OTFS-Based Efficient Handover Authentication Scheme with Privacy-Preserving for High Mobility Scenarios

Dawei Li, Di Liu, Yu Sun*, Jianwei Liu

School of Cyber Science and Technology, Beihang University, Beijing 100191, China
* The corresponding author, email: sunyv@buaa.edu.cn

Cite as: D. Li, D. Liu, *et al*., "Otfs-based efficient handover authentication scheme with privacy-preserving for high mobility scenarios," *China Communications*, vol. 20, no. 1, pp. 36-49, 2023. **DOI**: 10.23919/JCC.2023.01.004

Abstract: Handover authentication in high mobility scenarios is characterized by frequent and short-term parallel execution. Moreover, the penetration loss and Doppler frequency shift caused by high speed also lead to the deterioration of network link quality. Therefore, high mobility scenarios require handover schemes with less handover overhead. However, some existing schemes that meet this requirement cannot provide strong security guarantees, while some schemes that can provide strong security guarantees have large handover overheads. To solve this dilemma, we propose a privacy-preserving handover authentication scheme that can provide strong security guarantees with less computational cost. Based on Orthogonal Time Frequency Space (OTFS) link and Key Encapsulation Mechanism (KEM), we establish the shared key between protocol entities in the initial authentication phase, thereby reducing the overhead in the handover phase. Our proposed scheme can achieve mutual authentication and key agreement among the user equipment, relay node, and authentication server. We demonstrate that our proposed scheme can achieve user anonymity, unlinkability, perfect forward secrecy, and resistance to various attacks through security analysis including the Tamarin. The performance evaluation results show that our scheme has a small computational cost compared with other schemes and can also provide a strong guarantee of security properties.

Keywords: high mobility condition; handover authentication; privacy-preserving; Tamarin; OTFS

Received: Jul. 24, 2022
Revised: Aug. 24, 2022
Editor: Weijie Yuan

I. INTRODUCTION

Communication in high-mobility scenarios such as high-speed trains and airplanes has the specific characteristic that a large number of users perform handovers in parallel and frequently [1]. Current 5G networks use Orthogonal Frequency Division Multiplexing (OFDM) modulation, but user communications are also heavily affected by penetration loss and Doppler frequency shift due to the very fast running speeds of high-speed rail and airplanes [2]. These characteristics in high mobility scenarios often lead to poor communication quality for users, and even lead to handover failures, resulting in users being unable to obtain network services [3].

Orthogonal Time Frequency Space (OTFS) two-dimensional modulation [4, 5] is considered as a promising technology for future 6G communication. OTFS modulation uses Delay Doppler (DD) domain to place information symbols, which can effectively combat the Doppler frequency shift phenomenon during user communication [6]. Therefore, it is especially suitable for improving communication reliability in high mobility scenarios, such as high-speed rail communication networks with speeds exceeding 500 km/h and airborne communication networks with relative speeds exceeding 1200 m/s [7].

Under the above background, we design a user equipment handover authentication scheme in high mobility scenarios for future OTFS modulation-based 6G networks in this paper. We adopt the design idea of backward compatibility, that is, based on the current 5G network and facing the future 6G network. To achieve the goal of seamless handover for users, the mobile relay node (MRN) is widely used in current high-speed rail communication networks to assist communication [8]. MRN helps users perform handover authentication with access points and authentication servers, thereby providing users with continuous network connectivity.

The existing handover authentication schemes in high mobility scenarios have some limitations. For those schemes that can fully guarantee various security properties such as user anonymity, unlinkability, and perfect forward secrecy, large computational costs are often required [9, 10]. However, those schemes with low computational costs often cannot provide rich guarantees of security properties [11]. Another problem is that most of the existing schemes use complex cryptographic operations in the protocol design to ensure various security properties, which are not suitable for handover requirements in high mobility scenarios [12]. In addition to requiring longer handover times, they are also incompatible with schemes already deployed in existing communication networks, and additional overhead is required to deploy these schemes, which limits their practicality.

In this paper, we propose a handover authentication scheme suitable for high mobility scenarios that not only guarantee various security properties but also have a small computational cost. Based on the idea of the key encapsulation mechanism (KEM) and data encapsulation mechanism (DEM), we design the authentication and key agreement protocol and handover protocol in our scheme based on the existing 3GPP standards. The contributions of this paper are listed as follows:

- We propose a handover authentication scheme with high security, low computational cost, and compatibility with existing standards. The security of the scheme can be improved by establishing a temporary shared key between protocol entities for authentication and session key generation based on the KEM.
- The proposed scheme provides strong security guarantees, which can guarantee the anonymity of users, can resist linkability attacks based on error messages and replay attacks, can achieve key forward/backward secrecy, and can also resist man-in-the-middle attacks.
- We verified that the proposed scheme satisfies the above-mentioned security properties in the formal verification tool Tamarin, and we open-sourced the protocol model to provide support for researchers in the community.

The rest of the paper is organized as follows. Section II reviews and discusses related work. Section III presents the system architecture and threat model. Section IV details the handover authentication scheme we designed, which consists of an initial authentication phase and a handover phase. Section V discusses the security of our proposed scheme and analyzes the security properties satisfied by the protocol in Tamarin. Section VI compares the computational cost and functionality of our scheme and other schemes. The conclusions are presented in Section VII.

II. RELATED WORK

Recently, some handover authentication schemes for high mobility scenarios have been proposed. 3GPP has also formulated relevant standards [8, 13, 14] for the handover authentication of high-speed rail networks, but these schemes have some security problems such as replay attacks and linkability attacks. Cao *et al.* [15] proposed a group-to-route handover authentication scheme based on trajectory prediction. In this scheme, the MRN and the base stations on the route are aggregated into groups respectively to realize mutual authentication and key agreement between the two groups, which can improve the handover efficiency. However, although this scheme is resistant to majority protocol attacks, it cannot achieve security properties such as perfect forward secrecy and unlinkability. Ma *et al.* [1] proposed two group pre-handover authentication schemes in the 5G high-speed rail network scenario. The first scheme has less handover overhead but cannot achieve security properties such as user anonymity and perfect forward secrecy. The second scheme can achieve most security properties, but it has a significant handover overhead due to the extensive use of cryptographic operations on ellip-

tic curves. To improve the security of handover authentication under the 5G high-speed rail network, Li *et al.* [16] proposed a handover authentication scheme based on trusted MRN. By deploying a trusted execution environment in the MRN to assist user equipment in authentication, the security and handover efficiency of the scheme are improved. However, they did not perform a formal security analysis of the proposed protocol, nor did they evaluate the computational cost of the scheme.

Moreover, there are many handover authentication schemes for other network scenarios. Xue et al. [9] designed an efficient handover mechanism for IoT in spatial information networks. This scheme reduces authentication delay by enabling satellites to authenticate users and improves handover efficiency by supporting batch verification. Although this scheme can resist various attacks, it has a large computational cost. For the handover scenarios of different types of base stations in 5G heterogeneous networks, Cao et al. [17] proposed a privacy protection handover authentication scheme based on user capabilities combined with the software-defined network (SDN) technique. This scheme has a small handover overhead and can provide good security protection, but there is no discussion on perfect forward secrecy. Zhang et al. [10] proposed a universal handover scheme suitable for 5G heterogeneous networks based on the properties of the chameleon hash function and blockchain. The scheme considers multiple scenarios of inter-domain and intra-domain handover and can provide high-security assurance. However, this scheme does not discuss linkability attacks against users and has a high computational cost. Guo et al. [11] proposed an anonymous handover authentication scheme for fog computing, which achieves high handover efficiency by using lightweight cryptographic primitives and the cooperation of fog nodes. However, this scheme cannot achieve user traceability and does not discuss user linkability.

