Untraceable Authentication Protocol for IEEE802.11s Standard

Reham A. Abouhogail [1]

[1]Electrical Quantities Metrology Dept., National Institute of Standards (NIS), Egypt

Abstract: In the current paper, a new handover authentication protocol for IEEE802.11s Wireless mesh networks is presented. The new protocol divides the network into a number of cells, each cell contains a number of access points and based on the concept of ticket authentication, the mesh user takes a new ticket when enters the region of a new cell which decreases the handover latency. Moreover, in the current paper, a new idea for ticket generation is proposed, called Chain Ticket Derivation Function (CTDF), which uses the concept of a chain. Using CTDF in our proposed protocol raises the level of privacy for the users. The security analysis presented in the paper showed more strengths in our proposed scheme. Two formal verification tools, AVISPA and BAN logic are used to test the proposed protocol.

Keywords: IEEE802.11s; Fast handover; Authentication protocol; Ticket method; Privacy**.**

1. Introduction

Wireless mesh networks (WMNs) consist of mesh users and mesh points. The mesh points are divided into mesh access points and mesh gateways as shown in Figure.1. Mesh users can be fixed like desktops, and servers or movable like cell phones, tablets, and laptops. WMNs support internet access in case of wiring or connecting cables is hard or costly, and the time of deployment is critical [1]. WMNs support many important applications like internet access providing in rural zones, ad hoc networking in case of emergency and disaster rescue, provide people with the necessary information in airports, shopping centers, and public transportation, and in case of surveillance and security [1]. Aboba et. al. [2] proposed an extensible authentication protocol encapsulating transport layer security (EAP-TLS) to secure the transport layer in WMNs. This protocol satisfies mutual authentication between the mesh user and the access point. However, it suffers from high latency because each node has to connect to the authentication server to complete its authentication process [3]. Four-way handshake encryption is the used authentication method in 802.1X [4]. The four-way handshake contains four messages between the user and the access point. In the four-way handshake, there are four encryption algorithms, Master Session Key (MSK), Pairwise Temporal Key (PTK), Group Temporal Key (GTK), Group Master Key (GMK), Pairwise Master Key (PMK) [3]. The first derived key during 802.1X is the MSK. The PMK is generated from the MSK. For increasing security, the PMK isn't transmitted through the network. PTK is generated using PMK, and GTK is generated using GMK [5]. All unicast traffic between *U* and *AP* are encrypted using PTK, and all broadcast traffic between *AP* and the number of users are encrypted using GTK [3].

WMNs have some distinctive features compared with conventional wireless networks [6]:

Flexibility. WMN can be self-organized, and easy configured. All-access points are connected by multiple paths due to which it provides greater flexibility and the chances of disconnection from the network are minimal. In WMNs, all *AP*s can be connected with each other by different paths because of more flexibility. Moreover, the disconnection from the network is lower. So, the network availability in WMNs is more.

1. Self-Evolving. There's an algorithm in the mesh access points to select a suitable path for the wired and the wireless networks.
2. Self-Recovery. WMNs are self-recoverable. If an access point failed, there's another access point in its surrounding that will detect this failure and reorganize the problem according to the protocol in force.
3. Multi-hop [1]. WMNs allow multi-hopping to extend the coverage of the network. Especially, the wireless network.

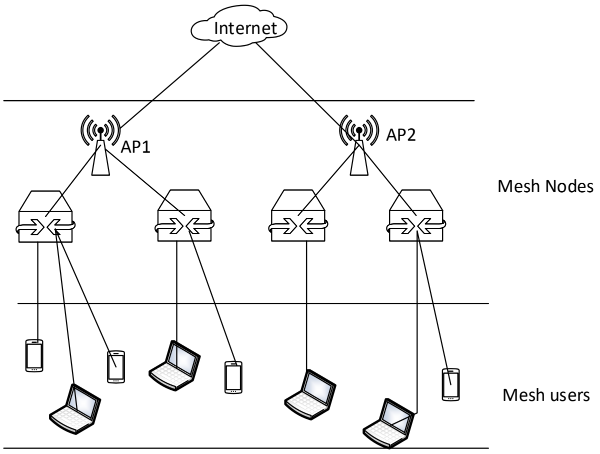


Figure 1. Example for a WMN.

On the other hand, there are some unsolved problems in WMNs [6]:

1. Latency. As the number of nodes increases in the network the number of required hops to complete the routing increase which leads to increase latency in the WMN.
2. Security. Because the routing in WMNs is done by various nodes these may lead to several vulnerabilities in the network. Moreover, there's a possibility of a set of rogue *AP*s in the network.
3. Scalability. Mesh networks are not scalable because network capability is decreased as more access points are increased.

The latest version of WMNs IEEE802.11s [7] does not support fast handover for mobile users. A mesh user, *U* to be authenticated by its new home mesh access point, *AP* must communicate with the authentication server, *AS* which may be located many hops away from *U*. This operation leads to long latency in the handover operation, which is not suitable for real-time applications like voice over IP (VoIP) or video conference. Nowadays, with the problem of Covid-19, real-time applications become a necessary measurement parameter and a required factor in many communication methods between people. So we work in the current paper to enhance the latency during the handover process in WMNs by proposing a new efficient handover authentication protocol. The new protocol is based on the ticket authentication method in handover; the mesh user does not need to connect with AS in each hop to minimize the latency during the handover process.

The main contributions of the current presented paper:

1) Fast handover. Our proposed protocol supports fast handover by dividing the network into several cells. Then select certain *AP* called *APc* from each cell to communicate with U before the handover starts instead of communicating with AS in each handover process.
2) Efficiency. The new proposed protocol uses light cryptographic functions during the handover authentication operation which is suitable for mobile devices.
3) Traceability. The new proposed protocol presents an untraceable route for any *U* involved in the system by changing the ticket dedicated to *U* for each new network cell.
4) Mutual authentication. The mutual authentication between the three shared entities in our proposed protocol, *AS*, *AP*, *U* is realized.

