates $ID_i = DID \oplus h(C_i^{x_s}(modp))$, $(r_i \oplus k) = W_i \oplus h(T_u \oplus A_i)$, and $V_i^* = h(ID_i||A_i||W_i||(r_i \oplus k)||T_u)$. Now S verifies that whether $V_i^* = V_i$, S authenticates U_i only if the conditions hold, else the process will be aborted.

ii S computes $G_i = h(ID_i||(A_i \oplus (r_i \oplus k)))$ and $M_i = h(G_i||T_s)$ and then sends it back to U_i.

iii After receiving $\{M_i, T_s\}$ from S, U_i verifies the freshness of timestamp T_s, if this holds, U_i computes $G_i^* = h(Id_i||(A_i \oplus (r_i \oplus k)))$ and $M_i^* = h(G_i^*||T_s)$ and then compares that whether $M_i^* = M_i$. If this holds, U_i authenticates S.

iv If the aforementioned authentications is performed, U_i and S calculate session key $SK_u = SK_s = h(Id_i||A_i||(r_i \oplus k)||(T_u \oplus T_s))$ separately for communication. Figure 2 shows Karuppiah et al.'s authentication protocol.

3.2. **Cryptanalysis of Karuppiah et al's protocol.** Here we describe the Karuppiah et al's protocol does provide PFS. Similarly, we assume that the attacker is E, and E can obtain the secret key x_s of S , the secret information of the smart card SC and the transmitted message from public channel. E can obtain the session key by following steps:

i E can calculates $ID_i = DID \oplus h(C_i^{x_s}(mod\ p))$, because the parameters $\{DID, C_i\}$ are from public channel and p is from SC.

ii E can obtain $A_i = h(ID_i \oplus x_s)$.

iii E computes $r_i \oplus k = W_i \oplus h(T_u \oplus A_i)$ by T_u from transmitted message.

iv After computing the parameters, E can obtain the session key $SK_u = SK_s = h(ID_i||A_i||(r_i \oplus k)||(T_u \oplus T_s))$, which means that the Karuppiah et al's protocol does not provide PFS.

4. **Method to detect a protocol is fulfilling PFS.** If an authentication protocol can provide the perfect forward secrecy, we can imply the following two conditions.

User U_i	*Cloud server* S
Inputs ID_i,PWD_i.	
$k = N_i \oplus h(ID_i \oplus PWD_i)$	
$N_t^* = k \otimes ID_i \otimes PWD_i$	
Verfies $N_t^* \stackrel{?}{=} N_t$	
$A_i = B_i \oplus h(ID_i \parallel h(PWD_i \parallel k))$	
$W_i = h(T_u \oplus A_i) \oplus (r_i \oplus k)$	
$C_i = g^{(r_i \oplus k)})(mod\ p)$	
$DID = ID_i \oplus h(y^{(r_i \oplus k)}(mod\ p))$	
$V_i = h(ID_i \parallel A_i \parallel W_i \parallel (r_i \oplus k) \parallel T_u)$	
$\xrightarrow{\{C_i, W_i, V_i, DID, T_u\}}$	
	$(T_s - T_u) \leq \Delta T$
	$ID_i = DID \oplus h(C_i^{x_s}\ (mod\ p$
	$(r_i \oplus k) = W_i \oplus h(T_u \oplus A_i)$
	$V_i^* = h(ID_i \parallel A_i \parallel W_i \parallel (r_i \oplus$
	Verifies $V_i^* \stackrel{?}{=} V_i$
	If true, S authenticates U_i
	$G_i = h(ID_i \parallel (A_i \oplus (r_i \oplus k)$
	$M_i = h(G_i \parallel T_s)$
	$\xleftarrow{\{M_i, T_s\}}$
$T_u' - T_s \leq \Delta T$	
$G_i^* = h(ID_i \parallel (A_i \oplus (r_i \oplus k)))$	
$M_i^* = h(G_i^* \parallel T_s)$	
Verifies $M_i^* \stackrel{?}{=} M_i$.	
If true, U_i authenticates S	
Shared session key is $SK_u = h(ID_i \parallel A_i \parallel (r_i \oplus k) \parallel (T_u \oplus T_s)) = SK_s$	

FIGURE 2. Karuppiah et al.'s protocol

1. This authentication protocol itself is secure against a passive attacker. On the contrary, a passive attacker who can extract the session key with non-negligible probability can also extract the session under a PFS challenge with the same non-negligible probability.
2. The session key cannot be calculated directly from the protocol's transcript (communication log) and the long-term secrets. This is also trivial as an attacker in a PFS challenge is given exactly the transcript and the long-term secrets, and its challenge is to extract the session key.

With these conditions, especially the second condition, we propose a method/benchmark to detect whether an authentication protocol satisfies PFS. The idea is to use graph analysis to assert that the session key cannot be directly derived from the transcript and the long-term secrets. The steps of our benchmark are as follows.

1. To examine whether an authentication protocol is fulfilling PFS, we construct a directed graph by constructing a dependency graph amount of the session key and other variables/long-term secrets. Taking the session key SK as the central node of the graph, whenever there is a *viable computation* from some variables to the session key, these variables would become nodes on the graph, and directed edges will be drawn from those nodes to a node representing the session key. We perform this operation recursively on every node in the graph.

