information

Article

5G-IPAKA: An Improved Primary Authentication and Key Agreement Protocol for 5G Networks

Yuelei Xiao [1,2,*] and Yang Wu [3]

[1] School of Modern Posts, Xi'an University of Post and Telecommunications, Xi'an 710061, China
[2] Shaanxi Provincial Information Engineering Research Institute, Xi'an 710075, China
[3] School of Computer, Xi'an University of Post and Telecommunications, Xi'an 710061, China; wu1903220154@163.com
* Correspondence: xiaoyuelei@xupt.edu.cn; Tel.: +86-85383289

Abstract: The 3rd generation partnership project (*3GPP*) has been enhancing the security of the 5G AKA (authentication and key agreement) protocol. However, there may still be some shortcomings in the latest version of the 5G AKA protocol. According to the analysis of the latest version of the 5G AKA protocol, this paper points out seven of its shortcomings. To overcome these shortcomings, an improved primary authentication and key agreement protocol for 5G networks is proposed, which is named 5G-IPAKA. Compared with the latest version of the 5G AKA protocol, the main improvements include that the pre-shared key between the user equipment (*UE*) and the home network (*HN*) is replaced with a derivation key as the pre-shared key, the challenge—response mechanism for the serving network (*SN*) is added, the mutual authentication and key confirmation occurs between the *UE* and the *SN*, and the message authentication code (*MAC*) failure procedure is replaced with a timeout mechanism on the *HN*. Then, the 5G-IPAKA protocol is proven secure in the mixed strand space model for mixed protocols. Further discussion and comparative analysis show that the 5G-IPAKA protocol can overcome the above shortcomings of the latest version of the 5G AKA protocol, and is better than the recently improved 5G AKA protocols. Additionally, the 5G-IPAKA protocol is efficient and backward-compatible.

Keywords: AKA (authentication and key agreement); 5G AKA; 5G-IPAKA; mixed strand space model; pre-shared key; challenge-response; timeout mechanism

check for updates

Citation: Xiao, Y.; Wu, Y. 5G-IPAKA: An Improved Primary Authentication and Key Agreement Protocol for 5G Networks. *Information* **2022**, *13*, 125. https://doi.org/10.3390/info13030125

Academic Editor: Lorenzo Mucchi

Received: 13 February 2022
Accepted: 28 February 2022
Published: 2 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

With the continuous popularization of 5G communication technology, in the near future, the 5G network, as an important communication infrastructure, will penetrate into diverse vertical fields, such as in transportation, medical treatment, and industry, and will also support various information interactions between people, people and things, and things and things [1]. In the 5G network, three different primary authentication and key agreement protocols are defined in the related 3rd generation partnership project (*3GPP*) specifications [2–4], including the 5G AKA (authentication and key agreement) protocol, the EAP-AKA' protocol, and the 5G EAP-TLS protocol. The first two protocols are based on the shared key cryptography, while the last one is based on the public key cryptography. These protocols all aim to provide mutual authentication of subscribers and networks. Currently, they are in the process of standardization.

The 5G AKA protocol [2–4] was developed directly from the evolution packet system (*EPS*)-AKA protocol of the long-term evolution (*LTE*)/4G network [3], so it inherited certain security vulnerabilities from the EPS-AKA protocol, such as impersonation attacks, man-in-the-middle attacks (*MitM*), and denial of service (*DoS*) attacks [5–11]. In [12], the authors analyzed the 5G AKA protocol of TS 33.501 v0.7.0. They discovered a protocol vulnerability that would enable an attacker to impersonate another user in a serving network (*SN*). Based

on the Tamarin model checker [13], Basin et al. [14] investigated the security properties of the 5G AKA protocol of TS 33.501 v15.1.0, and several major issues were revealed, which were related to user localization, the leakage of activity, the impact of active attackers, and the presence of malicious *SN* while roaming. In [15], the authors pointed out that the 5G AKA protocol suffers from link ability attacks, and proposed a new authentication scheme by making use of the Diffie–Hellman key exchange algorithm to generate the session key. This scheme was successful in preventing link ability attacks along with an MitM attack.