2. Related work

To improve handover latency in WMNs, several protocols have been proposed. Based on the used cryptographic primitives in mutual authentication operation between the user and the *AP*, the authentication protocols for wireless networks are divided into two categories as mentioned in [8]: symmetric key- based protocols and public-key- based protocols. First: the symmetric key- based protocols:

This type as in [1, 9, 10, 11, 12] uses symmetric key algorithm as our proposed protocol which decreases the required computation overhead. Here, we introduce only the most relevant protocols to our proposed one. The proposed protocol in [8] uses one group key for all base stations (BSs). The AS dedicates a group key (K_G) to all BSs. Before a handover operation happens, the current home base station generates a symmetrically encrypted ticket for the roaming user using K_G. The encrypted ticket contains the identity of the user, the Pairwise Master Key (PMK), and the expiration time. Upon handoff, the user sends his ticket to the new target BS, which decrypts the received ticket using K_G and gets PMK. Then, using PMK, the user and the new BS can authenticate each other. In this scheme, K_G is known by all BSs. So, the security of this scheme will be under risk if one of these BSs is compromised. The user uses the same secret (the same PMK) with all BSs. So, the forward and backward secrecy is not satisfied. Li et al.'s [1] proposed two authentication protocols, which are the initial login authentication protocol (LAP) and the handover authentication protocol (HAP). They presented the definition of ticket and trust model according to their authentication protocols are dependent. They also describe the three types of tickets used in their proposed protocols, client tickets, MAP tickets, and transfer tickets. Generally, these tickets are used for the mutual authentication between the user and the *AP*. The transfer ticket especially helps build trust between a new *AP* and *U*. *U* sends the transfer ticket to the new *AP* as a requirement for handover authentication. After LAP completed, the user and the *AP* use the PMK to generate the PTK as defined in the IEEE802.11i security standards [13]. The PMK is updated periodically. However, the new PMK is generated using the old PMK with some plaintext information. Furthermore, if an *AP* is compromised all the other *AP*s will be affected. Thus, there's a domino effect problem. Moreover, we can see a privacy problem, because the identity of the user and the identity of the *AP* are sent as plain text. The adversary can track down a certain user. Another problem in Li et al.'s protocol is that the expiration time and the date of generation of the transfer ticket are sent as a plain [14]. The user, *U* can change them and produce the matched MAC to be sent with them, because the used key to produce the MAC (KMAC) is known to *U*. A Privacy and Fast Handover Authentication Protocol (PF-HAP) is proposed in [4] based on the ticket authentication method. PF-HAP contains three phases: The login phase, the pre-handover phase, and the handover phase. During the login phase, the *AS*, the home *AP* and *U* share a PMK for the user *U*. Furthermore, the *AS* assigns a random number RMU to the user, *U* to be used as an alternative identity to *U*. PF-HAP preserves the user privacy but *U* is not protected from the traceability. Because RMU is not changed during handover between the different *AP*s. After the home *AP* authenticates *U* in the login phase, it sends an encrypted message in the pre-handover phase to its neighbors contains the important information to help them to authenticate *U* easily and in minimum time. They are RMU, PMK for this user, and the identity of the current home *AP*, IDHMP. In the handover phase, the target home *AP*, TMP can authenticate *U* by determining the PMK which is related to this RMU then follow several steps including decrypting the received ticket from *U*. The used ticket in PF-HAP is symmetrically encrypted which gives the protocol more robustness. However, this protocol uses a single group key for all *AP*s which can cause a security problem, if one of the *AP* is a malicious one. PF-HAP proposed a partial solution to this problem by update the group key periodically.

Second: the public key - based protocols:

This type overcomes the problem of the necessity to involve a third party as in [15, 16] Because the contact with *AS* is a requirement in the symmetric key –based protocols. Moreover, most of the symmetric key –based protocols have a problem with privacy. However, the public key- based protocols suffer from the heavy computation overhead which is not suitable for the limited capability of mobile devices. In the current paper, an authentication protocol for IEEE801.11s using a new method for ticket generation is proposed. This new method in generating the tickets gives the new protocol some characteristics that made it distinguished from its peers. As will be detailed in the next Section.

3. Untraceable Authentication Protocol for IEEE802.11s Standard (UAP for IEEE802.11s)

In our proposed protocol, the network is divided into cells; each cell contains some access points, *AP*s which are the nearer to each other. Each cell C*i* has its cell key, K_{Ci}. These cells intersect with each other in some access points which are called the common access points, *APc* as shown in Figure 2. So, each cell has some common access points

(*APc*) common between itself and its neighbor's cells. The number of *APc* in each cell is more than or equal to the number of the cell's neighbors. Maybe there are more than one *APc* are common for two neighbors cells to overcome the problem if one of these *APc*s fails and goes offline. Any common access point knows the two cell keys for the two cells in which this *APc* is a member in both of them. In our proposed protocol, the Authentication server, *AS* does the following jobs:

1. Divides the network into suitable group networks, each group of networks called a cell.
2. Updates the cell keys and distribute them to the different cells.
3. Authenticates the users for their first login in to the system.
4. Issues the first ticket for the user.
5. In case of *APc* fails and goes offline, *AS* can replace it and does its work; where the user communicates with *AS* if the expected response from any *AP* is delayed.
6. In case of any illegal operation for the user, *AS* can trace the movement of *U*.

Note that: *AS* is the only one that can trace the movement of the users by a complex method as will be described later. The *APc* has the responsibility of issuing new tickets for the users when they leave their current cell *Cx* to another cell *Cy*, where; *APc* is a shared access point between the two cells *Cx* and *Cy*.

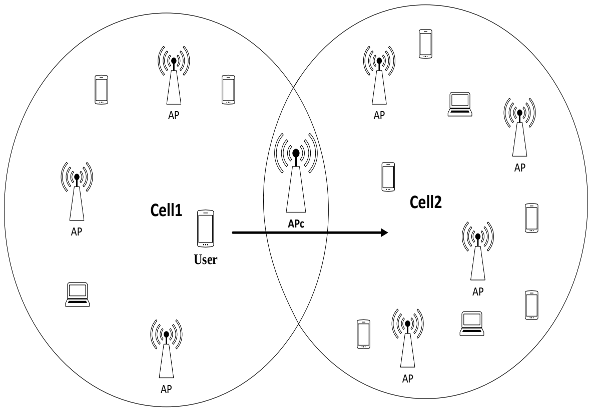


Figure 2. The network model of the proposed protocol.

The new generated ticket is generated using chain ticket derivation function (CTDF) as will be described in detail in the following.

The Chain Ticket Derivation Function (CTDF):

The first ticket, T_{U1} that is dedicated to the user, *U* by *AS* is a simple hash function *H*. Its input parameters are a random number R_U, the expiration time of the ticket t_{exp}, the cell key for the first cell the user *U* will enter, K_{C1}. Equation (1) presents the generation of T_{U1}. Then, the generation of the next ticket as in Equation (2) is based on the chain concept as shown in Figure. 3.

$$T_{U1} = H(R_U, t_{exp}, K_{C1}) \tag{1}$$

$$T_{Ui+1} = H(T_{ui}, \mathrm{t}_{exp}, K_{Cn}) \tag{2}$$

Where *n* is the number of the current network cell, *i* is the number of tickets which is dedicated to the user *U*.

So, the new ticket is the output of the hash function for the previous ticket with the expiration time of the ticket with the current cell key. Note that the identity of the user is not included in issuing the new ticket. Only the previous ticket which satisfies complete privacy and prevents traceability for users.

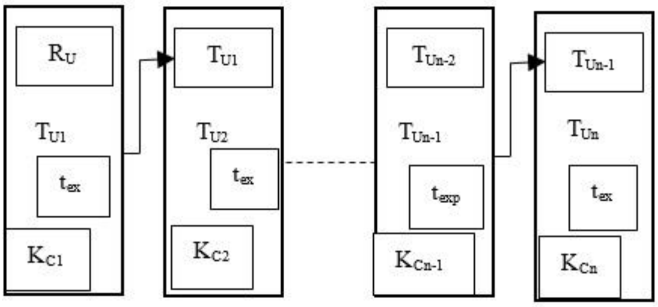


Figure 3. The network model of the proposed protocol.

