



SEAI: Secrecy and Efficiency Aware Inter-gNB Handover Authentication and Key Agreement Protocol in 5G Communication Network

Shubham Gupta¹ · Balu L. Parne² · Narendra S. Chaudhari³ · Sandeep Saxena⁴

Accepted: 16 August 2021 / Published online: 27 August 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

Recently, the Third Generation Partnership Project (3GPP) has initiated the research in the Fifth Generation (5G) network to fulfill the security characteristics of IoT-based services. 3GPP has proposed the 5G handover key structure and framework in a recently published technical report. In this paper, we evaluate the handover authentication mechanisms reported in the literature and identify the security vulnerabilities such as violation of global base-station attack, failure of key forward/backward secrecy, de-synchronization attack, and huge network congestion. Also, these protocols suffer from high bandwidth consumption that doesn't suitable for energy-efficient mobile devices in the 5G communication network. To overcome these issues, we introduce Secrecy and Efficiency Aware Inter-gNB (SEAI) handover Authentication and Key Agreement (AKA) protocol. The formal security proof of the protocol is carried out by Random Oracle Model (ROM) to achieve the session key secrecy, confidentiality, and integrity. For the protocol correctness and achieve the mutual authentication, simulation is performed using the AVISPA tool. Also, the informal security evaluation represents that the protocol defeats all the possible attacks and achieves the necessary security properties. Moreover, the performance evaluation of the earlier 5G handover schemes and proposed SEAI handover AKA protocol is carried out in terms of communication, transmission, computation overhead, handover delay, and energy consumption. From the evaluations, it is observed that the SEAI handover AKA protocol obtains significant results and strengthens the security of the 5G network during handover scenarios.

Keywords 5G communication · Key-secrecy · Security attacks · Computation overhead · Random oracle model

✉ Shubham Gupta
guptashubham396@gmail.com

Extended author information available on the last page of the article

1 Introduction

With the advancement of IoT-based services and applications, the academicians and researchers of 3GPP have recommended 5G communication technology of the cellular network from the recent past [1–3]. The 5G technology suggests advanced aspects related to LTE-A network as non-3GPP inter-working, the formative arrangement of User Plane (UP) operations which are described as logical networks (user and control plane operations) with different potentials [4]. Further, User Equipment (UE) may broadcast Non-Access Stratum (NAS) information to the core network of the 5G for session and mobility administration, that hasn't been attained in preceding cellular network technologies [5, 6]. Moreover, these attributes identifies various aspects in the security framework of the 5G handover network. There are different handover services and applications as a vehicular management system, e-health care, and multimedia services, etc. because of the portability of numerous IoT devices/equipment in the 5G network [7–10].

Although, a key structure of 5G handover suffers from authentication complexities and various security susceptibilities [11]. In the handover key structure, an attacker can breach the secret session keys from genuine base-stations. Nonetheless, the partition of secret keys among base-stations avoids these issues at the time of handover. However, this approach neglects the negotiated key in one particular gNB from the other one. The source Next Generation (5G) Base-Station Node (gNB_s) broadcasts session key to the target Next Generation (5G) Base-Station Node (gNB_t). The gNB_t obtains a fresh session key by adopting a one-way operation and obtains key backward secrecy (KBS). The KBS restrains gNB's from generating the preceding keys from the established key. Contrarily, the gNB's might learn the entire keys used in earlier sessions of handover. Correspondingly, the KFS (forward secrecy) is preserved to provide that the communicating participants place various specifications in obtaining the new key for subsequent gNB. Moreover, the current gNB doesn't form subsequent keys. The structure of the 5G handover key fails to establish KFS if an attacker negotiates an honest base-station. In this situation, gNB_t doesn't provide fresh session keys because of de-synchronization. Hence, it demonstrates the security deficiencies in the handover key structure, and an attacker may negotiate prior keys between gNB and UE. The potential attacks may be sustained before the aforesaid modifications of the current key as the key specifications are obtained from preceding keys [12]. Furthermore, inter-gNB handover scheme in 5G networks degrades the transmission overhead because of numerous rounds of information transmission among the communicating participants. Hence, it is recommended to introduce a cost-efficient and attack resilient inter-gNB handover protocol in the 5G network.

1.1 Fundamental Security Properties of Handover Protocol

The security properties of the 5G handover are required to establish mutual authentication and shared secret key compliance between the communicating participants to satisfy the integrity for subsequent handover. The proposed 5G inter-gNB handover protocol must conclude the following properties.

- The protocol should maintain the privacy of the communicating participants during the authentication process. Only the home network can obtain the permanent identity of mobile devices.

- The protocol should maintain forward/backward secrecy with key re-freshness in each new handover authentication connection even if an attacker knows the private keys.
- The protocol must establish robust secrecy during the authentication to reduce the possible attacks in the 5G network.
- It is known that the UE is a low power resource device and the network channel has controlled frequency. Therefore, the protocol must be designed in a form that mandates the reduced overhead.

To achieve the necessary security properties during the handover process, 3GPP has introduced the handover mechanism [11]. However, the protocol incurs security vulnerabilities such as 1) several messages correspondence are needed to communicate with the AMF (serving network). Therefore, the 5G network reduces the transmission efficiency. 2) The 5G handover key derivation structure proposed by 3GPP brings out various gNB keys based on the horizontal/vertical key approach. Hence, the researchers have proposed various handover protocols in 5G communication networks [13–17]. Unfortunately, authentication complexity, high communication, and computation overhead are observed in these protocols. In addition, these protocols are susceptible to several security attacks. Hence, these handover protocols are not much suitable for efficient handover authentication in the 5G communication network.

To overcome these issues, we introduce Secrecy and Efficiency Aware Inter-gNB (SEAI) handover AKA protocol in 5G network. The proposed protocol avoids the problem of key escrow without involving any third party in establishing the secret keys. Also, the UE/gNB shows a secret correspondence of their identity by collision avoidance hash function and chooses secret keys in the handover initialization stage. The protocol doesn't execute the time-consuming exponentiation operations and shows less overhead. Moreover, the protocol doesn't transmit the secret keys over the public channel to preserve the handover key authentication.

1.2 Core Technical Improvements

To overcome the above-raised issues, we propose the Secrecy and Efficiency Aware Inter-gNB (SEAI) handover AKA protocol in 5G communication network. The main improvements of the protocol compared to previous handover schemes are:

1. We investigate the current 5G handover key structure and analyze its security deficiencies such as bogus base-station attack and synchronization failure.
2. We introduce the SEAI handover AKA protocol to overcome the security deficiencies from the current handover protocols of the 5G communication network. In the proposed protocol, gNB_i and UE establish mutual authentication at the time of handover execution without broadcasting the secret keys in the air. Moreover, the protocol mandates the KFS and KBS.
3. The confidentiality, integrity, and session key secrecy in the SEAI handover AKA protocol are proven secure by adopting ROM. Also, the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool presents correctness and verification of the protocol. Moreover, the attack and security analysis are provided for numerous security specifications. The analysis represents that the protocol averts the potential attacks.

4. The performance estimation of current and proposed handover protocols is calculated on the basis of communication, computation, and transmission overhead. The estimation results represent that the SEAI handover AKA protocol is efficient and secure compared to the previously proposed handover schemes.
5. The handover delay & key size is computed for the proposed and existing handover protocols based on hop count, number of users. Also, we analyze the protocols based upon the energy consumption during the handover authentication process.

The rest of the article is formed as follows: Sect. 2 illustrates the network model of 5G handover, key hierarchy, handover structure, and the existing handover methodologies. The security susceptibilities of the 5G handover protocol are discussed in Sect. 3. Section 4 discusses the proposed SEAI handover AKA protocol in the 5G network. The formal security proof using ROM, correctness, and informal analysis of the protocol are presented in Sect. 5. Section 6 demonstrates the performance estimation of 5G handover AKA protocols. Lastly, Sect. 7 concludes the article.

2 Overview and Existing Methodologies

The 5G network derives a fundamental security architecture of the LTE-A network. 3GPP has done some security design contributions in the 5G network after the performance and practical operations. Although, a novel handover authentication framework is required to mandate these modifications for the 5G network. In this section, we demonstrate the overview of the 5G handover framework, handover key structure, and key hierarchy. To obtain mutual authentication and overcome the bandwidth consumption from the 5G network, researchers and academicians have introduced numerous handover methodologies. We illustrate these protocols based on their security features and issues in this section also.

2.1 Network Model of 5G Communication Network in Handover

The communication in 5G network framework is established by the following participants as Access and Mobility Management Function (AMF)/Security Anchor Function (SEAF), Authentication Credential Repository and Processing Function (ARPF), Session Management Function (SMF), Policy Control Function (PCF), and Authentication Server Function (AUSF) as shown in Fig. 1 [18–20]. In this framework, UE establishes the connection with various gNBs and AMF maintains secure communication using Key_{AMF} . Further, UE verifies the AUSF while subscription information is kept by the ARPF. For the authentication with UE, the ARPF stores the secure symmetric key S_{key} . Also, ARPF computes the authentication vectors (AVs) by executing the cryptographic operations with the security parameters. The Security Policy Control Function (SPCF) consists of security to the SMF and AMF. The security credentials has the key length, integrity and confidentiality algorithm, and AUSF information. The Non-access Stratum (NAS) and AS layers maintain their communication traffic to establish gNB security [21]. Whenever UE communicates in the 5G network, the AS layer establishes the secrecy between the UE, NAS layer, and gNB. In addition, the N3-UP (path of user plane signaling) and N2-CP (path of control plane signaling) are established between UE & User Plane Function (UPF) and UE & AMF respectively [22]. These new updates are the autonomous paths for user/control planes and key algorithms (integrity and encryption).

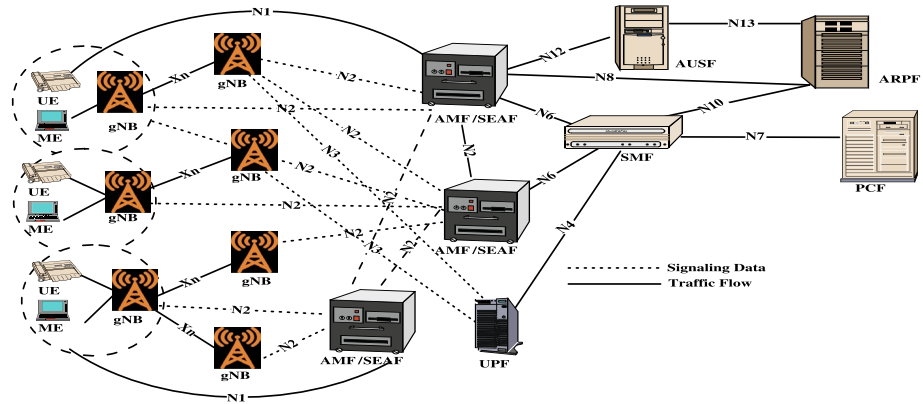


Fig. 1 A handover framework of 5G communication network

2.2 Key Hierarchy

The 5G network key hierarchy is designed for the efficient structure of numerous keys among the participating entities in the communication [11]. The first transition key Key_{AUSF} is computed by the ARPF to maintain secret communication between UE and ARPF. From this key, another transition key Key_{SEAF} is computed between UE and AUSF to determine Key_{AMF} . In addition, the key Key_{gNB} is retrieved at AMF and send to the gNB. The UE establishes authentication compliance with AMF in support of AUSF/ARPF. The AMF and UE compute the Key_{AMF} using Key_{SEAF}/Key_{AUSF} after obtaining the mutual authentication. The Key_{AMF} is valid for the certain period computed for the successive AKA process and generates four sub-keys from it. The two sub-keys Key_{NASenc} and Key_{NASint} are computed for encryption verification and integrity respectively. UE and AMF derives the third sub-key Non-3GPP access Inter-working Function (Key_{N3IWF}) from Key_{AMF} for non-3GPP access. Moreover, UE and gNB generate the fourth sub-key Key_{gNB} that computes another four keys. Firstly, two keys Key_{RRCenc} and Key_{RRCint} are required to authenticate the Radio Resource Control (RRC) signaling encryption and its integrity respectively. In addition, the keys Key_{UPenc} and Key_{UPint} are required to verify the UP data traffic encryption and integrity respectively. Also, Key_{gNB} is renewed during handover whenever the UE enters into the coverage area of another gNB.

2.3 Handover Structure

In this section, we will demonstrate the Xn-based (inter-gNB) 5G handover structure. In the inter-gNB handover, AMF and UE obtain the authentication process to fulfill the security properties. For secure communication during handover, gNB_s generates the Key_{NG-RAN} (preceding Key_{gNB}) for gNB_t . Also, Key_{gNB} is concatenated at handover key chaining before the subsequent AKA process [11]. By using the one-way hash, gNB_s generates the next Key_{gNB} from the present gNB and applies the current key from AMF. Then, AMF transmits these information to gNB_t after accomplishing the inter-gNB handover and apply it for subsequent handover. NH Chaining Counter (NCC) and Next Hop (NH) are the key parameters in handover key chaining. AMF

sets up the next NH parameters generated from Key_{AMF} for respective handover repeatedly. The communication mechanism of 5G inter-gNB handover is shown in Fig. 2 [11]. It is analyzed that the gNB_s obtains the specific key parameters $\{NH_{NCC}, NCC\}$ from the preceding handover. The counter of NH key update is NH_{NCC} . The gNB_s computes $Key_{NG-RAN'}$ from NH key and Key_{gNB} by performing horizontal and vertical key operations respectively for gNB_t . The horizontal and vertical key operations are $Key_{NG-RAN'} = KDF(\eta || NH_{NCC})$ and $Key_{NG-RAN'} = KDF(\eta || Key_{gNB})$ respectively, where $\eta = ARFC - DL || PCIA$, $NH_{NCC'} = KDF(Key_{gNB} || Key_{AMF})$ (original value of NH), $NH_{NCC} = KDF(NH_{NCC-1} || Key_{AMF})$, NH_{NCC-1} (preceding value of NH), absolute radio frequency channel-down link (ARFC-DL), and physical cell identity allocation (PCIA). In the horizontal handover, gNB_s doesn't achieve the specific NH key, and $\{NH_{NCC}, NCC\}$ are appeared before the completion of inter-gNB 5G handover. On the other hand in vertical handover, gNB_s has specific NH key derived in 5G inter-gNB handover, and AMF and UE could fetch the NH only.

The gNB_s transmits $\{NCC, Key_{NG-RAN'}\}$ to gNB_t in inter-gNB handover. It is analyzed that the gNB_s executes the vertical operation and future keys between gNB_t and UE. In this handover, the AMF and gNB_t transmit their handover request/response to UE. Later, UE verifies the acknowledged NCC from the equipped NCC. If it authenticates, UE performs vertical operation from the current Key_{gNB} to generate $Key_{NG-RAN'}$. Or, UE tries to integrate the NCC by generating NH key regularly, until it authenticates and executes the horizontal key operation to derive $Key_{NG-RAN'}$. Moreover, the gNB_t transmits the path change request to the AMF in inter-gNB 5G handover after the handover accomplishment with UE. Then, AMF increases NCC value by one and derives the specific NH key. Also, AMF transmits the $\{NH_{NCC+1}, NCC + 1\}$ to gNB_t for further handover.

