ANNEXURE A

PPSE: Privacy Preservation and Security Efficient AKA Protocol for 5G Communication Networks

Balu L. Parne[1], Shubham Gupta[2], Kaneesha Gandhi[3], Shubhangi Meena[4]
[1,3,4]*Department of Computer Engineering, Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, Gujrat, India.*
[2]*Department of Information Security and Communication Technology, Norwegian University of Science and Technology, Norway.*
[1]blparne@coed.svnit.ac.in, [2]shubham.gupta@ntnu.no, [3]kaneesha98@gmail.com, [4]shubhangimeena.cse@gmail.com

Abstract—**The authentication and key agreement (AKA) protocol strengthens the security of mobile communication networks. The 5G-AKA protocol is standardized by the 3rd generation partnership project (3GPP) for future mobile communication networks. However, it is observed that this protocol is vulnerable to numerous security weaknesses. Moreover, the protocol suffers from huge computation and communication overhead. To overcome these issues, several AKA protocols were introduced by the researchers. However, none of the protocols took care of the identity preservation and single key disclosure issue over the communication channel. In this article, we propose the Privacy Preservation and Security Efficient (PPSE-AKA) protocol that preserves the user's identity and protects the shared secret key. The mutual authentication is formally verified by using the AVISPA tool. The security analysis demonstrates that the protocol fulfills all the privacy requirements and dodges the potential attacks. The performance of the protocol is evaluated for the previously established schemes and observed that the PPSE-AKA protocol exhibits a cutting edge competition to them.**

Index Terms—**Authentication, AVISPA, Communication Overhead, Privacy Preservation, Symmetric Key.**

I. Introduction

The 5G is a novel universal wireless technology after 1G, 2G, 3G, and 4G networks. The 5G is considered the future of technological power that is designed to give people new experiences in terms of high multi-Gbps peak data speeds, more reliability, low latency, and more uniform user interaction. From just calling to sending messages and from sending messages to sending videos, audio, mail, etc. over an Internet, technology evolved to a great extent. The whole network is highly robust that virtually connects everything including machines, objects, and devices. Along with continuous evolution of the 5G technology integration with more and more devices, it is not surprising that LTE-Advanced will continue to change in a backward-compatible manner to achieve the best performance. This would in turn maximize benefit from the massive economies of scope established around the 3GPP LTE/LTE-Advanced ecosystem [1], [2].

To protect the privacy and integrity of users involved in communications across these networks, 3GPP designed the AKA protocol that mutually authenticates a device consisting of the Universal Subscriber Identity Module (USIM) card and established keys to encrypt these communications [3], [4]. However, taking into consideration the attacks and breaches involved with the security of the various AKA protocols and their versions, 5G networks are vulnerable to the International Mobile Subscriber Identity (IMSI)-Catcher Attack [5]. Hence, it is recommended to re-approach the 5G-AKA scheme for implementing the identity preservation of the user equipment/ mobile device (UE/MD) during the communication process.

A. Shortcomings of the existing 5G-AKA scheme

The existing 5G-AKA scheme effectively uses the challenge-response scheme to prevent the network from malignant cellular attacks. In this, four communication units are thoroughly involved: UE, Authentication Credential Repository and Processing Function (ARPF), Security Anchor Function (SEAF), and Authentication Server Function (AUSF). The Home Network (HN) maintains residency for both AUSF and ARPF whereas the SEAF performs inside the Serving Network (SN) [6]. The 5G-AKA scheme achieves most of the security requirements. However, there are some of the security weaknesses found in the technical report [7] which are as follows:

1) All the mobile communication technologies are vulnerable to the IMSI-catcher attack [5], [8]. This attack occurs due to the transmission of permanent identity merely as plain text. Suppose, a temporary identity is transmitted, an adversary may directly dispatch a Permanent-ID-Request to receive the original identity of UE.
2) An attacker may implant the bogus base station and enforce the UE to identity request by adopting the previous cellular communication (3G/4G). Until the entire systems enhancement to the 5G, nothing much can be fixed to handle the identity privacy-preservation.
3) The UE and ARPF implement the sequence number (SQN) between them. But if an attacker makes a false registration then synchronization problem will occur in the existing protocol [9].
4) There is no implicit scheme to achieve freshness for the ARPF/AUSF. Thus, the protocol is susceptible to a replay attack. An attacker may re-transmit a formerly encrypted IMSI to the ARPF/AUSF and investigates for several responses. Based on this, a device whose IMSI is undiscovered to the adversary can be traced.