For the more recently 5G AKA protocol, the authors in [16] found a new attack type. They claimed that the protection mechanism of the sequence number (*SQN*) can be defeated under specific replay attacks due to its use of exclusive-OR (*XOR*) and a lack of randomness. In [17], the authors modeled all key components of the 5G AKA protocol (i.e., the user equipment, the serving network, and the home network) according to the definition in the 3GPP specification document. They discovered an attack that exploits a potential race condition and additionally showed that solving the race condition for the honest case does not necessarily prevent the attack. In [18], the authors investigated the privacy properties of the 5G AKA protocol using the Bana–Comon logic [19,20]. They discovered a novel de-synchronization attack and proved that their proposed protocol guarantees the privacy properties. In [21], the authors proposed a novel version of the 5G AKA protocol to prevent active attacks and gain resistance against malignant serving networks. Unfortunately, there is a possibility of an *SN* impersonation, so this scheme does not eliminate the vulnerability towards the MitM attack. Further, Gharsallah et al. in [22] also attempted to launch a revised version of the 5G AKA protocol. However, their proposed protocol suffers from privacy preservation, as the device identities are clearly transmitted in the air, which leads to numerous security attacks.

As time goes on, more attacks on the 5G AKA protocol were found due to the insecure channel between different network domains in the legacy mobile network. In [23], the authors discovered an attack exploiting subscription concealed identifier (*SUCI*) to track a subscriber in the 5G network, which is directly caused by the insecure air channel. To cover this issue, they proposed a secure authentication scheme by utilizing the existing public key infrastructure (*PKI*) mechanism. Further, they found a location sniffing attack, which can be implemented by an attacker through inexpensive devices [24]. Similarly, they proposed a fix scheme based on the existing PKI mechanism. In [25], the authors modeled the 5G AKA protocol with symbolic modeling using ProVerif based on three and four entities models, and then proposed their security consideration. Further, Mariya et al. [26] proposed an enhanced version of the authentication and key agreement protocol for 5G system that surmounts the limitations existing in the 5G AKA protocol. Parne et al. [27] introduced a protocol that preserves the privacy of the user identity and overcomes the identified problems of the 5G AKA protocol. Similarly, 3GPP has also been used to enhance the security of the 5G AKA protocol [2–4].

However, there may still be some shortcomings in the latest version of the 5G AKA protocol. To solve this problem, we first point out these possible shortcomings. Then, we propose an improved primary authentication and key agreement protocol for 5G networks, named 5G-IPAKA. Finally, we prove that the 5G-IPAKA protocol is secure and that it is efficient and backward-compatible.

The main contributions of this paper are as follows:

- By analyzing the latest version of the 5G AKA protocol, we point out that the protocol still has seven shortcomings;
- We propose a new 5G-IPAKA protocol by improving the latest version of the 5G AKA protocol from four aspects;
- We formally analyze the security of the 5G-IPAKA protocol in the mixed strand space model for mixed protocols [28]. As a result, the 5G-IPAKA protocol is secure in the mixed strand space model;
- Through discussion and analysis, we are able to overcome the above shortcomings of the latest version of the 5G AKA protocol;

- Through discussion and a comparative analysis, we show that the new 5G-IPAKA protocol is better than the recently improved 5G AKA protocols in overcoming the various shortcomings, and is efficient and backward-compatible.

The rest of this paper is organized as follows. Section 2 provides an overview of the latest version of the 5G AKA protocol. In Section 3, we point out seven shortcomings of the latest version of the 5G AKA protocol. Section 4 describes our proposed 5G-IPAKA protocol. Section 5 provides a formal verification of the 5G-IPAKA protocol in the mixed strand space model. In Section 6, we present the discussion and analysis, and conclude the paper in Section 7.

2. Overview of the 5G AKA Protocol

According to [2–4], the steps of the latest version of the 5G AKA protocol in the 3GPP standard version v17.4.0 of TS 33.501 are illustrated in Figure 1.