The proposed protocol is divided into two phases, the login phase, and the handover phase. The handover phase is divided into two types of handover. The first type is the Intra-Domain handover. The second type is the Inter-Domain handover.

First the login phase:

After the mobile user, *U* finishes the EAP [17] full authentication with the *AS* server, the MSK, 512 bit is generated. A PMK is derived from the MSK when the user *U* logins to the system for the first time. Following are the explanation of this phase and the contents of the messages with their orders to complete this phase as shown in Figure.4.

- A user *U* sends to *AS* through their secure channel to join the network using his identity. *AS* assigns a random number R_U to *U*.
- *AS* generates the first ticket for *U*, T_{U1} as in Equation (3) with an expiration time, t_{exp}.
- *AS* sends to *U* a message as in Equation (4). *U* stores his R_U and T_{U1}.
- *AS* stores in its database R_U, *C1*, t_{exp}.
- *AS* does *AP*'s work in case of its failure. But, the authentication time will increase. *U* sends to *AS* his R_U and T_{U1} if he has failed to be authenticated.

$$T_{U1} = H_{PMK_{\mathrm{U}}}(R_U, \mathrm{t}_{exp}, K_{\mathrm{C1}}) \tag{3}$$

Where $H_{PMK_{\mathrm{U}}}$ is the hash function using $\mathrm{PMK_U}$, K_{C1} is the cell key for cell number 1, *C1*. *C1* is the cell network which *U* is going to enter its region area, and t_{exp} is the expiration time for the ticket T_{U1}, and the current time. *U* uses this ticket during his time in *C1* even if he changes the access point.

$$AS \rightarrow U: R_U, T_{U1} \tag{4}$$

AS sends T_{U1}, t_{exp} and PMK encrypted by K_{C1} to all the *AP*s exist inside *C1* as in Equation (5).

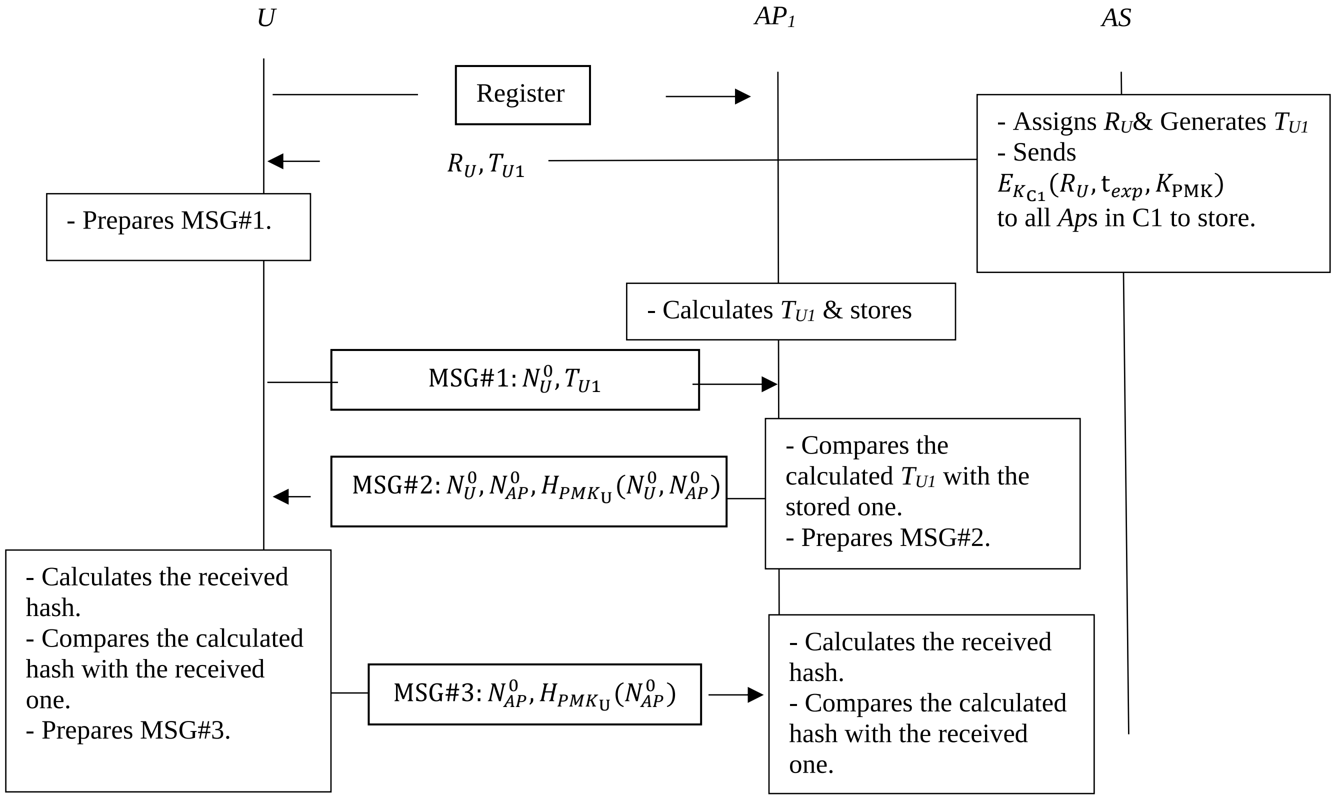


Figure 4. The Login phase.

$$\mathrm{AS} \rightarrow C1: E_{K_{\mathrm{C1}}}(T_{U1}, \mathrm{t}_{exp}, PMK) \quad (5)$$

All the *AP*s inside *C1* prepares themselves to authenticate *U* when arrives by decrypting Equation (5) using K_{C1} and get T_{U1}, t_{exp}, and PMK. Then, all the *AP*s inside *C*1 store these data in their database for certain time period T_s. T_s represents the valid time where *U* allowed to enter *C1* without the need to repeat the same steps. As T_s increases the memory allocated for storing these data increases but *U* will have the ability to switch between two cells without repeat steps for a longer period time.

Now, *U* is ready to be authenticated by any *AP* inside *C1* by the following steps:

U sends to *AP1* (the first access point for *U*) MSG#1 as in Equation (6).

In case of the received ticket equals the stored ticket, AP_1 prepares MSG#2 and sends it to *U*. Otherwise, AP_1 closes the communication.

After *U* receives MSG#2, *U* calculates the received hash value using his $\mathrm{PMK_U}$. If the received hash value equals the calculated value, *U* authenticates AP_1 and sends MSG#3 to AP_1. Otherwise, *U* closes this communication.

