sensors

MDPI

Article

Privacy Preserving Multi-Party Key Exchange Protocol for Wireless Mesh Networks

Amit Kumar Roy [1], Keshab Nath [2], Gautam Srivastava [3,4], Thippa Reddy Gadekallu [5] and Jerry Chun-Wei Lin [6,*]

1 Department of Computer Science and Engineering, National Institute of Technology Mizoram, Aizawl 796012, India; amitroy.cse@nitmz.ac.in
2 Department of Computer Science and Engineering, Indian Institute of Information Technology, Kottayam 686635, India; keshabnath@iiitkottayam.ac.in
3 Department of Mathematics and Computer Science, Brandon University, Brandon, MB R7A 6A9, Canada; srivastavag@brandonu.ca
4 Research Centre for Interneural Computing, China Medical University, Taichung 40402, Taiwan
5 School of Information Technology, Vellore Institute of Technology, Vellore 632014, India; thippareddy.g@vit.ac.in
6 Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, 5020 Bergen, Norway
* Correspondence: jerrylin@ieee.org

Abstract: Presently, lightweight devices such as mobile phones, notepads, and laptops are widely used to access the Internet throughout the world; however, a problem of privacy preservation and authentication delay occurs during handover operation when these devices change their position from a home mesh access point (HMAP) to a foreign mesh access point (FMAP). Authentication during handover is mostly performed through ticket-based techniques, which permit the user to authenticate itself to the foreign mesh access point; therefore, a secure communication method should be formed between the mesh entities to exchange the tickets. In two existing protocols, this ticket was not secured at all and exchanged in a plaintext format. We propose a protocol for handover authentication with privacy preservation of the transfer ticket via the Diffie–Hellman method. Through experimental results, our proposed protocol achieves privacy preservation with minimum authentication delay during handover operation.

Keywords: handover; authentication; privacy; tickets; computation cost; communication cost

Citation: Roy, A.K.; Nath, K.; Srivastava, G.; Gadekallu, T.R.; Lin, J.C.-W. Privacy Preserving Multi-Party Key Exchange Protocol for Wireless Mesh Networks. *Sensors* **2022**, *22*, 1958. https://doi.org/10.3390/s22051958

Academic Editor: Joel J. P. C. Rodrigues

Received: 28 January 2022
Accepted: 26 February 2022
Published: 2 March 2022

1. Introduction

As compared to conventional networks such as LAN and MANET, wireless mesh networks (WMN) have become the most promising network presently due to their advanced features. Due to their capacity to be self-organized and self-healing, WMN are the most favorable network [1]. Advanced features of WMN allow continuous network access to the end-users. Three mesh entities, namely gateway routers (GW), mesh routers (MR), and mesh clients (MC) form the architecture of WMN as shown in Figure 1. Mesh routers are also called mesh access points (MAP), which forward the mesh client's request to the gateway router (GW) for Internet access [2–5]. Due to the non-static nature, mesh clients can change their position from a home mesh access point to a foreign mesh access point. As a result, a secure handover authentication process should be carried out among mesh entities. Successful authentication during handover permits the client to join and access the Internet under a foreign MAP [6–8]. In the past, numerous protocols were proposed for handover authentication that are based on tickets [9,10]; however, they came up with certain issues and limitations, which are discussed in Section 2. The proposed multi-party key exchange protocol presented in this paper offers the privacy of the transfer ticket. Our proposed multi-party key exchange protocol is an extension of the Diffie–Hellman protocol [11]. The

privacy of a transfer ticket is preserved throughout the login authentication process (LAP) and handover authentication process (HAP) among the mesh entities. The protocol does not require any MAC key generation and master key generated from the AS, as it was required in the existing protocols. Mostly in existing protocols, the transfer ticket is issued by the authentication server (AS), but in our proposed work, it is issued by the mesh access point (MAP) within one hop; therefore, the presence of the AS is not considered in our proposed protocol throughout the handover process.

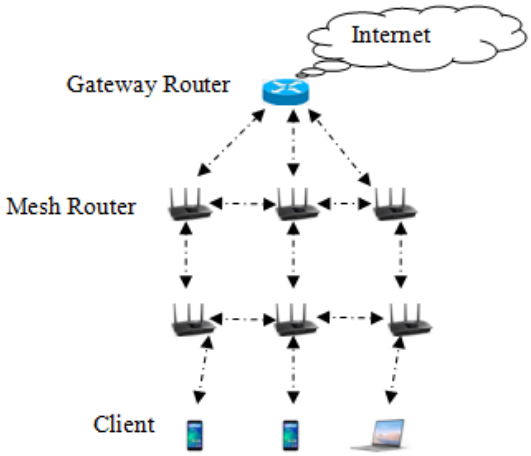

Figure 1. Architecture of WMN.

Discussion and Contribution

In this paper, we present an efficient authentication protocol during handover operation along with privacy preservation of tickets shared over the insecure channel. Our proposed protocol is analyzed against the Li et al. protocol because in this protocol authentication is carried out via tickets and computations are performed mostly by the TA and MAP [12]; secondly, the amount of communication cost is minimized (i.e., three-way handshake performed) during the handover authentication process; lastly, involvement of the third party during handover operation is omitted. All these properties make the protocol to be lightweight for mobile users in WMN; however, we analyzed the existing protocol in detail in Section 3 and found certain drawbacks. In this paper, our main contributions can be described as follows:

- We propose a multi-party key exchange protocol to generate a common secret key (CSK), which is shared among the MAP within a group. This common key is used for encrypting and decrypting the transfer tickets shared during the handover operation and offers privacy to the transfer tickets.
- We consider only symmetric key-based operations during the handover operation, which results in minimal computational cost.
- We achieve a complete handover authentication process with minimal communication cost, i.e., one-way handshake, which is efficient compared to two-way and three-way handshake of existing protocols.

The remainder of the paper is as follows: Section 2 describes the related works. In Section 3, the analysis of existing work and its drawbacks are discussed in detail. The proposed multi-party key exchange protocol and Diffie–Hellman protocol are discussed in Section 4. Proposed protocols during LAP and HAP are discussed in Section 5. Section 6 describes the experimental results and Section 7 concludes the paper.

2. Related Work

In this section, we discuss some of the existing protocols related to our proposed protocol. The existing protocols were mainly concerned with the handover authentication process carried out through a ticket-based approach.

Kassab et al. [13] proposed a secure protocol for proactive authentication for the IEEE 802.11F standard network during handover. During the handover process, the client sends a request message to the foreign access point to join the network. On acceptance, the foreign access point sends the message to the authentication server. The authentication server verifies the message. On successful verification, the authentication server issues an acceptance message to the foreign access point, which allows the client to join the network under the foreign access point; however, certain limitations were found in the protocol during the handover process. Limitations such as authentication delay due to verification of the request message by AS were required in a multi-hop fashion. Li et al. [14] proposed a handover protocol where re-authentication is strongly considered for the IEEE 802.11i standard network. Firstly, for mutual authentication, the complete process of authentication was formed among the mobile station and AS. Secondly, the AS issued a list of handover tickets of the neighboring access point to the mobile station. These lists of handover tickets allowed the mobile station to re-authenticate itself to the neighboring AP's during the handover operation; however, storing this list of tickets consumed massive storage space at mobile stations, which are usually resource constrained. Li et al. [15] proposed a protocol during handover based on broadcast authentication. The protocol allowed the client to be authenticated by the authentication server. During the handover operation, the authentication server issued and broadcast the tickets to each mesh access point, which allowed the clients to authenticate during the handover process; however, a massive authentication delay occurs due to multi-hop authentication required from the authentication server. He et al. [16] proposed a handover authentication protocol with a two-way handshake to complete the handover authentication process. The protocol was based on pre-shared pseudo identities (PIDi) generated by the AS to the mesh clients. However, a pseudo-identity involves the bilinear pairing operation, which results in high computational cost. Moreover, this approach pre-shared pseudo identities (PIDi) to the clients, putting extra load on clients' constrained resources. Xu et al. [17] proposed a protocol for wireless mesh network during handover authentication. The protocol allowed the authentication server (AS) to pre-distribute the tickets to the clients. These tickets were used during the re-authentication process. The client forwards the ticket to the intended mesh router based on its identity. Later, the mesh router verified the ticket sent by the client and on successful verification the client is re-authenticated; however, storing these pre-distributed tickets consumed massive storage space on the client side, which is resource constrained. Rathee et al. [18] proposed a secure protocol for WMN during handover operation. The protocol generates two keys, namely, the master key and group key shared between the authentication server (AS), mesh router, and mesh clients to authenticate each other. Then, the AS issued the ticket to the client and mesh router to authenticate each other during the handover process; however, the protocol comes up with certain limitations. First, during the handoff phase, target FMAP verifies the MC by comparing the tickets in step 2 but the protocol lacks the ability to verify the target FMAP by the MC side. Second, without verifying the target FMAP, the temporary session key is generated by both sides in step 3. Overall the protocol performs a 3-way handshake without completing the authentication process from the MC side. Third, a massive message was exchanged which leads to high communication costs during handover operation. Second, messages were exchanged in a plaintext format over the insecure channel, which violates the integrity of the message easily. Fourth, AS verifies the ticket and the client in a multi-hop fashion that leads to authentication delay. Wang et al. [19] proposed a batch handover authentication protocol based on the pre-distribution of handover keys to minimizing the authentication delay. The protocol preserved the privacy of the client where the identity of the foreign mesh router (MRj) and timestamp of the client (TMCi) was unknown to the attacker; however, storing these pre-distributed tickets consumed massive storage space at the client side, which are resource-constrained. Rekik et al. [20] proposed an optimized, secure authentication protocol based on extensible authentication protocol (EAP) for handover authentication;