Considering all the above issues with the existing 5G-AKA scheme, it is needed to revisit the existing approach and design an efficient & secure protocol for the 5G communication network.

B. Contribution and Approach

In this paper, we have introduced a protocol that preserves the privacy of the user identity and overcomes all the above-identified problems of the existing 5G-AKA protocol. The main contribution of the works are summarized as follows:

- The basic architecture of the 5G-AKA protocol is preserved while designing the proposed solution. There is no need for any additional physical devices to be added to the existing infrastructure.
- A shared symmetric key is applied between the communication entities over a network. So, this approach supports the energy-efficient devices in the communication network.
- All the communication entities are mutually authenticated with each other so it helps to overcome the man in the middle (MitM), redirection, DoS, and replay attacks.
- The formal verification of the protocol is carried out using the Automated validation of Internet Security protocol Application (AVISPA) tool. It demonstrates that the protocol fulfills all the security requirements and avoids the identified attacks.
- The proposed protocol transmits small size packets and uses a minimum number of functions which helps to reduce the communication and computation overhead on the communication entities.

Rest of the article is arranged as follows: Related work is highlighted in Section II. Section III includes the PPSE-AKA protocol for the 5G networks. Section IV describes the security analysis of the PPSE-AKA protocol. Section V illustrates the performance evaluation. Section VI concludes the article.

II. Related Work

The security & privacy in the 5G communication network is essential and it is widely used in the IoT based applications nowadays. So, this engaged several academicians/ researchers to participate in this area and designed the AKA protocols.

Author claims the vulnerability in the 5G-AKA scheme such as DoS, impersonation, and MitM attacks [9]. In response to this, the authors of [9] established an efficient and secure protocol in 5G network. The goal of this scheme is to accomplish mutual authentication between the user entity and home network, and to ensure the fulfillment of IoT's security demands. This scheme, however, could not eliminate the impersonation attack which was observed due to the clear exchange of security parameters. This leads to the possibility of impersonating the UE and hence to the subsequent attacks. Further, Gharsallah et al. in [10], attempts to launch a revised version of the 5G-AKA protocol. This scheme is particularly applied for machine-type communications (MTC) devices where each UE needs to have a unique identity. This protocol enables UE to share a secret key and reference parameter with full trust on HN. However, the protocol suffers from privacy-preservation as the device identities are clearly transmitted in the air that leads to numerous security attacks.

Liu et al. published a fresh authentication protocol by making use of the Diffie-Hellman key exchange algorithm for generating the session key [11]. This scheme was successful in preventing link-ability attacks along with a MitM attack. However, there is a problem of high computation overhead due to the usage of public-key cryptosystem. In addition to this, synchronization loss may lead to key disagreement problems faced by the AUSF and UE. A novel 5G authentication protocol was proposed by Braeken et al. in order to prevent active attacks and gain resistance against malignant serving networks [3]. Unfortunately, there is a possibility of having SN impersonation so, this scheme does not eliminate the vulnerability towards the MitM attack.

In correspondence to the issues, we establish the PPSE-AKA protocol which identifies the preservation of user privacy by securing its identity. We use a symmetric key based mechanism between UE, SN, and HN. The mutual authentication is verified by using the AVISPA tool.

III. Proposed Protocol

A. Registration Phase

UE_i and HN have the shared secret key K_i. First, the UE_i sends the request number (RN_i) to the HN. HN receives the request number and responds to the UE_i with *simcode* $Simcode_{iRand}$. It is assumed that the $Simcode_{iRand}$ is generated and stored at AuC when a USIM card gets triggered. The HN stores the $Simcode_{iRand}$ in its database prior to sending it to UE_i. The objective of this code is to check the stored $Simcode_{iRand}$ at HN and retrieve respective key K_i. The $Simcode_{iRand}$ is fixed as a tag to key K_i at AuC. The computation of $Simcode_{iRand}$ at HN is not openly available and is private in nature.

B. Authentication and Key Agreement Phase-1

In this phase, the UE requests for authentication to the HN through SN. The message transmission is shown in Fig. 1.