2. Then, we color all nodes that are either long-term secrets or have been sent directly in the communication.
3. Next, all incoming edges of colored nodes are deleted. Judge whether the protocol meets the PFS through the graph composed of the remaining nodes.
4. Finally, after the above steps, we say that the protocol satisfies PFS if there is a root in the formed graph that is not colored. Otherwise, we say that the protocol does not meet PFS if all the roots of the formed graph are colored.

Note that some of the functions/operations in the protocol are invertible so that the dependency is bi-directional. For example, $a = h(b, c)$ we say a is depended by b and c if h is a one-way function. However, if h is a invertible function (e.g. $h(x, y) = x \oplus y$), the variables a, b, c are depended from each other. Moreover, some variables can be calculated by two or more different equations. For example, $x = g^{ab} \mod p$, x can be computed via $(g^a)^b \mod p$ or $(g^b)^a \mod p$ where g^a and g^b are some other values used/sent in the transcript. Thus, *alternative dependency* will be needed. The variable x in this example will be dependent on either of the set $\{g, a, b, p\}$ or $\{g^a, b, p\}$ or $\{g^b, a, p\}$. The node x will be colored if *any* one of the roots of these dependencies are all colored.

4.1. **Examples.** The following dependency graphs shown how the session keys can be derived in different protocols as all the roots of the session key are colored.

4.1.1. *Verifying the Radhakrishnan et al.'s protocol.* First, the calculation to SK requires variables $\{V_4, N_u, r_s\}$ because the formula $SK = h(V_4 \parallel N_u \parallel r_s)$. We add these variables around SK, draw the arrows from variables to SK, and continuously analyze the newly added variables according to our rules recursively. The calculation of r_s requires $\{V_4, V_6, N_u\}$, the calculation of N_u needs $\{V_2, b, d\}$, the calculation of V_4 needs $\{d, ID_i\}$ or $\{r, B_u, PW_i\}$, the calculation of ID_i needs $\{V_3, d, b, N_u\}$, and the calculation of b needs d. By analogy, the relationship calculation diagram can be shown in Figure 3.

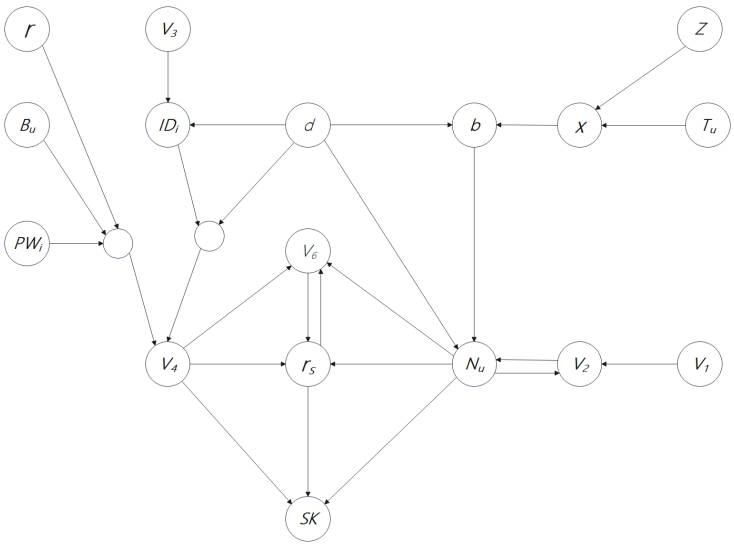

FIGURE 3. Step 1

(2) Next, we color all nodes which are either long-term secrets or have been sent directly in the communication. There nodes are $\{V_2, V_3, V_4, V6, Z, T_u, ID_i, d\}$. E can obtain d and $\{V_2, V_3, V_6, Z, T_u\}$ through the common channel. Because V_4 can be calculated by different equations, for example $V_4 = h(d \parallel ID_i)$, and the ID_i can be derived from the

obtained common channel nodes d and V_3, the ID_i and V_4 also need to be colored. The coloring result is shown in Figure 4.

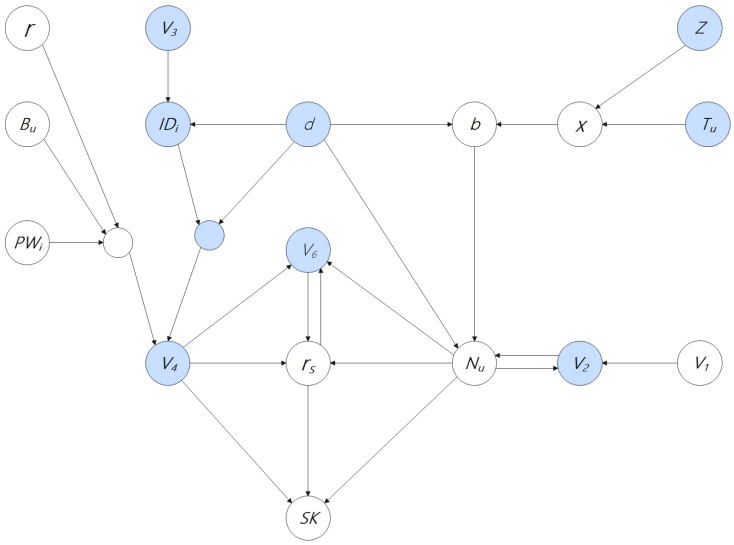

FIGURE 4. Step 2

(3) Finally, all the incoming edges of the colored nodes are removed, and whether the protocol satisfies PFS is judged by the graph composed of the remaining nodes. As shown in Figure 5, all root nodes in the graph are colored. SK can be calculated. Therefore, the protocol does not satisfy PFS.