III. SYSTEM MODEL

3.1 System Architecture

The system architecture is shown in Figure 1. The scenario we consider is handover authentication under high mobility conditions. The entities in the system

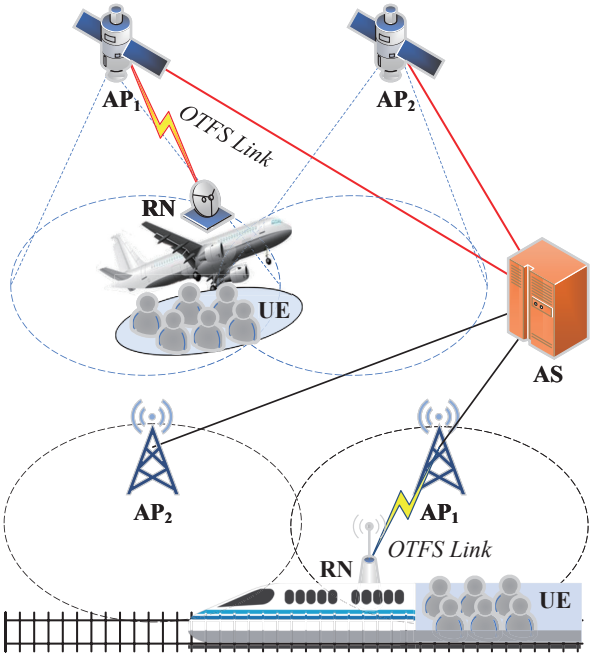

Figure 1. *System architecture.*

architecture include user equipment (UE), relay nodes (RN), access points (AP), and authentication servers (AS). The UE is user equipment that has a network access requirement on a high-speed train or an airplane. The RN is fixed on the train or plane and helps the UE to connect to the ground network. The AP is the access point for the UE to connect to the core network, which can be a terrestrial base station or a satellite. The AS is a server in the core network that authenticates the UE and generates keys and other information.

Since the UE is in a high-speed mobile scenario, frequent handovers are required. A large number of UEs performing handovers in parallel will cause huge communication delays, resulting in the unavailability of network services. Therefore, a relay node is introduced to solve this problem [8]. Before the departure of the train or the plane, the UE and RN complete the initial authentication with the AS. When moving at high speed, only the RN performs the handover procedure, and the RN acts as an access point to provide network services for the UE.

3.2 Threat Model

The channel between UE and RN and the channel between RN and AP are wireless channels, where the channel between RN and AP is a radio link based on

OTFS modulation. We assume that there are passive or active attackers in the wireless channel, possibly malicious users in our scenario. Attackers can eavesdrop, modify, and replay messages transmitted on the channel, or inject messages generated by themselves into the channel. We assume that the channel between AP and AS is a secure channel, which can guarantee the confidentiality, integrity and resistance to replay attacks of messages transmitted in the channel. The channel between the ground AP and the AS is a secure wired link, and the channel between the satellite AP and the ground AS is a secure OTFS link.

We further assume that the attacker cannot compromise the RN, AP, and AS entities in the network, which means that the attacker cannot steal the keys stored in the entities, such as the AS's private key and the long-term shared key with the UE and RN. But the attacker can have a legitimate UE and communicate with entities in the network.

3.3 OTFS Link

We discuss the basic input-output relationships of OTFS systems following the literature [18–22]. The information symbol $x[k,l]$ is represented as a point in the two-dimensional delay-Doppler (DD) domain, where $0 < k < N-1$, $0 < l < M-1$ represents the Doppler index and delay index, respectively. Symbol $x[k,l]$ transforms into time-frequency (TF) domain through inverse symplectic finite Fourier transform (ISFFT),

$$X[n,m] = \frac{1}{\sqrt{MN}} \sum_{k=0}^{N-1} \sum_{l=0}^{M-1} x[k,l] e^{j2\pi(\frac{nk}{N} - \frac{ml}{M})}, \quad (1)$$

where $0 < n < N-1$, $0 < m < M-1$ represents the time index and subcarrier index, respectively.

Symbol $X[n,m]$ is mapping through Heisenberg transform to the time domain continuous signal $s(t)$,

$$s(t) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} X[n,m] g_{tx}(t-nT) e^{j2\pi m \triangle f(t-nT)}, \quad (2)$$

where T and $\triangle f$ respectively represent the sampling interval in time and frequency, and the function $g_{tx}()$ is the pulse-shaping filter.

After the signal is transmitted through the time-varying wireless channel, the received signal $r(t)$ is represented by

$$r(t) = \iint h(\tau,\nu) s(t-\tau) e^{j2\pi\nu(t-\tau)} d\tau d\nu + \varphi(t), \quad (3)$$

where $\varphi(t)$ is the noise signal, and $h(\tau,\nu)$ represents the channel response in the DD domain,

$$h(\tau,\nu) = \sum_{i=1}^{P} h_i \delta(\tau-\tau_i)\delta(\nu-\nu_i), \quad (4)$$

where P represents the number of paths in the channel, and h_i, τ_i, and ν_i represent the channel coefficient, delay, and Doppler shift of a certain path, respectively.

The time domain signal $r(t)$ is transformed into the symbol $Y[n,m]$ of the TF domain by the Wigner transform,

$$Y[n,m] = \int r(t) g_{tx}^{*}(t-nT) e^{-j2\pi m \triangle f(t-nT)} dt, \quad (5)$$

and then the symbol $Y[n,m]$ is transformed into the symbol $y[k,l]$ in the DD domain by the symplectic finite Fourier transform (SFFT).

$$y[k,l] = \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} Y[n,m] e^{-j2\pi(\frac{nk}{N} - \frac{ml}{M})}. \quad (6)$$

A secure OTFS link can be achieved through physical layer encryption. The two communicating entities obtain channel state information (CSI) through OTFS channel estimation [18, 23, 24] and generate a physical layer key on this basis, and then encrypt the transmitted information symbols before sending the signal $s(t)$. We assume in the threat model that the link between the satellite AP and the AS is a secure OTFS link, while the link between the RN and the AP is not assumed to provide physical layer security guarantees.

IV. PROPOSED SCHEME

4.1 Overview of the Scheme

Our proposed scheme consists of two parts: the authentication phase and the handover phase. The entities involved in the authentication and key agreement (AKA) phase are UE, RN, AP, and AS. The UE and

Table 1. *Notations used in the scheme.*

Notation	Description
UE	User Equipment
RN	Relay Node
AP	Access Point / Base Station
AS	Authentication Server
K_{AP}	Session key between RN and AP
$K_{S\text{-}RN}$	Session key between UE and RN
$k/k_{ue}/k_{rn}$	Long-term shared key
$k_{UE}/k_{RN}/k_s$	Shared key established by entities
PK_{AS}/sk_{AS}	The AS's public-private key pair
PK_{RN}/sk_{RN}	The RN's public-private key pair
ID_{AS}/ID_{AP}	Identifier of AS/AP
RES/XRES	Response / Expected response
r_s/RAND	Random numbers
RAND$'$	Encrypted random number
SUPI	Subscriber permanent identifier
SUCI	Subscriber concealed identifier
GUTI	Temporary identifier
AK	Anonymity key
AV	Authentication vector
SQN	Sequence number
MAC	Message authentication code
AUTN	Authentication token

the RN implement mutual authentication with the AS based on the improved 5G-AKA protocol [14], respectively. At the same time, the RN and the UE perform mutual authentication with the help of the AS. Finally, the RN and the AP negotiate a session key K_{AP}, and the RN and the UE negotiate a session key $K_{S\text{-}RN}$, which is used to ensure the security of communication between them.