Step-1 ***(UE_i to SN)***

$\{SUCI, MAC_{UE}, Actcode_{iRand}, T_{UE}, V_{UE}, S_{UE}\}$

1) UE_i generates the delegation key (DK) $DK_i = f_1(T_{UE})_{K_i}$; where f_1: HMAC-SHA256 and T_{UE} is the time-stamp of UE_i
2) UE computes the one-time *activation code*

$$Actcode_{iRand} = LCS_n(T_{UE} \oplus Simcode_{iRand}) \quad (1)$$

,where LCS_n is the left circular shift by n, where n is the value of first eight digit of the Subscription concealing identifier (SUCI) (convert SUCI from bits to decimal).

3) UE computes the SUCI from Subscription permanent identifier (SUPI)

$$SUCI = f_2(SUPI||T_{UE})_{DK_i} \quad (2)$$

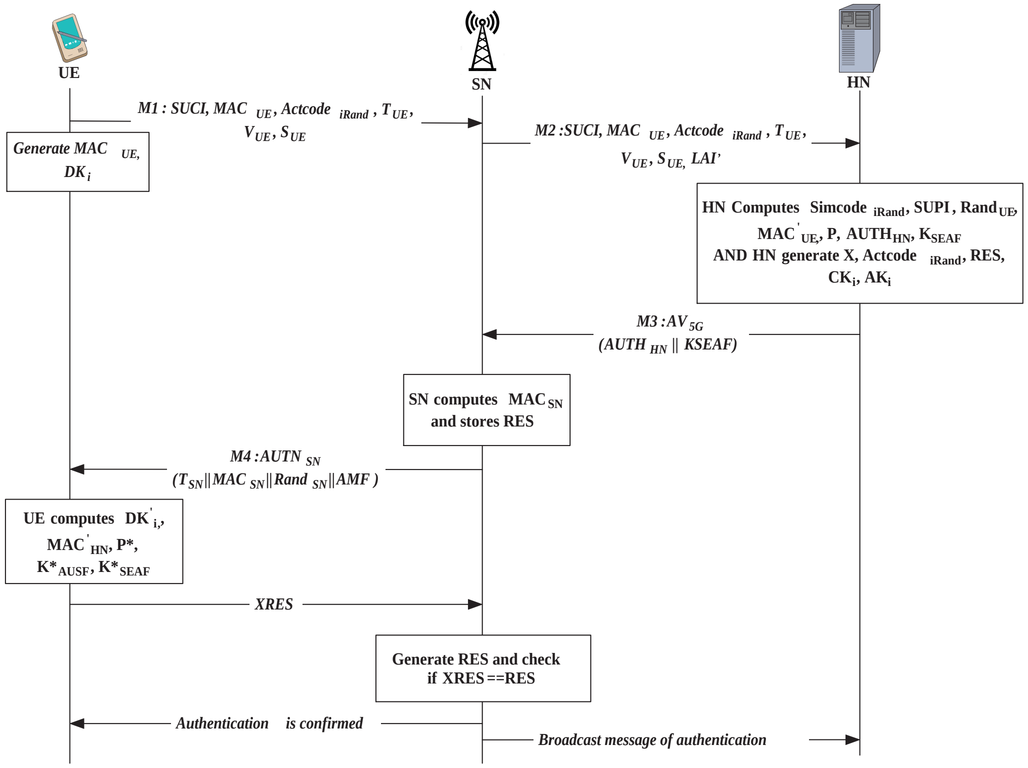

UE
SN
HN
M1 : SUCI, MAC UE, Actcode iRand , T UE, V UE, S UE
Generate MAC UE, DK i
M2 :SUCI, MAC UE, Actcode iRand , T UE, V UE, S UE, LAI'
HN Computes Simcode iRand, SUPI , Rand UE, MAC' UE, P, AUTH HN, K SEAF AND HN generate X, Actcode iRand, RES, CK i, AK i
M3 :AV 5G (AUTH HN || KSEAF)
SN computes MAC SN and stores RES
M4 :AUTN SN (T SN||MAC SN||Rand SN||AMF)
UE computes DK' i,, MAC' HN, P*, K* AUSF, K* SEAF
XRES
Generate RES and check if XRES ==RES
Authentication is confirmed
Broadcast message of authentication

Fig. 1: Phase-1: PPSE-AKA protocol

where f_2 is the reversible symmetric function (AES-CTR: AES- counter mode), and the plain-text and DK_i compute the cipher-text. The computation of key DK_i is available only at the UE_i and HN.