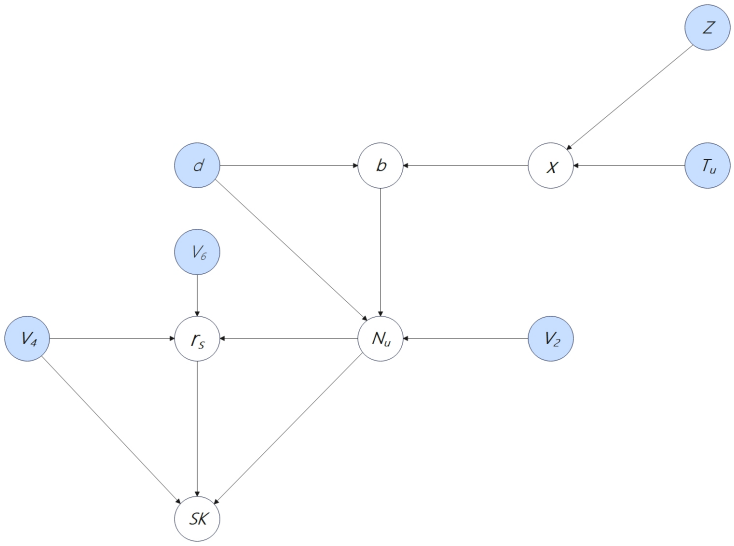

FIGURE 5. Final Result of Radhakrishnan et al.'s protocol

4.1.2. *Verifying the Karuppiah et al.'s protocol.* (1) First, the calculation to SK requires variables $\{T_u, T_s, k, A_i, ID_i, r_i\}$ because the formula $SK = h(ID_i \parallel A_i \parallel (r_i \oplus k) \parallel (T_u \oplus T_s))$. We add these variables around SK, draw the arrows of variables to SK, and continuously analyze the newly added variables according to our rules recursively. The calculation of A_i requires $\{B_i, rpwd_i\}$ or $\{x_s, ID_i\}$, the calculation of r_i needs $\{k, W_i, T_u, A_i\}$,

the calculation of k needs $\{N_i, ID_i, PWD_i\}$ or $\{W_i, A_i, T_u\}$, and the calculation of ID_i requires $\{DID, C_i, X_s\}$.

(2) Next, we color all nodes which are either long-term secrets or have been sent directly in the communication. These nodes are $\{C_i, DID, x_s, W_i, T_u, T_s\}$. This is because ID_i and A_i can be calculated by different equations, and there is a feasible calculation. So $\{ID_i, A_i\}$ are also colored.

(3) Finally, all the incoming edges of the colored nodes are removed, and whether the protocol satisfies PFS is judged by the graph composed of the remaining nodes. The final result is shown in Figure 6. All root nodes in this graph are colored. SK can be calculated. Therefore, the protocol does not satisfy PFS.

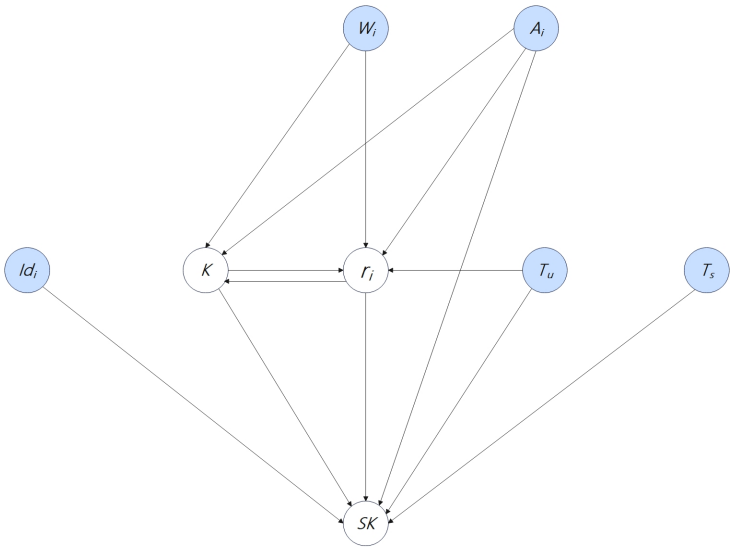

FIGURE 6. Final Result of Karuppiah et al.'s protocol

5. **Another Protocol satisfies PFS.** This section briefly reviews Chen et al.'s protocol [71] which is published recently. This protocol has been demonstrated to provide PFS. We use the proposed method to show that their protocol indeed provides PFS.

Chen et al.'s protocol [71] consists of three phases: initialization phase, registration phase, and login and authentication phase. Symbols and notations used in this protocol are presented in Table 3.

5.1. **The organization of the protocol.**

5.1.1. *Pre-deployment phase.* Before S and D_i are authenticated, a shared key K is first assigned to S and D_i, so that D_i can encrypt its own identity in the later registration phase. The shared key between the two is only known to S and D_i, and is inaccessible to other devices and personnel.

5.1.2. *Registration phase.* Figure 7 shows the IoT device registration phase. The detailed steps are described as follows.

i D_i chooses an identity ID_i and password PW_i, then encrypts the identity ID_i of D using a symmetric encryption algorithm to gain a pseudo-identity of the IoT device $RID_i = Enc_k(ID_i)$. Thereafter, the IoT device sends the registration request R_q and pseudo-identity RID_i to S through a secure channel.