The entities involved in the handover phase are only RN, AP_1, AP_2 and AS. The UE and the RN use the session key $K_{S\text{-}RN}$ negotiated in the authentication phase for secure communication. The RN negotiates a new session key K_{AP_2} with the next target access point AP_2 in advance with the help of the AP_1 and the AS. Once the RN enters the service range of AP_2, the RN can use the new session key K_{AP_2} to securely communicate with AP_2, thereby realizing seamless handover for the UE. The notations used in our scheme are presented in Table 1.

4.2 Design Idea

Our design principle is to be compatible with the existing 5G network while facing the future 6G heterogeneous network. Therefore, we design our handover authentication scheme based on the existing 3GPP technical standards [8, 14, 13]. Before the plane takes off or the train starts, both the UE and the RN respectively complete the mutual authentication with the AS as the user equipment. The UE and the RN negotiate a session key for subsequent communication, and the RN and the AS negotiate a session key for communicating with the AP and completing the handover procedure.

To ensure the anonymity of the UE, 3GPP TS 33.501 [14] adopts the Elliptic Curve Integrated Encryption Scheme (ECIES) [25] to encrypt the SUPI of the UE to obtain SUCI. ECIES is a hybrid encryption scheme that includes a key encapsulation mechanism (KEM) and a data encapsulation mechanism (DEM) [26]. In each session, we use KEM to establish a shared key between protocol entities, and use DEM with the established shared key to encrypt and decrypt the communication data between entities. We generate authentication materials based on the established shared key to help entities complete mutual authentication and generate session keys between entities. Note that the shared key established each session is not the same as the long-term shared key stored between entities during the registration phase. The use of ECIES can not only realize the compatibility of our designed scheme with existing 5G networks but also help improve the security and efficiency of our scheme.

4.3 Initial Authentication Phase

In the authentication and key agreement phase, the UE completes mutual authentication with the AS with the help of the RN, and the UE also verifies the identity of the RN. The RN then completes mutual authentication with the AS and authenticates the UE with the help of the AS. Subsequently, the session key K_{AP} between RN and AS and the session key $K_{S\text{-}RN}$ between RN and UE will be derived. The procedure of the protocol is shown in Figure 2, and the detailed steps are shown below.

***Step 1*:** UE $\rightarrow$ RN: (GUTI, ID_{AS})

The UE takes the public key PK_{AS} as input, and uses the key encapsulation mechanism to generate a shared key k_{UE} with the AS. The UE then uses the data encapsulation mechanism to encrypt its identifier $SUPI_{UE}$ to obtain $SUCI_{UE}$, which is used for authentication by the AS. In addition, the UE uses the public key PK_{RN} as an input to generate a shared key k_s with the RN, and generates a random number r_s together with the $SUCI_{UE}$ as an input to obtain the temporary identifier GUTI. The UE then sends the GUTI and the ID_{AS} of its authentication server to the RN.

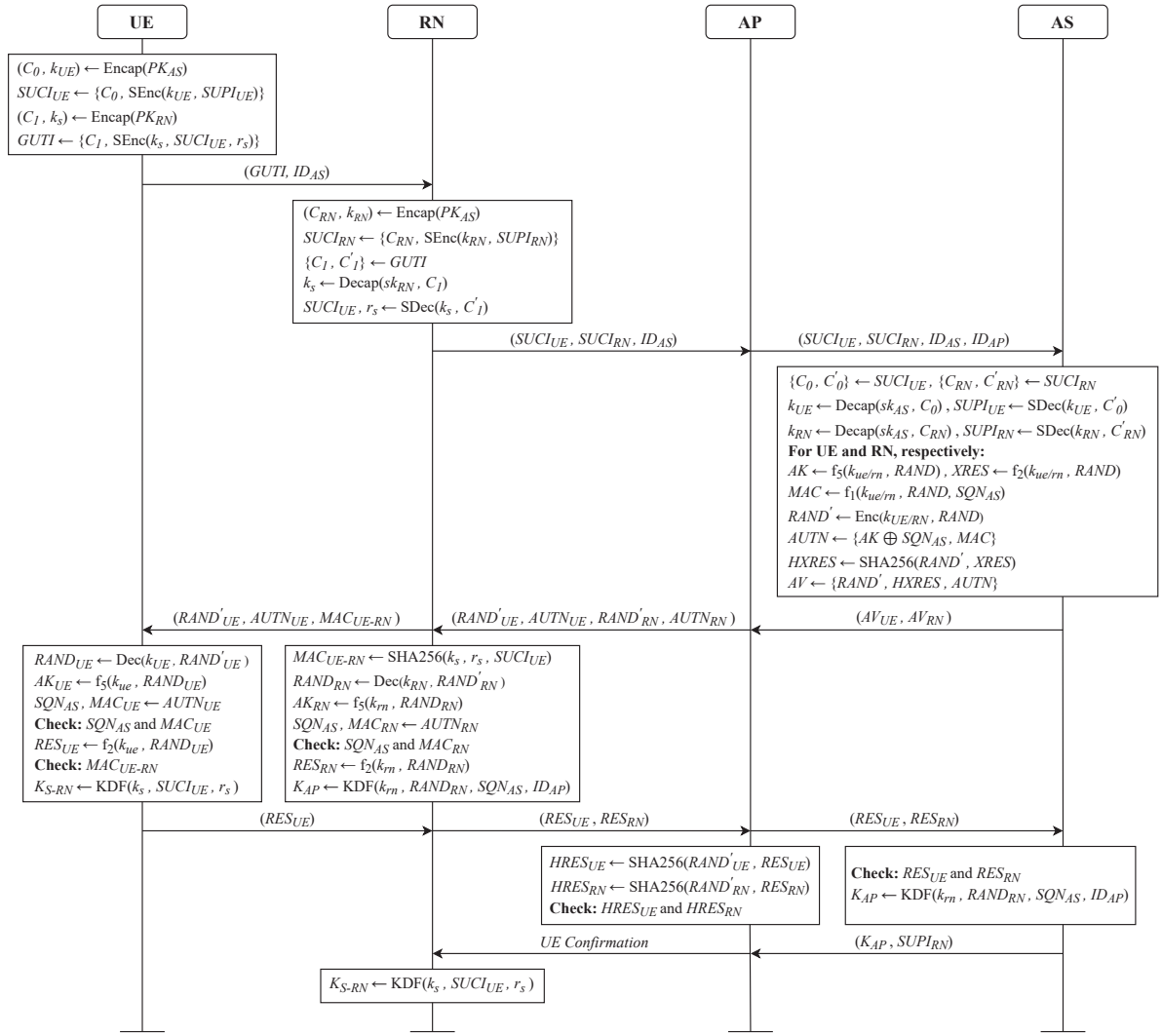

Figure 2. *Mutual authentication and key agreement protocol.*

***Step 2*:** RN → AP: (SUCI_{UE}, SUCI_{RN}, ID_{AS})

The RN takes the public key PK_{AS} as input, generates a shared key k_{RN} with the AS, and then encrypts its identifier SUPI_{RN} to obtain SUCI_{RN}, which is used for authentication by the AS. Subsequently, the RN parses the received GUTI into C_1 and $C_1^{'}$, decrypts C_1 with its private key sk_{RN} to obtain the shared key k_{s} with the UE, and then decrypts $C_1^{'}$ with the key k_{s} to obtain SUCI_{UE} and random number r_{s}. The RN then sends SUCI_{RN}, SUCI_{UE}, and ID_{AS} to the AP over the OTFS link.