After AP_1 receives MSG#3, AP_1 calculates the received hash value using $\mathrm{PMK_U}$. If the received hash value equals the calculated value, AP_1 authenticates *U*. Otherwise, AP_1 closes the communication with *U*.

$$\text{MSG\#1: } U \rightarrow AP_1: N_U^0, T_{U1} \quad (6)$$

$$\text{MSG\#2: } AP_1 \rightarrow U: N_U^0, N_{AP}^0, H_{PMK_{\mathrm{U}}}(N_U^0, N_{AP}^0) \quad (7)$$

$$\text{MSG\#3: } U \rightarrow AP_1: N_{AP}^0, H_{PMK_{\mathrm{U}}}(N_{AP}^0) \quad (8)$$

When *U* ends his time in the region of AP_1 and wants to move to another *AP*, AP_2 for example, he has to make an Intra-Domain handover operation. When *U* ends his tour inside *C1* and wants to move to another cell, he has to make an Inter-Domain handover operation. In the following, the description of the two types of handover operation will be presented:

Second the handover phase:

The Intra-Domain handover: In this handover, *U* moves to a new *AP* in the same cell. Then, both of *U* and the new *AP* follow the same steps in the login phase. *U* and the new *AP* exchange messages similar to MSG#1, MSG#2, and MSG#3 that were shown in Equations (6, 7, and 8).

The Inter-Domain handover: It is the handover from a cell to another cell. Before this type of handover happens, it's expected that *U* enters an area for a common *AP* (*AP*c) in his current cell. *U* sends MSG#1 as in Equation (9) which is similar to MSG#1 in intra domain handover. The steps of this type of handover are presented by the Equations (9, 10, and 11). In this case, *AP*c sends to *U* a new ticket T_{U2} during the normal steps for mutual authentication between each other as in Equation (10). Also, *AP*c has to send to its neighbors in the same new cell the required information to authenticate *U* when arrives. *AP*c sends this data symmetrically encrypted as in Equation (12). When *AP*c's neighbors receive these data, they decrypt the message and store the result in their database for later use. *AP*c sends a new ticket for each new user enters its region. *AP*c can check if this is a new user from the previous stored data.

$$\text{MSG\#1: } U \rightarrow AP_{C1}: N_U^1, T_{U1} \quad (9)$$

$$\text{MSG\#2: } AP_{C1} \rightarrow U: N_U^1, N_{MP}^1, T_{U2}, H_{PMK_{\mathrm{U}}}(N_U^1, N_{AP}^1, T_{U2}) \quad (10)$$

$$\text{MSG\#3: } U \rightarrow AP_{C1}: N_U^1, N_{AP}^1, H_{PMK_{\mathrm{U}}}(N_U^1, N_{AP}^1) \quad (11)$$

$$AP_{C1} \to C2: E_{K_{\mathrm{C2}}}(T_{U2}, \mathrm{t}_{exp}, K_{\mathrm{PMK}}) \tag{12}$$

The previous procedures will be considered as the first Inter-Domain handover operation for *U*. So, *U* can be authenticated easily by any *AP* inside the new cell, *C*2. In the following, a description of the two types of handover is presented as a general case.

General Intra-Domain handover operation procedures:

The following procedures will be considered as the general case for any Intra-Domain handover operation for *U* as shown in Figure. 5.

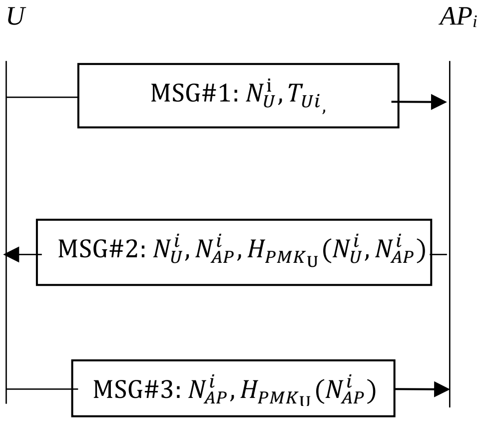


Figure 5. General Intra-Domain handover operation procedures.

(1) *U* sends to *AP*i (*AP*i is not a common access point) MSG#1 as in Equation (13).

(2) In case of the stored T_{Ui} equals the received T_{Ui}, *AP*i prepares MSG#2 and sends it to *U* as in Equation (14). Otherwise, *AP*i closes the communication.

(3) After *U* receives MSG#2, *U* calculates the received hash value using his PMKU. If the received hash value equals the calculated value, *U* authenticates *AP*i and sends MSG#3 to *AP*i as in Equation (15). Otherwise, *U* closes this communication.

(4) After *AP*i receives MSG#3, *AP*i calculates the received hash value using PMKU. If the received hash value equals the calculated value, *AP*i authenticates *U* Otherwise, *AP*i closes the communication with *U*.

$$\text{MSG\#1: } U \to AP_{\mathrm{i}}: N_U^{\mathrm{i}}, T_{Ui} \tag{13}$$

$$\text{MSG\#2: } AP_i \to U: N_U^i, N_{AP}^i, H_{PMK_{\mathrm{U}}}(N_U^i, N_{AP}^i) \tag{14}$$

$$\text{MSG\#3: } U \to AP_{\mathrm{i}}: N_{AP}^i, H_{PMK_{\mathrm{U}}}(N_{AP}^i) \tag{15}$$

General Inter-Domain handover operation procedures:

Also, generally, we can say that if *U* enters into an area for a common *AP* *AP*ci, This *AP*ci has to send a new ticket T_{Ui+1} to *U* as in Equation (17) and stores this new ticket with this user's data. Also, this *AP*ci has to send a message to other APs in the new cell (*C*i+1) and the old cell *C*i too (because *U* can keep residence in *C*i and not move to *C*i+1) as in Equation (19) and updates its stored ticket for this user. *U* follows the normal procedures as in Equations (16 and 18) and as shown in Figure. 6.

When *U* receives a new ticket from *AP*ci, he will use the new ticket T_{Ui+1} in his next handover authentication. If *Ui* still exists inside the same *AP*ci after updating its ticket to the new ticket T_{Ui+1}. *AP*ci can still authenticate it. But, as a member in the neighbor cell *Ci*+1 because *AP*ci updates its data too and waits for *Ui* as other *AP*s in *Ci*+1. The previous procedures will be considered as the general case for any Inter-Domain handover operation for *U*. So as we can see there's a high level of privacy for the users and this may cause problems later in case of any illegal operation issued from these users. So there must be a suggested solution to restore the previous movement of users in this special case.