Table 3. Notations in Chen et al.'s protocol

Notations	**Descriptions**
D,S	IoT device, server
D_i	i^{th} IoT device
ID_i	Identity of D_i
RID_i	$D_i's$ pseudo identity
PW_i	$D_i's$ password
K_s	Private key of S
K	Shared key between S and D
C_i,R_i	Response pair of D_i
PUF	Physically unclonable function
SID_i	pseudo identity of S, $(i = 1, 2, 3 \cdots)$
T_1, T_2, T_3	Time stamp
T', T''	Permissible transmission delay
N_i,N_s	Nonce geenrated by D_i and S
E	The adversary
$h(\cdot)$	Cryptographic one-way hash function
$\oplus$ $\|$	Bitwise XOR operation, concatenation operator
SK	Session key

ii Next, S generates the validity period $ETime$ and subsequently decrypts the pseudo identity to gain the device identity after receiving the registration information $ID_i = Des_k(RID_i)$. In addition, S encrypts the private key of S and the identity of D_i to gain $A = h(ID_i \parallel K_s \parallel ETime)$ and $B = h(ID_i \parallel K_s)$. Therefore, S generates C_i for D_i and also generates a series of pseudo-identities $SID_i(i = 1, 2, 3 \cdots)$ for its use. Then, S sends the calculated $\{A, SID_i, B, C_i\}$ to D_i through a secure channel.

iii Finally D_i encrypts the received C_i using a PUF to gain $R_i = PUF(C_i)$. Subsequently, it encrypts its own identity and password to gain V, which is used by D_i during the login phase. The parameters $\{ID_i, SID_i, A, B, V\}$ are stored in its own memory, and R_i is sent to S. S stores $\{C_i, R_i, RID_c, ETime\}$ in its own memory.

D_i		S
choose ID_i, PW_i		
compute $RID_i = Enc_k(ID_i)$		
Generate: R_q		
	$\xrightarrow{R_q, RID_i}$	
		Pick Etime
		compute $ID_i = Des_k(RID_i)$
		Compute $A = h(ID_i \parallel K_s \parallel ETime)$
		$B = h(ID_i \parallel K_s)$
		Generate C_i
		Generate a set of shadow identities SID
	$\xleftarrow{A, B, C_i, SID_i}$	
compute $R_i = PUF(C_i)$		
$V = h(ID_i \parallel B \parallel PW_i)$		
Store $\{ID_i, SID_i, B, A, V\}$ into D's memory		

Figure 7. Registration phase of Chen et al.'s protocol

5.1.3. *Login and authentication phase.* The detailed steps are listed in Figure 8 which shows the authentication phase with S and the login phase of D_i.

D_i	S
Input ID_i, PW_i	
Generate N_i	
$RID_i = Enc_k(ID_i)$	
$V_1 = h(ID_i \parallel B \parallel PW_i)$	
Verify $V_1 \stackrel{?}{=} V$	
Compute $c = RID_i \parallel N_i$	
$U = h(B \oplus N_i \oplus A \oplus T_1)$	
$\xrightarrow{c,SID_i,U,T_1}$	
	Verify $T_2 - T_1 \leq T'$
	Find RID_i through SID_i
	Compute $ID_i = Des_k(RID_i)$
	$N_i = c \oplus RID_i$
	$B = h(ID_i \parallel K_s)$
	$A = h(ID_i \parallel K_s \parallel ETime)$
	$U' = h(B \parallel N_i \parallel A \parallel T_1)$
	Verify $U' \stackrel{?}{=} U$
	Generate N_s
	Compute $G = N_i \oplus (N_s \parallel C_i)$
	$W = h(A \parallel B \parallel N_i \parallel N_s)$
	$SK = h(ID_i \parallel N_s \parallel N_i \parallel C_i \parallel R_i)$
	$\xleftarrow{G,W,T_2}$
Verify $T_3 - T_2 \leq T''$	
$(N_s \parallel C_i) = N_i \oplus G$	
$W' = h(A \parallel B \parallel N_i \parallel N_s)$	
Verify $W' \stackrel{?}{=} W$, abort if false	
$R_i = PUF(C_i)$	
$SK = h(ID_i \parallel N_s \parallel N_i \parallel C_i \parallel R_i)$	

FIGURE 8. Login and authentication phase of Chen et al.'s protocol

5.2. **Cryptanalysis of Chen et al.'s protocol.** Here we use the proposed benchmark to verify that the protocol meets PFS. According to our rules, taking the session secret key SK as the center, the relevant variables that can be used to calculate SK are extended to the four sides of the graph in turns.

(1) First, the calculation to SK requires variables $\{ID_i, N_s, N_i, C_i, R_i\}$. Because the formula $SK = h(ID_i \parallel N_s \parallel N_i \parallel C_i \parallel R_i)$. We add these variables around SK, draw the arrows of variables to SK, and continue to analyze the newly added variables according to our rules recursively. The calculation of R_i requires C_i, the calculation of ID_i needs $\{K, RID_i\}$, the calculation of N_i needs $\{C, ID_i, RID_i\}$, the calculation of N_s needs $\{C_i, N_i, G\}$, and the calculation of C_i needs $\{N_i, N_s, G\}$, the calculation of G needs $\{N_i, N_s, C_i\}$.