***Step 3*:** AP → AS: (SUCI_{UE}, SUCI_{RN}, ID_{AS}, ID_{AP})

After receiving the message from the RN, the AP selects the authentication server corresponding to the user according to the identifier ID_{AS}. The AP adds its identifier ID_{AP} to the message and sends the message to the AS.

***Step 4*:** AS → AP: (AV_{UE}, AV_{RN})

Upon receiving the authentication request sent by the AP, the AS first parses the SUCI_{UE} and SUCI_{RN}, and uses its private key sk_{AS} to decrypt the message to obtain the shared key k_{UE} with the UE and the shared key k_{RN} with the RN. The AS then decrypts SUCI_{UE} and SUCI_{RN} using the key $\text{k}_{\text{UE/RN}}$ to obtain SUPI_{UE} and SUPI_{RN}, respectively.

Subsequently, the AS calculates the parameters required for generating the authentication vector for the UE and the RN, respectively. The AS first generates a random number RAND, and then uses the key derivation function f_5 to generate the anonymity key AK through the long-term shared key $\text{k}_{\text{UE/RN}}$ with the UE or RN. The AS then uses the message authentication

functions f_2 and f_1 to generate the expected response XRES and the message authentication code MAC, respectively. SQN_{AS} is the sequence number generated by the AS for UE and RN to verify the freshness of the message. The definition of functions f_1, f_2, and f_5 can refer to 3GPP TS 33.102 [13].

Finally, the AS encrypts RAND with the key $\text{k}_{\text{UE/RN}}$ to obtain RAND′, and calculates the hash value of RAND′ and XRES to obtain HXRES. The AS then generates the authentication token AUTN and constructs the authentication vectors AV_{UE} and AV_{RN} corresponding to the UE and the RN according to the RAND′, HXRES, and AUTN, and sends them to the AP.

***Step 5*:** AP → RN: (RAND'_{UE}, AUTN_{UE}, RAND'_{RN}, AUTN_{RN})

After receiving the authentication vector sent by the AS, the AP stores the RAND′ and HXRES for subsequent authentication of the UE and the RN, and sends the RAND′ and AUTN corresponding to the UE and the RN to the RN through the OTFS link.

***Step 6*:** RN → UE: (RAND'_{UE}, AUTN_{UE}, $\text{MAC}_{\text{UE-RN}}$)

Once the RN receives the message from the AP, it first calculates the message authentication code $\text{MAC}_{\text{UE-RN}}$ according to the shared key k_{s}, which is used for the UE to authenticate the RN. The RN then sends a message (RAND'_{UE}, AUTN_{UE}, $\text{MAC}_{\text{UE-RN}}$) to the UE and starts to authenticate the AS.

The RN decrypts RAND'_{RN} through the shared key k_{RN} to obtain RAND_{RN}, and then calculates AK_{RN} through the long-term shared key k_{RN} with the AS. The RN parses SQN_{AS} and MAC_{RN} from AUTN_{RN} using AK_{RN}, and checks whether MAC_{RN} satisfies Eq. (7) and whether SQN_{AS} is fresh. If the check passes, the RN successfully authenticates the AS. Otherwise, the RN terminates the authentication process.

$$MAC_{RN} = f_1(k_{rn}, RAND_{RN}, SQN_{AS}). \quad (7)$$

Subsequently, the RN generates the authentication response RES_{RN} according to Eq. (8), which is used by the AS and the AP to authenticate the RN. The RN then calculates the session key K_{AP} with the AP according to Eq. (9).

$$RES_{RN} = f_2(k_{rn}, RAND_{RN}), \quad (8)$$

$$K_{AP} = KDF(k_{rn}, RAND_{RN}, SQN_{AS}, ID_{AP}). \quad (9)$$

***Step 7*:** UE → RN: (RES_{UE})

Once the UE receives the message from the RN, the UE needs to authenticate the RN and AS. The UE first decrypts RAND'_{UE} with the shared key k_{UE} to obtain RAND_{UE}, and then uses the long-term shared key k_{UE} with the AS to generate AK_{UE} based on RAND_{UE}. The UE parses SQN_{AS} and MAC_{UE} from AUTN_{UE} using AK_{UE}, and checks whether MAC_{UE} satisfies Eq. (10) and whether SQN_{AS} is fresh.

$$MAC_{UE} = f_1(k_{ue}, RAND_{UE}, SQN_{AS}). \quad (10)$$

If the UE successfully authenticates the AS, the UE generates the authentication response RES_{UE} according to Eq. (11) and sends the RES_{UE} to the RN.

$$RES_{UE} = f_2(k_{ue}, RAND_{UE}). \quad (11)$$

The UE authenticates the RN by checking whether the $\text{MAC}_{\text{UE-RN}}$ satisfies Eq. (12). If the authentication is successful, the UE calculates the session key $\text{K}_{\text{S-RN}}$ with the RN according to Eq. (13).

$$MAC_{UE-RN} = SHA256(k_s, r_s, SUCI_{UE}), \quad (12)$$

$$K_{S-RN} = KDF(k_s, SUCI_{UE}, r_s). \quad (13)$$

***Step 8*:** RN → AP: (RES_{UE}, RES_{RN})

After the RN receives the RES_{UE} sent by the UE, it adds its authentication response RES_{RN} and sends (RES_{UE}, RES_{RN}) to the AP through the OTFS link for authentication.

***Step 9*:** AP → AS: (RES_{UE}, RES_{RN})

Based on the received RES and the previously stored RAND′, the AP calculates the hash values HRES_{UE} and HRES_{RN} for the UE and RN according to Eq. (14) and (15), respectively. The AP then compares the calculation result with the previously stored HXRES_{UE} and HXRES_{RN}, and if they are consistent, the UE and RN are successfully authenticated from the AP's point of view. The AP then sends the (RES_{UE}, RES_{RN}) to the AS for further authentication.

$$HRES_{UE} = SHA256(RAND'_{UE}, RES_{UE}), \quad (14)$$

$$HRES_{RN} = SHA256(RAND'_{RN}, RES_{RN}). \quad (15)$$

***Step 10*:** AS → AP: (K_{AP}, $SUPI_{RN}$)

The AS compares the RES_{UE} and RES_{RN} received from the AP with the $XRES_{UE}$ and $XRES_{RN}$ it generated before. If they are consistent, the UE and the RN are successfully authenticated from the perspective of the AS.

The AS then generates the session key K_{AP} between the AP and the RN according to Eq. (9), and sends it to the AP together with the $SUPI_{RN}$.