however, the protocol requires multi-hop authentication from the AS, which results in an authentication delay.

To improve the handover authentication process, privacy was considered in Tsai et al. [21] protocol, Fu et al. [22] protocol, and Zhu et al. [23] protocol. These protocols preserved the privacy of the clients with a three-way handshake to complete the handover authentication process; however, to complete the three-way handshake protocol, it suffered from high computational cost. Yang et al. [24] proposed an efficient handover authentication protocol with a two-way handshake to complete the handover authentication process. The protocol was based on the group signature performed by the group manager (mesh access point). The roaming client is required to forward the group signature to the foreign mesh access point (FMAP) to validate its authentication; however, the protocol was based on bilinear pairing, which results in high computational cost. Table 1 compares the existing protocols with different parameters during handover operation.

Table 1. Comparison of protocols during handover operation.

Protocol	Θ_C Issued by	Privacy	Authent. Process	Compt. Cost	Commt. Cost	Authent. Delay
Kassab et al. [13]	AS	Yes	Multi-hop	High	High	High
Li et al. [14]	AS	Yes	Multi-hop	High	High	High
Li et al. [15]	AS	Yes	Multi-hop	High	High	High
He et al. [16]	AS	Yes	Multi-hop	High	Low	Low
Xu et al. [17]	AS	Yes	Multi-hop	High	High	High
Rathee et al. [18]	AS	No	Multi-hop	High	High	High
Wang et al. [19]	AS	Yes	Multi-hop	High	High	High
Rekik et al. [20]	AS	Yes	Multi-hop	High	High	High
Tsai et al. [21]	AS	Yes	Multi-hop	High	High	High
Fu et al. [22]	AS	Yes	Multi-hop	High	High	High
Zhu et al. [23]	AS	Yes	Multi-hop	High	High	High
Yang et al. [24]	AS	Yes	Multi-hop	High	Low	Low
Li et al. [12]	MAP	No	One-hop	Low	High	Low
Proposed	MAP	Yes	One-hop	Low	Low	Low

3. Analysis of Existing Protocol

In this section, we investigate in detail the existing protocol proposed by Li et al. [12] and discuss the security threat present in the protocol. The protocol considered a trust model, which employed a ticket agent (*TA*). The *TA* issues the MAP ticket and user ticket to authenticate each other during the login process and handover process. In the mesh network, *TA* acts as a centralized authority. The following shows the various faiths built among the mesh entities.

- **TA-MAP**: On a request of MAP ticket, faith is built between TA and the MAP.
- **TA-user**: On a request of user ticket, faith is built between TA and the user.
- **MAP-user**: Through MAP ticket and user ticket, faith is built between MAP and the user.
- **MAP$_1$-MAP$_2$**: Among neighboring MAPs, faith is built through their public key certificate. Faith among neighboring MAP allows the user to connect to any neighboring MAP.

3.1. Types of Ticket Issued to MAP and User for Mutual Authentication

- **User tickets (T_C)**: Faith between user and MAP is built through user ticket. The legality of user is proved to MAP through T_C. T_C contains the following elements

$$T_C = \{I_C, I_A, \tau_{exp}, P_C, Sig_A\} \quad (1)$$

where,
I_C = User identity.
I_A = *TA* identity.

τ_{exp} = expiry time of T_C.
P_C = User's public key.
Sig_A = TA digital signature.

- **MAP ticket (T_M)**: Builds faith between MAP and User. The legality of MAP is proved to user through T_M. T_M contains the following elements

$$T_M = \{ I_M, I_A, \tau_{exp}, P_M, Sig_A \} \quad (2)$$

where,
I_M = MAP identity.
I_A = TA identity.
τ_{exp} = expiry time of T_M.
P_M = MAP's public key.
Sig_A = TA digital signature.

- **Transfer tickets (Θ_C)**: Builds faith between user and FMAP (e.g., MAP_2). After, the mutual trust/faith is built between user and home MAP, Θ_C is generated by a home MAP (e.g., MAP_1). User proved its legality to MAP_2 through Θ_C. Θ_C contains the following elements

$$\Theta_C = \{ I_C, I_M, I_A, \tau_{exp}, V_{K_{MAC}}(I_C, I_M, I_A, \tau_{exp}) \} \quad (3)$$

where,
I_C = User identity owning Θ_C.
I_M = MAP identity issuing Θ_C.
I_A = TA identity.
τ_{exp} = expiry time of Θ_C.

3.2. The Login Authentication Protocol (LAP)

Assume that the trust agent (TA) issued a user ticket (T_C) to user C and MAP ticket (T_M) to MAP_1. Now the user and MAP_1 exchanged the tickets for mutual authentication. Steps for exchanging the tickets for mutual authentication are as follows:

$$C \rightarrow MAP_1 : I_C \quad (4)$$

$$MAP_1 \rightarrow C : T_{M1} \quad (5)$$

$$C \rightarrow MAP_1 : E_{P_{M1}}(T_C, N_{C1}, N_{C2}) \quad (6)$$

$$MAP_1 \rightarrow C : E_{P_C}(N_{M1}, N_{M2}) \quad (7)$$

$$C \rightarrow MAP_1 : N_{M2} \quad (8)$$

$$MAP_1 \rightarrow C : N_{C2}, (\Theta_C) \quad (9)$$

Step 1: For Internet access, the identity (I_C) of C is broadcast as a request message to MAP_1.

Step 2: On the acceptance of the request message, MAP_1 send its ticket (T_{M1}) to user C. After receiving T_{M1} by C, T_{M1} is verified through signature (Sig_A) and through expiry time τ_{exp} exists in T_{M1}.

Step 3: If verification of T_{M1} is successful, then the public key P_{M1} of MAP_1 is extracted from T_{M1} by C. Then User C encrypts the ticket T_C, nonces N_{C1}, and N_{C2} by using the public key P_{M1} and sends to MAP_1. On acceptance of the message, MAP_1 decrypts the message with its private key and verifies the ticket T_C. Verification is achieved through signature (Sig_A) and expiry time τ_{exp} present in T_C. MAP_1 ignores the ticket T_C, if the verification fails.

Step 4: After verification is successful, public key P_C of C is extracted from T_C by MAP_1. Later, MAP_1 encrypts two nonce N_{M1} and N_{M2} using P_C and forwards the encrypted message to user. Meanwhile, MAP_1 compute its shared MAC key $K_{MAC} = N_{C1} \parallel N_{M1}$ and pairwise master key $PMK_0 = N_{C1} \parallel N_{M1}$. On the acceptance of an encrypted message, this message is decrypted by user C with its own private key to gain N_{M1} and N_{M2}. Later, user C computes its shared MAC key $K_{MAC} = N_{C1} \parallel N_{M1}$ and pairwise master key $PMK_0 = N_{C1} \parallel N_{M1}$. The nonces N_{C1}, N_{C2}, N_{M1}, and N_{M2} are secured through asymmetric cryptography.