4) Then, UE_i generates the token value (V_{UE}) to compute the symmetric signature (S_{UE}) to maintain the non-repudiation and non-reusability.

$$V_{UE} = Rand_{UE} \oplus SUPI \tag{3}$$

,where $Rand_{UE}$ is the random number of UE_i.

$$S_{UE} = Rand_{UE} + DK_i \oplus Actcode_{iRand} \tag{4}$$

5) Finally, UE_i computes the message authentication code MAC_{UE}.

$$MAC_{UE} = f_1^*(Rand_{UE}||S_{UE}||LAI)_{DK_i} \tag{5}$$

where, LAI is the location area identity. LAI is the identifier of BSS (bas-station subsystem) location. UE_i and BSS are physically connected to each other. f_1^* is the HMAC-SHA1.

Step-2 (SN to HN)
$\{SUCI, MAC_{UE}, Actcode_{iRand}, T_{UE}, V_{UE}, S_{UE}, LAI^{'}\}$
The SN passes the received message to the HN with $LAI^{'}$.

Step-3 (HN to SN) AV_{5G}

1) Upon receiving the message from the SN, the HN first checks whether $T_{UE} - T_{HN} > |\triangle T|$. If it exceeds the threshold time value $|\triangle T|$, it means that authentication message is replayed and HN cancels the authentication request.

2) Then, the HN generates the $Simcode_{iRand}$ from the obtained $Actcode_{iRand}$ and compares with the $Simcode_{iRand}$. If they are equal, the K_i and DK_i keys of that $Simcode_{iRand}$ are obtained. We get from eq. (1) as:

$$Simcode_{iRand} = RCS_n(T_{UE} \oplus Actcode_{iRand})$$

,where RCS_n is the right circular shift by n (convert SUCI from bits to decimal).

3) Now, HN can compute the SUPI from eq. (2) as:

$$SUPI = f_2(SUCI||T_{UE})_{DK_i}$$

4) Then HN computes the $Rand_{UE}$ from eq. (3) as

$$Rand_{UE} = V_{UE} \oplus SUPI$$

5) HN receives the MAC_{UE} and checks the $LAI^{'}$. The SN knows the LAI of UE_i and SN forwards its own $LAI^{'}$ to the HN. To check the MAC_{UE}, HN computes the $MAC_{UE}^{'}$. Then, the HN can confirm whether the $LAI^{'}$ transferred by SN is same as the known by the UE_i. Otherwise, HN declines the authentication request.

$$MAC_{UE}^{'} = f_1^*(Rand_{UE}||S_{UE}||LAI^{'})_{DK_i}$$

,where the computation of S_{UE} is same as the eq. (4). HN compares the MAC_{UE} (eq.5) and $MAC_{UE}^{'}$. If they holds, the UE_i is authenticated by the HN. If $MAC_{UE}^{'}$ is bogus, HN cancels the authentication request of the UE_i.

6) HN computes a new random token value $P = DK_i \oplus SUPI$ and generates $X = S_{UE} - Actcode_{i\,Rand} \oplus P$. (from eq. (4)). Now, HN checks if $V_{UE} \overset{?}{=} X$ or not. If it is verified, HN maintains the non-repudiation and non-reusability for UE_i.

7) HN generates the $Actcode_{i\,Rand}{}' = f_1^*(Actcode_{i\,Rand}, P)_{DK_i}$ and MAC_{HN} as

$$MAC_{HN} = f_1(P||E(Actcode_{i\,Rand}{}'))_{DK_i} \qquad (6)$$

, where $E()_{DK_i}$ is used to encrypt the transmitted message over the public channel such as modified AES with 256 bit key.

8) Now, HN generates the response message $RES = f_2(Rand_{UE} \oplus SUPI)_{DK_i}$.