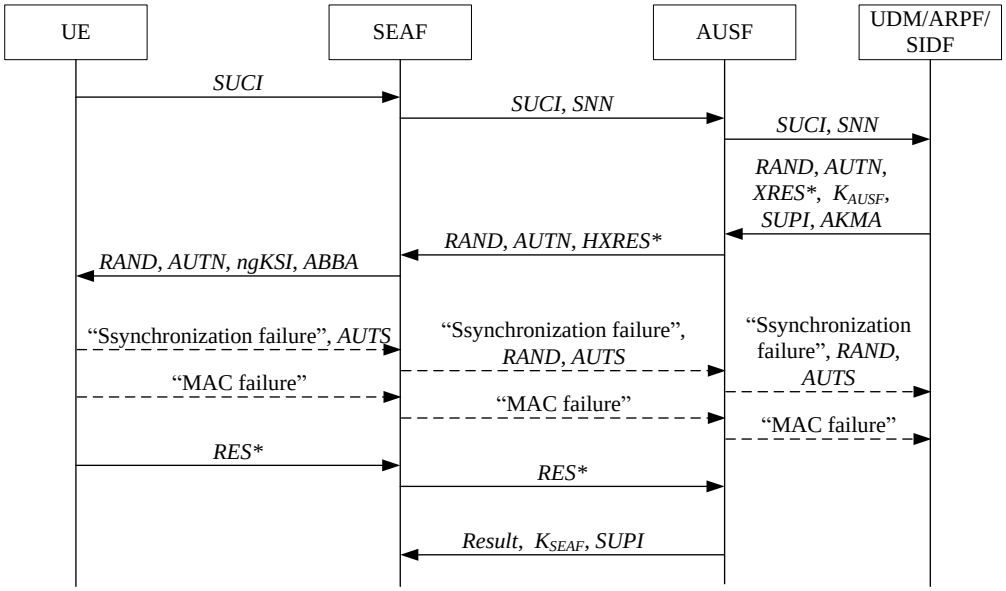

Figure 1. The steps of the latest version of the 5G AKA protocol.

In Figure 1, the universal subscriber identity module (*USIM*) and the mobile equipment (*ME*) are located in the user equipment (*UE*), and the security anchor function (*SEAF*) is located in the *SN*. The authentication server function (*AUSF*), the unified data management (UDM), the authentication credential repository and processing function (*ARPE*), and the subscriber identity de-concealing function (*SIDF*) are located in the home network (*HN*). The messages between the *SN* and the *HN* are usually protected. The detailed steps of the latest version of the 5G AKA protocol are as follows:

1. When the *SEAF* initiates an authentication with the *UE*, the *UE* sends $SUCI$ to the *SEAF*, where the *UE* includes the ME and the USIM. $SUCI$ denotes a *SUCI* of the *UE* and $SUCI = x \cdot G||\{SUPI\}_{EK}||MAC_{UE}$, where $SUPI$ denotes the subscription permanent identifier (*SUPI*) of the *UE*, $x \cdot G$ and x are an ephemeral public–private key pair of the *UE* for Diffie–Hellman exchange, $y \cdot G$ and y are the ephemeral public–private key pair of the *HN* for Diffie–Hellman exchange, $EK||ICB||MK = KDF(x \cdot y \cdot G)$ and $MAC_{UE} = HMAC(MK, \{SUPI\}_{EK})$, EK is an encryption key, ICB is an initial counter block (*ICB*), MK is a message authentication code (*MAC*) key, MAC_{UE} is a *MAC* of the *UE*, $KDF()$ is a key derivation function, and $HMAC()$ is a hash function for computing *MAC*;
2. Upon receiving $SUCI$, the *SEAF* sends $SUCI$ and SNN to the *AUSF*. SNN denotes the serving network name (*SNN*) of the *SN*;