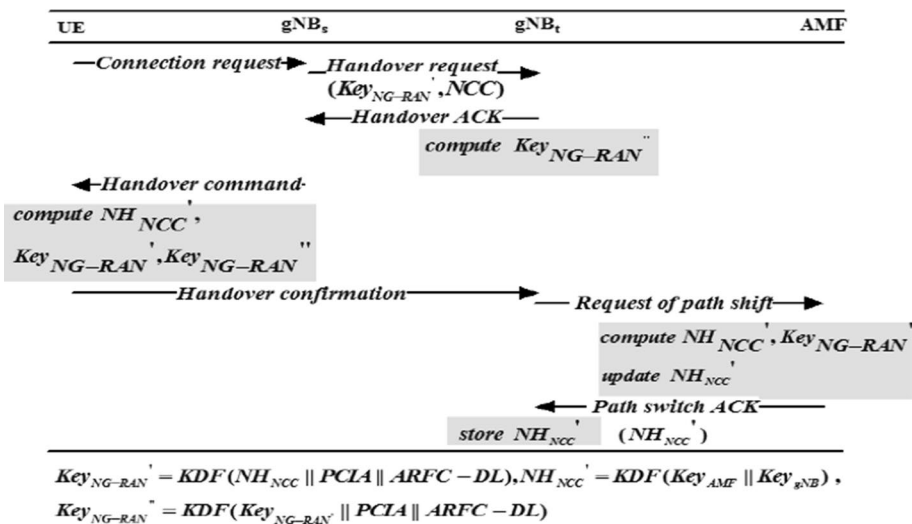


Fig. 2 Inter-gNB 5G handover mechanism

2.4 Existing Methodologies

Cao et al. [13] discussed the privacy-preserving handover authentication protocol for 5G HetNets using the Software Defined Network (SDN). The protocol obtains the mutual authentication and key agreement between base-stations and mobile devices without any other entities. Also, the protocol overcomes the system authentication complexity and minimizes bandwidth consumption. However, similar to the 3GPP-5G handover AKA protocol, the protocol fails to avoid the de-synchronization of communicating entities that lead to DoS attack because of sequence number (SQN) mismatch. In the protocol, it is considered that the SQN is maintained between base-station and UE. In one registration, the value of SQN is used for entire the n connections and increases the value by one at UE/base-station. An adversary may attempt a bogus registration attempt by using previous messages and SQN value become inconsistent. If the genuine UE attempts to create the connection with the target base-station, the session keys and message authentication code are not matched. Therefore, the genuine UE will be unauthorized to access the network during handover. To avoid the above issues, Sharma et al. [14] proposed the handover authentication protocol that maintains the privacy-preservation and key secrecy. Also, the protocol avoids all the security susceptibilities and withstands security attacks. However, numerous message correspondence with the base-station and terminal (UE) carries handover breach and increases the overhead because the serving network is very far from base-station. Hence, the protocol incurs authentication complexity. Also, the source base-station computes numerous keys for target base-stations that enhance the probability of dodging the secret keys.

Zhang et al. [15] introduced the Elliptic Curve Cryptography (ECC)-based handover authentication protocol by using chameleon hash function key pairs to avoid the authentication complexity. However, the protocol obtains all the security characteristics but suffer from identity privacy preservation and MitM attack. Also, the protocol exhibits a huge network and transmission overhead due to the additional use of point multiplication key operations. Han et al. [16] designed the efficient handover AKA to enhance security properties and maintain mutual authentication. Also, the protocol incurs less overhead and establishes the key secrecy. However, the protocol suffers from DoS attack similar to Cao's protocol. Due to the use of Extensible Authentication Protocol (EAP)-AKA [23], the proposed protocol suffers from identity privacy preservation and security vulnerabilities such as redirection and MitM attack. Recently, Kumar et al. [17] designed the ECC-based handover authentication protocol for 5G-wireless LAN networks. The protocol obtains mutual authentication and most of the security properties such as key forward/backward secrecy, anonymity. However, the protocol fails to preserve the identity of the communicating participants and suffers from redirection, MitM attack. In addition, the protocol incurs huge communication and computational overhead due to the additional use of point multiplication functions during the handover authentication process.

From the existing handover methodologies, it is noticed that these protocols are susceptible to various known attacks and exhibit huge network overhead. Also, the protocols fail to provide the key secrecy and suffer from authentication complexity. Therefore, the above-discussed protocols are not well suited for efficient handover development in the 5G communication network. To avoid these problems, we introduce the SEAI handover AKA protocol in the 5G network to obtain necessary security requirements. The SEAI protocol is free from the problem of key escrow as there is no entanglement of

any third party in establishing the secret keys. Also, the communicating participants send their identity securely in the handover process and don't transmit the secret keys in the public channel during the handover agreement. The protocol operates the key operations using the point multiplication functions and enhances its efficiency compared to the existing protocols. Moreover, the protocol avoids potential attacks and provides all the security properties.

3 Security Weaknesses in 5G Handover Mechanism

This section illustrates the security susceptibilities in the 5G handover mechanism proposed by the 3GPP and other various researchers. These security problems represent various adversities in the steady communication of the 5G handover network. Let consider, an attacker ATT impersonates the genuine base-station (gNB) and implants the forged base-station gNB_{ATT} in the communication network. ATT may approach its stored parameters by massive attacks as gNB is implanted very far to the AMF.

3.1 De-synchronization Attack

ATT can install the gNB_{ATT} that performs the Denial-of-Service (Dos) and leads to de-synchronization during the 5G handover. The prime target of gNB_{ATT} is to build the bogus information of NCC and dodge the imminent keys. The ATT can impose to gNB_t to disturb the key forward secrecy by performing horizontal key operations. The value of NCC can be compromised by manipulating the information between gNB_s and gNB_t in the 5G handover mechanism. The gNB_{ATT} chooses a large prime number to impersonate the NCC and transmits to gNB_t during second handover response as shown in Fig. 2.

ATT sends the original and false NCC to UE for maintaining the synchronization. The NCC value in path shifting information is negligible than that obtained by gNB_{ATT} . In addition, the gNB_t and UE generate future handover keys on the basis of present Key_{gNB} in place of NH_{NCC+1} . Therefore, gNB_{ATT} may not obtain the following Key_{gNB} because of forward secrecy failure. The gNB acquires the following key of Key_{NG-RAN} from Key_{gNB} because ATT can know ARFC-DL and PCIA. Moreover, ATT impersonates the UE by sending the original value of NCC and executes de-synchronization. ATT can damage the NCC by disguising the information AMF to gNB_t . The gNB_t fails to accommodate to the fresh value of NCC because bogus information has a lesser value of NCC compared to the initial one. To overcome the above security concerns, the Internet Protocol Security scheme is applied in path shifting and its confirmation message. Although, numerous links of IPsec with gNB_s are prescribed to establish in these transmitted messages with AMF. ATT may deploy the de-synchronization by information flooding/drop to block the gNB_t from recovering the NCC . Accordingly, the gNB_t may not modify the NCC and synchronization of the keys is not established. ATT may know the secret handover information from the communicating parties from gNB_{ATT} and degrades the network efficiency.

3.2 Verification Failure

The 5G inter- gNB handover mechanism needs various request/response message communication rounds with the AMF and gNB_s/gNB_t that suffers from handover explosion. Also, it increases the overhead because the AMF is installed far from gNB . Hence, the

5G handover network suffers from authentication complexity/verification failure. The gNB_s generates legitimate keys for numerous gNB_t from the current one by using required specifications in the 5G handover mechanism. For explanation, gNB_s may obtain the Key_{NG-RAN} between the UE and gNB_t from Key_{NG-RAN} . Once the gNB_s is attacked, the ATT knows all the subsequent keys. Therefore, the key backward secrecy is not obtained in the current 5G handover communication.

4 Proposed SEAI Handover AKA Protocol

In this section, we discuss the SEAI handover AKA protocol to avoid the security deficiencies from the previously proposed handover protocols. The proposed protocol has three stages: a) establishment stage; b) handover initialization stage and c) handover authentication stage. The methodology of Elliptic Curve Cryptography (ECC) is illustrated in the establishment stage. UE is authenticated at AMF and gNB_s defines the handover request/response information to UE for preceding communication in the initial authentication stage. Moreover, the gNB_t and UE executes the handover authentication stage when UE arrives in the area of gNB_t . The used notations and their meaning in the proposed protocol are reported in Table 1.

4.1 Establishment Stage

In order to achieve the authentication between gNB_t and UE in the SEAI handover AKA protocol, we are applying ECC [24]. Let λ be a security parameter, a prime number w and an elliptic curve $E(F_w)$ over F_w with w elements. Here, two elements a, b are designated in E over F_w of an equation $b^2 + x_1ab + x_3b = a^3 + x_2a^2 + x_4a + x_5$, where $x_1, x_2, x_3, x_4, x_5 \in F_w$. Suppose, q is a prime order in $E(F_w)$ with point P , where $q \nmid \#E(F_w)$.

Table 1 Used notations and their meaning in the proposed SEAI protocol

Notation	Meaning
$m_{AMF} / m_{UE}, n_{UE} / n_{gNB_t} / m_{ARPF} / x_{AUSF}$	Secret key of AMF/UE/ gNB_t /ARPF/AUSF
$M_{AMF} / M_{UE}, N_{UE} / N_{gNB_t} / M_{ARPF} / X_{AUSF}$	Public key of AMF/UE/ gNB_t /ARPF/AUSF
$XRESV / RESV'$	Expected response/actual expected response at AMF/UE
$XRES' / RES'$	Expected response/actual response at AUSF/UE
$Key_{AMF}, Key_{SEAF}, Key_{AUSF}$	Generated key at AMF/UE, AUSF, and ARPF respectively
$IKKey / CKKey$	Integrity/cipher key
$ID_{gNB_t} / ID_{gNB_s} / SEAF_{ID}$	Identity of $gNB_t/gNB_s/SEAF$
$ngKSI$	Key set identifier function of 5G communication network
$AUTN$	Authentication token value
RHI_{UE}	Received handover information by UE from gNB_s
T_{exp}	Time of expiration of RHI_{UE}
$Key_{gNB_s}^{UE} / Key_{gNB_t}^{UE}$	Computed session key between gNB_s/gNB_t and UE
$MAC_{gNB_t} / MAC_{UE}, MAC_{cfm}$	Message authentication information of gNB_t/UE , confirmation of handover
$inau_i$	Information of authentication of entity i

Moreover, finite field of integers modulo prime q is the Z_q and Z_q^* is multiplicative subgroup of Z_q . Also, the cyclic group C has the generator P . The ARPF initializes the SEAI handover AKA stage as following.

1. The ARPF selects the secure one way collision resistant hash functions:

- $H_1 : \{0, 1\}^* \times C \longrightarrow Z_q^*$
- $H_2 : \{0, 1\}^* \times Z_q^* \longrightarrow Z_q^*$
- $H_3 : \{0, 1\}^* \times C^2 \times \{0, 1\}^* \longrightarrow Z_q^*$
- $H_4 : \{0, 1\}^* \times C^2 \times \{0, 1\}^* \times Z_q^* \times C^2 \times \{0, 1\}^* \times C \longrightarrow \{0, 1\}^\lambda$
- $H_5 : C \times \{0, 1\}^\lambda \times \{0, 1\}^\lambda \times \{0, 1\}^* \times Z_q^* \times C^2 \times \{0, 1\}^* \longrightarrow \{0, 1\}^\lambda$

2. Furthermore, ARPF distributes/publishes these system specifications/public parameters $PK = \{KDF, P, C, w, q, H_1, H_2, H_3, H_4, H_5\}$ to all the entities that establish the communication in initial and handover authentication stage.

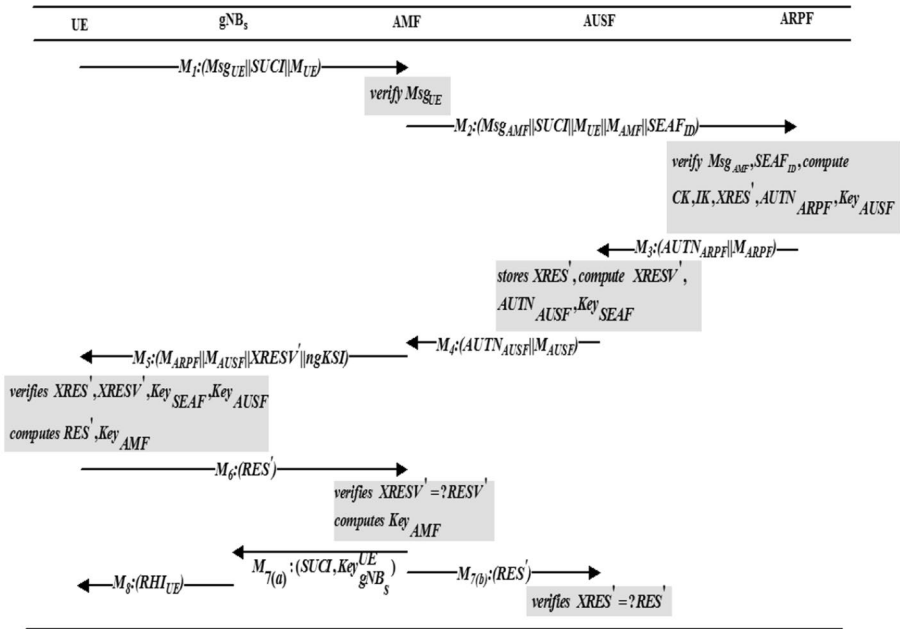
As the protocol believes in the elliptic curve discrete logarithmic problem (ECDLP) assumption [25, 26]. It is admitted that the ECDLP computation is not feasible in polynomial-time and the key of ECC (size: 256 bits) obtains the same secrecy as RSA (size: 3072 bits).

1. *Note-(a):* Let, C be a group of q prime order and point P . $xP \in C$ is an element, where $x \in Z_q^*$. It is computationally difficult to derive x from xP and P .
2. *Note-(b):* Let, C be a group of q prime order and point P . $xP, yP, P \in C$ are the elements where $x, y \in Z_q^*$. It is computationally difficult to derive the xyP by using any polynomial time algorithm.