Step 5: After calculating a shared MAC key and pairwise master key, user C sends the nonce N_{M2} to MAP_1. On the acceptance of a nonce N_{M2}, MAP_1 verifies a nonce N_{M2} with a nonce issued by MAP_1 itself earlier in Equation (7). MAP_1 ignores the nonce, if N_{M2} does not match with the earlier nonce.

Step 6: After successful verification till step 5, MAP_1 generates a transfer ticket Θ_C. Then, MAP_1 sends to user C the nonce N_{C2} and transfer ticket Θ_C. User C after receiving the N_{C2} and Θ_C, verifies the nonce N_{C2} by checking with the nonce issued earlier by C itself in Equation (6). User C ignores the message if the N_{C2} does not match. Finally, step 1 to step 6 concludes the login authentication protocol. Later, Θ_C allows the user C to initiate the handover authentication process from home MAP_1 to foreign MAP.

3.3. The Handover Authentication Protocol (HAP)

To initiate an efficient handover operation, MAP_1 pre-distributes the shared keys to all its neighboring MAP. These keys are shared between the user and MAP_1 during the login authentication process. It is assumed that all the MAP contain its neighboring MAP public key certificates. On successful completion of the login authentication process, MAP_1 pre-distributes the encrypted shared keys, which includes I_C, I_{M1}, key K_{MAC}, and pairwise master key PMK_0 to its neighboring MAP_x. The encryption is performed via public key P_x of neighboring MAP_x. After receiving the encrypted shared keys, MAP_x uses its private key to decrypt it. Finally, the new authentication process is carried out with user C through these shared keys. During the handover process from MAP_1 to MAP_x, user C performs the following steps:

$$C \rightarrow MAP_x : \Theta_C, N_C, V_{K_{MAC}}(N_C) \tag{10}$$

$$MAP_x \rightarrow C : N_M, V_{K_{MAC}}(N_C, N_M) \tag{11}$$

$$C \rightarrow MAP_x : N_M, V_{K_{MAC}}(N_M) \tag{12}$$

Step 1: User C sends Θ_C, new nonce N_C and MAC $V_{K_{MAC}}$ (N_C) to foreign MAP_x shown in Equation (10). On the acceptance of the message, MAP_x verifies the accuracy of $V_{K_{MAC}}$ (N_C) by using previously received K_{MAC} from the home MAP_1. If the verification is successful, MAP_1 checks the elements in Θ_C to verify the legality of Θ_C. Likewise, only user C with K_{MAC} knowledge could generate a valid pair of (N_C, $V_{K_{MAC}}$ (N_C)).

Step 2: If the validation of Θ_C is successful, MAP_x send a nonce N_M and $V_{K_{MAC}}$ (N_C, N_M) to user C shown in Equation (11).

Step 3: On the acceptance of a message, user C sends N_M and $V_{K_{MAC}}$ (N_M) to MAP_x shown in Equation (12). On the acceptance of N_M and $V_{K_{MAC}}$ (N_M), MAP_x verifies the $V_{K_{MAC}}$ (N_M). On successful verification, the user's identity is approved as legal and concludes the HAP.

Discussion: We analyze the Li et al. [12] protocol in detail and found certain limitations and security threats in the protocol, which are highlighted below:

Two different authentication protocols are considered in the existing protocol: 1. To initiate mutual authentication, login authentication protocol (LAP) is considered. 2. To

initiate the handover process, handover authentication protocol (HAP) is considered as shown in Figure 2. Both LAP and HAP rely on certain keys such as pairwise master key and group transient key for authentication between users and MAP. Within the network, users are offered constraint power; therefore, the exchange of these keys should be minimized. Both LAP and HAP protocols suffered from security threats. Firstly, throughout LAP the information T_{M1}, N_{M2}, N_{C2} and Θ_C are shared in a plaintext format as $MAP_1 \rightarrow C$: T_{M1} shown in Equation (5), $C \rightarrow MAP_1$: N_{M2} as shown in Equation (8) and $MAP_1 \rightarrow C$: N_{C2}, Θ_C as shown in Equation (9). As a result, an intruder could easily acquire this information and misuse it.

Secondly, Θ_C are shared in the plaintext format as $C \rightarrow MAP_x$: Θ_C, N_{C2}, $V_{K_{MAC}}$ (N_C) during HAP as shown in Equation (10). As a result, an intruder could easily tamper the elements of Θ_C such as I_C, I_M, I_A, τ_{exp} and violates the integrity of transfer ticket (Θ_C); therefore, an intruder could easily eavesdrop on these exchanged messages at the time of the authentication process. Further, the intruder could replay these messages and try to obtain successful authentication as a user to access the network.

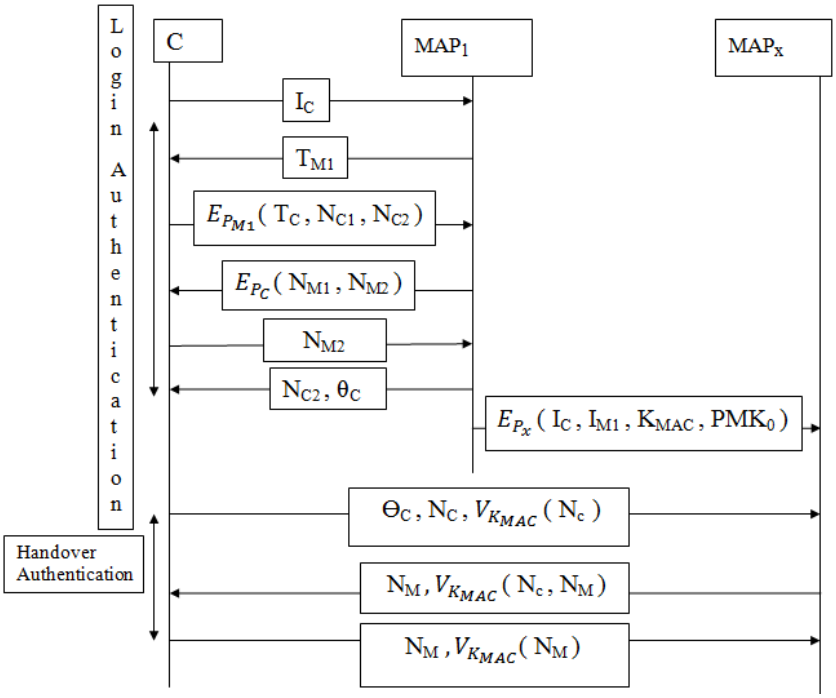

Figure 2. Li et al. [12] protocol during LAP and HAP.

4. Proposed Multi-Party Key Exchange Protocol

The proposed multi-party key exchange protocol is an extension of the Diffie–Hellman approach, which is performed within a group by the ticket agent (TA) and the MAP, where the ticket agent (TA) is known as a group controller (GC). The ticket agent (TA) generates the common secret key (CSK) and shares it in an encrypted form among neighboring MAP. Further, the CSK is employed for encryption and decryption of the transfer ticket during LAP and HAP between MAP and users. The detailed procedure for multi-party key exchange protocol is presented in Algorithm 1.

Algorithm 1: Muti-party key exchange algorithm.

Input: g = Generator, p = Set of prime number, n = Private key of TA, m = Private key of MAP

Output: Common Secret Key(CSK)

1 int i = 0, $i \in g$
2 int countPrimitiveRoots(int p)
3 **for** $(inti = 2; i < p; i++)$ **do**
4 **if** $(gcd(i, p) == 1)$ **then**
5 g[x] = i; `// generator`
6 x++;
7 return rand(g); `// g is a primitive root of p`
8 **end**
9 **end**
10 Compute, c = $g^{\sum_{TA=1}^{k} rand(n)}$ **mod** p;
`// Public key of TA`
11 Compute, d = $g^{\sum_{MAP=1}^{k} rand(m)}$ **mod** p;
`// Public key of MAP`
12 Compute, **power(d, n, p)** and **power(c, m, p)**; `// Secret key by TA and MAP`
13 F= $(\prod_{MAP_i})$ mod p;
14 Compute, $E_{SK_i}(CSK_i)$ and $D_{SK_i}(CSK_i)$;
`// Encrypted/Decrypted common secret key by TA and MAP`

In Algorithm 1, line 3 to line 7 returns the primitive roots less than the modulo prime p and the value is stored in an array g[]. In line 10 of Algorithm 1, the public keys for the i^{th} TA is computed as $TA_i = g^{n_i} \bmod p$. In line 11 of Algorithm 1, the public keys for the i^{th} MAP is computed as $MAP_i = g^{m_i} \bmod p$. After the generation of public keys by TA_i and MAP_i, both parties exchanged their public keys. In line 12 of Algorithm 1, the secret keys are computed by both the parties, where the value of n and m are chosen randomly. TA_i computes the secret key as $SK_i = MAP_i^{n_i} \bmod p$ and MAP_i computes the secret key as $SK_i = TA_i^{m_i} \bmod p$. Both parties generates the same secret keys in line 12. This keys are further used for encrypting and decrypting the common secret key (CSK) as shown in line 14. In line 13, the common secret key generated by the TA_i is computed as $CSK_i = (\prod_{MAP_i})$ mod p. TA generates the common secret key (CSK) by adding all the public keys received from each MAP's using product of sum operation ($\prod$). Later, CSK_i is used for encrypting and decrypting the transfer ticket throughout the LAP and HAP.