***Step 11*:** AP → RN: (UE Confirmation)

The AP receives the session key K_{AP} sent by the AS and uses it to communicate with the RN as the session key. Finally, the AP generates a UE authentication success message through the key K_{AP} and sends it to the RN over the OTFS link.

***Step 12*:** RN generates $K_{S\text{-}RN}$

After the RN accepts the message sent by the AP to confirm the successful authentication of the UE, it generates the session key $K_{S\text{-}RN}$ with the UE according to Eq. (13).

After the initial authentication and key agreement procedures are completed, the UE and the RN establish the session key $K_{S\text{-}RN}$, and the RN and the AP establish the session key K_{AP}, which are respectively used to ensure the security of their respective communication channels.

4.4 Handover Phase

In the handover phase, the RN establishes a new session key K_{AP_2} with AP_2 with the assistance of AP_1 using the session key K_{AP_1} and the shared key k_{RN}. The K_{AP_1} is the session key established by the RN, AP, and AS in the authentication phase, and the k_{RN} is the shared key established by the RN and AS in the authentication phase. The procedure of the handover protocol is shown in Figure 3, and the detailed steps are shown below.

***Step 1*:** RN → AP_1: ($SUCI^*$, AUTN, ID_{AS})

The RN generates a random number r_h together with the identifier $SUPI_{RN}$ as input, and generates a new temporary identifier $SUCI^*$ using the shared key k_{RN} established with the AS during the authentication phase. Then, the RN uses the function f_1 to generate the message authentication code MAC based on the long-term shared key k with the AS, the random number r_h, and the sequence number SQN_{RN}. Finally, the RN uses the key derivation function f_5 to calculate the anonymity key AK, generates the authentication token AUTN, and then sends a handover request message containing ($SUCI^*$, AUTN, ID_{AS}) to AP_1 over the OTFS link.

***Step 2*:** AP_1 → AS: ($SUCI^*$, AUTN, ID_{AS}, ID_{AP_1})

AP_1 adds its identifier ID_{AP_1} to the message received from the RN, and then forwards the handover request message to the corresponding AS.

***Step 3*:** AS → AP_2: ($GUTI^*$, K_{AP_2})

After receiving the handover request message sent by AP_1, the AS first decrypts $SUCI^*$ with the shared key k_{RN} established with the RN to obtain the identifier $SUPI_{RN}$ of the RN and the random number r_h. Then the AS generates AK based on the r_h and the long-term shared key k with the RN, and uses AK to parse AUTN to obtain SQN_{RN} and MAC. The AS checks whether the MAC satisfies Eq. (16), and then checks whether the SQN_{RN} is fresh.

$$MAC = f_1(k, r_h, SQN_{RN}). \quad (16)$$

If the AS successfully authenticates the RN, the AS takes $SUPI_{RN}$ and r_h as input and generates a new temporary identifier $GUTI^*$ for the RN according to Eq. (17). The AS then selects a new access point AP_2 and calculates the session key K_{AP_2} between the RN and AP_2 according to Eq. (18), and sends $GUTI^*$ and K_{AP2} to the AP_2.

$$GUTI^* = KDF(SUPI_{RN}, r_h, K_{AP_1}), \quad (17)$$

$$K_{AP_2} = KDF(k_{RN}, r_h, SQN_{RN}, ID_{AP_2}). \quad (18)$$

***Step 4*:** AP_2 → AS: Confirmation

AP_2 stores the received RN identifier $GUTI^*$ and session key K_{AP_2} for subsequent communication with the RN, and then sends a key confirmation message and other related messages to the AS.

***Step 5*:** AS → AP_1: ($AUTN^*$, ID_{AP_2})

The AS generates a message authentication code MAC^* according to Eq. (19), which is used for the RN to perform authentication. Note that the identifier ID_{AP_2} of AP_2 also participates in the calculation of the MAC^*. Finally, the AS constructs $AUTN^*$ based on AK, SQN_{AS}, and MAC^*, and then sends it to the AP_1

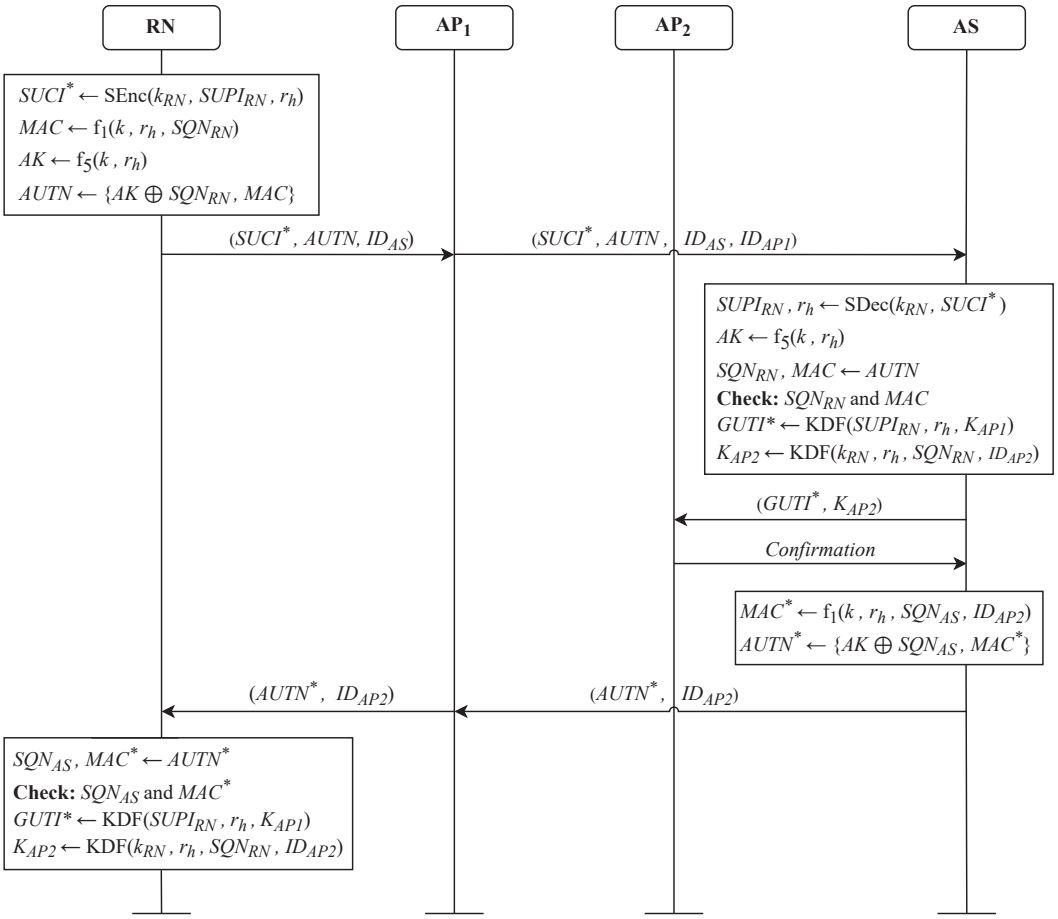

Figure 3. *Handover protocol.*

together with $\mathrm{ID}_{\mathrm{AP}_2}$.

$$MAC^* = f_1(k, r_h, SQN_{AS}, ID_{AP_2}). \quad (19)$$

***Step 6*:** $\mathrm{AP}_1 \rightarrow$ RN: (AUTN^*, $\mathrm{ID}_{\mathrm{AP}_2}$)

Once the AP_1 receives the handover response message sent by the AS, the AP_1 forwards the message to the corresponding RN over the OTFS link.