(2) Next, we color all nodes which are either long-term secrets or have been sent directly in the communication. There nodes are $\{K, C, G\}$. But he cannot gain RID_i, so RID_i is

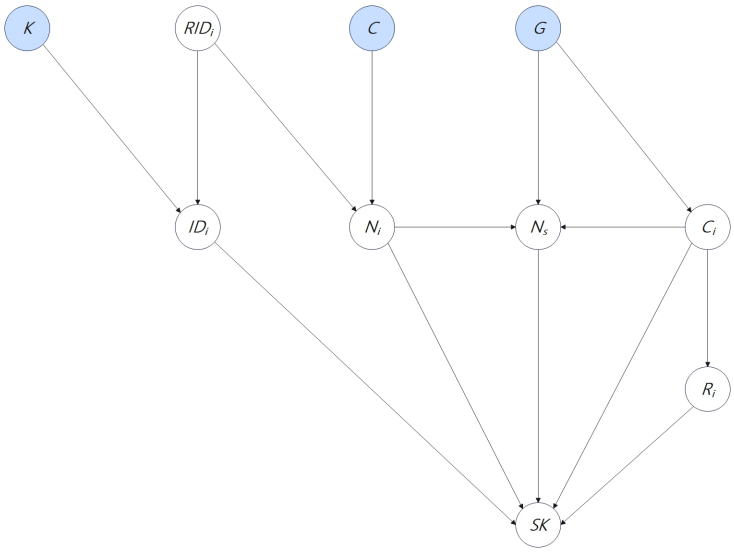

FIGURE 9. Final Result

not colored.

(3) Finally, all the incoming edges of the colored nodes are removed, and whether the protocol satisfies PFS is judged by the graph composed of the remaining nodes. The final result is shown in Figure 9, where a root node RID_i in the graph is not colored. Although attacker A knows the secret key K, he cannot gain RID_i, and SK cannot be calculated. Therefore, the protocol satisfies PFS.

6. **Conclusion.** In this paper, we first review two authentication protocols, Radhakrishnan et al.'s protocol, and Karuppiah et al.'s protocol. We also explain why these two protocols do not fit PFS. Then we propose a method named AuthPFS. It can verify if an authentication protocol can provide PFS. We also utilize AuthPFS to show that another authentication protocol does provide PFS. With AuthPFS, we hope researchers can use it to verify whether their authentication protocols satisfy PFS.

REFERENCES

[1] E. K. Wang, Z. Cao, T.-Y. Wu, and C.-M. Chen, "Mapmp: A mutual authentication protocol for mobile payment." *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, no. 4, pp. 697–707, 2015.

[2] S. Bojjagani, V. Sastry, C.-M. Chen, S. Kumari, and M. K. Khan, "Systematic survey of mobile payments, protocols, and security infrastructure," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–46, 2021. [Online]. Available: https://doi.org/10.1007/s12652-021-03316-4

[3] K.-H. Yeh, C. Su, J.-L. Hou, W. Chiu, and C.-M. Chen, "A robust mobile payment scheme with smart contract-based transaction repository," *IEEE ACCESS*, vol. 6, pp. 59 394–59 404, 2018.

[4] C.-M. Chen, X. Deng, W. Gan, J. Chen, and S. Islam, "A secure blockchain-based group key agreement protocol for iot," *The Journal of Supercomputing*, vol. 77, no. 8, pp. 9046–9068, 2021.

[5] S. M. Sajjad, M. R. Mufti, M. Yousaf, W. Aslam, R. Alshahrani, N. Nemri, H. Afzal, M. A. Khan, and C.-M. Chen, "Detection and blockchain-based collaborative mitigation of internet of things botnets," *Wireless Communications and Mobile Computing*, vol. 2022, 1194899, 2022.

[6] E. K. Wang, R. Sun, C.-M. Chen, Z. Liang, S. Kumari, and M. K. Khan, "Proof of x-repute blockchain consensus protocol for iot systems," *Computers & Security*, vol. 95, p. 101871, 2020.

[7] C.-M. Chen, X. Deng, S. Kumar, S. Kumari, and S. Islam, "Blockchain-based medical data sharing schedule guaranteeing security of individual entities," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–10, 2021.

[8] H.-M. Sun, C.-F. Hung, and C.-M. Chen, "An improved digital rights management system based on smart cards," in *2007 Inaugural IEEE-IES Digital EcoSystems and Technologies Conference*. IEEE, 2007, pp. 308–313.

[9] T.-Y. Wu, J. C.-W. Lin, Y. Zhang, and C.-H. Chen, "A grid-based swarm intelligence algorithm for privacy-preserving data mining," *Applied Sciences*, vol. 9, no. 4, p. 774, 2019.

[10] C.-T. Li, T.-Y. Wu, and C.-M. Chen, "A provably secure group key agreement scheme with privacy preservation for online social networks using extended chaotic maps," *IEEE ACCESS*, vol. 6, pp. 66 742–66 753, 2018.

[11] Z. Bao, W. Shi, S. Kumari, Z.-y. Kong, and C.-M. Chen, "Lockmix: a secure and privacy-preserving mix service for bitcoin anonymity," *International Journal of Information Security*, vol. 19, no. 3, pp. 311–321, 2020.

[12] K. Wang, C.-M. Chen, Z. Tie, M. Shojafar, S. Kumar, and S. Kumari, "Forward privacy preservation in iot-enabled healthcare systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 1991–1999, 2021.