Reason to Considered Diffie–Hellman Key Exchange Protocol

We considered an extension of the Diffie–Hellman protocol [11] in our proposed protocol, which allows multiple users to securely exchange the keys over an insecure channel. Further, the keys are used for encrypting and decrypting the message. The difficulty and complexity of discrete logarithms to compute directly reflect the advantage of the Diffie–Hellman algorithm. The difficulty and complexity to crack the Diffie–Hellman protocol can be discussed as follows

- Discrete logarithms can be defined as a primitive root that belongs to the prime number p whose powers modulo p produce 1 to $p1$ integers; therefore, if a consider as prime number p, then $a^1 \bmod p, a^2 \bmod p, \ldots, a^{p1} \bmod p$ are distinct and contain integers from 1 to $p1$ in some permutation.
- Discrete Logarithm Problem: It is considered as a multiplicative cyclic group. Where $G = (g)$ is the generator of the cyclic group with element h of G; therefore, search unique integer x, where $g^x = h$, and x is the discrete logarithm of h with base g.
- Computational Diffie–Hellman Problem (CDH): It is defined as a cyclic group (G) with generator g and $g^{x1}, g^{x2} \in G$; therefore, known values are y1 = g^{x1} and y2 = g^{x2}

whereas $x1$ and $x2$ are unknown, hence search $y = g^{x1}g^{x2}$. CDH assumption is considered in most of the security of the cryptosystem. CDH assumption is associated with discrete logarithm assumption, where computing the discrete logarithm for value base a generator g is hard.

5. Proposed Protocol during Login Authentication Protocol (LAP) and Handover Authentication Protocol (HAP)

To overcome the limitations present in [12], we proposed a multi-party key exchanged protocol shown in Section 4. We consider the existing ticket types in our proposed protocol with a change in transfer ticket Θ_C elements. Changed elements in Θ_C are given as

$$\Theta_C = \{I_C, I_M, I_A, N_i, \tau_{exp}\} \tag{13}$$

where,
I_C = Identity of the user owning Θ_C.
I_M = Identity of the MAP issuing Θ_C.
I_A = TA Identity.
τ_{exp} = Θ_C's expiry time.
N_i = nonce to prevent replay attack.

5.1. Proposed Protocol for Login Authentication Protocol (LAP)

Initially, the user ticket and the MAP ticket were issued by the TA. Both MAP_1 and user exchanged their tickets for mutual authentication. Order of tickets exchanged between MAP_1 and User are as follows.

$$C \rightarrow MAP_1 : (I_C, P_C) \tag{14}$$

$$MAP_1 \rightarrow C : E_{P_C}(T_{M1}, N_{M1}) \tag{15}$$

$$C \rightarrow MAP_1 : E_{P_{M1}}(T_C, N_{M1}) \tag{16}$$

$$MAP_1 \rightarrow C, FMAP : E_{CSK_i}(\Theta_C) \tag{17}$$

Step 1: Identity and public key of user C is broadcast as a request message to MAP_1 to allow Internet access in Equation (14).

Step 2: After the message received, MAP_1 extracts the users public key P_C. MAP_1 uses the public key to encrypt the ticket T_{M1} and a nonce N_{M1} and sends to user C in Equation (15). On the acceptance of encrypted T_{M1} and a nonce N_{M1}, the user decrypts it by using its private key. After decryption, the user verifies a T_{M1} through Sig_A and τ_{exp} that resides within T_{M1}.

Step 3: After successful verification of T_{M1}, the public key P_{M1} of MAP_1 is extracted by the user from T_{M1}. The user encrypts T_C and nonce N_{M1} using P_{M1} and send towards MAP_1 in Equation (16). On the arrival of $E_{P_{M1}}$ (T_C, N_{M1}), MAP_1 decrypts the message and verifies the parameters of T_C. Further, the nonce N_{M1} is verified by MAP_1 to check the similarity of the nonce issued by the MAP_1 in Equation (15). If the verification is successful, then the authentication process is successful between the user and the MAP_1.

Step 4: After successful authentication, when user C wants to migrate, it informs to the MAP_1 to which FMAP the user wants to join. Thereafter, the MAP_1 generates and sends the encrypted transfer ticket as E_{CSK_i} (Θ_C) to user C and FMAP in Equation (17). Later, user C forwards the encrypted transfer ticket $E_{CSK_i}(\Theta_C)$ to FMAP to authenticate itself.

5.2. Proposed Protocol for Handover Authentication Protocol (HAP)

The common secret key (CSK) described in Section 4 is shared among the neighboring MAP's beforehand the handover process took place to offer privacy. After the completion of mutual trust between the client and HMAP (i.e., MAP_1), transfer ticket (Θ_C) is issued by MAP_1 to the client and FMAP during the login process as described in Equation (17) of Section 5.1. Later, when the client wants to join the foreign mesh access point (FMAP) during the handover process, the client sends the transfer ticket in an encrypted form as E_{CSK_i} (Θ_C) to the foreign mesh access point to prove its authenticity as

$$C \rightarrow \text{FMAP} : E_{CSK_i}(\Theta_C) \tag{18}$$

Step 1. User C sends E_{CSK_i} (Θ_C) to foreign mesh access point (FMAP) as shown in Equation (18). After receiving E_{CSK_i} (Θ_C), foreign mesh access point (FMAP) tries to decrypt it.

If (successful in decrypting, i.e., D_{CSK_i} (Θ_C)) then

FMAP verifies the contents of the transfer ticket for successful authentication, i.e., Θ_C sent by HMAP previously during the login process is equal to Θ_C sent by the user during the handover process. If both the contents of Θ_C are similar then the user is authenticated successfully by the foreign mesh access point.

Else

If (unsuccessful in decrypting) then

a user fails to authenticate itself to FMAP, as FMAP could not verify the transfer ticket (Θ_C) without decrypting it. Finally, FMAP concludes that the transfer ticket (Θ_C) was not issued from the corresponding HMAP with whom FMAP had shared the common secret key. Figure 3 shows the handover process of the proposed protocol. Figure 4 shows the proposed login authentication protocol (LAP) and handover authentication protocol (HAP).

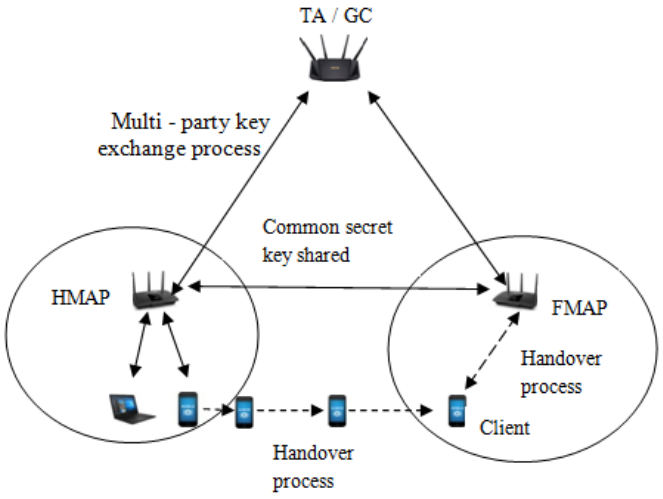

Figure 3. Proposed handover process.