9) HN generates the cipher-key $CK_i = f_3(Rand_{UE})_{DK_i}$, anonymity-key $AK_i = f_4(Rand_{UE})_{DK_i}$, and integrity-key $IK_i = f_5(Rand_{UE})_{DK_i}$. The computation purpose of these keys is to maintain a secure communication between UE_i, HN and between UE_i, SN (key agreement). Also, HN computes $K_{AUSF} = KDF(CK_i||IK_i||P \oplus AK_i)$ and $K_{SEAF} = KDF(K_{AUSF}||P)$, where KDF is the key-derivation function, K_{AUSF} is the key for Authentication Server Function, and K_{SEAF} is the key for security anchor function. These functions will be used in generating the authentication vectors to SN. Here, HN computes the new delegation key for SN $DK_i{}' = f_1(Rand_{UE}||P)_{DK_i}$.
HN derives the authentication token $AUTN_{HN} = (MAC_{HN}||DK_i{}'||RES||AMF)$, where AMF is the authentication management field (shows the keys that generate AVs). Then, HN computes the authentication vectors $AV_{5G} = (AUTN_{HN}||K_{SEAF})$ and sends to the SN as an authentication response.

Step-4 (SN to UE_i) $AUTN_{SN}$

1) After receiving the AV_{5G}, SN increments its time-stamp value (T_{SN}) by one and selects a random number $Rand_{SN}$ and computes the MAC_{SN} as

$$MAC_{SN} = f_1(MAC_{HN}||Rand_{SN}||T_{SN})_{DK_i{}'} \qquad (7)$$

2) SN collects all the above values and stores the RES. And, sends the $AUTN_{SN} = T_{SN}||MAC_{SN}||Rand_{SN}||AMF$ to the UE_i.

Step-5 (UE_i to SN) $XRES$

1) Now, the UE checks whether $T_{SN} - T_{UE} > |\triangle T|$. If it exceeds the threshold time value $|\triangle T|$, UE_i declines the authentication request.

2) Then after, UE_i computes $DK_i{}' = f_1(Rand_{UE}||P)_{DK_i}$. It also computes MAC'_{HN} and compare with MAC_{HN} (shown in eq. (6)). For the verification of MAC'_{HN}, UE_i computes the $P^* = DK_i \oplus SUPI$. It checks whether $P^* \overset{?}{=} P$ or not. If the verification is correct, the UE_i verifies the HN. Otherwise, UE_i terminates the connection and ask for the fresh request. Then, UE_i computes the $Actcode_{i\,Rand}{}'$ and attempts to generate the MAC'_{HN}

$$MAC'_{HN} = f_1(P^*||D(Actcode_{i\,Rand}{}'))_{DK_i} \qquad (8)$$

, where $D()_{DK_i}$ is used to decrypt the transmitted message over the public channel such as modified AES with 256 bit key. If MAC'_{HN} and MAC_{HN} are same, the HN is authenticated by the UE_i. otherwise, UE_i declines the authentication request. Also, UE_i retrieves the CK_i, IK_i and AK_i same as HN.

3) Then UE_i checks for received value MAC_{SN} (shown in eq. (7)) and attempts to compute MAC'_{SN}. If MAC'_{SN} and MAC_{SN} are same, the SN is authenticated by the UE_i. otherwise, UE_i declines the authentication request. Then, UE_i computes the K^*_{AUSF}, K^*_{SEAF} and checks with K_{AUSF}, K_{SEAF} respectively.

4) Now, SN generates the $XRES = f_2(Rand_{UE} \oplus SUPI)_{DK_i}$. If $XRES = RES$, then UE_i is authenticated at SN and SN sends the broadcast message of authentication to the HN.

Therefore, the AKA mechanism is accomplished.

C. Authentication and Key Agreement Phase-2

If the UE_i is authenticated at HN (*Phase-1*), UE_i executes the *Phase-2* for n connections. When the UE_i communicates in same SN, the SN uses the $DK_i{}'$ (send by HN in Phase-1 (*Step-3*)) to validate UE_i for each call. Therefore, it is not needed to send another set of new authentication vectors from HN to SN. The SN authenticate the UE_i on the basis of received AV_{5G}. Each time the AV_{5G} are updated whenever the UE_i transmits authentication request. In *Phase-2*, the SN and UE_i mutually authenticate to each other.