4.2 Handover Initialization Stage

In this stage, UE is verified at AUSF and AMF followed by ARPF [4]. During the verification process, some handover specifications are confined to message authentication requests/responses of the original 5G-AKA protocol. These specifications in 5G-AKA don't mitigate the efficiency of the network. In the SEAI handover AKA protocol, the AMF sends the secret keys to gNB_s and then, gNB_s broadcasts the information to UE for subsequent handover after accomplishing the UE's verification. The descriptive explanation of the handover initialization is exhibited in Fig. 3 and step-wise discussion is as follows:

- **Step-1:** $m_{UE} \in Z_q^*$ is private key chosen by the UE and computes $M_{UE} = m_{UE}.P$. Then, UE sends the message $SUCI, M_{UE}, Msg_{UE}, M_{AMF}$ to AMF and initiate the authentication mechanism with ARPF. The Subscription Permanent Identifier (SUPI) is never broadcasted in the communication channel and Subscription Concealed Identifier (SUCI) is the privacy-preserving identifier containing the concealed SUPI. Only ARPF uses the Subscriber Identity De-concealing Function (SIDF) and decrypts the SUCI to achieve the original SUPI.
- **Step-2:** AMF authenticates the message from the UE and verifies Msg_{UE} . After this, it chooses $m_{AMF} \in Z_q^*$ (private key) and derives public key $M_{AMF} = m_{AMF}.P$. Finally, AMF sends the $SUCI, M_{ARPF}, M_{AMF}, M_{UE}, Msg_{AMF}, SEAF_{ID}$ to the ARPF.
- **Step-3:** Msg_{AMF} is verified at the ARPF and authentication of UE is accomplished. Then, ARPF authenticates $SEAF_{ID}$ and checks the $SEAF_{ID}$ of UE. The $SEAF_{ID}$ is



$$\begin{aligned}
 &Msg_{UE} = H_1(SUCI || M_{AMF} || m_{UE}), Msg_{AMF} = H_1(SUCI || m_{AMF} || M_{ARPF} || SEAF_{ID}), CKey = H_2(SUCI || M_{UE} || m_{ARPF}), \\
 &XRES' = H_4(SUCI || M_{UE} || m_{ARPF} || M_{AUSF}), RES' = H_4(SUCI || m_{UE} || M_{ARPF} || M_{AUSF}), Key_{AUSF} = KDF(SUCI || CKey || IKey || M_{ARPF}), \\
 &AUTN_{ARPF} = (XRES' || Key_{AUSF} || M_{AMF}), Key_{SEAF} = (SUCI || Key_{AUSF} || M_{AUSF}), XRESV' = H_5(RESV' || M_{AMF} || m_{AUSF} || M_{ARPF}), \\
 &RESV' = H_5(RES' || m_{AMF} || M_{AUSF} || M_{ARPF}), AUTN_{AUSF} = (XRESV' || Key_{SEAF} || M_{ARPF}), Key_{AMF} = (SUCI || Key_{SEAF} || M_{UE}), \\
 &Key_{gNB_s}^{UE} = KDF(ID_{gNB_s} || Key_{AMF} || rspec), RHI_{UE} = E\{SUCI || ID_{gNB_s} || Key_{gNB_s}^{UE} || T_{eq}\}, IKey = H_3(SUCI || M_{UE} || m_{ARPF})
 \end{aligned}$$

Fig. 3 Handover initialization stage

verified if they are same, otherwise; ARPF rejects an authentication request. Moreover, the ARPF chooses $m_{ARPF} \in Z_q^*$ and derives $M_{ARPF} = m_{ARPF} \cdot P$. It generates the $IKey, CKey, Key_{AUSF}, AUTN_{ARPF}, XRES'$ and transmits the $M_{ARPF}, AUTN_{ARPF}$ to the AUSF.

- **Step-4:** AUSF keeps $XRES'$ and generates the $Key_{SEAF}, AUTN_{AUSF}, XRESV'$. Then, it transmits the $M_{AUSF}, AUTN_{AUSF}$ to the AMF.
- **Step-5:** AMF sends the $M_{ARPF}, M_{AUSF}, ngKSI, XRESV'$ to the UE. Then, UE generates the $XRES', XRESV', Key_{AMF}, Key_{AUSF}, Key_{SEAF}$. It compares these derived values with the obtained ones. UE verifies and confirms the authenticity of AUSF and ARPF, if these value matches. Moreover, UE computes RES' and sends to AMF.
- **Step-6:** AMF obtains $RESV'$ and compares with $XRESV'$. If it verifies, AMF confirms the UE's verification and generates Key_{AMF} . Further, AMF transmits RES' to AUSF and $Key_{gNB_s}^{UE}, SUCI$ to the gNB_s .
- **Step-7:** The AUSF achieves the RES' and compares with $XRES'$. If they match successfully, authentication of the UE is accomplished at AUSF. Moreover, gNB_s retrieves the RHI_{UE} from $Key_{gNB_s}^{UE}$, and sends to UE for subsequent handover. Here, $rspec$ is the related specifications of gNB_s as $ID_{gNB_s}, ECI, frequency, PCI$. Then, UE retrieves $Key_{gNB_s}^{UE}$ and securely stores RHI_{UE} .

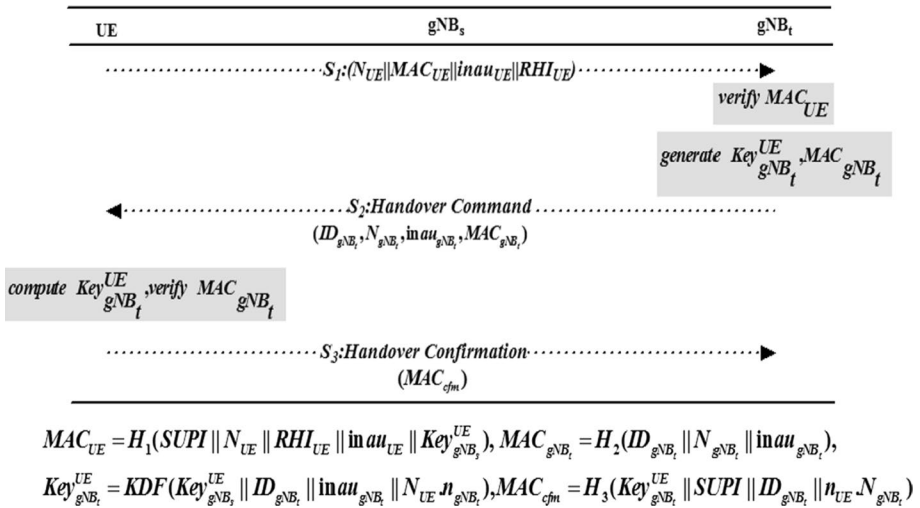


Fig. 4 Handover authentication stage

4.3 Authentication Stage of Handover

When UE moves into the range of gNB_t , the gNB_t and UE initiate mutual authentication and key agreement mechanism. Here, UE uses the RHI_{UE} which is retrieved in the handover initialization stage. The inter-gNB handover follows the traditional handover authentication mechanism. Figure 4 represents the flow of the authentication messages in the SEAI handover AKA mechanism. The illustration of the handover authentication steps is shown below.

- **Step-1:** When UE is in the area of gNB_t , it obtains public parameters of associated gNBs and another specifications such as cell ID (ECI), PLMN-ID, location area identity (LAI), PCI of gNB_t . After this, UE chooses a random nonce $n_{UE} \in Z_q^*$ and generates $N_{UE} = n_{UE} \cdot P$. Then, UE retrieves MAC_{UE} and sends the $N_{UE} || RHI_{UE} || MAC_{UE} || inau_{UE}$ to gNB_s ; where, the $inau_{UE}$ has the related specifications as $ECI, PLMN_{ID}, PCI$ of gNB_t and targeted LAI .
- **Step-2:** Now, gNB_t retrieves the $Key_{gNB_t}^{UE}$ by applying RHI_{UE} . It also confirms the authenticity of RHI_{UE} from T_{exp} . If it is not verified, gNB_t rejects the handover query. After this, gNB_t computes and checks the MAC_{UE} by using $Key_{gNB_t}^{UE}$; If it verifies, gNB_t accepts the acknowledged MAC_{UE} that is transferred from genuine UE. Or, authentication is rejected.
- **Step-3:** After this, gNB_t chooses a random nonce $n_{gNB_t} \in Z_q^*$ and retrieves $n_{gNB_t} \cdot P = N_{gNB_t}$. Moreover, it generates the MAC_{gNB_t} for UE and session key $Key_{gNB_t}^{UE}$. Also, it sends the handover message $MAC_{gNB_t} || N_{gNB_t} || ID_{gNB_t} || inau_{gNB_t}$ to the UE. The $inau_{gNB_t}$ has the specifications as $ID_{AMF}, ECI, PLMN_{ID}$, and PCI .
- **Step-4:** Now, UE calculates the $Key_{gNB_t}^{UE}$ and checks the MAC_{gNB_t} . If it is incorrect, UE transmits the authentication failure response to gNB_t . On the other hand, UE accepts the gNB_t and transmits successful handover acknowledgement (MAC_{cfm}) to gNB_t with the $Key_{gNB_t}^{UE}$. Then, gNB_t approves the handover confirmation with the UE.

5 Security Analysis

This section discusses that the proposed protocol fulfills the security requirements in the ROM. The used assumptions and security model are shown in this proof. The correctness of the protocol is obtained from the AVISPA tool. Also, the informal analysis of protocol is discussed for various security attacks.

5.1 Security Model

For the resistance of identified attacks in the SEAI protocol, we are using a provable security mechanism. We are showing the security proof based on the modeling introduced by [27].

5.1.1 Participants

The protocol Π executes with numerous number of associated participants in 5G network where the participant could be a client $W \in \omega$ or server $N \in \eta$. The set η is considered that only a single server is involved at one time. Every participants could have numerous instances (oracles) in distinct executions of Π . We indicate the i_{th} instance of W and N in sessions as Π_W^i and Π_N^i respectively. Each instance Π_W^i/Π_N^i has its session identity sid_W^i/sid_N^i (set of identities that shows the message flow sending/receiving in this instance), partner identity pid_W^i/pid_N^i (set of identities which are executed in this instance), and session key as sk_W^i/sk_N^i . The instances Π_W^i, Π_N^i can be accepted if it maintains the $sid_W^i/sid_N^i, sk_W^i/sk_N^i$, and pid_W^i/pid_N^i . $\Pi_{W_1}^i/\Pi_{W_2}^j$ are acknowledged as a partner if (i) both are successfully accepted; (ii) $sid_{W_1}^i = sid_{W_2}^j$; (iii) $sk_{W_1}^i = sk_{W_2}^j$; (iv) $pid_{W_1}^i = pid_{W_2}^j$.

5.1.2 Attacker Model

It is considered that the attacker ATT completely controls the network, which initiates the communication sessions among the participants [28]. The ATT can execute the following queries as:

Execute($\Pi_{W_1}^i, \Pi_{W_2}^j, \Pi_N^k$): The query forms passive attacks where an adversary dodges the legitimate operations among the instances of client $\Pi_{W_1}^i, \Pi_{W_2}^j, \Pi_N^k$. The result of the query is the exchange of messages at the time of the genuine operation of Π .

Send_Client(Π_W^i, m): The attacker may use this query to trace the message and update it or forward to the client Π_W^i . The result of the query is the information that the client Π_W^i might compute upon acceptance of message m . Moreover, an attacker is granted to start the protocol by appealing to **Send_Client**($\Pi_{W_1}^i, (W_1, Start)$).

Send_Server(Π_N^i, m): The query builds active attacks counter to server. The result of the query is the information that the server Π_N^i might compute upon acceptance of message m .

Reveal(Π_W^i): The query builds identified session key attack. An attacker executes the query to achieve the secret keys of instance Π_W^i .

Corrupt(W): The query sends the long-term secret/private keys to an attacker for participant W .

Test(Π_W^i): An attacker can build this type of query only one time to a fresh instance. On the response of the query, random number $e \in \{0, 1\}$ is chosen. If $e = 1$, session key obtained by Π_W^i is send. Or, return the consistently chosen random number.

5.1.3 Fresh Instances

An instance Π_W^i is fresh if following condition satisfies: (i) Π_W^i is accepted; (ii) Π_W^i or its corresponding partner hasn't run the **Reveal** query after acceptance; (iii) client's corresponding partner with Π_W^i , hasn't run the **Corrupt** query.

5.1.4 Protocol Security

The security of proposed protocol Π is formed by game $Game^{protocol}(\Pi, \mathcal{ATT})$. As running this game, \mathcal{ATT} can execute several queries to Π_W^i and Π_N^i . If \mathcal{ATT} asks a **Test**(Π_W^i) query, and Π_W^i is fresh and accepted, \mathcal{ATT} generates the e . The objective of \mathcal{ATT} is know e correctly in test query. The advantage of \mathcal{ATT} can be written as:

$$Adv_{\Pi}^{protocol}(\mathcal{ATT}) = |2Pr[e = e'] - 1| \tag{1}$$

The protocol Π is secure if $Adv_{\Pi}^{protocol}(\mathcal{ATT})$ is negligibly higher than $O(q_{se})$, where q_{se} is the number of **Send** queries.

5.1.5 Assumption

The CDH assumption can be stated by two experiments, $Exp1^{CDH-real}(\Phi)$ and $Exp2^{CDH-rand}(\Phi)$. Adversary Φ is obtained with xP, yP, xyP in the $Exp1^{CDH-real}(\Phi)$; and xP, yP, zP in the $Exp2^{CDH-rand}(\Phi)$, where $x, y, z \in Z_q^*$. The advantage of Φ in breaching the CDH assumption, $Adv_q^{CDH}(\Phi) = \max\{|Pr(Exp1^{CDH-real}(\Phi) = 1) - Pr(Exp1^{CDH-rand}(\Phi) = 1)|\}$

5.2 Security Proof

Theorem: Let proposed protocol Π runs the q_{se} number of **Send** queries, q_{ex} number of **Execute** queries, and q_{hash} number of hash queries. Then CDH assumption holds the following

$$Adv_{\Pi}^{protocol}(\mathcal{ATT}) \leq \frac{(q_{se} + q_{ex}^2)}{q} + \frac{q_{hash}}{2^l} + 2q_{ex}Adv_q^{CDH}(\Phi) + 4maxima\{\frac{q_{se} + q_{ex}}{2^l}, \frac{q_{hash}}{l}\}.$$

Proof: The proof has a combination of games, initiating from real attack G_1 and finishing at game G_5 where an attacker has no power. In each game, we set $Succ_i$ as event that \mathcal{ATT} knows e correctly in test query.

Game G_1 : This is the real attack by \mathcal{ATT} in protocol. In this game, the entire instances of participants are formed as real run/execution in ROM. As per the definition of $Succ_i$, we have

$$Adv_{\Pi}^{protocol}(\mathcal{ATT}) = |2Pr[Succ_1] - \frac{1}{2}| \tag{2}$$

Game G_2 : This is very similar game to *Game G_1* except the simulation of hash oracles h by constructing hash records h_{rec} with input/output entries. By executing h *inp* query, the output result is generated from the h_{rec} , otherwise randomly select the *Output* $\in \{0, 1\}^l$ and

transmit to the ATT with storing new entry of input/output in h_{rec} . Moreover, we simulate the oracles of the entire queries. As per the knowledge of ATT , the game G_2 is indistinguishable from real attack (game G_1). Therefore,

$$Pr[Succ_2] = Pr[Succ_1] \tag{3}$$

Game G_3 : Here, we simulate the entire instances of game G_2 , except we omit the game by which collisions may appear on transcripts as $(Msg_{UE}, Msg_{AMF}), (MAC_{UE}, MAC_{gNB_i})$, and hash values in the protocol. As per the definition of birthday paradox, in the result of h instances, the probability of collisions is $\frac{q_{hash}}{2^{l+1}}$. Also, collisions probability in the transcripts is no more than $\frac{(q_{se} + q_{ex}^2)}{2q}$. Therefore,

$$|Pr[Succ_3] - Pr[Succ_2]| \leq \frac{(q_{se} + q_{ex}^2)}{2q} + \frac{q_{hash}}{2^{l+1}} \tag{4}$$

Game G_4 : Here, we change queries to the $Send_Client$ instances. Also, select a random session initiated by legitimate clients UE and gNB_i for partner oracles Π_{UE}^i and $\Pi_{gNB_i}^j$.