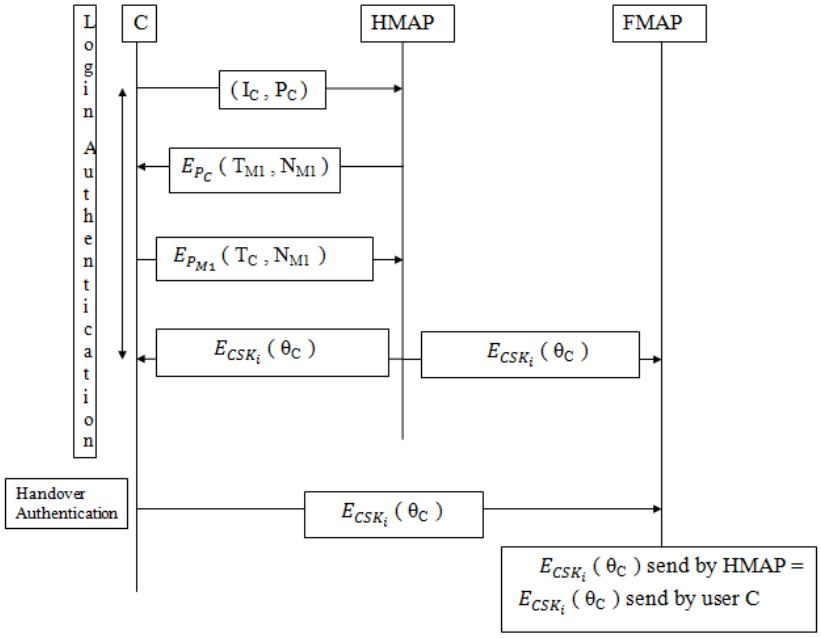

Figure 4. Proposed protocol during LAP and HAP.

6. Experimental Results

Implementation and experimental results of our proposed protocol is described in this section. Table 2 shows the experimental model setup, where Network Simulator 3 (NS3) is considered for simulating the proposed protocol as existing protocols have considered the same simulation tool. Other simulation parameters as mentioned in Table 2 is setup based on the existing protocols setup. Table 3 shows the simulation results gained during the login process. Table 4 shows the simulation results gained during the handover process. In both Tables 3 and 4, d represents the average delay transmission within a single hop.

Table 2. Experimental model setup.

Notation	Description
Platform	NS3
Traffic	CBR/UDP
Routing Protocol	AODV
Simulation Area	1000 × 1000 m
MAC protocol	IEEE 802.11
Total MAP	4
Placement of nodes	Randomly
Network size	50, 100, 150, 200
MAPs Transmission range	250 m
Clients Transmission range	100 m

6.1. Security Analysis of Proposed Login Authentication Protocol (LAP) and Handover Authentication Protocol (HAP)

In this section we analyze the security of our proposed protocol with respect to the following features:

Mutual Authentication: During login operation in Section 5.1, mutual authentication allowed the user and MAP_1 to verify each others identity. The verification is performed with their respective ticket's exchanged. Sig_A ensures the authentication of the tickets. Later, MAP_1 encrypts the message through E_{P_C} as E_{P_C} (T_{M1}, N_{M1}) shown in Equation (15) and the user encrypts the message through $E_{P_{M1}}$ as $E_{P_{M1}}$ (T_C, N_{M1}) shown in Equation (16). In Section 5.1, encryption of the messages shown in Equations (15) and (16)through public key ensures that only the user C and MAP_1 can decrypt the message.

Privacy preservation: In the Li et al. [12] protocol during LAP and HAP, the information such as T_{M1}, N_{M2}, N_{C2} and Θ_C are shared in a plaintext format as shown in Equations (5), (8)–(10). As a result, an intruder could easily tamper with the information exchanged during LAP and HAP. Our proposed protocol offers privacy to the exchanged information and prevents from tampering, such as E_{P_C} (T_{M1}, N_{M1}) as shown in Equation (15) and $E_{P_{M1}}$ (T_C, N_{M1}) as shown in Equation (16) during LAP. Privacy of the transfer ticket (Θ_C) is also preserved such as E_{CSK_i} (Θ_C) during LAP in Equation (17) and during HAP in Equation (18); therefore, both mutual authentication and privacy preservation prevents intruders to tamper with the integrity of the exchanged messages and also prevents a replay attack. As a result, the transmitted information could neither be captured by intruders throughout the authentication process, nor could any information be replayed to access the network as a user.

6.2. Result Analysis of Proposed Protocol

We considered four performance metrics to compute the overall performance of our proposed protocol. Comparison of proposed protocol with existing protocols is performed in terms of computation and communication cost, login delay, and handover delay.

Computational cost is computed as the time required in processing the various security operations given in column 1, row 2, 3, 4, 5, and 6 of Table 3 during login operation and column 1, row 2, 3, 4, 5, and 6 of Table 4 during the handover operation [25–27]. Total computation cost comparison during login operation is given in row 7 of Table 3 (i.e., 69.45 vs. 69.54 vs. 69.44 vs. 69.44 vs. 104.16). Total computation cost comparison during handover operation is shown in row 7 of Table 4 (i.e., 0.011 vs. 0.105 vs. 34.78 vs. 69.44 vs. 69.47). Figure 5 shows the total computational cost required during the login authentication process. Figure 6 shows the total computational cost required during the handover authentication process.

Communication cost is the total message exchanged between mesh entities during login operation and handover operation. Figure 7 shows the total communication cost required during the login authentication process. Total communication cost is the number of messages exchanged during the login operation given in column 1, row 6 of Table 3 (i.e., 4 vs.6 vs. 9 vs. 5 vs. 7). Figure 8 shows the total communication cost required during the handover authentication process. Total communication cost is the number of messages exchanged during handover operation given in column 1, row 6 of Table 4 (i.e., 1 vs. 3 vs. 5 vs. 4 vs. 2).

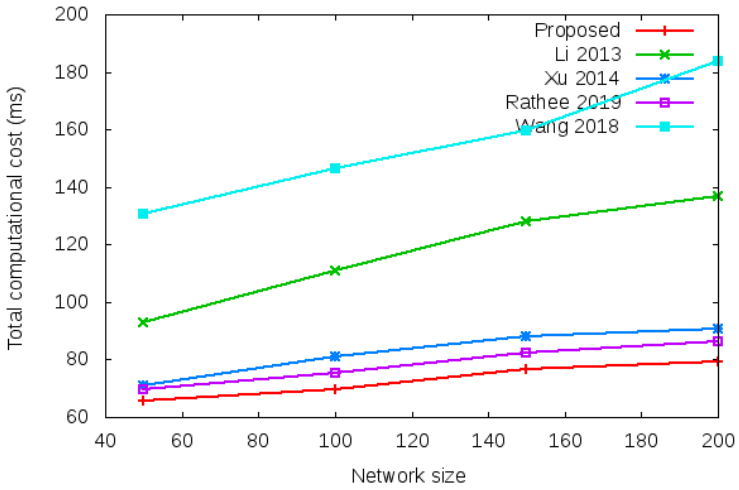

Figure 5. Total computational cost of proposed protocol vs. existing protocols with different network size of 50, 100, 150, and 200 nodes during login authentication process.

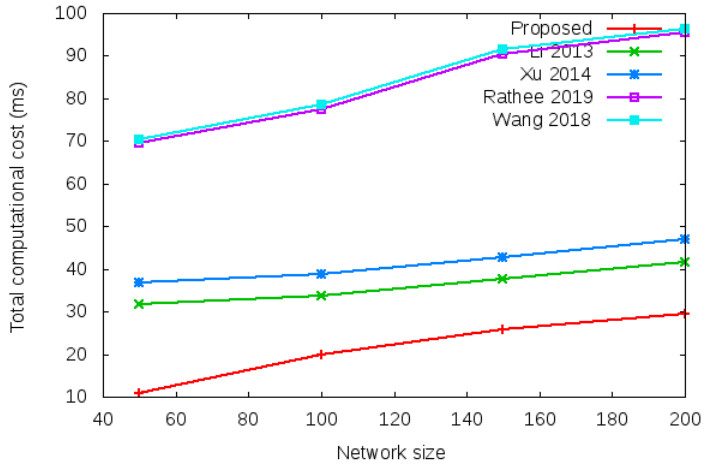

Figure 6. Total computational cost of proposed protocol vs. existing protocols with different network size of 50, 100, 150, and 200 nodes during handover process.