Step-1 (UE_i to SN) $\{SUCI, MAC_{UE}, T_{UE}\}$

1) The UE_i increases the value of T_{UE} by 1 (the value of T_{UE} continues from the *Phase-1*). Then, UE_i computes the MAC_{UE} and there is no shared key in between UE_i and SN.

$$MAC_{UE} = f_1^*(T_{UE}||SUCI||LAI)_{DK_i{}'} \qquad (9)$$

The computation of $SUCI$ is same as *Step-1* in *Phase-1*.

Step-2 (SN to UE_i) $AUTN_{SN}$

2) The SN increases the value of T_{SN} by 1 and compares the T_{UE}. If $T_{UE} - T_{SN} > |\triangle T|$, SN declines the request. After this, SN computes the $MAC_{UE}{}'$ and checks whether $MAC_{UE} \overset{?}{=} MAC_{UE}{}'$. If they match, UE_i is authenticated at SN.
Now, SN computes MAC_{SN}

$$MAC_{SN} = f_1(T_{SN}||Rand_{SN}||MAC_{HN})_{DK_i{}'} \qquad (10)$$

where, $Rand_{SN}$ is newly computed random number by SN and MAC_{HN} is received in *Phase-1*. SN generates the $AUTN_{SN} = MAC_{SN}||T_{SN}||Rand_{SN}||AMF$.

Step-3 (UE_i to SN) $XRES$

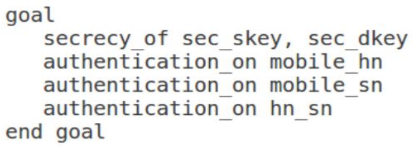
3) After receiving $AUTN_{SN}$, UE_i computes $MAC_{SN}{'}$ and if it matches with received MAC_{SN}, then SN is authenticated at UE_i. Also, UE_i computes the $CK_i = f_3(Rand_{SN})_{DK_i{'}}$, $AK_i = f_4(Rand_{SN})_{DK_i{'}}$, and $IK_i = f_5(Rand_{SN})_{DK_i{'}}$.
   Now, UE_i generates the $XRES = f_2(Rand_{SN})_{DK_i{'}}$ and send it to the SN. The SN computes the $XRES^* = f_2(Rand_{SN})_{DK_i{'}}$ and verifies the $XRES$. Also, it computes the CK_i, IK_i, AK_i.

For each successful authentication, SN increases the value of T_{SN} and transmits to the UE_i. To maintain the synchronization between UE_i and SN, UE_i sets the incremented T_{SN} to its T_{UE}. These synchronized values of random time-stamps would defeat all the possible attacks from the network.

IV. Security Analysis

This section includes the security analysis of the proposed protocol and protection from the security attacks. The privacy preservation of user identity, key secrecy, and mutual authentication of the communication entities are the security features of our PPSE-AKA protocol. These goals are identified as shown in Fig. 2. The proposed PPSE-AKA protocol is examined by the formal verification tool named AVISPA [10], [13]. The result of an AVISPA simulation of OFMC back-end shows that the communication entities achieve mutual authentication and are SAFE from the network attacks as shown in Fig. 3.