[13] H. Ma, T.-Y. Wu, M. Chen, R.-H. Yang, and J.-S. Pan, "A parse tree-based nosql injection attacks detection mechanism." *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 4, pp. 916–928, 2017.

[14] T.-Y. Wu, C.-M. Chen, X. Sun, S. Liu, and J. C.-W. Lin, "A countermeasure to sql injection attack for cloud environment," *Wireless Personal Communications*, vol. 96, no. 4, pp. 5279–5293, 2017.

[15] S. Gul, R. U. Khan, M. Ullah, R. Aftab, A. Waheed, and T.-Y. Wu, "Tanz-indicator: A novel framework for detection of perso-arabic-scripted urdu sarcastic opinions," *Wireless Communications and Mobile Computing*, vol. 2022, 9151890, 2022.

[16] Y. Ma, Y. Peng, and T.-Y. Wu, "Transfer learning model for false positive reduction in lymph node detection via sparse coding and deep learning," *Journal of Intelligent and Fuzzy Systems*, vol. 43, no. 2, pp. 2121–2133, 2022.

[17] J. Gao, H. Zou, F. Zhang, and T.-Y. Wu, "An intelligent stage light-based actor identification and positioning system," *International Journal of Information and Computer Security*, vol. 18, no. 1-2, pp. 204–218, 2022.

[18] A. Shafiq, M. F. Ayub, K. Mahmood, M. Sadiq, S. Kumari, and C.-M. Chen, "An identity-based anonymous three-party authenticated protocol for iot infrastructure," *Journal of Sensors*, vol. 2020, 8829319, 2020.

[19] C.-M. Chen, Y. Huang, K.-H. Wang, S. Kumari, and M.-E. Wu, "A secure authenticated and key exchange scheme for fog computing," *Enterprise Information Systems*, vol. 15, no. 9, pp. 1200–1215, 2021.

[20] T.-Y. Wu, C.-M. Chen, K.-H. Wang, and J. M.-T. Wu, "Security analysis and enhancement of a certificateless searchable public key encryption scheme for iiot environments," *IEEE ACCESS*, vol. 7, pp. 49 232–49 239, 2019.

[21] M. Yavari, M. Safkhani, S. Kumari, S. Kumar, and C.-M. Chen, "An improved blockchain-based authentication protocol for iot network management," *Security and Communication Networks*, vol. 2020, 8836214, 2020.

[22] C.-M. Chen and S. Liu, "Improved secure and lightweight authentication scheme for next-generation iot infrastructure," *Security and Communication Networks*, vol. 2021, 6537678, 2021.

[23] K.-H. Wang, C.-M. Chen, W. Fang, and T.-Y. Wu, "On the security of a new ultra-lightweight authentication protocol in iot environment for rfid tags," *The Journal of Supercomputing*, vol. 74, no. 1, pp. 65–70, 2018.

[24] C.-M. Chen, C.-T. Li, S. Liu, T.-Y. Wu, and J.-S. Pan, "A provable secure private data delegation scheme for mountaineering events in emergency system," *IEEE ACCESS*, vol. 5, pp. 3410–3422, 2017.

[25] T.-Y. Wu, L. Yang, Q. Meng, X. Guo, and C.-M. Chen, "Fog-driven secure authentication and key exchange scheme for wearable health monitoring system," *Security and Communication Networks*, vol. 2021, 8368646, 2021.

[26] H. Xiong, Y. Hou, X. Huang, Y. Zhao, and C.-M. Chen, "Heterogeneous signcryption scheme from ibc to pki with equality test for wbans," *IEEE Systems Journal*, vol. 16, no. 2, pp. 2391–2400, 2021.

[27] J. Chen, F. Zou, T.-Y. Wu, and Y.-p. Zhou, "A new certificate-based aggregate signature scheme for wireless sensor networks." *Journal of Information Hiding and Multimedia Signal Processing*, vol. 9, no. 5, pp. 1264–1280, 2018.

[28] T.-Y. Wu, L. Yang, Z. Lee, S.-C. Chu, S. Kumari, and S. Kumar, "A provably secure three-factor authentication protocol for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2021, 5537018, 2021.

[29] Q. Mei, H. Xiong, Y.-C. Chen, and C.-M. Chen, "Blockchain-enabled privacy-preserving authentication mechanism for transportation cps with cloud-edge computing," *IEEE Transactions on Engineering Management*, 2022.

[30] K. Mahmood, M. F. Ayub, S. Z. Hassan, Z. Ghaffar, Z. Lv, and S. A. Chaudhry, "A seamless anonymous authentication protocol for mobile edge computing infrastructure," *Computer Communications*, vol. 186, pp. 12–21, 2022.

[31] S. A. Chaudhry, "Comments on" a secure, privacy-preserving, and lightweight authentication scheme for vanets"," *IEEE Sensors Journal*, vol. 22, no. 13, pp. 13 763–13 766, 2022.

[32] M. A. Khan, I. Ullah, A. Alkhalifah, S. U. Rehman, J. A. Shah, M. I. Uddin, M. H. Alsharif, and F. Algarni, "A provable and privacy-preserving authentication scheme for uav-enabled intelligent transportation systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3416–3425, 2021.

[33] T.-Y. Wu, Z. Lee, L. Yang, J.-N. Luo, and R. Tso, "Provably secure authentication key exchange scheme using fog nodes in vehicular ad hoc networks," *The Journal of Supercomputing*, vol. 77, no. 7, pp. 6992–7020, 2021.