- **Send_Client**($\Pi_{UE}^i, (gNB_i, Start)$) is requested and send output $SUCI, MAC_{UE}, RHI_{UE}$ to the ATT .
- **Send_Client**($\Pi_{UE}^i, (AUTN_{AUSF}, XRESV')$) is requested, randomly select $x \in Z_q^*$ and generates $N_{UE} = x.P$. Then, UE computes $MAC_{UE} = H_1(SUPI || x || RHI_{UE} || inau_{UE} || K_{gNB_i}^{UE})$ and $RHI_{UE} = E\{SUCI || ID_{gNB_i} || K_{gNB_i}^{UE} || T_{exp}\}$ as real protocol. Then, send the output as $N_{UE} || RHI_{UE} || MAC_{UE} || SUCI || inau_{UE}$ to ATT .
- **Send_Client**($\Pi_{gNB_i}^j, (N_{UE} || RHI_{UE} || MAC_{UE} || SUCI || inau_{UE})$) is requested and randomly select $y \in Z_q^*$ and generates $y.P = N_{gNB_i}$. Also, computes $MAC_{gNB_i} = H_2(ID_{gNB_i} || y || inau_{gNB_i})$ and $Key_{gNB_i}^{UE} = Key_{gNB_i} = KDF(Key_{gNB_i}^{UE} || ID_{gNB_i} || inau_{gNB_i} || N_{UE}.y) = xy.P$. Then, it sends the output $MAC_{gNB_i} || N_{gNB_i} || ID_{gNB_i} || inau_{gNB_i}$ to ATT .
- **Send_Client**($\Pi_{UE}^i, (MAC_{gNB_i} || N_{gNB_i} || ID_{gNB_i} || inau_{gNB_i})$) is requested, compute $Key_{UE} = xy.P$, $MAC_{cfm} = H_3(Key_{gNB_i}^{UE} || SUPI || ID_{gNB_i} || x.N_{gNB_i})$ and session key $Key_{gNB_i}^{UE}$ in real protocol. Then it send MAC_{cfm} to ATT .

Hence, it is observed that the game is indistinguishable from game G_3 . So,

$$Pr[Succ_4] = Pr[Succ_3] \tag{5}$$

Game G_5 : Here, we update the simulation queries of $Send_Client$ instances for randomly chosen session in G_3 . In this game, we choose another way for computing the value of Key_{gNB_i}, Key_{UE} so it will be autonomous for handover acknowledgment value and keys. When **Send_Client**($\Pi_{gNB_i}^j, (N_{UE} || RHI_{UE} || MAC_{UE} || SUCI || inau_{UE})$) and **Send_Client**($\Pi_{UE}^i, (MAC_{gNB_i} || N_{gNB_i} || ID_{gNB_i} || inau_{gNB_i})$) are requested $Key_{gNB_i} = Key_{UE} = T_z(\psi)$ (for UE and gNB_i), where $z \in Z_q^*$. The difference between game G_5 and G_4 is:

$$|Pr[Succ_5] - Pr[Succ_4]| \leq q_{ex} Adv_{\psi, q}^{CDH}(\Phi) \tag{6}$$

By considering a successful attacker ATT to analyze G_5 and G_4 , we make the CDH fixer Φ . The difference between G_5 and G_4 is the way of calculation of Key_{gNB_i}, Key_{UE} for chosen session. Firstly, Φ obtains the CDH value (xP, yP, Z) . As G_5 and G_4 , the fixer Φ chooses a

verifying session for Π_{UE}^i and $\Pi_{gNB_t}^j$ initiated legitimate clients UE and gNB_t respectively. When **Send_Client**($\Pi_{UE}^i, (AUTN_{AUSF}, XRESV')$) is requested, the Φ sets $N_{UE} = x.P$. In addition, when, **Send_Client**($\Pi_{gNB_t}^j, (N_{UE} || RHI_{UE} || MAC_{UE} || SUCI || inau_{UE})$) and **Send_Client**($\Pi_{UE}^i, (MAC_{gNB_t} || N_{gNB_t} || ID_{gNB_t} || inau_{gNB_t})$) are requested, Φ sets $y.P = N_{gNB_t}$ and $Key_{gNB_t}^{UE} = Z$.

The analyzer ATT selects a random session for the test queries (**Test**(Π_{UE}^i), **Test**($\Pi_{gNB_t}^j$)), then the probability is $\frac{1}{q_{ex}}$. Hence, the Φ simulates all instances query without having information of x, y . From this, analyzer ATT may generate $N_{UE} = x.P, y.P = N_{gNB_t}$ but not the correct Key_{gNB_t}, Key_{UE} . In case, $Z = xyP$, this setting for the analyzer is similar to G_4 . In case, $Z = zP$, this setting for the analyzer is similar to G_5 .

Lastly, if analyzer ATT interacts with G_4 , the fixer Φ decides that $Z = xyP$. And, if ATT interacts with G_5 , the fixer Φ decides that $Z \neq xyP$. Hence, eq. (6) holds. In this game, the keys Key_{gNB_t}, Key_{UE} are independent and random with secret keys. Therefore, three possibilities can be arises where an attacker analyzes the random and secret session keys as:

Case-1: Attacker queries ($zP, SUCI, ID_{gNB_t}$) to h . Then, this event obtains in $\frac{2q_{hash}}{l}$.

Case-2: Attacker requests **Send_Client** query excepting **Send_Client**($\Pi_{gNB_t}^j, m$) and impersonates UE to gNB_t . If an attacker, tries to impersonate UE in random session by generating MAC_{UE} and got success, it will make the discrepancy but the probability is less than to $\frac{1}{2^l}$. As there are maximum $2(q_{se} + q_{ex})$ sessions, then the total probability that this event is obtained will be less than to $\frac{2(q_{se} + q_{ex})}{2^l}$.

Case-3: Attacker requests **Send_Client** query excepting **Send_Client**(Π_{UE}^i, m) and masquerades the gNB_t to UE. Similar to *Case-2*;, the probability of this event is obtained less than to $\frac{2(q_{se} + q_{ex})}{2^l}$. Therefore, from above three cases;

$$|Pr[Succ_3]| = \frac{1}{2} + 2maxima\left\{ \frac{q_{se} + q_{ex}}{2^l}, \frac{q_{hash}}{l} \right\} \tag{7}$$

By combining the eq. from (1) to (7), the results are:

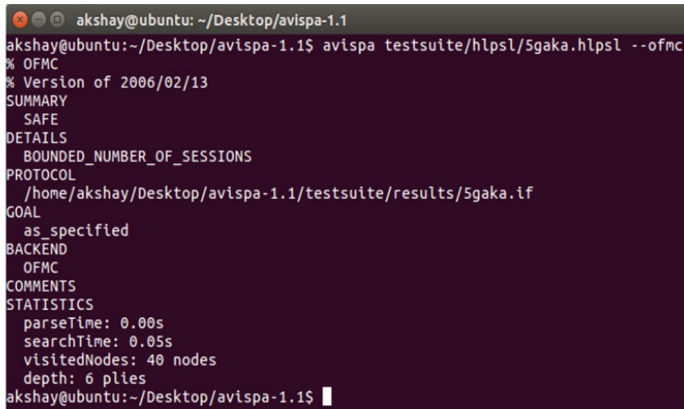
$$\begin{aligned} Adv_{\Pi}^{protocol}(ATT) &= 2Pr[Succ_1] - \frac{1}{2} | \\ &\leq (|Pr[Succ_2] - Pr[Succ_3]| + \\ &\quad |Pr[Succ_4] - Pr[Succ_5]| + \\ &\quad 2maxima\left\{ \frac{q_{se} + q_{ex}}{2^l}, \frac{q_{hash}}{l} \right\}) \\ &\leq \frac{(q_{se} + q_{ex}^2)}{q} + \frac{q_{hash}}{2^l} \\ &\quad + 2q_{ex}Adv_q^{CDH}(\Phi) + \\ &\quad 4maxima\left\{ \frac{q_{se} + q_{ex}}{2^l}, \frac{q_{hash}}{l} \right\} \end{aligned}$$

```

goal
  secrecy_of sec_ue_nuei, sec_gnb_ngnbi, sec_kuei_gnbs, sec_kuei_gnbt
  authentication_on gnb_uei
  authentication_on ue_gnbi
end goal

```

Fig. 5 Objectives of the SEAI handover protocol



```

akshay@ubuntu: ~/Desktop/avispa-1.1
akshay@ubuntu:~/Desktop/avispa-1.1$ avispa testsuite/hlpsl/5gaka.hlpsl --ofmc
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/akshay/Desktop/avispa-1.1/testsuite/results/5gaka.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 0.05s
  visitedNodes: 40 nodes
  depth: 6 plies
akshay@ubuntu:~/Desktop/avispa-1.1$

```

Fig. 6 Output of OFMC back-end

5.3 Correctness of the Protocol

The proposed SEAI-AKA handover protocol is simulated using the AVISPA tool to prove its correctness. The protocol is programmed coded in classic High-Level Protocol Specification Language (HLPSL) to define its characteristics [29]. AVISPA tool simulates the protocol in numerous backends as On-the-Fly Model Checker (OFMC) and SAT-based Model-Checker (SATMC). There are two participants titled gNB and UE in the protocol. We have programmed the fundamental role of these participants in HLPSL and simulated the mechanism by adopting the AVISPA tool. The HLPSL program of the communicating participants is demonstrated in the *Appendix-A*. Also, the objectives of the protocol are described in Fig. 5.

The simulation of the protocol is implemented by applying the OFMC backend with a restricted number of terms. Essentially, the OFMC simulates handover protocol, and then attacker fetches the information from preceding executions. Therefore, OFMC obtains the session complexity and avoids replay attack without executing different sessions between communicating participants. Also, OFMC checks whether the genuine participants can execute the protocol by seeking the passive attacker and broadcasts the instructions of a few sessions to the attacker between genuine participants [30]. The test outputs show that the protocol dodges replay attack. The output of OFMC back-end model is represented in Fig. 6. The keyword **SAFE** in result proves the correctness of the protocol. Moreover, the protocol averts from the MitM attack by adopting the tests of OFMC back-end. Therefore, the SEAI handover AKA protocol gains the essential security characteristics and dodges the known attacks from the 5G network.

5.4 Informal Analysis

In this section, we discuss various malicious attacks to show that the SEAI handover protocol is not vulnerable to the probable attacks.

- KFS/KBS:** To preserve the KFS/KBS, the secret keys must not be acknowledged in the preceding and successive sessions even if it is compromised. In the protocol, UE achieves the RHI_{UE} and $Key_{gNB_s}^{UE}$ from gNB_s and AMF respectively in a secure communication even if ATT generates the required public keys. Moreover, ATT aims to achieve MAC_{UE}/MAC_{gNB_t} for self-verification at any participant. However, ATT can't obtain these authentication values as n_{UE} and n_{gNB_t} are random values at unique communication of handover. ATT needs the information of private keys to generate the preceding and following session keys of $Key_{gNB_t}^{UE}$. However, it fails to obtain these values as ECDLP is computationally hard. Also, the protocol doesn't follow the key chain framework and interaction with gNB_s . Therefore, ATT will never have the information of earlier/subsequent private keys.
- Key Escrow Problem:** The UE or gNB_t select the secret keys in each handover authentication. To compute these secret keys, there is no association of the third party such as a key generation center (KGC)/private key generator (PKG). Therefore, the protocol avoids the key escrow problem.
- DoS Attack:** The ATT may transmit a large number of false handover requests to UE or gNB_t in the authentication stage to drain its network bandwidth. In the protocol, gNB_t obtains the $Key_{gNB_t}^{UE}$, MAC_{gNB_t} , and transfers the sequence message S_2 to the UE (as presented in Fig. 4). UE generates $Key_{gNB_t}^{UE}$ and authenticates MAC_{gNB_t} . After this, it sends the MAC_{cfm} to gNB_t . If the authentication is not successful, an authentication reject information is send to UE. As per the ECDLP infeasibility assumption, it is impractical for ATT to obtain the secret keys of the communicating participants. Hence, the proposed protocol avoids the DoS attack.
- Privacy-Preservation:** In the proposed protocol, UE transmits the SUCI to the ARPF followed by AMF as SUPI can't be transmitted over the communication channel and SUCI is applied to form this. The ARPF decrypts the SUCI value by SIDF. Hence, the identity of the UE is achieved in the proposed protocol. In addition, the ID_{gNB_s} is never transmitted from AMF to UE for computing the $Key_{gNB_s}^{UE}$, RHI_{UE} . Suppose, ATT computes the ID_{gNB_t} transmitted from gNB_t to UE and attempts to compute the bogus MAC_{gNB_t} . However, an attacker can't derive the private keys due to the computationally infeasibility assumption of ECDLP. Therefore, only legitimate UE can accept the ID_{gNB_t} from gNB_t .
- Replay Attack:** In the authentication stage of handover mechanism, replay attack couldn't be initiated as each corresponding message has the chosen private keys. Let consider, ATT transmits duplicate informations to gNB_t/UE . Then, the communicating participants instantly verify that the information is achieved previously by them as secret/random keys are unique in every communication of handover. Also, ATT couldn't obtain the genuine $Key_{gNB_t}^{UE}$. Therefore, the protocol dodges the replay attack.
- Redirection Attack:** The ATT can initiate the redirection attack if it masquerades/impersonates UE or maintains the bogus gNB correctly. Moreover, no ATT could decrypt the identity of UE excluding the ARPF. Therefore, it can't obtain the original identity of the UE. Also, ATT fails to obtain identity of gNB_t and compute

MAC_{gNB_t} . gNB_s sends the LAI to gNB_t when the UE arrives in the range of gNB_t . Hence, protocol averts the redirection attack from the 5G network.

- MitM Attack:** ATT can't implant the MitM attack at the authentication stage of protocol. It is noted that the $Key_{gNB_t}^{UE}$ is verified at UE and gNB_t successfully. Suppose, ATT corrupts the N_{UE} , N_{gNB_t} and generates the $N_{UE,ATT}$, $N_{gNB_t,ATT}$, where $N_{UE,ATT} = n_{UE,ATT} \cdot P$ and $N_{gNB_t,ATT} = n_{gNB_t,ATT} \cdot P$. Therefore, ATT generates the $N_{UE,ATT}$ at gNB_t but, the $Key_{gNB_t,ATT}^{UE}$ is not generated correctly as $Key_{gNB_t,ATT}^{UE} = KDF(Key_{gNB_s}^{UE} || ID_{gNB_t} || inau_{gNB_t} || N_{UE,ATT} \cdot n_{gNB_t})$. Similarly, ATT obtains $N_{gNB_t,ATT}$ at UE but, the $Key_{gNB_t,ATT}^{UE}$ is not generated correctly as $Key_{gNB_t,ATT}^{UE} = KDF(Key_{gNB_s}^{UE} || ID_{gNB_t} || inau_{gNB_t} || n_{UE} \cdot N_{gNB_t,ATT})$. As, the ATT doesn't have the information of UE's/ gNB_t secret key, so it is not possible for to obtain valid MAC_{UE} / MAC_{gNB_t} . Hence, ATT can't achieve the authentic handover message to execute MitM attack in the network.
- Eavesdropping Attack:** In the handover establishment stage, the UE and AMF authenticate to each other. AMF transmits the $Key_{gNB_s}^{UE}$ to gNB_s and then gNB_s broadcasts RHI_{UE} to the UE. The chosen secret keys are private in all over the handover operations. Hence, ATT couldn't compute the secret session keys even though he/she calculates the universal/public specifications of the UE and gNB_s . In the handover authentication stage, the universal and handover specifications are transmitted between gNB_t and UE in the public channel.