3. If the *SEAF* is entitled to use SNN, then the *AUSF* stores the receiving SNN and sends $SUCI$ and SNN to the UDM;
4. The UDM invokes the *SIDF* regardless of whether $SUCI$ is received. Then, the *SIDF* de-conceals $SUCI$ to gain $SUPI$ before the UDM can process the request. Based on $SUPI$, the UDM/ARPF chooses the authentication method;
5. When 5G AKA is selected, the UDM/ARPF generates $RAND$, calculates $AUTN$ and $XRES*$, and derives K_{AUSF}, and then creates a 5G home environment authentication vector (5G HE AV) from $RAND$, $AUTN$, $XRES*$, and K_{AUSF}. $RAND$ is an unpredictable challenge of the *HN*. $AUTN$ is an authentication token of the *HN* and $AUTN = SQN \oplus AK||AMF||MAC$, where SQN is a fresh sequence number generated by the *HN*, AK is an anonymity key and $AK = f_5(K, RAND)$, *AMF* is the authentication management field (*AMF*) and the separation bit of the *AMF* is set 1, MAC is a *MAC* of the *HN* and $MAC = f_1(K, SQN||RAND||AMF)$, K is a long-term key between the *UE* and the *HN*, $f_1()$ is a message authentication function, and $f_5()$ is a key-generating function. Here, $XRES* = KDF(CK||IK, SNN\ ||RAND||XRES)$, where CK is a cipher key and $CK = f_3(K, RAND)$, IK is an integrity key and $IK = f_4(K, RAND)$, $XRES$ is an expected response and $XRES = f_2(K, RAND)$, $f_2()$ is a message authentication function, and $f_3()$ and $f_4()$ are two key-generating functions. K_{AUSF} is a key derived from CK and IK, and $K_{AUSF} = KDF(CK||IK, SNN||SQN \oplus AK)$;
6. The UDM sends the 5G HE AV to the *AUSF* together with $SUPI$. When an authentication and key management for applications (*AKMA*) subscription is used, the UDM also sends $AKMA$ to the *AUSF*. $AKMA$ denotes the *AKMA* indication and routing indicator;
7. The *AUSF* stores the $XRES*$ temporarily together with the received $SUPI$;
8. The *AUSF* generates a 5G AV from the 5G HE AV received from the UDM/ARPF by computing $HXRES*$ from $XRES*$, computing K_{SEAF} from K_{AUSF}, replacing $XRES*$ with $HXRES*$, and replacing K_{AUSF} with K_{SEAF} in the 5G HE AV, where $HXRES* = SHA256(RAND||XRES*)$, $K_{SEAF} = KDF(K_{AUSF}, SNN)$, and $SHA256()$ is a hash function;
9. The ASUF creates a 5G serving environment authentication vector (5G SE AV) by removing K_{SEAF} from the 5G AV, then sends the 5G SE AV (i.e., $RAND$, $AUTN$, and $HXRES*$) to the *SEAF*;
10. The *SEAF* stores $HXRES*$, and then sends $RAND$, $AUTN$, $ngKSI$, and $ABBA$ to the *UE*. Here, $ngKSI$ is used by the *UE* and the access and mobility management function (*AMF*) to identify the K_{AMF} and the partial native security context that is created if the authentication is successful. $ABBA$ denotes the anti-bidding down between architectures (*ABBA*) parameter;
11. In the *UE*, the ME forwards $RAND$ and $AUTN$ to the USIM. Upon receipt of $RAND$ and $AUTN$, the USIM first computes the anonymity key AK and retrieves the sequence number $SQN = (SQN \oplus AK) \oplus AK$. Next, the USIM computes $XMAC = f_1(K, SQN||RAND||AMF)$ and compares this with MAC, which is included in $AUTN$. Then, the USIM verifies that the received SQN is in the correct range. If $XMAC$ is the same as MAC and SQN is in the correct range, then the USIM computes a response $RES = f_2(K, RAND)$, CK, and IK, and then returns RES, CK, and IK to the ME. The ME then computes $RES* = KDF(CK||IK, SNN||RAND||RES)$, K_{AUSF}, and K_{SEAF};
12. The *UE* sends $RES*$ to the *SEAF*;
13. The *SEAF* computes $HRES* = SHA256(RAND||RES*)$ and compares this with $HXRES*$. If they coincide, then the *SEAF* considers the authentication successful from the serving network point of view; if not, then the *SEAF* considers the authentication unsuccessful;
14. The *SEAF* sends the received $RES*$ to the *AUSF*;
15. The *AUSF* compares the received $RES*$ with the stored $XRES*$. If $RES*$ and $XRES*$ are equal, then the *AUSF* considers the authentication successful from the home

network point of view. Then, the *AUSF* informs the UDM about the authentication result;

16. The *AUSF* indicates to the *SEAF* whether the authentication was successful or not from the home network point of view (i.e., *Result*). If the authentication was successful, then the ASUF also sends K_{SEAF} and $SUPI$ to the *SEAF*.