[34] T.-Y. Wu, X. Guo, L. Yang, Q. Meng, and C.-M. Chen, "A lightweight authenticated key agreement protocol using fog nodes in social internet of vehicles," *Mobile Information Systems*, vol. 2021, 3277113, 2021.

[35] S. Hussain, Y. B. Zikria, G. A. Mallah, C.-M. Chen, M. D. Alshehri, F. Ishmanov, and S. A. Chaudhry, "An improved authentication scheme for digital rights management system," *Wireless Communications and Mobile Computing*, vol. 2022, 1041880, 2022.

[36] T.-Y. Wu, Y.-Q. Lee, C.-M. Chen, Y. Tian, and N. A. Al-Nabhan, "An enhanced pairing-based authentication scheme for smart grid communications," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–13, 2021.

[37] S. A. Chaudhry, K. Yahya, S. Garg, G. Kaddoum, M. Hassan, and Y. B. Zikria, "Las-sg: An elliptic curve based lightweight authentication scheme for smart grid environments," *IEEE Transactions on Industrial Informatics*, 2022. [Online]. Available: https://doi.org/10.1109/TII.2022.3158663

[38] C.-M. Chen, L. Chen, Y. Huang, S. Kumar, and J. M.-T. Wu, "Lightweight authentication protocol in edge-based smart grid environment," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–18, 2021.

[39] T.-Y. Wu, T. Wang, Y.-Q. Lee, W. Zheng, S. Kumari, and S. Kumar, "Improved authenticated key agreement scheme for fog-driven iot healthcare system," *Security and Communication Networks*, vol. 2021, 6658041, 2021.

[40] C.-T. Li, T.-Y. Wu, C.-L. Chen, C.-C. Lee, and C.-M. Chen, "An efficient user authentication and user anonymity scheme with provably security for iot-based medical care system," *Sensors*, vol. 17, no. 7, p. 1482, 2017.

[41] T.-Y. Wu, L. Yang, J.-N. Luo, and J. Ming-Tai Wu, "A provably secure authentication and key agreement protocol in cloud-based smart healthcare environments," *Security and Communication Networks*, vol. 2021, 2299632, 2021.

[42] T.-Y. Wu, Z. Lee, M. S. Obaidat, S. Kumari, S. Kumar, and C.-M. Chen, "An authenticated key exchange protocol for multi-server architecture in 5g networks," *IEEE ACCESS*, vol. 8, pp. 28 096–28 108, 2020.

[43] L. Yang, Y.-C. Chen, and T.-Y. Wu, "Provably secure client-server key management scheme in 5g networks," *Wireless Communications and Mobile Computing*, vol. 2021, 4083199, 2021.

[44] T.-Y. Wu, X. Guo, Y.-C. Chen, S. Kumari, and C.-M. Chen, "Sgxap: Sgx-based authentication protocol in iov-enabled fog computing," *Symmetry*, vol. 14, no. 7, p. 1393, 2022.

[45] C.-M. Chen, K.-H. Wang, K.-H. Yeh, B. Xiang, and T.-Y. Wu, "Attacks and solutions on a three-party password-based authenticated key exchange protocol for wireless communications," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 8, pp. 3133–3142, 2019.

[46] K. Mahmood, S. A. Chaudhry, H. Naqvi, S. Kumari, X. Li, and A. K. Sangaiah, "An elliptic curve cryptography based lightweight authentication scheme for smart grid communication," *Future Generation Computer Systems*, vol. 81, pp. 557–565, 2018.

[47] C.-M. Chen, B. Xiang, K.-H. Wang, K.-H. Yeh, and T.-Y. Wu, "A robust mutual authentication with a key agreement scheme for session initiation protocol," *Applied Sciences*, vol. 8, no. 10, p. 1789, 2018.

[48] M. A. Akram, Z. Ghaffar, K. Mahmood, S. Kumari, K. Agarwal, and C.-M. Chen, "An anonymous authenticated key-agreement scheme for multi-server infrastructure," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, pp. 1–18, 2020.

[49] R. Tso, K. Huang, Y.-C. Chen, S. M. M. Rahman, and T.-Y. Wu, "Generic construction of dual-server public key encryption with keyword search on cloud computing," *IEEE ACCESS*, vol. 8, pp. 152 551–152 564, 2020.

[50] T.-Y. Wu, Q. Meng, S. Kumari, and P. Zhang, "Rotating behind security: A lightweight authentication protocol based on iot-enabled cloud computing environments," *Sensors*, vol. 22, no. 10, p. 3858, 2022.

[51] M. A. Khan, I. Ullah, M. H. Alsharif, A. H. Alghtani, A. A. Aly, and C.-M. Chen, "An efficient certificate-based aggregate signature scheme for internet of drones," *Security and Communication Networks*, vol. 2022, 9718580, 2022.

[52] T. Wu, X. Guo, Y. Chen, S. Kumari, and C. Chen, "Amassing the security: An enhanced authentication protocol for drone communications over 5g networks," *Drones*, vol. 6, no. 1, p. 10, 2021.

[53] K. Xue, C. Ma, P. Hong, and R. Ding, "A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 316–323, 2013.

[54] C.-C. Chang and H.-D. Le, "A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 1, pp. 357–366, 2015.

[55] M. S. Farash, "Security analysis and enhancements of an improved authentication for session initiation protocol with provable security," *Peer-to-Peer Networking and Applications*, vol. 9, no. 1, pp. 82–91, 2016.