The analysis of SEAI handover AKA protocol and existing 5G protocols is presented in Table 2 based on numerous security characteristics. It can be defined that the current 5G handover protocol achieves the mutual authentication between the communicating participants in the authentication mechanism. Although, the protocol doesn't obtain the KFS/KBS and deteriorates from authentication complication. Also, the protocol fails to avoid DoS attack. The Cao's-AKA protocol doesn't obtain the KFS/KBS and defeats from DoS, redirection, and eavesdropping attack. Also, Sharma's-AKA protocol fails to achieve the key secrecy and avoid system complexity. Additionally, the protocol is vulnerable to redirection attack. Zhang's-AKA protocol can't preserve the identity during the handover authentication; hence, it is susceptible to several security attacks. Similar to Zhang's protocol,

Table 2 Comparative scrutiny of the handover protocols

Handover protocols	Security characteristics									
	SC_1	SC_2	SC_3	SC_4	SC_5	SC_6	SC_7	SC_8	SC_9	SC_{10}
5G Handover [11]	Yes	No	Yes	No	Yes	No	No	Yes	No	No
Cao's-protocol [13]	Yes	No	Yes	No	Yes	Yes	Yes	Yes	No	No
Sharma's-protocol [14]	Yes	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes
Zhang's-protocol [15]	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
Han's-protocol [16]	Yes	Yes	Yes	No	No	No	Yes	Yes	No	No
Kumar's-protocol [17]	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No
SEAI protocol	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

SC_1 : Establish mutual authentication; SC_2 : Retain KFS/KBS; SC_3 : Overcomes the key escrow problem; SC_4 : Defeats the DoS attack; SC_5 : Privacy-Preservation of the identity; SC_6 : Defeats the MitM attack; SC_7 : Avoids the authentication complexity; SC_8 : Defeats replay attack; SC_9 : Defeats the redirection attack; SC_{10} : Avoids the eavesdropping attack

Han's-AKA protocol has numerous security weaknesses and can't establish identity privacy preservation. Furthermore, Kumar's-AKA protocol obtains most of the security characteristics but can't prevent the MitM and eavesdropping attack from the communication network. Different from the current protocols, the proposed SEAI handover AKA protocol executes the key procedures adopting the ECC. The protocol accomplishes the KFS/KBS in the authentication mechanism. Moreover, the protocol resist all the potential attacks and free from the authentication complication. Therefore, the proposed protocol is relatively better compared to the existing protocols as it gains all the crucial security characteristics.

6 Performance Estimation

The performance of the proposed SEAI handover AKA protocol is estimated for the existing 5G handover schemes in terms of computation, communication, and transmission overhead. Additionally, we compute the handover delay, key size, and energy consumption for the handover protocols based on various parameters. The analysis represents that the proposed protocol gains all security objectives with adequate competence.

6.1 Computation Overhead

For the estimation of computation overhead of handover protocols at the handover authentication and initialization stage, elapsed time of various security functions is executed at OpenSSL written in C library [31] operating on 4 GB memory machine with Intel Core i5-7200U 4 GHz processor for gNB and 2.50 GHz processor for UE. Hence, the elapsed time (in ms) is: point multiplication (T_{pmul})= 0.441, hash (T_{hh})=0.0087, AES encryption/decryption (T_{aes})=0.071, modular exponentiation (T_{moe})=0.629, arithmetic operation (T_{art})=0.0021, multiplication operation (T_{mul})=0.0033 (for gNB); T_{pmul} : 1.023, T_{hh} =0.0194, T_{aes} =0.109 ms, T_{moe} =1.277 ms, T_{art} =0.0074 ms, T_{mul} =0.0091 ms (for UE). The computational overhead of current and proposed handover protocols is presented in Table 3. Also, the graphical presentation is shown for the comparison of handover protocols in terms of computation overhead in Figs. 7 and 8.

The current 3GPP-5G handover protocol accepts the hash operations and symmetric cryptography that generates the overhead at each communicating participant in inter-gNB handover. However, the protocol fails to avoid the de-synchronization that derives the DoS attack and complex handover process. In the Cao's-AKA protocol, UE and base-station execute the hash operation for integrity and AES for encryption/decryption operations. The protocol shows less overhead compared to the proposed scheme however, Cao's handover protocol is not secure against eavesdropping and redirection attacks. Also, the Han's-AKA protocol has less computation overhead compared to the SEAI handover AKA protocol as it executes only hash operations during handover operations but suffers from DoS and MitM attack. Both the Zhang's-AKA and Kumar's-AKA protocol operate the handover authentication using point multiplication, arithmetic, and multiplication operations. Moreover, the Sharma's-AKA protocol execute the handover authentication by time-consuming modular exponentiation operations. Hence, these protocols aren't recommended for the development of efficient handover authentication protocol in the 5G communication network. Different from above schemes, the proposed SEAI handover AKA protocol establishes mutual authentication and key agreement between the gNB_i and UE by adopting

Table 3 Estimated analysis of handover protocols

Handover protocols	Computation and communication overhead			
	$COMM^{UE}$ (in bits)	$COMM^{gNB}$ (in bits)	$COMM^{UE}$ (in ms)	$COMP^{\lambda}$ (in ms)
5G handover [11]	-	1152	-	$2T_{hh}$
Cao's- <i>AKA</i> protocol [13]	384	640	$3T_{aes} + 4T_{hh}$	$3T_{aes} + 3T_{hh}$
Sharma's- <i>AKA</i> protocol [14]	1044	832	$2T_{moe} + 4T_{hh}$	$3T_{moe} + 4T_{hh}$
Zhang's- <i>AKA</i> protocol [15]	1124	1124	$4T_{pmul} + 2T_{hh} + 2T_{art} + 3T_{mul}$	$2T_{pmul} + 3T_{hh} + 3T_{art} + 3T_{mul}$
Hant's- <i>AKA</i> protocol [16]	636	448	$4T_{hh}$	$3T_{hh}$
Kumar's- <i>AKA</i> protocol [17]	1048	2412	$3T_{pmul} + 2T_{hh} + 2T_{art} + 2T_{mul}$	$2T_{pmul} + 4T_{hh} + 4T_{art} + 3T_{mul}$
SEAI Protocol	832	512	$T_{pmul} + 2T_{hh}$	$T_{pmul} + 2T_{hh}$

$COMM^{UE}$: UE's communication overhead in initial authentication stage; $COMM^{gNB}$: gNB's communication overhead in initial authentication stage; α : Communication overhead at inter-gNB handover; $COMP^{UE}$: UE's computation overhead in initial authentication stage; $COMP^{gNB}$: gNB's computation overhead in initial authentication stage; β : UE's computation overhead in inter-gNB handover; λ : gNB's computation overhead in inter-gNB handover

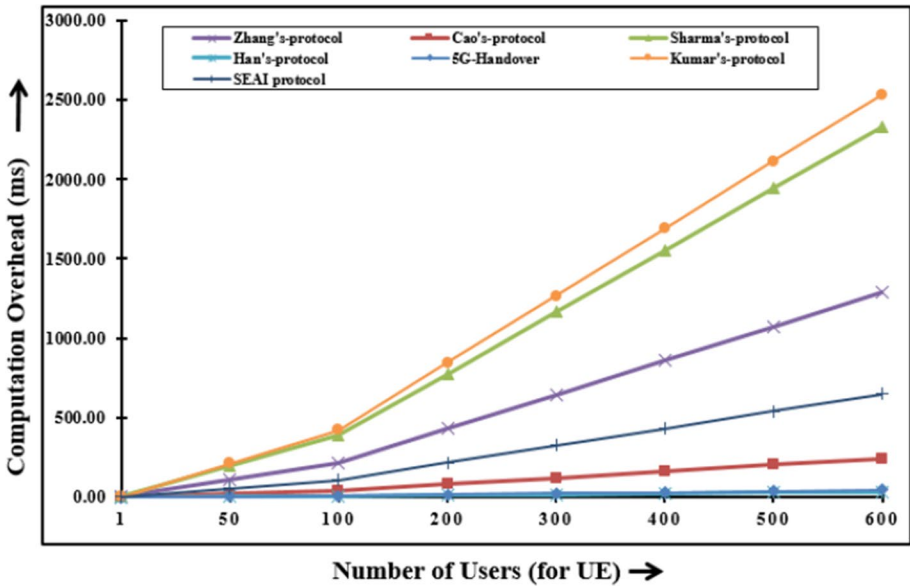


Fig. 7 Computation overhead of handover protocols at UE

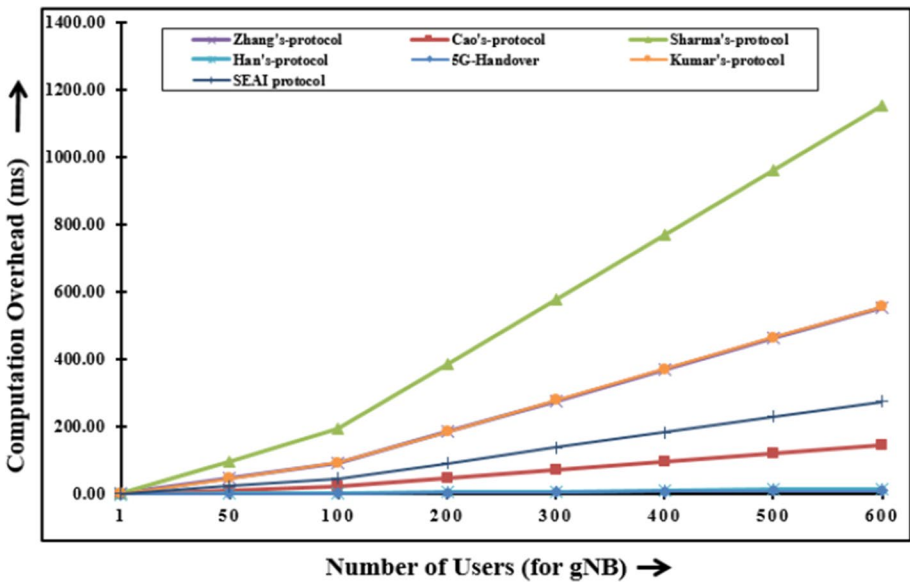


Fig. 8 Computation overhead of handover protocols at gNB

point multiplication operation. Moreover, the protocol avoids the loss of key secrecy and potential security susceptibilities. Hence, it obtains a significant security & privacy compared to the current handover schemes with competitive overhead.

6.2 Communication Overhead

In order to measure the communication overhead of the current and proposed protocols, we fix $|p| = 1024$ and $|q| = 256$ because the ECC key indicates identical security. The $|n| = |\#E(F_n)| = 256$ and $E(F_n):\#E(F_n) = 256$ bits prime order q . Moreover, Table 4 represents the specification list and their costs/value [32]. To estimate the overhead, we measure the broadcasted information between the communicating participants in the current and proposed handover AKA protocols. In Table 3, the overhead of the protocols is measured. Also, the graphical presentation is shown for the comparison of handover protocols in terms of communication overhead in Fig. 9.

Although, the overhead of the SEAI handover AKA protocol is larger than the 3GPP-5G handover mechanism. However, the 3GPP-5G protocol deteriorates from key negotiation issue, DoS attack, and authenticity complexity. In the Cao's-AKA protocol, UE communicates to the target and future base-station for accomplishing mutual authentication respectively. The UE and base-stations share the message authentication codes, capability messages, and handover tickets in 1884 bits. Although, the protocol incurs less communication overhead during the handover initialization stage compared to SEAI handover AKA scheme because keys and identity are generated directly from the handover module. Also, the protocol suffers from lack of forward key secrecy and DoS attack. In Sharma's-AKA protocol, the terminal and new/previous hub communicate with each other during handover authentication. The terminal transmits the sequence number, message authentication code, and various handover request/response. At the same time, the authentication server communicates with new and previous hubs in 2978 bits. Han's-AKA protocol follows the EAP-AKA scheme during the initial authentication of UE and base-station. In the handover stage, the UE and base-station obtain the authentication parameters and use additional counter hash values. Also, the protocol fails to preserve the identity during the authentication process.

The Zhang's-AKA protocol establishes mutual authentication between the communicating participants. Firstly, UE transmits its one-time trapdoor hash key, secret, public keys, expiration time, and identity. Then, the target base-station sends its handover specifications to the UE with a shared secret key, and UE approves handover acknowledgment by transmitting the secret key. Similar to Zhang's-AKA protocol, Kumar's-AKA protocol accomplishes mutual authentication between the communicating participants. Firstly, UE transmits its secret, public keys, passwords, and pseudo-identity. Then, the target base-station sends its random number, secret keys, and public parameters to UE with a shared secret

Table 4 Specifications for communication overhead

Specifications	Cost (in bits)
$SUCI/PLMN_{ID}/ID_{gNB}/ECV/PCI$	128
$Key_{NG-RAN}^*/Key_{NG-RAN}^{**}$	256
$Key_{gNB_1}^{UE}/Key_{gNB_1}^{UE}$	128
$Key_{AUSF}/Key_{SEAF}/NH_{NCC}/Key_{AMF}$	256
RES/XRES	160
$N_{re}/Timestamp(T_{cur}/T_{exp})$	64
LAI/POS/NAI	40
MAC/CMAC/Hash	256

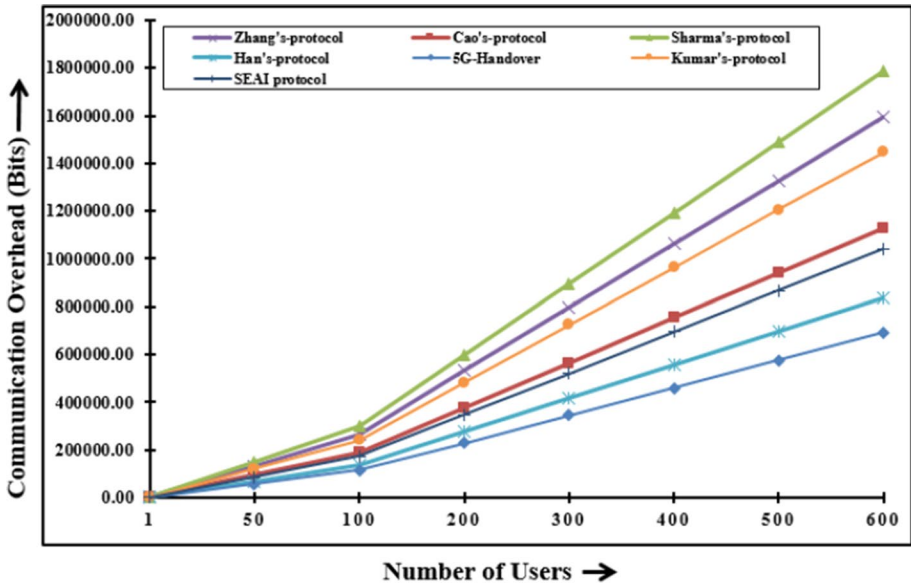


Fig. 9 Communication overhead of handover protocols

key, and UE accepts the handover message successfully. The prime objective of the proposed SEAI handover AKA protocol is to avoid the overhead at the communicating participants and evolve the security capabilities at the time of handover. Hence, we designed the handover protocol by adopting the ECC procedure. Our protocol setups the session key secrecy and $Key_{gNB_t}^{UE}$ is attained between gNB_t and UE without any ambiguous handover system. The UE and gNB_t maintain the secure mutual authentication in the protocol and there is no transmission of the secret session key in the public channel. Thus, the protocol is very efficient and secure compared to the current handover schemes.