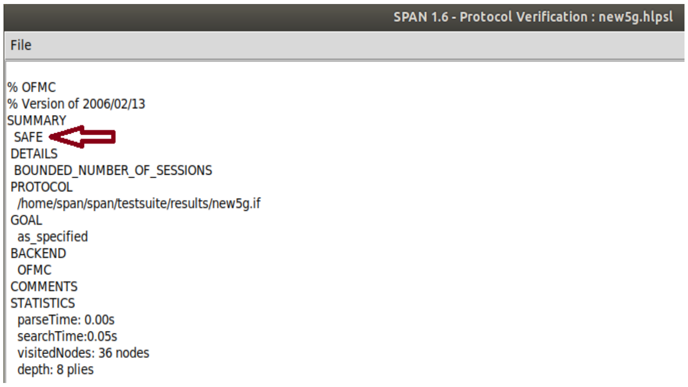
- **Privacy preservation of user identity:** The user identity (SUPI) is protected in the proposed protocol by encrypting it with a shared symmetric key between the UE and HN. While encrypting the time-stamp value of an UE is appended with the $SUPI$. It is impossible for an intruder to guess the time stamp value. In addition, we have ensured the secrecy of the shared symmetric key by using $Simcode_{iRand}$ and $Actcode_{iRand}$ parameters. So, the intruder neither gets access to the key nor to the actual user identity.
- **Secrecy maintenance of the shared secret key:** UE_i and HN have the shared secret key K_i. The security of the AKA protocol completely depends upon the confidentiality of the K_i. The one-time activation code $Actcode_{iRand}$ is computed at UE_i and is not publicly accessible. The purpose of this code is to obtain and check the $Simcode_{iRand}$. Similarly, the computation of $Simcode_{iRand}$ at HN is not openly available. These ($Actcode_{iRand}$) and $Simcode_{iRand}$ helps to protect the shared symmetric key between the UE and HN. In addition to this, the delegation key DK_i is used to update the symmetric shared secret key after each successful authentication.
- **Mutual Authentication of communication entities:** The mutual authentication begins with the UE and the HN authenticating each other. The message authentication code generated and transmitted among communication entities helps to maintain the integrity of user identity and achieves mutual authentication between them. In short, if the UE_i is authenticated at HN (Phase-1), UE_i executes the Phase-2 for n connections. The SN mutually authenticates the UE_i on the basis of received AV_{5G}.
- **Resistance from Security Attacks:** The proposed PPSE-AKA protocol uses the time-stamp and random number values which helps to protect it from the replay attacks. The identity of the SN is verified at HN using the MAC transmitted by the UE so it helps to protect the communication entities from the MitM and redirection attacks. The identity of the entities over the communication network is protected as well as they are verified by the other party so there is no scope for the unauthorized entity to send a random request. Moreover, the MAC_{UE} transmitted by the UE and MAC_{HN} transmitted by the HN ensures the communication occurs with legitimate entities only, and that prevents the DoS attack over the network.



```
goal
   secrecy_of sec_skey, sec_dkey
   authentication_on mobile_hn
   authentication_on mobile_sn
   authentication_on hn_sn
end goal
```

Fig. 2: Goals of the PPSE-AKA Protocol

The proposed PPSE-AKA protocol is formally verified by the AVISPA tool to ensure that the various security objectives presented above are achieved and the output of the OFMC back-end is shown in Fig. 3. The results conclude that the PPSE-AKA protocol has achieved all the above-mentioned security goals and mutual authentication of communication entities.



```
% OFMC
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/new5g.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime:0.05s
 visitedNodes: 36 nodes
 depth: 8 plies
```

Fig. 3: AVISPA Simulation Result of OFMC back-end

V. Performance Analysis

This section evaluates the performance in terms of communication and computation cost of the 5G-AKA protocols. The standardized notations and their meaning with size (in bits) of the 5G-AKA schemes are depicted in TABLE II.

To accomplish mutual authentication and key compliance in the AKA schemes, different messages are transmitted on

TABLE I: Comparative Analysis of the 5G AKA protocols

AKA Protocols	Mutual Authentication	Privacy Preservation	Secure from Redirection Attack	Secure from MiTM Attack	Secure from DoS Attack	Secrecy of Symmetric Key	Communication Overhead (in bits)	Computation Overhead
5G-AKA Protocol [12]	No	No	No	No	No	No	2960	15
SEL-AKA [10]	Yes	No	Yes	Yes	Yes	No	3856	11
Novel 5G-AKA [3]	Yes	No	Yes	Yes	Yes	No	2040	15
Generic-AKA [9]	Yes	Yes	Yes	Yes	Yes	No	2896	13
PPSE-AKA	Yes	Yes	Yes	Yes	Yes	Yes	2120	15

TABLE II: Notation and their interpretation with size

Notation	Interpretation	Size(in bits)
K_i	Shared secret key between UE_i and HN	128
$DK_i/DK_i^{'}$	Delegation key computed by K_i/DK_i	128
$SUCI$	Subscription concealing identifier	128
$SUPI$	Subscription permanent identifier	128
$MAC_x/MACID$	Message authenticated code of entity x	64
$Actcode_{i_{Rand}}$	One-time activation code of shared secret K	64
$Simcode_{i_{Rand}}$	Sim-code for UE_i	64
T_x	Time-stamp of entity x	64
$V_{UE}/S_{UE}/X$	Token value/ Signature of UE_i/ Signature of HN	128
P/P_*	Token value of HN/ UE_i	128
$Rand_x/RANDID/RAND$	Random number of entity x	128
$LAI/SN-name$	Location area identity/ Serving network-name	40
$AUTN$	Authentication token	Variable
AMF	Authentication management field	48
IK/CK	Integrity/cipher key	128
AK	Anonymity key	48
K_{AUSF}	Authentication server Anchor function Key	256
K_{SEAF}	Authentication Security Anchor function Key	256
$RES/XRES/HXRES_*/HRES_*$	Response/ Expected response	64