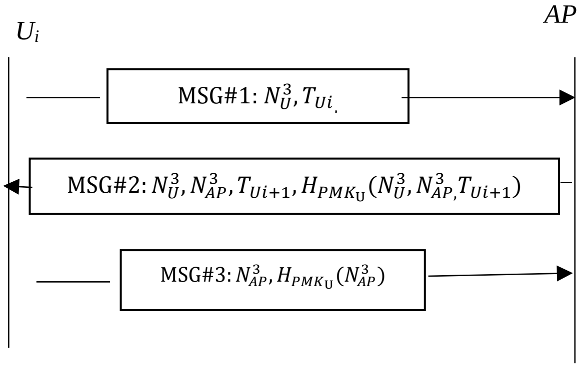


Figure 6. General Inter-Domain handover operation procedures.

$$\text{MSG\#1: } U \to AP_{\mathrm{Ci}}: N_U^3, T_{Ui} \tag{16}$$

$$\text{MSG\#2: } AP_{Ci} \to U: N_U^3, N_{AP}^3, T_{Ui+1}, H_{PMK_{\mathrm{U}}}(N_U^3, N_{AP}^3, T_{Ui+1})$$
(17)

$$\text{MSG\#3: } U \to AP_{\mathrm{Ci}}: N_{AP}^3, H_{PMK_{\mathrm{U}}}(N_{AP}^3) \tag{18}$$

$$AP_{Ci} \to (C_i, C_{i+1}): E_{(K_{\mathrm{Ci}}, K_{\mathrm{Ci+1}})}(T_{Ui+1}, \mathrm{t}_{exp}, K_{\mathrm{PMK}}) \tag{19}$$

A suggested solution to restore the previous movement of users:

As we know, *AS* knows the cell key, K_C for the different cells in the different periods, $t_0, t_1, \dots t_n$. So *AS* can build a table like Table 1 easily.

Table 1.

Cell number	Time period	The corresponding Cell Key
C0	t_0	K_{C0}
C1	t_1	K_{C1}
. .		. .
Cn	t_n	K_{Cn}

AS stores in its database the construction of the network. The construction of the network contains the distribution of the common points inside the different cells. Any *AP*c stores in its database the new generated tickets with the generation time for these tickets for a certain period. So, *AS* can restore these data from these common points and builds table as Table 2 easily. The third column in Table 2 is the generated tickets by each *AP*c. For example, $T_0, T_1, \dots$.. are the generated tickets by *AP*c0. The fourth column in Table 2 is the t_{exp} for these generated tickets in order. For example, t_{exp0} is the expiration time for T_0, and t_{exp1} is the expiration time for T1, etc.

As *AS* stores in its database R_U, *C*1, t_{exp} for *U*, so from Equation (1) *AS* can calculate the first ticket for *U*, T_{U1}. Then from Table 1, Table 2, and Equation (2) *AS* can calculate the calculated tickets for certain user *U*. Then from Table 1, *AS* can determine the path for this user. For Example, assume that *U* was in *C*1, and *C*1 has a common *AP* with *C*0 and *C*2 only for simplicity. *U* may be going to move towards *C*0 or *C*2. Therefore, to calculate the next ticket for *U*, T_{U2}, *AS* once uses K_{C0} to calculate T_{U2}, then searches in Table 2 for the generated tickets by *AP*c0, and once uses K_{C1} to calculate T_{U2} and searches in Table 2 for the generated tickets by *AP*c1 if *AS* does not find T_{U2} in the generated tickets by *AP*c0, and so on in the other cases. So from the previous analysis, *AS* is the only part which has the authority to determine the path of users.

Table 2.

Cells	Common access points	The generated tickets	$t_{exp:}$ The time of generation & the expiration time
*C*0 & *C*1	AP_{c0}	T_0, $T_{1,....}$	t_{exp0}, $t_{exp1,....}$
*C*1 & *C*2	AP_{c1}	T_2, T_3, $T_{4,.....}$	t_{exp2}, t_{exp3}, ….
. .	. .	. .	. .
*C*n-1& *C*n	…..	T_{m-1}, T_m, T_{m+1}	$t_{exp,m-1}$, $t_{exp,m}$, $t_{exp.m+1}$

4. Security analysis and verification

In this Section, security analysis and verification of the proposed protocol are presented.

4.1 Security analysis

1) Mutual Authentication

The user *U* gets his required PMK_U after he has finished the registration phase with the *AS*. *AS* and the shared *AP*c send this PMKU to the other *AP*c in the cell encrypted with cell key. The proposed tickets in our protocol can be calculated only by the legitimate *AP*c. The new *AP* can authenticate *U* by recalculating the ticket using the correct K_C and the correct PMK_U. So illegitimate users can't send MSG #1, and MSG #3 in our proposed protocol because they don't know PMK_U. Also, illegitimate *AP*c can't send MSG#2 in correct form, because they don't know PMK_U. So from previous, both of *U* and *AP* have a mutual authentication with each other.

2) Privacy

In our scheme, the user roams inside the network using his ticket, and this ticket is changed for each new cell. So the user privacy is preserved in our protocol.

3) Replay attack

The intruder in the replay attack interprets the message and resends it to the receivers to persuade them that this message was transmitted from the legal sender [18]. Assume an attacker can catch MSG#1 and resend it in another time, it will be impossible for him to prepare MSG#3, which is considered an important step to complete the authentication phase. PMK_U key is important information to prepare MSG#3 which is unknown according to the illegal users.

4) Denial of service attack

Sometimes the access points receive many spam messages which may cause that these access points don't work with the required efficiency, which is called denial of service attack. In our proposed protocol, the *AP* authenticates the user after MSG#3. But, after *AP* receives MSG#1 in the handover phase, *AP* can verify the correctness of the received ticket by executing a simple hash function. *AP* closes the session with this user if this check fails and does not complete the protocol. So denial of service attack has a less effect in our proposed protocol.

5) Domino effect

During the roaming of the user inside the network, if one of these *AP*c is a compromised *AP*. In some proposed protocols as in [1], the protocol will fail due to the propagation of this error in each handover step. Our proposed protocol has immunity against this type of attack. Because if there's a compromised *AP*, *AP*m and this *AP*m knows the Cell key, K_C for a certain cell, this problem will be solved because of two reasons:

1. K_C is changed from cell to cell.
2. K_C for the same cell is updated after each certain time by *AS*.

6) Forgery attack

In our proposed protocol, the ticket is an output of a hash function. The cell key is a requirement to generate the ticket. Because K_C is an input parameter to this hash function as in Equation (2). So the proposed protocol has immunity against the forgery attack.

7) Forward and backward secrecy

This property is satisfied if the adversary can't calculate future session keys or acquire previous ones using a compromised key. In the proposed protocol, if the current K_C for a certain cell is intercepted, the adversary can't detect the new K_C for this cell because it's generated randomly by *AS* then distributed to the corresponding cell. Moreover, K_C is different from cell to cell. So as soon as *U* changes his current cell, K_C will be changed. Other proposed protocols [4] suffer from using the same key for all the *AP* in the network.

8) Fake Access Point attack

It's a type of attack which tries repeatedly to reach user data. This is done by making a broadcast similar to the SSID (Service Set Identifier) by the attacker. Then, the attacker allows the users to communicate with this SSID [19]. Our proposed protocol has immunity against this type of attack because it satisfies mutual authentication between the two shared parties (*U* and *AP*) as mentioned before.

9) Illegal tickets

If any malicious *AP*c calculates an illegal ticket to send it to the other *AP*s in the cell, it will be detected. Because any *AP* before authenticates *U*, it has to calculate the new ticket as in Equation (3) using PMK and the cell key (*Kc*) which is changed from cell to cell and updated by *AS* periodically. Then, it compares the calculated ticket with the received one as described in Section 3.