6.3 Transmission Overhead

It is studied that the conventional cost of the message authentication between i) gNB_s/gNB_t and UE is ρ unit; ii) gNB_s and gNB_t is σ unit; and iii) AMF and gNB_s/gNB_t is Δ unit to measure transmission overhead of the proposed and current handover protocols. As the gNB is implanted a very long distance from AMF; hence the overhead of σ unit has the scope as $0 < \sigma < \rho$. Also, the overhead of ρ is greater than the cost of Δ . The transmission overhead of proposed and existing handover AKA protocols is demonstrated in Table 5. Hence, it is noticed that the overhead of proposed SEAI handover AKA protocol is less compared to most of the existing protocols. Although, Kumar’s scheme has less transmission overhead but suffers from huge communication and computation overhead because of additional point multiplication operations during handover. In the handover authentication stage of proposed protocol, 3 communication messages are required between gNB_t and UE. Although, only 2 messages are enough to establish mutual authentication between gNB_t and UE. The third correspondence message is transmitted from the UE to approve the handover key agreement with gNB_t .

Table 5 Transmission overhead of protocols

	5G handover [11]	Cao's-AKA proto-col [13]	Sharma's-AKA protocol [14]	Zhang's-AKA proto-col [15]	Han's-AKA proto-col [16]	Kumar's-AKA protocol [17]	SEAI protocol
$TO_{gNB_s/gNB_t - AMF/SN}^u$	2ρ	5ρ	4ρ	3ρ	3ρ	2ρ	2ρ
TO_{UE-gNB_s/gNB_t}^v	3σ	6σ	12σ	3σ	3σ	3σ	4σ
$TO_{gNB_s-gNB_t}^z$	2Δ	2Δ	2Δ	0	0	0	0

u: Transmission overhead between $gNB_s/gNB_t - AMF/SN$; *v*: Transmission overhead between gNB_s/gNB_t and UE; *z*: Transmission overhead between gNB_s and gNB_t

6.4 Handover Delay

In this section, the handover delay is computed for the proposed SEAI handover AKA protocol and other existing schemes when the user is executing various handover between base-station/nodes. The handover delay for each handover scheme in A by parameter HD^A as $f_{HD_m^A}^*(s) = \sum_{t \in T^A} P_t f_{HD_t^A}^*(s)$ [33, 34]. In this scenario, t is the authentication or re-authentication process that is executed in each scheme. P_t is the ratio for executing the mechanism t , and T^A is the handover scheme. Here, suppose A is the A_{5G} then $T^A = \{gNB_s, gNB_t, gNB_s, gNB_t, \dots\}$, and A is the A_{Cao} then $T^A = \{BS_2, BS_3, BS_2, BS_3, \dots\}$, and A is the A_{Sharma} then $T^A = \{pHub, nHub, pHub, nHub, \dots\}$. Also, A is the A_{Zhang} then $T^A = \{AP_t, AP_t, AP_t, AP_t, \dots\}$, and A is the A_{Han} then $T^A = \{BS_t, BS_t, BS_t, BS_t, \dots\}$, A is the A_{Kumar} then $T^A = \{AP, MBS, AP, MBS, \dots\}$, and A is the A_{SEAI} then $T^A = \{gNB_t, gNB_t, gNB_t, gNB_t, \dots\}$. Furthermore, ϕ_t^A is the set that has the delay factors in the protocol A . The Laplace transformation of HD^A is $f_{HD_t^A}^*(s) = \sum_{i \in \phi_t^A} HD_i(s) = (\prod_{i \in \phi_t^A} f_{HD_i}^*)$. The Laplace transformation of HD^{5G} is $f_{HD^{5G}}^*(s) = f_{HD^{5G}}^*(s) + HD_{gNB_s}^{5G}(c)$, HD^{Cao} is $f_{HD^{Cao}}^*(s) = f_{HD^{Cao}}^*(s) + HD_{BS_2}^{Cao}(s)$, and HD^{SEAI} is $f_{HD^{SEAI}}^*(s) = f_{HD^{SEAI}}^*(s) + HD_{gNB_t}^{SEAI}(s)$. Additionally, the Laplace transformation of HD^A can be written as $E(HD^A) = \int_0^\infty f_{HD^A}(x)dx$ [35]. For the handover AKA protocols, it can be written as $E(HD^{5G}) = -\frac{d}{ds} f_{HD^{5G}}^*(s)|_{s=0}$.

Figure 10 represents the handover delay of the SEAI handover AKA protocol and existing schemes concerned by increasing the hop count between the base-station/nodes and server. The handover delay of the proposed protocol is far less compared to the existing schemes because of executing a similar re-authentication process in each hop. Figure 11 shows the performance of the SEAI handover AKA protocol compared to the existing schemes in terms of the number of users and handover delay in milliseconds. As the

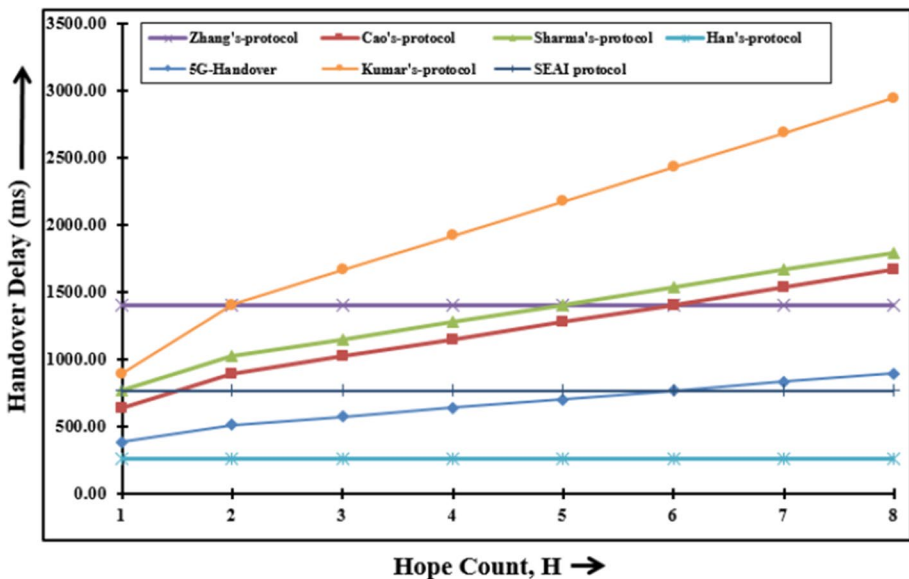


Fig. 10 Handover delay with hop count

number of users is increasing in each scheme, the handover delay is also increased. The proposed protocol obtains comparatively less handover delay to the Kumar's, Sharma's, Cao's, and Zhang's handover schemes. The proposed SEAI handover AKA scheme reduces the handover delay by 14%, 25%, 30%, and 60% compared to Kumar's, Sharma's, Cao's, and 3GPP-5G handover AKA schemes respectively.

6.5 Key Size

In this section, the size of the key is determined which are computed at the execution of handover AKA schemes. The size of computed and transferred keys has an important impact on the storage overhead as other parameters such as private/public key pair, time-stamp, identification parameters have a similar impact compared to an alternative approach. The sum of the key size is calculated for all the handover AKA protocols based on hop count as shown in Fig. 12. From, the Fig. 12, it is observed that the SEAI handover AKA protocol has a very competitive key size with an increasing number of hop counts compared to Han's protocol. The key size of the SEAI handover AKA protocol will be the same with an increasing number of hop counts. In the Kumar's, Cao's, and Sharma's handover AKA schemes, the key size is larger compared to the other protocols, and key size is increased at the following re-authentication processes. Additionally, in the Kumar's, Cao's, and Sharma's handover AKA protocols, the users roam to the previously visited base-station/node (hops (H) 2 to 8), and some additional keys may be generated in the home server and during the re-authentication process. Also, the keys are generated and stored at every hop count.

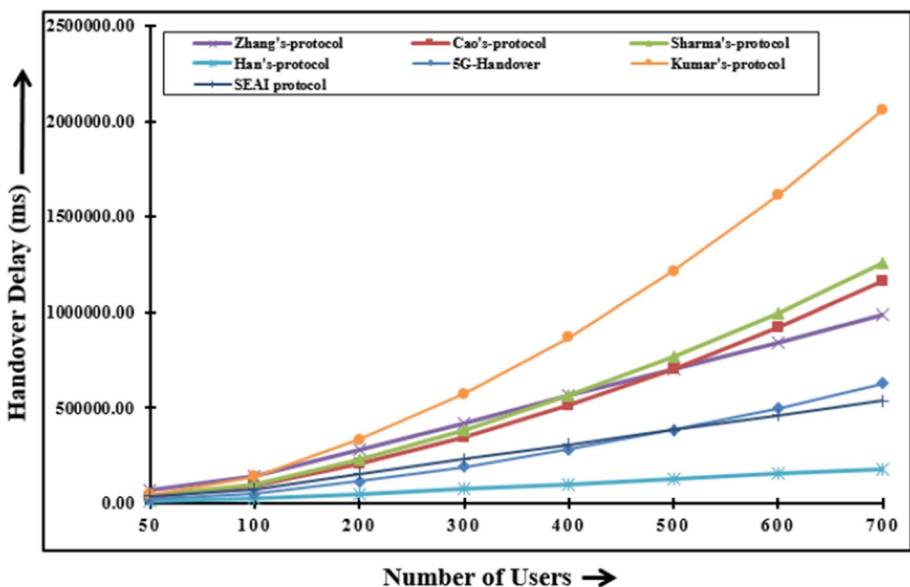


Fig. 11 Handover delay with number of users

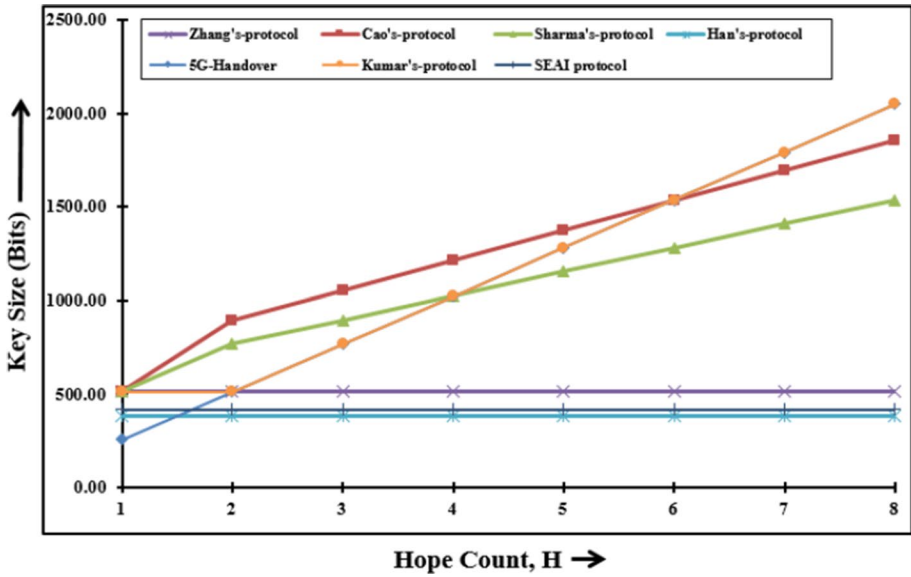


Fig. 12 Key size with hop count

Similarly, the Figs. 13 and 14 represent the key size of the handover AKA protocols for the number of users and user movements. Also, it can be noticed that the SEAI handover AKA protocol has far better key size results compared to the existing handover schemes.

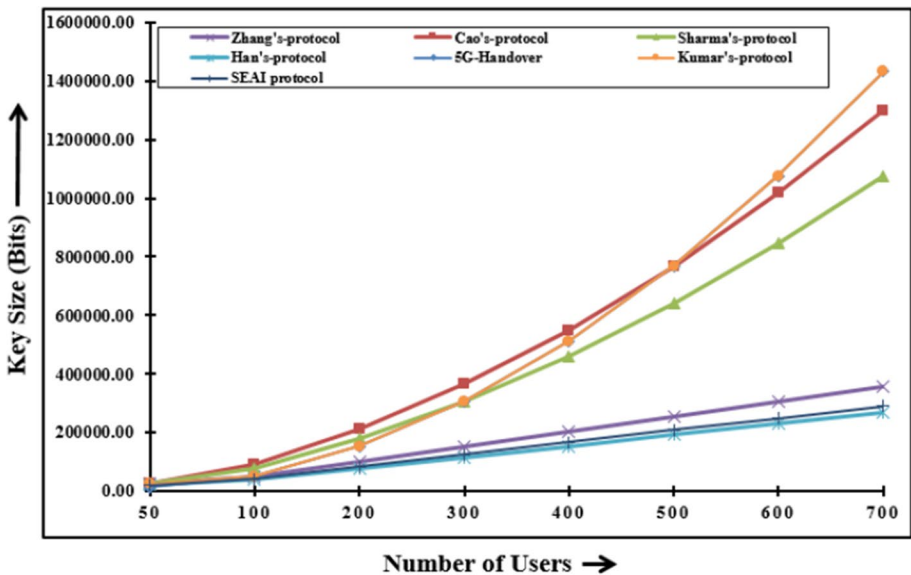


Fig. 13 Key size with number of users

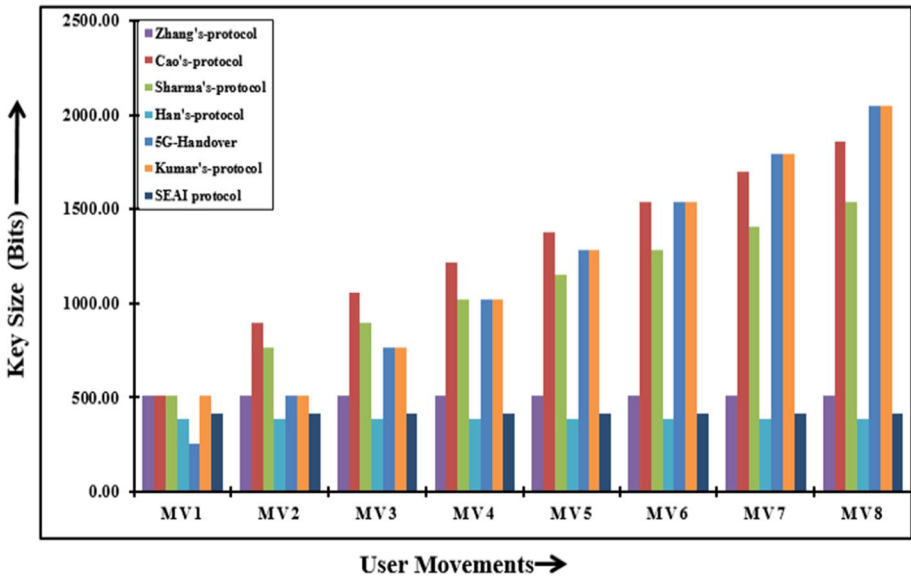


Fig. 14 Key size during user movements

6.6 Average Handover Cost

To evaluate the average handover cost of the handover AKA protocols, the wireless network model and mobility model are adopted as per [36, 37] respectively. It is considered that the network model is the 5G, WLAN-5G inter-networking domain and sizes of each subnet are similar. The average handover rate (α_j) is calculates as $\alpha_j = (v.P(i))/(\Pi.L(i))$, where j is the user group index, v is the UE’s average velocity (varies from 2 to 4km/h) in the 5G and WLAN-5G communication network. The perimeter $P(i)$ of the respective network can be computed as $P(i) = (12i + 6).R$. Here i is the cells number, R is the radius of subnet. The roaming area $L(i)$ is computed as $L(i) = (2.6R^2)(3i(i + 1) + 1)$. Therefore, the average handover cost (AHC) can be calculated as $AHC_t = \alpha_j.C_t$. The cost of each scheme $C_t = C_{t,s} + C_{t,p}$, where $C_{t,s}$ and $C_{t,p}$ is the signaling and processing cost respectively. The $AC_{t,s}$ for each scheme can be computed for each handover protocol as:

$$\begin{aligned}
 SEAI_{C_{t,s}} &= 3C_{ws} + 1H \\
 5G_{C_{t,s}} &= 5C_{ws} + 2H \\
 Cao_{C_{t,s}} &= 8C_{ws} + 2H \\
 Sharma_{C_{t,s}} &= 12C_{ws} + 2H \\
 Zhang_{C_{t,s}} &= 4C_{ws} + 1H \\
 Han_{C_{t,s}} &= 8C_{ws} + 1H \\
 Kumar_{C_{t,s}} &= 3C_{ws} + 2H
 \end{aligned}$$

where $C_{t,s}$ is the transmission cost of wireless links. The calculation of each scheme $C_{t,p}$ is the execution cost of each node $C_{n,p}$. For example, $C_{t,p}$ for 3GPP-5G handover scheme can be shown as $C_{5G,p} = C_{UE,p} + C_{gNB_s,p} + C_{gNB_r,p}$, where, $C_{UE,p} = 4C_{Key} + C_{Enc} + C_{Dec} + C_{Ver}$,

$C_{gNB_s,p} = 2C_{Key} + C_{Hash}$, and $C_{gNB_r,p} = C_{Key} + C_{Enc} + C_{Dec} + C_{Ver}$. The $C_{Key}, C_{Enc}, C_{Dec}, C_{Ver}, C_{Hash}$ are the costs of key computation, encryption, decryption, verification, and hash operation respectively. Therefore, $C_{t,p}$ for all the handover AKA schemes can be computed as:

$$\begin{aligned}
 SEAI_{C_{t,p}} &= 3C_{Key} + C_{Enc} + C_{Dec} + 2C_{Ver} + 7C_{Hash} \\
 5G_{C_{t,p}} &= 7C_{Key} + 2C_{Enc} + 2C_{Dec} + 2C_{Ver} + C_{Hash} \\
 Cao_{C_{t,p}} &= 7C_{Key} + 3C_{Enc} + 3C_{Dec} + 3C_{Ver} + 7C_{Hash} \\
 Sharma_{C_{t,p}} &= 8C_{Key} + 2C_{Enc} + 2C_{Dec} + 2C_{Ver} + 8C_{Hash} \\
 Zhang_{C_{t,p}} &= 5C_{Key} + 2C_{Enc} + 2C_{Dec} + 2C_{Ver} + 4C_{Hash} \\
 Han_{C_{t,p}} &= 6C_{Key} + 2C_{Enc} + 2C_{Dec} + 2C_{Ver} \\
 Kumar_{C_{t,p}} &= 7C_{Key} + 2C_{Enc} + 2C_{Dec} + 2C_{Ver} + 6C_{Hash}
 \end{aligned}$$

The value of i is considered 10, C_{ws} is set to 10. The costs such as $C_{Key}, C_{Enc}, C_{Dec}, C_{Ver}, C_{Hash}$ are set to one unit. The results achieved from the handover cost evaluations of each schemes are shown in Figs. 15, 16, and 17 at varying value of v from 2 to 4km/h. Also, the value of R is 0.1 km and H is 1 to 7 hop count. As the values of v and H increase, the average cost of existing handover AKA schemes is also increases compared to the SEAI handover AKA protocol. Therefore, the proposed protocol can be recommended for the IoT-enabled services in various handover scenarios as the handover cost is significantly reduced. Moreover, the AHC increases from 60 to 357 when H increases from 1 to 7 in the 3GPP-5G handover AKA scheme. However, the AHC remains the same with varying values of v and H in the proposed scheme. The reduction of handover cost in the SEAI handover AKA

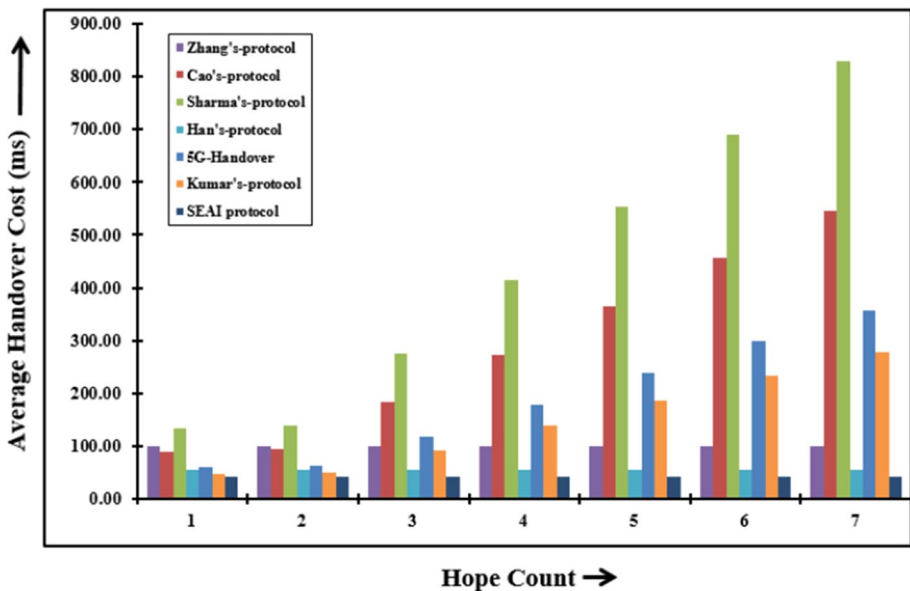


Fig. 15 Average handover cost at $v = 2$ km/h

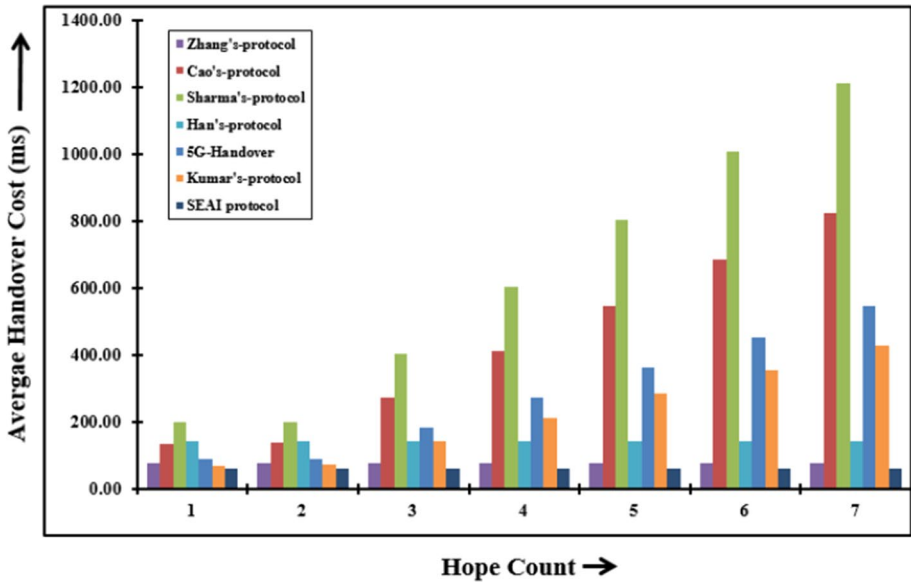


Fig. 16 Average handover cost at $v = 3$ km/h

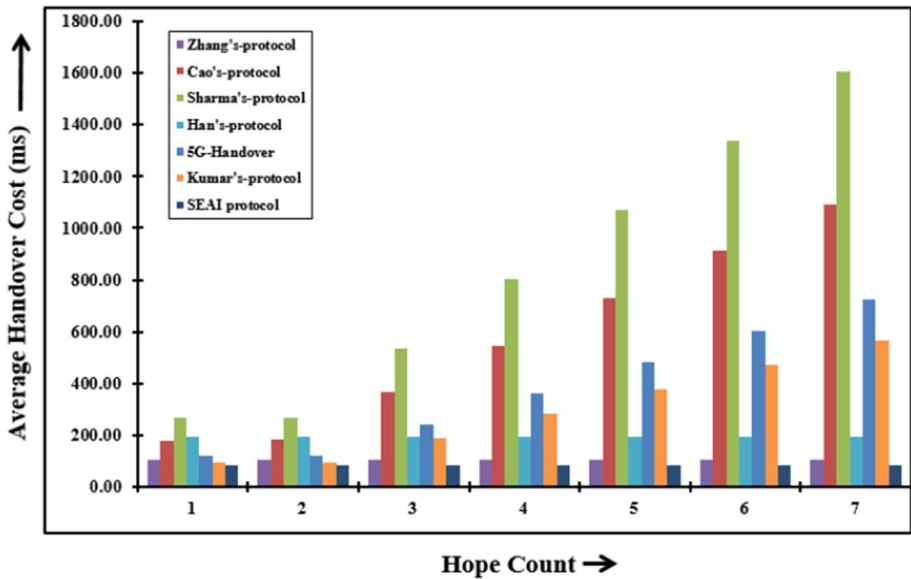


Fig. 17 Average handover cost at $v = 4$ km/h

scheme raises 34%, 23%, and 15% compared to the 3GPP-5G, Cao's, Sharma's handover AKA protocol respectively.

6.7 Energy Consumption

The current cellular networks manage massive users; hence, the computation of energy consumption is one of the essential performance estimation metrics. The reduction of the computed keys and exchanged messages at the authentication process represent the energy consumption [38, 39]. Generally, the total energy consumption in wireless networks can be computed as $Total_{Energy} = N.M + FC$, where N is the total bits transmitted/received by the UE, M is the incremental value, and FC is the fixed cost. The fixed and incremental value are coefficients which are obtained in [40]. The energy consumption is computed as per number of bits received and transmitted by the UE as $Energy_{trans} = 0.48N + 431$; $Energy_{rec} = 0.12N + 316$. The above-mentioned equations are adopted to compute the energy consumed by UE in each user movement. The calculations are utilized in the proposed and existing handover AKA protocols. For instance, the energy consumption of SEAI handover AKA scheme is $Energy_{trans}=1088$; $Energy_{rec}=928$. Figure 18 shows that the energy consumption in the previously proposed handover schemes is increased when UE roams into another base-station/node (inter/intra handover) in the 5G or WLAN-5G communication networks. Moreover, the proposed handover AKA scheme reduces the energy consumption 78%, 31%, and 54% compared to the Cao's, Sharma's, and Kumar's protocol respectively.

7 Conclusion

In this article, we introduced the secrecy and efficiency aware inter-gNB handover AKA protocol in 5G communication network to avoid the potential security susceptibilities as key negotiation, DoS & bogus base-station attack, and huge authentication complexity. In the proposed SEAI handover AKA protocol, mutual authentication is accomplished with a

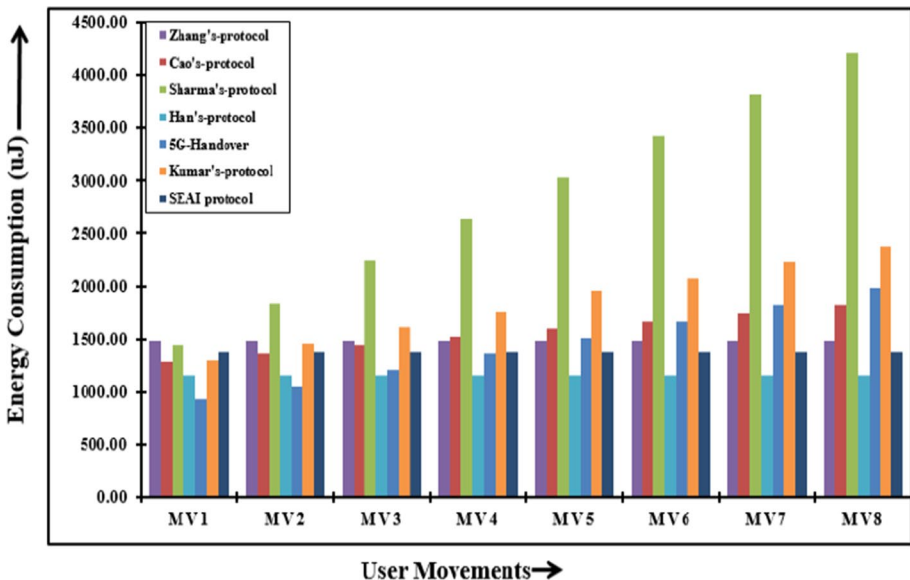


Fig. 18 Energy consumption in user movement

secret key between gNB and UE. Also, the protocol forms the forward/backward secrecy and averts the network complexities. In addition, simulation of the protocol is presented by the AVISPA tool to prove the correctness. To obtain the session key secrecy, confidentiality, and integrity, the formal security proof of the protocol is carried out by the ROM. The security analysis is examined with corresponding numerous security specifications and obtains the security across potential attacks. The performance estimation clarifies that the protocol is far valuable compared to the current 5G handover schemes based on various overhead analysis. Also, the handover delay, key size, and energy consumption of the proposed SEAI handover AKA protocol are very much competitive compared to the existing handover schemes. Hence, we expect that the proposed protocol will enhance the performance and security of the 5G communication network in numerous handover applications.