network lines. For communication cost, we find out the dispatched message size. The communication overhead of different previously established protocols is shown in the TABLE I. For calculating the computation overhead in the proposed model, we take into account the number of different cryptographic functions at the communication nodes. Here, for the convenience of maintaining uniformity, we take unit cryptographic functions. The TABLE I shows the computation overhead of various protocols for comparison with our scheme. The table gives a concrete proof that the proposed PPSE-AKA protocol stands on a competitive platform when compared to the existing schemes.

VI. Conclusion

In this article, privacy preservation and efficient AKA mechanism is illustrated for 5G communication networks. The protocol preserves the device identity and improves the authentication process in the cellular communication network. Moreover, the proposed scheme protects the single shared key between communication entities. The PPSE-AKA protocol is formally verified using the AVISPA tool. The analysis shows that the protocol achieves all the security goals and overcomes the potential attacks. Further, the performance evaluation proves that the scheme enhances the privacy & security with significant overhead. To the best of our wisdom, this is the first ever approach in the literature to preserve the device identity and single shared key between communicating participants.

References

[1] J. Lee, Y. Kim, Y. Kwak, J. Zhang, A. Papasakellariou, T. Novlan, C. Sun, and Y. Li, "Lte-advanced in 3gpp rel -13/14: an evolution toward 5g," *IEEE Communications Magazine*, vol. 54, no. 3, pp. 36–42, 2016.

[2] P. Schneider and G. Horn, "Towards 5g security," in *2015 IEEE Trustcom/BigDataSE/ISPA*, vol. 1. IEEE, 2015, pp. 1165–1170.

[3] A. Braeken, M. Liyanage, P. Kumar, and J. Murphy, "Novel 5g authentication protocol to improve the resistance against active attacks and malicious serving networks," *IEEE Access*, vol. 7, pp. 64 040–64 052, 2019.

[4] M. Bargh, R. Hulsebosch, E. Eertink, J. Laganier, A. Zugenmaier, and A. Prasad, "Umts-aka and eap-aka inter-working for fast handovers in all-ip networks," in *2007 IEEE Globecom Workshops*. IEEE, 2007, pp. 1–6.

[5] A. Koutsos, "The 5g-aka authentication protocol privacy," in *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2019, pp. 464–479.

[6] H. Khan and K. M. Martin, "A survey of subscription privacy on the 5g radio interface-the past, present and future," *Journal of Information Security and Applications*, vol. 53, p. 102537, 2020.

[7] 3rd Generation Partnership Project (3GPP) TS 33.501, "3gpp technical specification; security architecture and procedures for 5g system," vol. V.15.7.0, 2020.

[8] D. Strobel, "Imsi catcher," *Chair for Communication Security, Ruhr-Universität Bochum*, vol. 14, 2007.

[9] N. S. C. Shubham Gupta, Balu L. Parne, "A generic construction for efficient and secure aka protocol in 5g network," pp. 1–6, 2018.

[10] I. Gharsallah, S. Smaoui, and F. Zarai, "A secure efficient and lightweight authentication protocol for 5g cellular networks: Sel-aka," in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*. IEEE, 2019, pp. 1311–1316.

[11] F. Liu, J. Peng, and M. Zuo, "Toward a secure access to 5g network," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. IEEE, 2018, pp. 1121–1128.

[12] 3rd Generation Partnership Project (3GPP) TS 33.501, "Technical specification group services and system aspects; security architecture and procedures for 5g system," vol. V.15.0.0, March 2018.

[13] B. L. Parne, S. Gupta, and N. S. Chaudhari, "Segb: Security enhanced group based aka protocol for m2m communication in an iot enabled lte/lte-a network," *IEEE Access*, vol. 6, pp. 3668–3684, 2018.