10) Compromised Access point

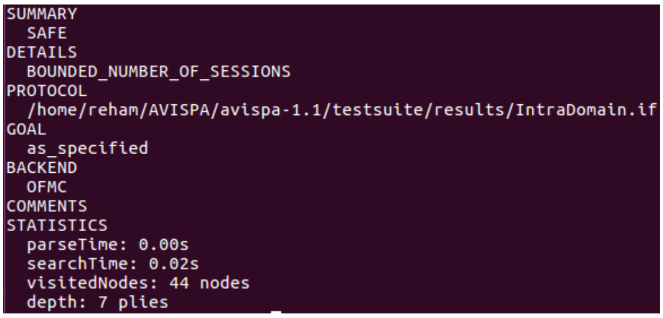
The normal *AP* stores the following information for *U* after authenticates it for Ts time: t_{exp}, PMK, and T_{Ui} after authenticating *U* for Ts time. So, in case *AP* is compromised, the users' stored information will be stolen. However, *U* uses T_{Ui} for his roaming inside only one cell. When *U* moves to another cell, he uses a new ticket T_{Ui+1}which was sent encrypted to other *AP*s as in Equation (12). So the attacker can trace the movement of *U* inside one cell only. When *U* leaves his current cell and goes to another cell, the dedicated *AP*c sends his new ticket to other *AP*s in the new cell encrypted.

11) If a malicious ticket is inserted in the chain

If any malicious *AP*c calculates an illegal ticket and sends it to the other *AP*s in the cell, it will be detected. Because, any *AP* before authenticates *U*, it has to compare the received ticket from the user with the stored ticket. *AP* gets the stored ticket after decrypting the received message from AP_C as in Equation (19) using the cell key (*Kc*) which is changed from cell to cell and updated by *AS* periodically.

4.2 Formal verification using AVISPA tool

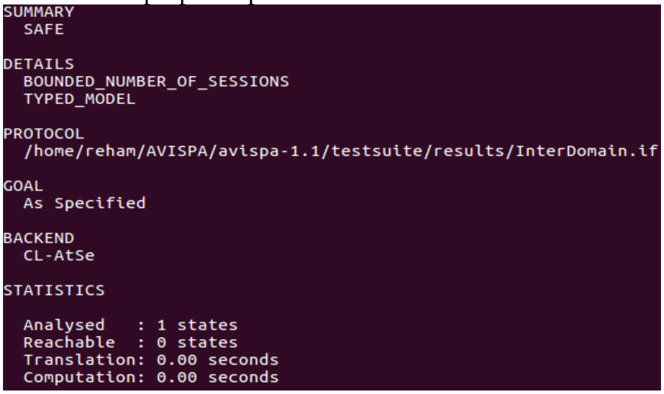
To test the security of our proposed protocol, we use a formal verification based on the Automated Validation of Internet Security and Applications (AVISPA) [20, 21] for the proposed protocol. AVISPA is considered the commonly used formal verification tool by developers and researchers of security protocols. This tool gives the ability to use four different verification methods (backends) without changing the protocol specification. The four backends are Constraint-Logic based Attack Searcher (CL-AtSe), On-the-fly Model-Checker (OFMC), SAT-based Model-Checker (SATMC), and Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP). These four backends present four different techniques for analysis. In our verification using AVISPA, High-Level Protocol Specification Language (HLPSL) is the used language for the description of our security protocols. DoleveYao attacker [22] is the implemented intruder in AVISPA. The model of the DoleveYao intruder gives it the ability to change messages, eavesdrop to messages, interrupt messages, and insert new messages. In our proposed scheme model in HLPSL, there will be four roles: *U*, *AP*, session and, environment, where *U* and *AP* are the basic roles. The basic roles are used to represent the two participants, the user *U* and the access point *AP*, while the session and environment are composition roles. The session role expresses a single session of the proposed protocol, while the environment role expresses the composition of the number of cases of session roles with cases of basic roles, *U*, and *AP* with knowing the presence of the DoleveYao attacker. We use OFMC and TA4SP to test our protocol.



```
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/reham/AVISPA/avispa-1.1/testsuite/results/IntraDomain.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.02s
  visitedNodes: 44 nodes
  depth: 7 plies
```

Figure 7. Test result using OFMC for Intra-Domain Handover.

We tested our protocol in two cases, the inter-domain case and the intra-domain case. The test results are as shown in Figures 6 and 7, the protocol is safe. Figure 6 presents the result of testing the protocol in case of intra-domain using OFMC, while Figure 7 represents testing the protocol in case of the inter-domain using TA4SP. Therefore, the test shows that no revealed attacks like a man-in-the-middle attack, and replay attack in our proposed protocol.



```
SUMMARY
  SAFE

DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
  TYPED_MODEL

PROTOCOL
  /home/reham/AVISPA/avispa-1.1/testsuite/results/InterDomain.if

GOAL
  As Specified

BACKEND
  CL-AtSe

STATISTICS

  Analysed   : 1 states
  Reachable  : 0 states
  Translation: 0.00 seconds
  Computation: 0.00 seconds
```

Figure 8. Test result using TA4SP for Inter-Domain Handover.

4.3 Formal verification using BAN logic

In this Subsection, we will test our proposed protocol using BAN logic [23] to ensure that its functions work correctly before the real implementation. Moreover, BAN logic is useful in verifying authentication protocols [24]. But before present the necessary proof for our protocol, we have to describe the used rules in BAN.

Rules of BAN Logic

Rule 1: the interpretation rule, $\dfrac{P \mid\equiv (Q \mid\sim (X,Y))}{P \mid\equiv (Q \mid\sim X), P \mid\equiv (Q \mid\sim Y)}$

Rule 2: the message meaning rule, $\dfrac{P \mid\equiv P \xleftrightarrow{K} Q, P \triangleleft [X]_K}{P \mid\equiv Q \mid\sim X}, P \neq Q$

Rule 3: the nonce verification rule, $\dfrac{P \mid\equiv \#(X), P \mid\equiv Q \sim X}{P \mid\equiv Q \mid\equiv X}$

Rule 4: the jurisdiction rule, $\dfrac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$

Rule 5: the freshness rule, $\dfrac{P \mid\equiv \#(X)}{P \mid\equiv \#(X,Y)}$

Rule 6: the synthetic rule, $P \mid\equiv (Q \mid\sim X) \rightarrow P \mid\equiv (Q \mid\sim (X,Y))$

Rule 7: (P⊨ (X, Y))/(P⊨(X), P⊨(Y))