Appendix: Fundamental Role of the Communicating Participants

Listing 1: HLPSTL code for UE

```

role ue(U, G:agent,
        SND, RCV: channel(dy),
        U_NUE, G_NgNB: public_key,
        P, RHLUE, K_UE_GNBs, K_UE_GNBt: text,
        H1, H2, H3, KDF: function)
played_by U def=

local
  State : nat,
  IDgNBs, IDgNBt, SUCI, Inau_ue, Inau_gNBt,
  N_ue, N_gNBt: text

const sec_ue_nuei, sec_gnb_ngnbi,
sec_kuei_gnbs, sec_kuei_gnbt,
uei_gnb, gnb_ue : protocol_id,
success: text

init State := 0

transition

1. State = 0 /\ RCV(start) =>
State' := 1 /\ N_ue' := new()
           /\ SND({SUCI.N_ue'.P.H1(SUCI.
           N_ue'.Inau_ue.RHLUE.
           K_UE_GNBs).Inau_ue.RHLUE})_
           (inv(U_NUE)))
           /\ secret(N_ue', sec_kuei_gnbs,
           {U,G})
           /\ witness(G, SUCI, ue_gnbi,
           RHLUE)

2. State = 1 /\ RCV({IDgNBt.N_gNBt'.P.
Inau_gNBt.H2(IDgNBt.N_gNBt'.
Inau_gNBt)})_(inv(G_NgNB))=>
State' := 2
           /\ SND(H3(K_UE_GNBt.SUCI.IDgNBt.
           N_ue'.N_gNBt.P))
           /\ secret(K_UE_GNBt,
           sec_gnb_ngnbi, {U,G})
           /\ secret(N_ue', sec_kuei_gnbt,
           {U,G})
           /\ witness(G, U, gnb_uei, N_gNBt')

3. State = 2
State' := 3 /\ RCV(success) =>
end role

```

Listing 2: HLPSTL code for *gNB*

```

role gnb(G, U:agent,
        SND, RCV: channel(dy),
        U_NUE, G_NgNB: public_key,
        P, RHLUE, K_UE_GNBs, K_UE_GNBt: text,
        H1, H2, H3, KDF: function)
played_by G def=

local
  State : nat,
  IDgNBs, IDgNBt, SUCI, Inau_ue, Inau_gNBt,
  N_ue, N_gNBt: text

const sec_ue_nuei, sec_gnb_ngnbi, sec_kuei_gnbs,
sec_kuei_gnbt, ue_gnbi, gnb_uei : protocol_id,
success: text
init State := 0
transition

1. State = 0 /\RCV({SUCI.N_ue'.P.H1(SUCI.N_ue'.
Inau_ue.RHLUE.K_UE_GNBs).
Inau_ue.RHLUE})_(inv(U_NUE))=>
State' := 1 /\N_gNBt' := new()
           /\SND({IDgNBt.N_gNBt'.P.Inau_gNBt.
           H2(IDgNBt.N_gNBt'.Inau_gNBt)})_
           (inv(G_NgNB))
           /\secret(N_gNBt', sec_gnb_ngnbi,
           {G,U})
           /\secret(IDgNBt, sec_kuei_gnbt,
           {G,U})
           /\witness(G,U, gnb_uei, K_UE_GNBt)

2. State = 1 /\RCV(H3(K_UE_GNBt.SUCI.IDgNBt.
N_ue'.N_gNBt.P)) =>
State' := 2 /\SND(success)
           /\request(G,U, gnb_uei, N_ue')
end role

```

References

- Li, S., Da Li, X., & Zhao, S. (2018). 5G internet of things: A survey. *Journal of Industrial Information Integration*, 10, 1–9.
- Cao, J., Ma, M., Li, H., Ma, R., Yunqing Sun, P. Y., & Xiong, L. (2019). A survey on security aspects for 3GPP 5G networks. *IEEE Communications Surveys & Tutorials*, 22(1), 181–186.
- Zhang, S., Wang, Y., & Zhou, W. (2019). Towards secure 5G networks: A survey. *Computer Networks*, 162, 106871.
- 3GPP. (2018). 3GPP technical specification; security architecture and procedures for 5G system. https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/15.02.00_60/ts_133501v150200p.pdf.
- Zhang, X., Kunz, A., & Schröder, S. (2017). Overview of 5G security in 3GPP. In *IEEE conference on standards for communications and networking (CSCN)* (pp. 181–186).
- Agiwal, M., Roy, A., & Saxena, N. (2016). Next generation 5G wireless networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 18(3), 1617–1655.
- Khanna, A., & Kaur, S. (2020). Internet of things (IoT), applications and challenges: A comprehensive review. *Wireless Personal Communications*, 114, 1687–1762.
- Ullah, I., & Youn, H. Y. (2020). Intelligent data fusion for smart IoT environment: A survey. *Wireless Personal Communications*, 114(1), 409–430.
- Goudos, S. K., Dallas, P. I., Chatziefthymiou, S., & Kyriazakos, S. (2017). A survey of IoT key enabling and future technologies: 5G, mobile IoT, semantic web and applications. *Wireless Personal Communications*, 97(2), 1645–1675.
- Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., & Mustaqim, M. (2020). Internet of things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios. *IEEE Access*, 8, 23022–23040.
- 3GPP. (2020). 3GPP technical specification; security architecture and procedures for 5G system. https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/15.07.00_60/ts_133501v150700p.pdf.
- 3GPP. (2020). 3GPP technical specification; security architecture and procedures for 5G system. Retrieved from https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/15.06.00_60/ts_133501v150600p.pdf.
- Cao, J., Ma, M., Fu, Y., Li, H., & Zhang, Y. (2019). Cppha: Capability-based privacy-protection handover authentication mechanism for SDN-based 5G HetNets. *IEEE Transactions on Dependable and Secure Computing*, 18(3), 1182–1195.
- Sharma, V., You, I., Leu, F.-Y., & Atiqzaman, M. (2018). Secure and efficient protocol for fast handover in 5G mobile Xhaul networks. *Journal of Network and Computer Applications*, 102, 38–57.
- Zhang, Y., Deng, R., Bertino, E., & Zheng, D. (2019). Robust and universal seamless handover authentication in 5G HetNets. *IEEE Transactions on Dependable and Secure Computing*, 18(2), 858–874.
- Han, K., Ma, M., Li, X., Feng, Z., & Hao, J. (2019). An efficient handover authentication mechanism for 5G wireless network. In *IEEE wireless communications and networking conference (WCNC)* (pp. 1–8).
- Kumar, A., & Om, H. (2019). Design of a USIM and ECC based handover authentication scheme for 5G-WLAN heterogeneous networks. *Digital Communications and Networks*, 6(3), 341–353.
- Gupta, A., & Jha, R. K. (2015). A survey of 5G network: Architecture and emerging technologies. *IEEE Access*, 3, 1206–1232.
- Rudolph, H. C., Kunz, A., Iacono, L. L., & Nguyen, H. V. (2019). Security challenges of the 3GPP 5G service based architecture. *IEEE Communications Standards Magazine*, 3(1), 60–65.
- Pariikh, J., & Basu, A. (2020). Technologies assisting the paradigm shift from 4G to 5G. *Wireless Personal Communications*, 112, 481–502.
- 3GPP. (2018). 3GPP technical specification; digital cellular telecommunications system (phase 2+) (GSM); universal mobile telecommunications system (UMTS); LTE; 3GPP system architecture evolution (SAE); security architecture. Retrieved from https://www.etsi.org/deliver/etsi_ts/133400_133499/133401/14.06.00_60/ts_133401v140600p.pdf.
- Kim, J., Kim, D., & Choi, S. (2017). 3GPP SA2 architecture and functions for 5G mobile communication system. *ICT Express*, 3(1), 1–8.
- Arkko, J., Lehtovirta, V., & Eronen, P. (2009). Improved extensible authentication protocol method for 3rd generation authentication and key agreement (eap-aka'). *Network Working Group Request for Comments*, 5448, 1–29.
- Miller, V. S. (1985). Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques* (pp. 417–426). Springer.
- Johnson, D., Menezes, A., & Vanstone, S. (2001). The elliptic curve digital signature algorithm (ECDSA). *International Journal of Information Security*, 1(1), 36–63.

26. Majumder, S., Ray, S., Sadhukhan, D., Khan, M. K., & Dasgupta, M. (2020). ECC-COAP: Elliptic curve cryptography based constraint application protocol for internet of things. *Wireless Personal Communications, 116*, 1867–1896.
27. Abdalla, M., & Pointcheval, D. (2005) Interactive Diffie–Hellman assumptions with applications to password-based authentication. In *International conference on financial cryptography and data security* (pp. 341–356). Springer.
28. Chaudhry, S. A., Farash, M. S., Naqvi, H., Islam, S. K. H., & Shon, T. (2017). A robust and efficient privacy aware handover authentication scheme for wireless networks. *Wireless Personal Communications, 93*(2), 311–335.
29. AVISPA. (2005). AVISPA automated validation of internet security protocols. Retrieved from <http://www.avispa-project.org>.
30. Narwal, B., & Mohapatra, A. K. (2020). Seemaka: Secured energy-efficient mutual authentication and key agreement scheme for wireless body area networks. *Wireless Personal Communications, 113*(4), 1985–2008.
31. OPENSLL. (2018). OPENSLL-cryptography and SSL/TLS toolkit. Technical report. Retrieved from <https://www.openssl.org/>.
32. Gupta, S., Parne, B. L., & Chaudhari, N. S. (2019). SRGH: A secure and robust group-based handover aka protocol for MTC in LTE-A networks. *International Journal of Communication Systems, 32*(8), e3934.
33. Huang, K.-L., Chi, K.-H., Wang, J.-T., & Tseng, C.-C. (2013). A fast authentication scheme for WIMAX-WLAN vertical handover. *Wireless Personal Communications, 71*(1), 555–575.
34. Alezabi, K. A., Hashim, F., Hashim, S. J., Ali, B. M., & Jamalipour, A. (2020). Efficient authentication and re-authentication protocols for 4G/5G heterogeneous networks. *EURASIP Journal on Wireless Communications and Networking, 2020*, 1–34.
35. Kleinrock, L. (1976). *Computer applications. Queueing systems*. Wiley.
36. Lee, J.-H., & Chung, T.-M. (2008). A traffic analysis of authentication methods for proxy mobile IPV6. In *2008 international conference on information security and assurance (ISA 2008)* (pp. 512–517). IEEE.
37. Wang, W., & Akyildiz, I. F. (2000). Intersystem location update and paging schemes for multitier wireless networks. In *Proceedings of the 6th annual international conference on mobile computing and networking* (pp. 99–109).
38. Carman, D. W., Kruus, P. S., & Matt, B. J. (2000). Constraints and approaches for distributed sensor network security (final). *DARPA Project report, Cryptographic Technologies Group, Trusted Information System, NAI Labs, 1*(1), 1–39.
39. Zhang, L., Tang, S., & Zhu, S. (2016). An energy efficient authenticated key agreement protocol for SIP-based green VOIP networks. *Journal of Network and Computer Applications, 59*, 126–133.
40. Feeney, L. M., & Nilsson, M. (2001). Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In *Proceedings IEEE INFOCOM 2001. Conference on computer communications. Twentieth annual joint conference of the IEEE computer and communications society (Cat. No. 01CH37213)* (vol. 3, pp. 1548–1557).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Shubham Gupta received his B.Tech. in information technology and M.Tech in computer science & engineering from Uttar Pradesh Technical University, Lucknow, and University College of Engineering, RTU, Kota, India respectively. He has completed his Ph.D. degree in Computer Science & Engineering from Visvesvaraya National Institute of Technology, Nagpur, Maharashtra, India in September 2019. He worked as an European Research Consortium for Informatics and Mathematics (ERCIM) Post-doctoral research fellow at Norwegian University of Science and Technology, Trondheim, Norway. His research interest includes security and privacy in 5G communication network, machine type communication, wireless communication networks and mobile computing.



Balu L. Parne has done his under graduation from Shri Sant Gajanan Maharaj College of Engineering (SSGMCE), Shegaon that is affiliated to Sant Gadge Baba Amravati University (SGBAU), Amravati, Maharashtra, India and post-graduation from National Institute of Technology (NIT), Rourkela, Orissa, India. He has completed his Ph.D. degree in the Department of Computer Science and Engineering, Visvesvaraya National Institute of Technology (VNIT), Nagpur, Maharashtra, India in January 2019. Currently, he is working as the Assistant Professor in the Computer Engineering Department at Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, Gujarat, India. Prior to SVNIT Surat, he served as an Assistant Professor in the School of Computer Science and Engineering of Vellore Institute of Technology (VIT-AP) University, Amaravathi, Andhra Pradesh, India from May-2018 to November- 2019. His current area of research is Wireless Communication, Network security, Internet of Things, Mobile computing and its applications.



Narendra S. Chaudhari completed his undergraduate, postgraduate and doctoral studies at Indian Institute of Technology (IIT), Mumbai, Maharashtra, India, in 1981, 1983, and 1988 respectively. He has done significant research work on game AI, novel neural network models like binary neural nets and bidirectional nets, graph isomorphism problem, security of the wireless mobile communication, mobile computing and Internet of Things. He has been referee and reviewer for a number of premier conferences and Journals including IEEE Transaction, Neurocomputing, etc. Currently, he is the professor at Computer Science and Engineering Department in Indian Institute of Technology (IIT), Indore (M.P.), India and Vice-Chancellor of Uttarakhand Technical University, Dehradun, India. Also, he is fellow and recipient of Eminent Engineer award (Computer Engineering) of the Institution of Engineers, India (IE-India), Bharat Vidya Shiromani Award (with gold medal), as well as fellow of the Institution of Electronics and Telecommunication Engineers (IETE) (India), senior member of Computer Society of India, senior member of IEEE, USA, member of Indian

Mathematical Society (IMS), Cryptology Research Society of India (CRSI) and many other professional societies.



Sandeep Saxena has completed his B.Tech in CSE from UPTU Lucknow, MS in Information Security from IIIT, Allahabad, and Ph.D. in CSE from NIT Durgapur in CSE Department. Currently, he is an Associate Professor (CSE) in Galgotia College of Engineering and Technology, Greater Noida. He was working for various other departmental enhancements, research & development Activities, Syllabus Design, and other major development. He is an active member of a professional society like IEEE Senior member (USA), IAASSE, CRSI (INDIA), Life Member of CSI and other professional societies. He is a reviewer of several prestigious conferences/Journals /Transactions like IEEE, many SPRINGERS/other Scopus indexed International Journals. His research outlines an emphasis on Network Security, Cloud Security, Interconnection Networks & Architecture and Information security. He has organized various Session Chairs, organizing IEEE and Springer conference and many others.

Authors and Affiliations

Shubham Gupta¹  · Balu L. Parne² · Narendra S. Chaudhari³ · Sandeep Saxena⁴

Balu L. Parne
blparne@coed.svnit.ac.in

Narendra S. Chaudhari
nsc0183@yahoo.com

Sandeep Saxena
saxena.s.in@ieee.org

¹ Department of Computer Science and Engineering, SRM University Andhra Pradesh (AP), Amravati 522502, India

² Computer Engineering Department, Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat 395007, India

³ Department of Computer Science and Engineering, Indian Institute of Technology (IIT), Indore, Madhya Pradesh (M.P.) 453552, India

⁴ Department of Information Technology, Galgotias College of Engineering and Technology (GCET), Greater Noida 201306, India