***Step 7*:** RN generates $\mathrm{K}_{\mathrm{AP}_2}$

The RN uses AK to parse out $\mathrm{SQN}_{\mathrm{AS}}$ and MAC^* from the received AUTN^*, then checks whether MAC^* satisfies Eq. (19) and checks whether $\mathrm{SQN}_{\mathrm{AS}}$ is fresh. If both verifications are successful, the RN generates the temporary identifier GUTI^* according to Eq. (17) and calculates the session key $\mathrm{K}_{\mathrm{AP}_2}$ according to Eq. (18) for subsequent communication with AP_2.

After the handover process is completed, once the UE enters the service range of AP_2, the session key $\mathrm{K}_{\mathrm{AP}_2}$ negotiated in the handover phase can be used between the UE and AP_2 to ensure the security of the communication channel.

V. SECURITY ANALYSIS

In this section, we evaluate the security properties of the proposed scheme through the informal security analysis and formal verification tool Tamarin.

5.1 Informal Security Analysis

5.1.1 Mutual Authentication and Key Agreement

In the initial authentication phase, the UE/RN authenticates the AS by checking whether the received MAC is equal to the MAC value calculated by itself according to Eq. (10)/(7). The AS authenticates the UE/RN by checking whether the received RES is equal to the XRES calculated by itself. Only the legitimate AS has the private key $\mathrm{sk}_{\mathrm{AS}}$ and the long-term shared key $\mathrm{k}_{\mathrm{UE/RN}}$ consistent with the UE/RN. Therefore, only the legitimate AS can decrypt the SUCI to obtain the cor-

rect shared key $k_{UE/RN}$ for encrypting the RAND and generate the correct $MAC_{UE/RN}$. Likewise, only the legitimate UE/RN can decrypt the RAND and generate the correct $RES_{UE/RN}$. Therefore, the RN and AS can respectively calculate the session key K_{AP} according to Eq. (9) for subsequent communication between the RN and the AP. The RN authenticates the UE through the AS, and the UE authenticates the RN by checking the $MAC_{UE\text{-}RN}$. Only the legitimate RN has the private key sk_{RN} to decrypt the GUTI to get the correct k_s and r_s. In this way, the UE and RN can calculate the session key $K_{S\text{-}RN}$ according to Eq. (13) for subsequent communication.

In the handover phase, only the legitimate RN and AS have the same shared key k_{RN} and long-term shared key k, so that the correct MAC/MAC* can be calculated to pass mutual authentication. Therefore, the RN and AS can calculate the new session key K_{AP_2} according to Eq. (18) for subsequent communication between RN and AP_2.

5.1.2 User Anonymity and Unlinkability

To ensure the anonymity of the user equipment, the UE generates a temporary identifier GUTI based on its encrypted identifier $SUCI_{UE}$ and random number r_s in each session with the RN. Only the legitimate AS can get the actual identifier $SUPI_{UE}$ of the UE. Therefore, no attacker can reveal the actual identifier of the UE. The RN also uses the encrypted identifier in the session with the AS, and the attacker also cannot obtain the actual identifier of the RN.

For the linkability of the user equipment, in addition to generating a temporary identifier GUTI for the UE in each session, various linkability attacks against the UE are also prevented by encrypting the RAND using the established shared key k_{UE} [27]. If the transmitted RAND$'$ is modified or replayed by an attacker, the UE will output an error message when checking the correctness of the MAC_{UE}. This approach fundamentally solves linkability attacks against UEs based on failure messages [28], sequence number inference [29], and SUCI replay [30, 31].

5.1.3 Key Forward/Backward Secrecy

Based on the session key Eq. (9), (13), and (18) between entities, we can infer that even if the attacker knows the session key of the current session, he cannot guess the session key of the previous session as well as the future session. Because parameters such as r_s, r_h, RAND, and SQN are different in each session, the session key generated for each session is also different.

Similarly, even if the long-term shared key between the UE and the AS is stolen by the attacker, the attacker cannot derive the previous session key. Because the generation of K_{AP} requires the participation of the RAND of the current session, and the RAND also needs to be generated using the shared key k_{RN} established by the current session. The generation of $K_{S\text{-}RN}$ and K_{AP_2} also requires the participation of the shared keys established by them.

5.1.4 Resistance Against Replay Attacks

For the proposed scheme, replay attacks are prevented from two aspects. First, after verifying the legality of the entity by checking the MAC, the freshness of the message is also checked by the sequence number SQN. Second, random numbers such as r_s, r_h, and RAND are also introduced into the parameters for generating the session key, so that the session key is only bound to the current session.

5.1.5 Resistance Against Man-in-the-Middle Attacks

When the attacker cannot know the secret information such as shared keys $k_{UE/RN}$, $k_{UE/RN}$, k_s and random numbers r_s, r_h, RAND, etc., they cannot forge a valid authentication request to pretend to be a legitimate protocol entity. Also, the attacker cannot generate a legitimate session key. Therefore our proposed scheme is resistant to impersonation attacks. On this basis, the attacker cannot pretend to be a legitimate AS to communicate with the UE/RN at the same time as a legitimate UE/RN to communicate with the AS, so our proposed scheme can also resist the man-in-the-middle attack.

5.2 Formal Verification Based on Tamarin

We evaluate the security of our proposed handover protocol with Tamarin. Tamarin is a powerful symbolic verification tool for analyzing security protocols [32]. Tamarin takes the protocol and the security property as input. It outputs the conclusion that the protocol satisfies the security property or the

counter-example that the security property does not hold. Tamarin is used by researchers in the community to analyze many protocols, including the 5G-AKA protocol [28, 33, 34], and is a state-of-the-art tool for modeling operations such as Diffie-Hellman groups, asymmetric encryption, XOR, and more.

For the DEM paradigm, the $senc$ and $sdec$ functions from the built-in symmetric encryption theory can be used to model data encapsulation and decapsulation mechanisms. For the KEM paradigm, the built-in asymmetric encryption theory can be used, but the built-in asymmetric encryption functions cannot accurately describe the KEM paradigm. Because there are two output parameters of the Encap function, but the built-in function output parameter in Tamarin can only be one. To address this issue, we refer to the equation proposed by Wang et al. [27] to model the KEM paradigm in Tamarin, as shown in Eq. (20).

$$\begin{aligned} &Decap(sk, getcipher(Encap(pk(sk), R))) \\ &= getkey(Encap(pk(sk), R)), \end{aligned} \tag{20}$$

where R is a random number, indicating that the Encap function is a randomness algorithm. Therefore, the shared keys $k_{UE/RN}$ established in different sessions are also different. pk(sk) is a built-in asymmetric encryption function in Tamarin, representing a public-private key pair. The getcipher function takes the output of the Encap function as input and outputs the ciphertext encapsulating the shared key. The getkey function takes the output of the Encap function as input and outputs the shared key.