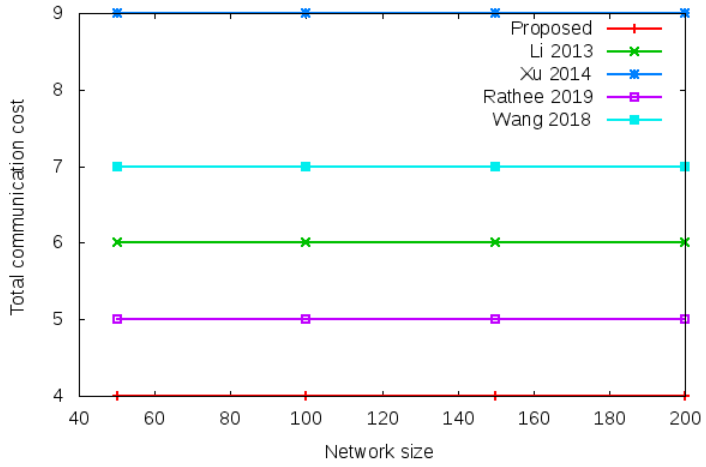

Figure 7. Total communication cost of proposed protocol versus existing protocols with different network size of 50, 100, 150, and 200 nodes during login process. Total communication cost comparison during login operation is given in column 1, row 6 of Table 3 (i.e., 4 vs. 6 vs. 9 vs. 5 vs. 7).

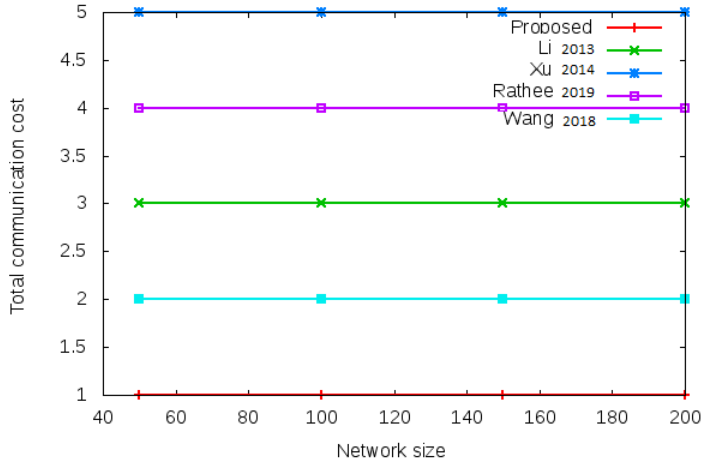

Figure 8. Total communication cost of proposed protocol versus existing protocols with different network size of 50, 100, 150, and 200 nodes during handover process. Total communication cost comparison during handover operation is given in column 1, row 6 of Table 4 (i.e., 1 vs. 3 vs. 5 vs. 4 vs. 2).

Table 3. Simulation results during login process.

Operation	Algorithm	Time	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
E_{pub_x} (m)	RSA	1.42	2	2	2	2	3
D_{prv_x} (m)	RSA	33.3	2	2	2	2	3
E_{CSK} (m)	AES	0.016	1	0	0	0	0
D_{CSK} (m)	AES	0.011	0	0	0	0	0
MAC	HMAC	0.015	0	2	0	0	0
Comput.cost (ms)	-	-	69.45	69.54	69.44	69.44	104.16
No. of messages	-	-	4	6	9	5	7
Login delay (ms)	-	-	69.45 + 4d	69.54 + 6d	69.44 + 9d	69.44 + 5d	104.16 + 7d

Table 4. Simulation results during handover process.

Operation	Algorithm	Time	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
E_{pub_x} (m)	RSA	1.42	0	0	1	2	2
D_{prv_x} (m)	RSA	33.3	0	0	1	2	2
E_{CSK} (m)	AES	0.016	0	0	0	0	0
D_{CSK} (m)	AES	0.011	1	0	0	0	0
MAC	HMAC	0.015	0	7	4	0	2
Comput.cost (ms)	-	-	0.011	0.105	34.78	69.44	69.47
No. of messages	-	-	1	3	5	4	2
Handover delay (ms)	-	-	0.011 + 1d	0.105 + 3d	34.78 + 5d	69.44 + 4d	69.47 + 2d

Login delay and handover delay are the time utilized during sending an authentication request and receiving the acceptance confirmation among mesh entities. The time utilized is the addition of computation cost and communication cost shown in the bottom row of Tables 3 and 4. Symbol d in the bottom row of Tables 3 and 4 denotes average delay transmission within a single hop. Figure 9 shows the login delay required during the login authentication process. Login delay is the time utilized during sending an authentication request and receiving the acceptance confirmation among mesh entities during the login process. The simulation result is shown in the bottom row of Table 3. Figure 10 shows the handover delay required during the handover authentication process. Handover delay is the time utilized during sending an authentication request and receiving the acceptance confirmation among mesh entities during the handover process. The simulation result is shown in the bottom row of Table 4.

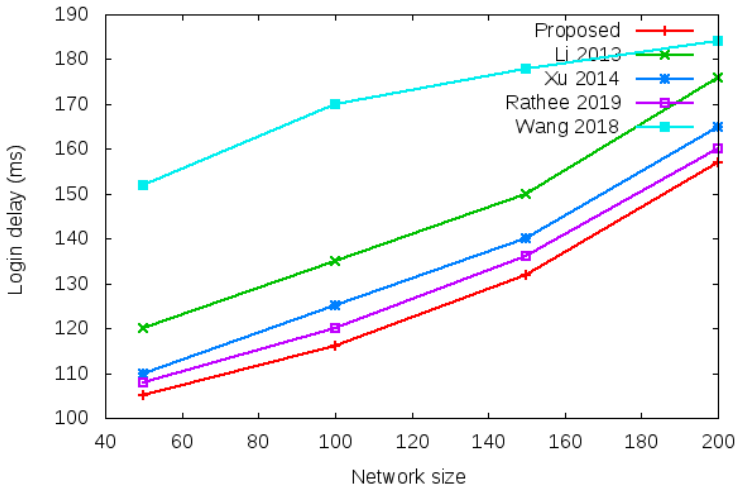

Figure 9. Login delay of the proposed protocol versus existing protocols with a different network size of 50, 100, 150, and 200 nodes based on total computational cost and total communication cost during the login authentication process.

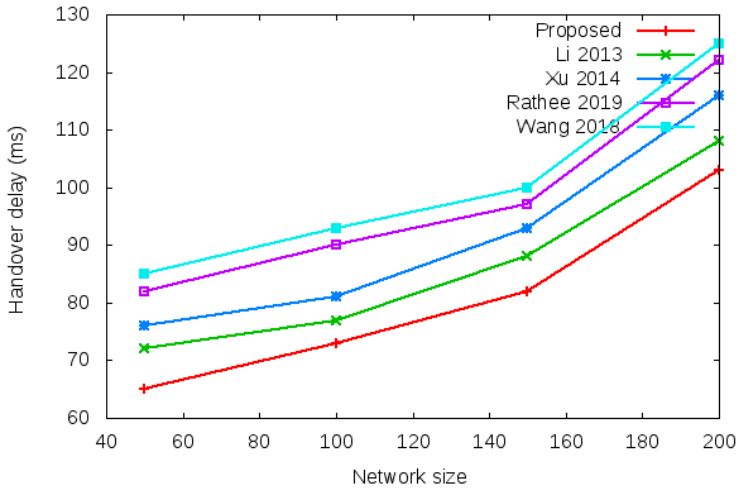

Figure 10. Handover delay of proposed protocol versus existing protocols with a different network size of 50, 100, 150, and 200 nodes based on total computational cost and total communication cost during the handover authentication process.

Table 5 and Figure 11 show the results of minimum login authentication delay with the network size of 100 to 600 mobile clients.