In step 11, if $XMAC$ and MAC are different, then the USIM indicates to the ME an *MAC* failure of $AUTN$. Then, the *UE* sends a "*MAC* failure" indication to the *SEAF*. Further, the *SEAF* sends the "*MAC* failure" indication to the *AUSF*. Finally, the ASUF sends the "*MAC* failure" indication to the UDM/ARPF.

In step 11, if SQN is not in the correct range, then the USIM computes $AUTS = SQN_{UE} \oplus AK^* || MAC - S$, and then sends $AUTS$ with a "synchronization failure" indication to the ME, where SQN_{UE} denotes the highest sequence number the USIM has accepted, $AK^* = f_5^*(K, RAND)$, $MAC - S = f_1^*(K, SQN_{UE} || RAND || AMF_0)$, AMF_0 is a dummy value of all zeros, $f_1^*()$ is a message authentication function, and $f_5^*()$ is a key-generating function. Then, the *UE* sends $AUTS$ with a "synchronization failure" indication to the *SEAF*. Further, the *SEAF* sends $RAND$ and $AUTS$ with a "synchronization failure" indication to the *AUSF*. Finally, the ASUF sends $RAND$ and $AUTS$ with a "synchronization failure" indication to the UDM/ARPF.

3. Shortcomings of the 5G AKA Protocol

According to the analysis of the above 5G AKA protocol, there are still some shortcomings in the latest version of the 5G AKA protocol, as follows:

- $SUCI$ **can be replayed without being found.** The *HN* cannot find out whether $SUCI$ is a replayed message because $SUCI$ does not contain the challenge of the *HN*. Similarly, the *UE* cannot find out whether $SUCI$ is a replayed message because $AUTN$ does not contain the challenge of the *UE* (i.e., x), which is included in $SUCI$ generated by the *UE*;
- **Mutual authentication between the *UE* and the *SN* cannot be established.** The *UE* cannot authenticate the *SN* because $AUTN$ does not contain SNN. Similarly, the *SN* cannot authenticate the *UE* for the following three reasons. Firstly, the *SN* does not verify $SUCI$, $AUTN$, $HXRES*$, $RES*$, and $AUTS$. Secondly, the second received message of the *SN* does not contain $SUPI$ to match with $SUCI$ in the first received message of the *SN*. Finally, the last received message of the *SN* does not contain $RAND$, meaning that $SUPI$ in the last received message of the *SN* cannot match with the *UE* identity in $AUTN$ and $HXRES*$, which are included in the second received message of the *SN*;
- K_{SEAF} **cannot reach an agreement.** The last received message of the *SN* does not contain $RAND$, so this message can be a replayed message, meaning that K_{SEAF} on the *SN* is not equal to K_{SEAF} on the *HN*. As a result, K_{SEAF} on the *SN* is not equal to K_{SEAF} on the *UE*;
- **The location privacy of the *UE* can be compromised.** Because $AUTN$ does not contain the challenge of the *UE* (i.e., x), the first received message of the *UE* can be a replayed message. If $SQN \subset AUTN$ is in the correct range, then the location of the *UE* can be compromised by reidentifying $RES*$. If $SQN \subset AUTN$ is not in the correct range, then the location privacy of the *UE* can be compromised by identifying the "synchronization failure" indication; that is to say, when the first received message of the *UE* is replayed, the legitimate *UE* response is $RES*$ or a "synchronization failure" indication, but any other *UE* response is a "*MAC* failure" indication. As a result, the location privacy of the legitimate *UE* can be compromised;
- **DoS attacks against the *SN* can be formed.** Because the received messages of the *SN* does not contain the challenge of the *SN*, these messages can be replayed messages. As a result, the penetrator can impersonate the *UE* and the *HN* to complete the entire 5G AKA protocol with the *SN*, forming DoS attacks against the *SN*;