We analyze the security properties satisfied by the proposed handover protocol in Tamarin, and the source code of our protocol model is available on Github [35]. The verification results of the protocol are shown in Figure 4. Our proposed scheme can achieve mutual authentication between the RN and the AS and achieve injective agreement on the session key. The attacker cannot obtain the real identifier SUPI of the RN under the condition that the RN or AS cannot be compromised, hence the scheme can achieve the anonymity of the user. Furthermore, the attacker cannot obtain the session key of the current session, and the scheme can achieve perfect forward secrecy. Even if the attacker obtains the long-term shared key in the current session, the session key established by the previous session is still secret.

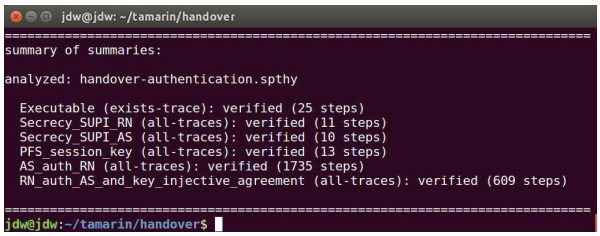

Figure 4. *Verification results of the handover protocol.*

Table 2. *The time required for cryptographic operations.*

Cryptographic Operations	Time Cost (us)
T_h	1.21
T_s	1.05
T_a	1.39
T_m	500
T_e	1000
T_p	8360

Table 3. *Comparison of computational costs.*

Scheme	Computational costs	Total (us)
[10]	$4.5T_m$	2250
[17]	$7T_h + 2T_s$	10.57
[9]	$9T_m + 4T_a + 6T_h$	4512.82
[11]	$16T_h$	19.36
[1]-1	$10T_h + 2T_s$	14.2
[1]-2	$9T_h + 2T_s + 3T_a + 9T_m$	4517.16
Our	$10T_h + 2T_s$	14.2

VI. PERFORMANCE EVALUATION

In this section, we analyze and compare the computational cost and functionality of our proposed scheme with previous ones.

6.1 Computational Cost

Our proposed scheme consists of an initial authentication phase and a handover phase, where the initial authentication phase occurs before the departure of the train or plane, and the handover phase occurs during high-speed movement. In such a scenario, the main factor affecting the user communication quality is the handover phase. Therefore we mainly evaluate the computational cost of the handover phase.

To facilitate comparison with other schemes, we consider the cost of the following operations as in other schemes, and other operations such as XOR are ignored. A one-way hash or MAC operation T_h, a symmetric encryption/decryption operation T_s, an

Table 4. *Comparison of functionality.*

Functionality	[10]	[17]	[9]	[11]	[1]-1	[1]-2	Our Scheme
Mutual Authentication and Key Agreement	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Anonymity	Yes	Yes	Yes	Yes	No	Yes	Yes
Unlinkability	-	Yes	Yes	-	No	Yes	Yes
Traceability	Yes	Yes	-	No	Yes	Yes	Yes
Key Forward/Backward Secrecy	Yes	-	Yes	Yes	No	Yes	Yes
Withstanding Protocol Attacks	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Formal Security Proof	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Computational Cost	High	Low	High	Low	Low	High	Low

elliptic curve point addition operation T_a, an elliptic curve scalar multiplication operation T_m, a modular exponentiation operation T_e, and a bilinear pairing operation T_p. Since the time required to perform these operations on computing platforms with different computing resources is different, we refer to the simulation time of predecessors to compare the computational cost, and do not distinguish the computing power between RN and AS. Table 2 shows the time required for the above cryptographic operations, tested in [1] using the OpenSSL library on an Intel(R) Core(TM) i7-7500U CPU 2.70 GHz processor.

The comparison results of the computational cost required in the handover phase are shown in Table 3. Due to the elliptic curve cryptography operations, the computational costs of schemes [10], [9], and [1]-2 are relatively large. In contrast, the computational cost of our scheme and schemes [17], [11], and [1]-1 is relatively small. We did not consider batch access in the comparison. In the scenario of group handover of user equipment, our scheme can achieve better performance. Since our scheme establishes a session key between UE and RN in the initial authentication phase. Therefore, in the handover phase, only the RN needs to negotiate a new session key with the target AP without the participation of the UE, so our scheme does not cause additional computational costs.

6.2 Functionality Comparison

Table 4 shows how our proposed scheme compares with other schemes in terms of functionality including security properties. The "-" symbol indicates that there is no discussion of such properties in the scheme. Overall, compared with other schemes, our proposed scheme has strong security guarantees and also has a small computational cost. Schemes [10], [9], and [1]-2 can provide similar security guarantees as our scheme, but at a higher computational cost. Schemes [17], [11], and [1]-1 have similar computational costs to our scheme but cannot provide the same strong security guarantees as our scheme.

As for the formal security analysis, Scheme [1] also analyzes the security properties satisfied by the scheme in Tamarin. But they did not prove the user's anonymity in Tamarin, that is, whether the attacker can obtain the user's real identifier SUPI. Schemes [9] and [10] use the AVISPA tool [36] to analyze the security properties of the protocol, and scheme [17] uses the Scyther tool [37] to analyze the security of the protocol. Scheme [11] theoretically analyzes the security of the scheme in the Real-Or-Random (ROR) model, without using formal verification tools such as Tamarin, AVISPA, or Scyther.

VII. CONCLUSION

In this paper, we propose an efficient handover authentication scheme with user privacy protection for the handover requirements of high mobility scenarios. According to the characteristics of high mobility scenarios, we negotiate the session key between UE and RN and the session key between RN and AP in the initial authentication phase based on the KEM mechanism for subsequent communication. Therefore, the UE and the AP can be decoupled in the handover phase, thereby greatly improving the handover efficiency and achieving the seamless handover of the UE. Our security analysis including the formal verification tool Tamarin demonstrates that our proposed protocol has strong security guarantees and is resistant to multiple attacks. Moreover, the performance evaluation compared with other schemes shows that our scheme can provide strong security guarantees while also having a small computational cost, which is very suitable for high mobility scenarios.

ACKNOWLEDGEMENT

This work was supported by Natural Science Foundation of China (No. 62002006, U2241213, U21B2021, 62172025, 61932011, 61932014, 61972018, 61972019, 61772538, 32071775, 91646203), and Defense Industrial Technology Development Program (No. JCKY2021211B017)

References

[1] R. Ma, J. Cao, *et al.*, "Ftgpha: Fixed-trajectory group pre-handover authentication mechanism for mobile relays in 5g high-speed rail networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 2, pp. 2126–2140, 2019.

[2] Y. Zhou and B. Ai, "Handover schemes and algorithms of high-speed mobile environment: A survey," *Computer Communications*, vol. 47, pp. 1–15, 2014.

[3] Q. Kong, R. Lu, *et al.*, "Achieve secure handover session key management via mobile relay in lte-advanced networks," *IEEE Internet of Things Journal*, vol. 4, no. 1, pp. 29–39, 2016.

[4] R. Hadani and A. Monk, "Otfs: A new generation of modulation addressing the challenges of 5g," *arXiv preprint arXiv:1802.02623*, 2018.

[5] W. Yuan, Z. Wei, *et al.*, "Integrated sensing and communication-assisted orthogonal time frequency space transmission for vehicular networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 6, pp. 1515–1528, 2021.

[6] Z. Wei, W. Yuan, *et al.*, "Orthogonal time-frequency space modulation: A promising next-generation waveform," *IEEE Wireless Communications*, vol. 28, no. 4, pp. 136–144, 2021.