[56] J. Kim and J. Song, "A dual key-based activation scheme for secure lorawan," *Wireless Communications and Mobile Computing*, vol. 2017, 6590713, 2017.

[57] C. Wang, G. Xu, and J. Sun, "An enhanced three-factor user authentication scheme using elliptic curve cryptosystem for wireless sensor networks," *Sensors*, vol. 17, no. 12, p. 2946, 2017.

[58] M. Wazid, A. K. Das, V. Odelu, N. Kumar, and W. Susilo, "Secure remote user authenticated key establishment protocol for smart home environment," *IEEE Transactions on Dependable and Secure Computing*, vol. 17, no. 2, pp. 391–406, 2017.

[59] S. F. Aghili, H. Mala, M. Shojafar, and P. Peris-Lopez, "Laco: Lightweight three-factor authentication, access control and ownership transfer scheme for e-health systems in iot," *Future Generation Computer Systems*, vol. 96, pp. 410–424, 2019.

[60] I. Altaf, M. Arslan Akram, K. Mahmood, S. Kumari, H. Xiong, and M. Khurram Khan, "A novel authentication and key-agreement scheme for satellite communication network," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 7, p. e3894, 2021.

[61] Z. Yang, J. He, Y. Tian, and J. Zhou, "Faster authenticated key agreement with perfect forward secrecy for industrial internet-of-things," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 10, pp. 6584–6596, 2019.

[62] Z. Ali, S. Hussain, R. H. U. Rehman, A. Munshi, M. Liaqat, N. Kumar, and S. A. Chaudhry, "Itssaka-ms: An improved three-factor symmetric-key based secure aka scheme for multi-server environments," *IEEE ACCESS*, vol. 8, pp. 107 993–108 003, 2020.

[63] R. Shashidhara, S. Bojjagani, A. K. Maurya, S. Kumari, and H. Xiong, "A robust user authentication protocol with privacy-preserving for roaming service in mobility environments," *Peer-to-Peer Networking and Applications*, vol. 13, no. 6, pp. 1943–1966, 2020.

[64] M. J. Sadri and M. R. Asaar, "An anonymous two-factor authentication protocol for iot-based applications," *Computer Networks*, vol. 199, p. 108460, 2021.

[65] F. Wu, X. Li, L. Xu, P. Vijayakumar, and N. Kumar, "A novel three-factor authentication protocol for wireless sensor networks with iot notion," *IEEE Systems Journal*, vol. 15, no. 1, pp. 1120–1129, 2020.

[66] E. Lara, L. Aguilar, and J. A. García, "Lightweight authentication protocol using self-certified public keys for wireless body area networks in health-care applications," *IEEE ACCESS*, vol. 9, pp. 79 196–79 213, 2021.

[67] M. A. Khan, B. A. Alzahrani, A. Barnawi, A. Al-Barakati, A. Irshad, and S. A. Chaudhry, "A resource friendly authentication scheme for space–air–ground–sea integrated maritime communication network," *Ocean Engineering*, vol. 250, p. 110894, 2022.

[68] T.-Y. Wu, Z. Lee, L. Yang, and C.-M. Chen, "A provably secure authentication and key exchange protocol in vehicular ad hoc networks," *Security and Communication Networks*, vol. 2021, 9944460, 2021.

[69] C.-M. Chen, Z. Li, S. A. Chaudhry, and L. Li, "Attacks and solutions for a two-factor authentication protocol for wireless body area networks," *Security and Communication Networks*, vol. 2021, 3116593, 2021.

[70] Z. Li, Q. Miao, S. A. Chaudhry, and C.-M. Chen, "A provably secure and lightweight mutual authentication protocol in fog-enabled social internet of vehicles," *International Journal of Distributed Sensor Networks*, vol. 18, no. 6, 2022. [Online]. Available: https://doi.org/10.1177/15501329221104332

[71] C.-M. Chen, X. Li, S. Liu, M.-E. Wu, and S. Kumari, "Enhanced authentication protocol for the internet of things environment," *Security and Communication Networks*, vol. 2022, 2022.

[72] T.-Y. Wu, L. Yang, Z. Lee, C.-M. Chen, J.-S. Pan, and S. Islam, "Improved ecc-based three-factor multiserver authentication scheme," *Security and Communication Networks*, vol. 2021, 6627956, 2021.

[73] C.-M. Chen, Z. Chen, S. Kumari, and M.-C. Lin, "Lap-ioht: A lightweight authentication protocol for the internet of health things," *Sensors*, vol. 22, no. 14, p. 5401, 2022.

[74] T.-Y. Wu, Q. Meng, L. Yang, X. Guo, and S. Kumari, "A provably secure lightweight authentication protocol in mobile edge computing environments," *The Journal of Supercomputing*, 2022. [Online]. Available: https://doi.org/10.1007/s11227-022-04411-9

[75] N. Radhakrishnan and M. Karuppiah, "An efficient and secure remote user mutual authentication scheme using smart cards for telecare medical information systems," *Informatics in Medicine Unlocked*, 2018.

[76] M. Karuppiah, A. K. Das, X. Li, S. Kumari, F. Wu, S. A. Chaudhry, and R. Niranchana, "Secure remote user mutual authentication scheme with key agreement for cloud environment," *Mobile Networks and Applications*, no. 11, pp. 1–17, 2018.