The mutual authentication is completed between *U* and *AP*, if for certain data *X*:
AP ⊨ U ⊨ X , AP ⊨ X: they mean that *U* believes that *X* is sent by *AP*; where symbol ⊨ means believes, and for certain data *Y*, AP ⊨ U ⊨ Y, AP ⊨ Y. We will present our verification proof in the intra-domain phase only. Because this is the general case. The target is to satisfy the following four Goals:
Goal 1: AP ⊨ U ⊨ T_{Ui}
Goal 2: AP ⊨ T_{Ui}
Goal 3: U ⊨ AP ⊨ (N_{AP}^2)
Goal 4: U ⊨ (N_{AP}^2)
Our proposed protocol in intra-domain phase (Equations 13, 14 and 15) can be transformed into the following formulas:

$$U \rightarrow AP\text{: } \#N_U^2, T_{Ui} \quad (20)$$

Equation (20) can be written as follows:

$$U \rightarrow AP\text{: } \#N_U^2, \left(T_{ui-1}, \mathrm{t}_{exp}, K_{Cn}\right)_{PMK_U} \quad (21)$$

$$AP \rightarrow U\text{: } \#N_U^2, \#N_{AP}^2, (\#N_U^2, \#N_{AP}^2)_{PMK_U} \quad (22)$$

$$U \rightarrow AP\text{: } \#N_{AP}^2, (\#N_{AP}^2)_{PMK_U} \quad (23)$$

The following initial assumptions are necessary to complete our test:

$$AP \mid\equiv U \xrightarrow{\text{PMK}} \text{AP} \quad (24)$$

$$U \mid\equiv AP \xrightarrow{\text{PMK}} \text{U} \quad (25)$$

$$AP \mid\equiv\ ^{\#}\ \mathrm{t}_{exp} \quad (26)$$

$$U \mid\equiv\ ^{\#}\ N_{AP}^2 \quad (27)$$

$$AP \mid\equiv \text{U} \Rightarrow T_{Ui} \quad (28)$$

$$U \mid\equiv \text{AP} \Rightarrow N_{AP}^2 \quad (29)$$

Using Equation (21) and Equation (24) and after applying the message meaning rule, we obtain:

$$AP \models U \mid\sim (T_{ui-1}, \mathrm{t}_{exp}, K_{Cn}) \quad (30)$$

Using Equation (23) and Equation (25) and after applying the message meaning rule, we obtain:

$$U \models AP \mid\sim (\#N_{AP}^2) \quad (31)$$

Using Equation (30) and applying the interpretation rule, we obtain:

$$AP \models U \mid\sim (T_{ui-1}, \#\ \mathrm{t}_{exp}) \quad (32)$$

Using Equation (26 and 32) and applying the freshness rule, we obtain:

$$AP \models \#(T_{ui-1}, \mathrm{t}_{exp}) \quad (33)$$

(34)

Using Equation (33 and 32) and applying the nonce verification rule, we obtain:

$$AP \models U \models (T_{ui-1}, \#\ \mathrm{t}_{exp}) \quad (34)$$

From Equation (34) and from rule 7

$$AP \models U \models (T_{ui-1}) \quad (35)$$

From Equation (35, and 28) and from the jurisdiction rule, we obtain:

$$U \models (T_{ui-1}) \quad (36)$$

Using Equation (27 and 31) and applying the nonce verification rule, we obtain:

$$U \models AP \models (\#N_{AP}^2) \quad (37)$$

From Equation (37) and from rule 7

$$U \models AP \models (N_{AP}^2) \quad (38)$$

From Equation (29, and 38) and from the jurisdiction rule, we obtain:

$$U \models (N_{AP}^2) \quad (39)$$

So from previous Equations Goals (1, 2, 3, and 4) are satisfied from Equations (35, 36, 38, and 39) respectively. So we can say that our proposed protocol works probably, free from any redundancy and free from any type of known attacks. Table 3 is a table of comparison between our proposed protocol and the most similar authentication protocols in the literature according to the used formal verification tool to test each one of them.

Table 3. Verification tool comparison with other similar schemes

	The scheme proposed in [8]	The scheme proposed in [25]	The scheme proposed in [26]	The scheme proposed in [4]	Our proposed protocol
Verification tool	AVISPA	AVISPA	BAN Logic	AVISPA	AVISPA & BAN Logic

5. Performance Analysis

In the current section, we shall present the performance of our proposed protocol by measuring some important parameters and show how it compares with other similar protocols. The selected similar protocols will be EAP-TLS [17], Anmin Fu.et al's protocol [8], Li.et al's protocol [1], and PF-HAP [4]. EAP-TLS is the standard authentication protocol in IEEE 802.11-based wireless networks. We will divide the performance measurements into two main performance parameters: the computation overhead, and the communication overhead.

5.1 Computation Overhead

The computation overhead represents the time consumption of the cryptographic operations for the two shared entities, *U* and *AP* in our case. The required cryptographic operations to complete our analysis will be public-key encryption (Epub), public key decryption (Dpub), generation of digital signature (Gsig), verification of digital signature (Vsig), calculation of MAC function (MAC), calculation of hash function (H), symmetric key encryption (Es), and symmetric key decryption (Ds), calculation of truncate function (Dot) and calculation of dot function (Tr). We used the experimental results which are presented by Long and Wu. in [27] to estimate the processing time for these cryptographic operations as shown in Table 4. However, Long and Wu didn't include the processing time of Dot and Tr. We will use the assumption that was presented in [25], that: Dot Equals *H*, and Tr is neglected.

Table 4. The processing time for various cryptographic operations [27].

Cryptographic operation	Used algorithm	Processing Time (in s)
H	SHA-2	0.009 *10-3
MAC	HMAC	0.015 *10-3
Es	AES	2.1 *10-3
Ds	AES	2.2 *10-3
Epub	RSA	1.42 *10-3
Dpub	RSA	33.3 *10-3
Gsig	ECDSA	11.6 *10-3
Vsig	ECDSA	17.2 *10-3

From Table 4 and by determining the number of different types of cryptographic performed operations by the selected protocols we can build Table 5. But, since the user only needs to run the login phase one time at the start, we will neglect the login phase in our comparison. In our comparison, we have presented the computation overhead in the case of intra-domain handover operation and inter-domain handover operation. We can observe from Table 5 that the computation overhead of the proposed scheme is the lowest one compared with other relatively similar schemes under comparison. This shows that the proposed scheme has an excellent efficiency. It's more applicable for real time applications.

Table 5. Performance comparison with other similar schemes.

	EAP-TLS	the scheme proposed in [8]	the scheme proposed in [1]	the scheme proposed in [4]	Ours	
					Intra-domain handover	Inter-domain handover
Computation overhead	Gsig +3Vsig+ Epub+Dpub++3H	Es+Ds+ 5MAC +2H+7Dot	6MAC	6H+ Ds	4H	5H
No. of messages	9	5	3	3	3	3
Processing Time (Sec)	97.962 $*10^{-3}$	4.44 $*10^{-3}$	0.09 $*10^{-3}$	2.25 $*10^{-3}$	0.036 $*10^{-3}$	0.045 $*10^{-3}$
Handover Delay Time (Sec)	(97.962+9dh) $*10^{-3}$	(4.44+5d) $*10^{-3}$	(0.09 +3d) $*10^{-3}$	(2.25 +3d) $*10^{-3}$	(0.036 +3d) $*10^{-3}$	(0.045 +3d) $*10^{-3}$

5.2 Communication Overhead

The communication overhead is estimated by the number of mutual messages between the two shared entities (*U* and *AP*) in the handover phase. To measure this type of overhead, we will need two extra parameters *d* and *h*. (*d*) is the average delay caused by one message through one-hop of transmission, and *h* represents the number of hops between the two shared entities. We will use the parameter *h* in the EAP-TLS protocol only. Because, it's a multi-hop protocol, which means that it's the only one that requires communication between *U* and *AS*.