Table 5. Comparison on minimum login authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	155	170	160	158	175
200	161	174	168	166	179
300	174	188	182	179	191
400	183	191	188	185	198
500	210	236	230	225	241
600	251	267	260	254	270

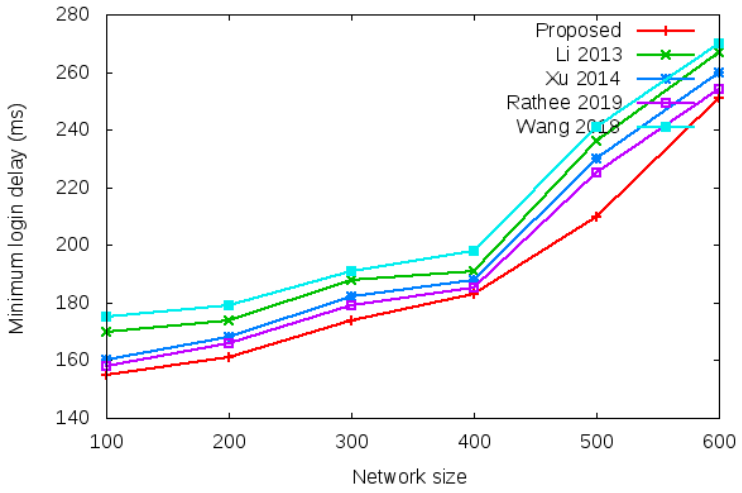

Figure 11. Minimum login authentication delay.

Table 6 and Figure 12 show the results of average login authentication delay with the network size of 100 to 600 mobile clients.

Table 6. Comparison on average login authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	190	209	205	198	213
200	205	234	226	215	238
300	231	251	242	239	261
400	264	276	270	266	282
500	284	297	293	288	301
600	311	326	321	316	337

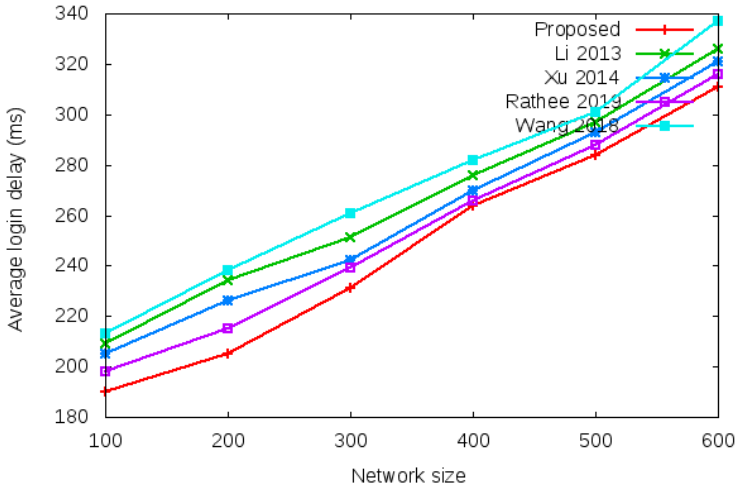

Figure 12. Average login authentication delay.

Table 7 and Figure 13 show the results of maximum login authentication delay with the network size of 100 to 600 mobile clients.

Table 7. Comparison on maximum login authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	210	229	223	216	235
200	225	245	238	231	254
300	244	261	254	249	273
400	279	293	289	282	298
500	309	326	319	313	338
600	345	361	357	349	372

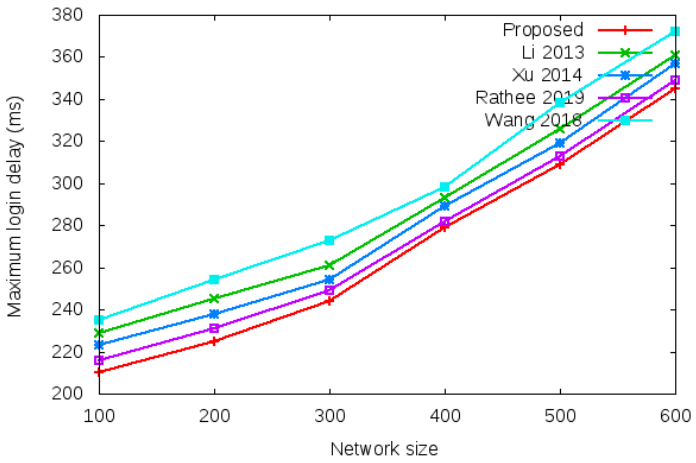

Figure 13. Maximum login authentication delay.

Table 8 and Figure 14 show the results of minimum handover authentication delay with the network size of 100 to 600 mobile clients.

Table 8. Comparison on minimum handover authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	72	75	79	82	85
200	75	78	84	90	93
300	95	99	108	115	121
400	126	138	146	152	157
500	134	141	148	154	159
600	141	147	154	161	169

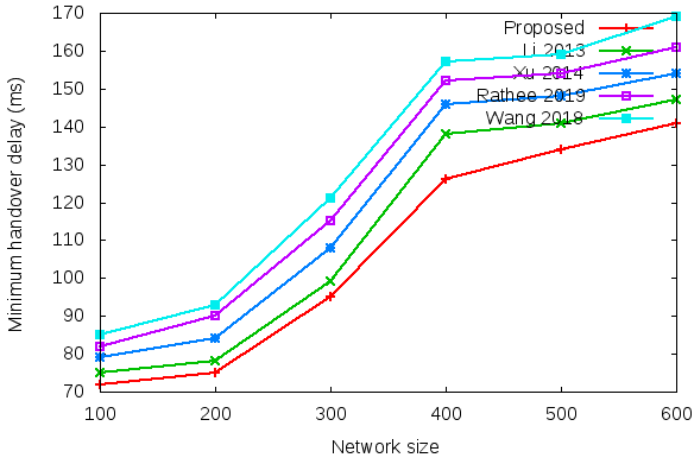

Figure 14. Minimum handover authentication delay.

Table 9 and Figure 15 show the results of average handover authentication delay with the network size of 100 to 600 mobile clients.

Table 9. Comparison on average handover authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	79	83	87	92	96
200	106	115	122	127	136
300	113	119	125	131	139
400	121	128	135	142	153
500	130	138	146	154	162
600	145	151	159	168	174

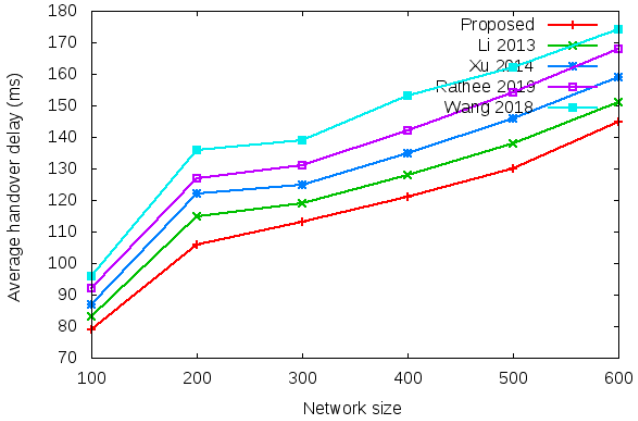

Figure 15. Average handover authentication delay.

Table 10 and Figure 16 show the results of maximum handover authentication delay with the network size of 100 to 600 mobile clients.

Table 10. Comparison on maximum handover authentication delay.

Number of Mobile Clients	Proposed	Li et al. [12]	Xu et al. [17]	Rathee et al. [18]	Wang et al. [19]
100	86	92	96	106	114
200	115	119	125	129	135
300	120	126	138	144	151
400	127	132	141	146	157
500	136	143	149	158	165
600	147	157	168	177	186

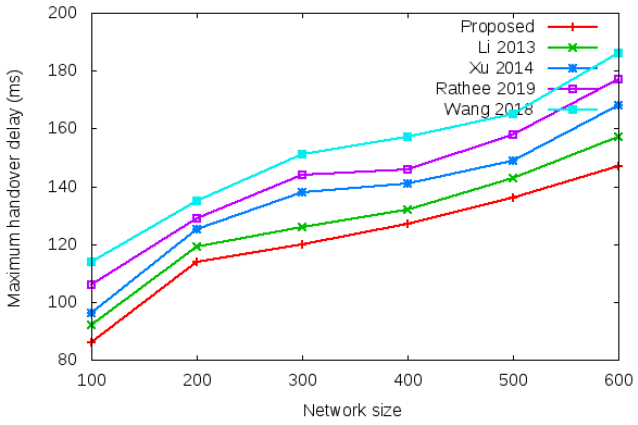

Figure 16. Maximum handover authentication delay.