[7] T. M. C. Chu, H.-J. Zepernick, *et al.*, "Performance assessment of otfs modulation in high doppler airborne communication networks," *Mobile Networks and Applications*, pp. 1–11, 2022.

[8] 3GPP, "Technical specification group radio access network; evolved universal terrestrial radio access (e-utra); study on mobile relay (release 12)," *Tech. Rep.*, 2014.

[9] K. Xue, W. Meng, *et al.*, "A secure and efficient access and handover authentication protocol for internet of things in space information networks," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5485–5499, 2019.

[10] Y. Zhang, R. H. Deng, *et al.*, "Robust and universal seamless handover authentication in 5g hetnets," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 858–874, 2019.

[11] Y. Guo and Y. Guo, "Fogha: An efficient handover authentication for mobile devices in fog computing," *Computers & Security*, vol. 108, p. 102358, 2021.

[12] T. Xu, C. Xu, *et al.*, "An efficient three-factor privacy-preserving authentication and key agreement protocol for vehicular ad-hoc network," *China Communications*, vol. 18, no. 12, pp. 315–331, 2021.

[13] 3GPP, "Technical specification group services and system aspects; 3G security; security architecture (release 14)," *Tech. Rep.*, 2017.

[14] 3GPP, "Technical specification group services and system aspects; security architecture and procedures for 5G system (release 16)," *Tech. Rep.*, 2019.

[15] J. Cao, M. Ma, *et al.*, "G2rha: Group-to-route handover authentication scheme for mobile relays in lte-a high-speed rail networks," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 9689–9701, 2017.

[16] Z. Li, D. Liu, *et al.*, "Seamless group pre-handover authentication scheme for 5g high-speed rail network," in *International Conference on Smart Computing and Communication*, pp. 308–317. Springer, 2021.

[17] J. Cao, M. Ma, *et al.*, "Cppha: Capability-based privacy-protection handover authentication mechanism for sdn-based 5g hetnets," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 3, pp. 1182–1195, 2019.

[18] W. Yuan, S. Li, *et al.*, "Data-aided channel estimation for otfs systems with a superimposed pilot and data transmission scheme," *IEEE Wireless Communications Letters*, vol. 10, no. 9, pp. 1954–1958, 2021.

[19] B. Liu, Z. Wei, *et al.*, "Channel estimation and user identification with deep learning for massive machine-type communications," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10 709–10 722, 2021.

[20] P. Raviteja, K. T. Phan, *et al.*, "Interference cancellation and iterative detection for orthogonal time frequency space modulation," *IEEE Transactions on Wireless Communications*, vol. 17, no. 10, pp. 6501–6515, 2018.

[21] Z. Wei, W. Yuan, *et al.*, "Transmitter and receiver window designs for orthogonal time-frequency space modulation," *IEEE Transactions on Communications*, vol. 69, no. 4, pp. 2207–2223, 2021.

[22] Y. Yang, Z. Bai, *et al.*, "Design and analysis of spatial modulation based orthogonal time frequency space system," *China Communications*, vol. 18, no. 8, pp. 209–223, 2021.

[23] Z. Wei, W. Yuan, *et al.*, "Off-grid channel estimation with sparse bayesian learning for otfs systems," *IEEE Transactions on Wireless Communications*, 2022.

[24] Z. Wei, W. Yuan, *et al.*, "Performance analysis and window design for channel estimation of otfs modulation," in *ICC 2021-IEEE International Conference on Communications*, pp. 1–7. IEEE, 2021.

[25] D. R. Brown, "Sec 1: Elliptic curve cryptography, version 2.0," *Standards for Efficient Cryptography*, pp. 1–144, 2009.

[26] V. Shoup, "A proposal for an ISO standard for public key encryption," *Cryptology ePrint Archive*, 2001.

[27] Y. Wang, Z. Zhang, *et al.*, "Privacy-preserving and standard-compatible AKA protocol for 5G," in *30th USENIX Security Symposium (USENIX Security 21)*, pp. 3595–3612. USENIX Association, 2021.

[28] D. Basin, J. Dreier, *et al.*, "A formal analysis of 5G authentication," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1383–1396, 2018.

[29] R. Borgaonkar, L. Hirschi, *et al.*, "New privacy threat on 3G, 4G, and upcoming 5G AKA protocols," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 3, pp. 108–127, 2019.

[30] A. Koutsos, "The 5G-AKA authentication protocol privacy," in *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, pp. 464–479. IEEE, 2019.
[31] P.-A. Fouque, C. Onete, *et al.*, "Achieving better privacy for the 3GPP AKA protocol," *Cryptology ePrint Archive*, 2016.
[32] S. Meier, B. Schmidt, *et al.*, "The TAMARIN prover for the symbolic analysis of security protocols," in *International Conference on Computer Aided Verification*, pp. 696–701. Springer, 2013.
[33] C. Cremers and M. Dehnel-Wild, "Component-based formal analysis of 5G-AKA: Channel assumptions and session confusion," in *26th Annual Network and Distributed System Security Symposium*. NDSS, 2019.
[34] L. Di, W. Ziyi, *et al.*, "Formal analysis and improvement methods of 5G AKA protocol based on tamarin," *Journal of Cryptologic Research*, vol. 9, no. 2, pp. 237–247, 2022.
[35] L. Di, "Handover protocol model in tamarin." [Online]. Available: https://github.com/tajiaodavid/Handover_Tamarin_model 2022.
[36] A. Armando, D. Basin, *et al.*, "Avispa: automated validation of internet security protocols and applications," *ERCIM News*, vol. 64, no. January, 2006.
[37] C. J. F. Cremers, *et al.*, *Scyther: Semantics and verification of security protocols*. Eindhoven University of Technology Eindhoven, Netherlands, 2006.

Biographies

Dawei Li received the B.S. degree from Beihang University, Beijing, China, in 2015. He received the Ph.D. degree from Beihang University in 2019. Now, he is a lecturer of the School of Cyber Science and Technology at Beihang University. His research interests include cryptography, blockchain and 5G security.

Di Liu received the B.S. degree from the School of Electrical Engineering, Southwest Jiaotong University, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Cyber Science and Technology, Beihang University, China. His research interests include cryptography and hardware security.

Yu Sun received the Ph.D. degree in Information & Communication Engineering from Beihang University, China in 2014 under the supervision of Prof. Zheng Zheng. He received his B.S. degree in the School of Electronic and Information Engineering, Beihang University, China in 2008. He is currently an assistant professor at the School of Cyber Science and Technology, Beihang University. His research interest is artificial intelligence, include privacy protection of machine learning, biometric identification and smart wireless communication. He has published about 15 papers in Biosystems Engineering, Computational Intelligence & Neuroscience, etc.

Jianwei Liu received the Ph.D. in communication engineering from Xidian University, China in 1998, and his B.S. and M.S. degrees in electronic engineering from Shandong University, China in 1985 and 1988. He is currently a professor and dean of School of Cyber Science and Technology, Beihang University. His current research interests include cryptographic protocol design, security on wireless and mobile network, computer network security, and Cryptography. He has published 6 books and more than 200 papers in his research fields. He is a senior member of the Chinese Institute of Electronics, and director of the Chinese Association for Cryptologic Research, and member of Teaching steering committee of information security of MOE, China. He has been awarded the first prize of technological invention of China.