6. Conclusions

Nowadays, development the methods of communications become an urgent requirement. Especially, with the current hard situation which the world faces by Corona virus. Therefore, the target of our paper is to improve the capabilities of IEEE802.11s standards to provide fast hand over for real-time applications such as video conference, distance learning, and VoIP with user privacy preservation. The presented performance analysis demonstrates that our protocol outperforms similar previously proposed protocols in computation and communication cost. Moreover, the presented security analysis shows that the proposed protocol has an immunity against various types of electronic attacks. A formal verification test is performed for the proposed protocol using AVISPA tool and BAN logic. The result of this test declares that the presented protocol is safe against various types of known attacks and achieves mutual authentication between the shared parties.

7. Acknowledgement

This work was supported by National Institute of Standards (NIS), EGYPT.

References

[1] Celia Li, Uyen Trang Nguyen, Hoang Lan Nguyen, Nurul Huda, "Efficient authentication for fast handover in wireless mesh networks", Computers & Security, Vol. 37, pp. 24-42, 2013.

[2] B. Aboba and D. Simon, "PPP EAP TLS authentication protocol", RFC 2716, 1999.

[3] Jaydip Sen, "Security and Privacy Issues in Wireless Mesh Networks: A Survey", Chapter · February 2013 DOI: 10.1007/978-3-642-36169-2_7 · Source: arXiv

[4] Reham A. Abouhogail, Mohammed S. Gadelrb, "A New Secure and Privacy Preserved Protocol for IEEE802.11s Networks", Computers & Security, Vol. 77, pp. 745-755, Aug. 2018.

[5] A. Alabdulatif, X. Ma, and L. Nolle, "Analysing and attacking the 4-way handshake of IEEE 802.11i standard", 2013 8th International Conference for Internet Technology and Secured Transactions, ICITST 2013, London, UK, pp. 382–387, 2013.

[6] Farooq Ahmed, Zain ul Abedin Butt, Asif Hussain Khan, Jabar Mehmood, et. al., "Wireless Mesh Network IEEE802.11s", International Journal of Computer Science and Information Security (IJCSIS), Vol. 14, No. 12, pp.803-809, December 2016.

[7] IEEE Std. 802.11-2012, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", 2012.

[8] A. Fu, Y. Zhang, Z. Zhu, Q. Jing and J. Feng, "An efficient handover authentication scheme with privacy preservation for IEEE 802.16m network", Computers and Security, pp. 741-749, June 2012.

[9] Huang CM, Li JW, "A cluster-chain-based context transfer mechanism for fast basic service set transition in the centralized wireless LAN architecture", Wirel. Commun. Mob. Comput., Vol. 9, No. 10, pp.1387-1401, Oct. 2009.

[10] Rafa ML, Fernando PG, Yoshihiro O, Fernando BH, Antonio FG, "A kerberized architecture for fast re-authentication in heterogeneous wireless networks", Mobile Networks and Applications, 2010a, Vol. 15, No. 3, pp. 392-412.

[11] Rafa ML, Yoshihiro O, Fernando PG, Gomez AF, "Analysis of handover key management schemes under IETF perspective", Computer Standards & Interfaces, Vol. 32, pp. 266-273, Oct. 2010.

[12] Zheng X, Sarikaya B. Handover keying and its uses. IEEE Netw Mar. 2009;23(2), pp27-34.

[13] IEEE. Part11: wireless medium access control (MAC) and physical layer specifications: medium access control (MAC) security enhancement 2003. IEEE Standard 802.11i/D10.0.

[14] Yan-Ming Lai, Pu-Jen Cheng, Cheng-Chi Lee, Chia-Yi Ku, "A New Ticket-Based Authentication Mechanism for Fast Handover in Mesh Network", PLOS ONE | DOI: 10.1371/journal.pone.0155064, Vol. 11, No.5, pp.1-18, 12 May 2016.

[15] Zhang C, Liu R, Ho PH, Chen A., "A location privacy preserving authentication scheme in vehicular networks", In: Proc. WCNC 2008, Las Vegas, NV, USA, pp. 2543-2548, Mar. 2008.

[16] Cai L, Machiraju S, Chen H, "CapAuth, "a capability-based handover scheme", In: Proc. INFOCOM 2010; San Diego, CA, USA, Mar. 2010.

[17] D. Simon, B. Aboba, R. Hurst, The EAP-TLS authentication protocol, 2008.

[18] Mahmoud Rajallah Asassfeh, Nadim Obeid, and Wesam Almobaideen, "Anonymous Authentication Protocols for IoT based-Healthcare Systems: A survey", International Journal of Communication Networks and Information Security (IJCNIS), Vol. 12, No. 3, pp. 302-314, Dec. 2020.

[19] F. KILINÇER, F. ERTAM and A. ŞENGÜR, “Automated Fake Access Point Attack Detection and Prevention System with IoT Devices”, BALKAN JOURNAL OF ELECTRICAL & COMPUTER ENGINEERING, Vol. 8, No. 1, pp. 50-56, January 2020.

[20] AVISPA v1.1, http://www.avispa-project.org/. Access on 8 July 2020.

[21] Vigano` L., “Automated security protocol analysis with the AVISPA tool”, Electrical Notes on Theoretical Computer Science, Vol. 155, pp.61-86, 12 May 2006.

[22] Dolev D, Yao A., “On the security of public key protocols”, IEEE Transactions on Information Theory; Vol. 29, No. 2, pp.198-208, March. 1983.

[23] Burrows, M.; Abadi, M.; Needham, R.,”A logic of authentication”, ACM Trans. Computer Systems, Vol. 8, No. 1, pp. 18–36, 1990.

[24] Reham Abdellatif Abouhogail, “A Comparative Analysis of Tools for Testing the Security Protocols”, I.J. Information Technology and Computer Science, Vol. 12, pp. 30-37, Dec. 2019.

[25] Xu L, He Y, Chen X, Huang X., “Ticket-based handoff authentication for wireless mesh networks”, Computer Network, Vol. 73, pp. 185–94, 2014.

[26] Reham Abdellatif Abouhogail, "Improving the Handoff Latency of the Wireless Mesh Networks Standard", International Journal of Security and Its Applications, Vol. 10, No. 5, pp.73-86, 2016.

[27] Long M, Wu CHJ., “Energy-efficient and intrusion-resilient authentication for ubiquitous access to factory floor information”, IEEE Transactions on Industrial Informatics.; Vol. 2, No. 1, pp.40–7, 2006.