7. Conclusions

Multi-party key exchange protocol preserves the privacy of the exchanged information shared during the login authentication process (LAP) and handover authentication process (HAP) to offer secure communication. The experimental results show that the proposed protocol achieves minimum authentication delay compared to existing protocols in terms of computation cost and communication cost. Through security analysis, it also proves that the proposed protocol offers a higher security level during the login authentication process (LAP) and handover authentication process (HAP) where no intruders can tamper with the exchanged information. In the future, the proposed protocol can be further extended to gain more efficiency and security during the login authentication process (LAP) and handover authentication process (HAP) for wireless mesh networks (WMN).

Author Contributions: Conceptualization, A.K.R.; Data curation, K.N.; Formal analysis, A.K.R. and K.N.; Funding acquisition, T.R.G. and J.C.-W.L.; Investigation, A.K.R.; Methodology, T.R.G.; Project administration, J.C.-W.L.; Supervision, G.S.; Validation, G.S.; Visualization, G.S.; Writing—review & editing, G.S. and J.C.-W.L. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is partially supported by the Western Norway University of Applied Sciences, Bergen, Norway.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

g	generator
mod p	modulo prime
HMAP	Home mesh access point
FMAP	Foreign mesh access point
TA/GC	Ticket agent/Group controller
AS	Authentication server
pub	Public key of x
prv	Private key of x
CSK	Common secret key
ID	mesh entities identity
LAP	Login authentication protocol
HAP	Handover authentication protocol
MAC	Message Authentication Code
E_{CSK} (m)	Encryption of message with CSK
D_{CSK} (m)	Decryption of message with CSK
T_C	User Ticket
T_M	MAP Ticket
Θ_C	Transfer ticket
Sig_A	Digital Signature of AS
MC	Mesh client
MR	Mesh router
GW	Gateway
PMK	Pairwise Master Key

References

1. Akyildiz, I.F.; Wang, X.; Wang, W. Wireless mesh networks: A survey. *Comput. Netw.* **2005**, *47*, 445–487. [CrossRef]
2. Seyedzadegan, M.; Othman, M.; Ali, B.M.; Subramaniam, S. Wireless mesh networks: WMN overview, WMN architecture. In Proceedings of the International Conference on Communication Engineering and Networks IPCSIT, Hong Kong, China, 25–27 November 2011; Citeseer: Princeton, NJ, USA, 2011; Volume 19, p. 2.
3. Franklin, A.A.; Murthy, C.S.R.; Zhang, Y.; Zheng, J.; Hu, H. An introduction to wireless mesh networks. In *Security in Wireless Mesh Networks*; Book Chapter; Zhang, Y., Ed.; IntechOpen Limited: London, UK, 2007; pp. 3–44.
4. Sen, J. Security and privacy issues in wireless mesh networks: A survey. In *Wireless Networks and Security*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 189–272.
5. Santhanam, L.; Xie, B.; Agrawal, D.P. Selfishness in mesh networks: Wired multihop MANETs. *IEEE Wirel. Commun.* **2008**, *15*, 16–23. [CrossRef]
6. Choudhary, K.; Gaba, G.S.; Butun, I.; Kumar, P. Make-it—A lightweight mutual authentication and key exchange protocol for industrial internet of things. *Sensors* **2020**, *20*, 5166. [CrossRef]
7. Wu, T.Y.; Lee, Z.; Yang, L.; Luo, J.N.; Tso, R. Provably secure authentication key exchange scheme using fog nodes in vehicular ad hoc networks. *J. Supercomput.* **2021**, *77*, 6992–7020. [CrossRef]
8. Chen, C.M.; Huang, Y.; Wang, K.H.; Kumari, S.; Wu, M.E. A secure authenticated and key exchange scheme for fog computing. *Enterp. Inf. Syst.* **2021**, *15*, 1200–1215. [CrossRef]
9. He, D.; Chan, S.; Guizani, M. Handover authentication for mobile networks: Security and efficiency aspects. *IEEE Netw.* **2015**, *29*, 96–103. [CrossRef]
10. Wang, K.; Wang, Y.; Zeng, D.; Guo, S. An SDN-based architecture for next-generation wireless networks. *IEEE Wirel. Commun.* **2017**, *24*, 25–31. [CrossRef]
11. Diffie, W.; Hellman, M. New directions in cryptography. *IEEE Trans. Inf. Theory* **1976**, *22*, 644–654. [CrossRef]
12. Li, C.; Nguyen, U.T.; Nguyen, H.L.; Huda, N. Efficient authentication for fast handover in wireless mesh networks. *Comput. Secur.* **2013**, *37*, 124–142. [CrossRef]
13. Kassab, M.; Bonnin, J.M.; Guillouard, K. Securing fast handover in WLANs: A ticket based proactive authentication scheme. In Proceedings of the 2007 IEEE Globecom Workshops, Washington, DC, USA, 26–30 November 2007; IEEE: New Jersey, NJ, USA, 2007; pp. 1–6.
14. Li, G.; Chen, X.; Ma, J. A ticket-based re-authentication scheme for fast handover in wireless local area networks. In Proceedings of the 2010 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Chengdu, China, 23–25 September 2010; IEEE: New Jersey, NJ, USA, 2010; pp. 1–4.

15. Li, G.; Ma, J.; Jiang, Q.; Chen, X. A novel re-authentication scheme based on tickets in wireless local area networks. *J. Parallel Distrib. Comput.* **2011**, *71*, 906–914. [CrossRef]
16. He, D.; Wang, D.; Xie, Q.; Chen, K. Anonymous handover authentication protocol for mobile wireless networks with conditional privacy preservation. *Sci. China Inf. Sci.* **2017**, *60*, 052104. [CrossRef]
17. Xu, L.; He, Y.; Chen, X.; Huang, X. Ticket-based handoff authentication for wireless mesh networks. *Comput. Netw.* **2014**, *73*, 185–194. [CrossRef]
18. Rathee, G.; Saini, H. Secure handoff technique with reduced authentication delay in wireless mesh network. *Int. J. Adv. Intell. Paradig.* **2019**, *13*, 130–154.
19. Wang, D.; Xu, L.; Wang, F.; Xu, Q. An anonymous batch handover authentication protocol for big flow wireless mesh networks. *EURASIP J. Wirel. Commun. Netw.* **2018**, *2018*, 200. [CrossRef]
20. Rekik, M.; Meddeb-Makhlouf, A.; Zarai, F.; Nicopolitidis, P. OAP-WMN: Optimised and secure authentication protocol for wireless mesh networks. *Int. J. Secur. Netw.* **2019**, *14*, 205–220. [CrossRef]
21. Tsai, J.L.; Lo, N.W. Provably secure anonymous authentication with batch verification for mobile roaming services. *Ad Hoc Netw.* **2016**, *44*, 19–31. [CrossRef]
22. Fu, A.; Zhang, Y.; Zhu, Z.; Jing, Q.; Feng, J. An efficient handover authentication scheme with privacy preservation for IEEE 802.16 m network. *Comput. Secur.* **2012**, *31*, 741–749. [CrossRef]
23. Zhu, H.; Lin, X.; Shi, M.; Ho, P.H.; Shen, X. PPAB: A privacy-preserving authentication and billing architecture for metropolitan area sharing networks. *IEEE Trans. Veh. Technol.* **2008**, *58*, 2529–2543.
24. Yang, G.; Huang, Q.; Wong, D.S.; Deng, X. Universal authentication protocols for anonymous wireless communications. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 168–174. [CrossRef]
25. Jemmali, M.; Denden, M.; Boulila, W.; Jhaveri, R.H.; Srivastava, G.; Gadekallu, T.R. A Novel Model Based on Window-Pass Preferences for Data-Emergency-Aware Scheduling in Computer Networks. *IEEE Trans. Ind. Inform.* **2022**. [CrossRef]
26. Jhaveri, R.H.; Ramani, S.V.; Srivastava, G.; Gadekallu, T.R.; Aggarwal, V. Fault-resilience for bandwidth management in industrial software-defined networks. *IEEE Trans. Netw. Sci. Eng.* **2021**, *8*, 3129–3139. [CrossRef]
27. Maddikunta, P.K.R.; Srivastava, G.; Gadekallu, T.R.; Deepa, N.; Boopathy, P. Predictive model for battery life in IoT networks. *IET Intell. Transp. Syst.* **2020**, *14*, 1388–1395. [CrossRef]