- **Attacks based on *MAC* failure can be performed.** Firstly, the penetrator can forge or tamper with the first received message of the *UE* to make the *UE* respond to a "*MAC* failure" indication, resulting in authentication failure. Secondly, the penetrator can directly send a "*MAC* failure" indication to the *SN* to cause authentication failure. Finally, the penetrator can also replay a "*MAC* failure" indication between the *SN* and the *HN* to cause authentication failure;
- **Perfect forward secrecy cannot be provided.** In the latest version of the 5G AKA protocol, if K is leaked, then the penetrator can calculate K_{AUSF} and K_{SEAF} based on those messages transmitted in the past run of the protocol. As a result, the penetrator can decrypt those encrypted communication messages transmitted in the past run of the protocol. Therefore, the latest version of the 5G AKA protocol cannot provide perfect forward secrecy.

4. Our Proposed 5G-IPAKA Protocol

In order to overcome the above shortcomings of the latest version of the 5G AKA protocol, we propose the 5G-IPAKA protocol, which is illustrated in Figure 2.

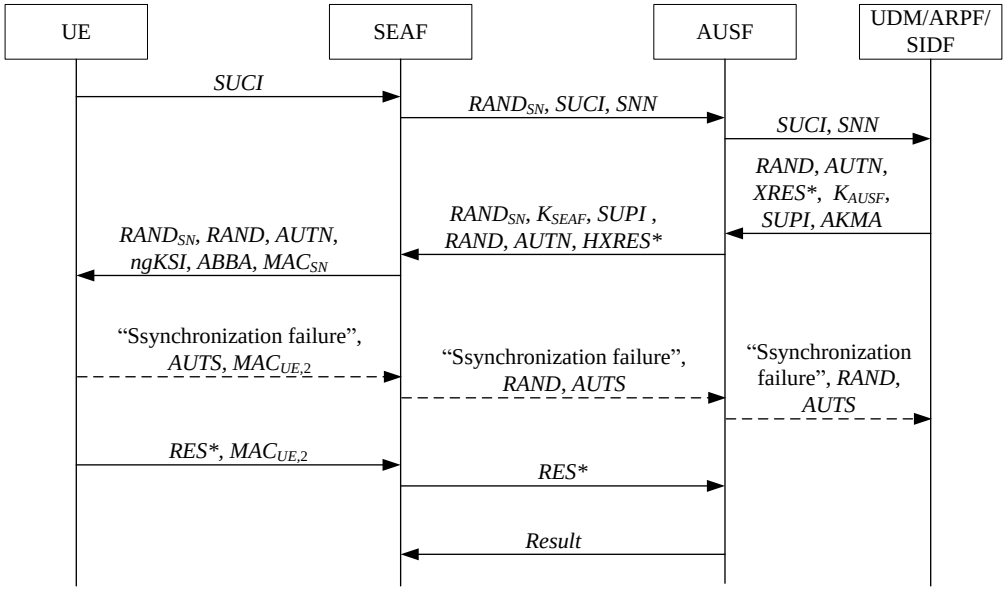

Figure 2. Our proposed 5G-IPAKA protocol.

In Figure 2, the detail steps of the 5G-IPAKA protocol are shown, as follows:

1. When the *SEAF* initiates an authentication with the *UE*, the *UE* sends $SUCI$ to the *SEAF*;
2. Upon receiving $SUCI$, the *SEAF* generates $RAND_{SN}$ and then sends $RAND_{SN}$, $SUCI$, and SNN to the *AUSF*, where $RAND_{SN}$ is an unpredictable challenge of the *SEAF*;
3. If the *SEAF* is entitled to use SNN, then the *AUSF* stores the receiving SNN and sends $SUCI$ and SNN to the UDM;
4. The UDM invokes the *SIDF* whether $SUCI$ is received or not. Then, the *SIDF* deconceals $SUCI$ to gain $SUPI$ before the UDM can process the request. Based on $SUPI$, the UDM/ARPF chooses the authentication method;
5. When 5G-IPAKA is selected, the UDM/ARPF generates $RAND$, calculates $AUTN$ and $XRES*$, and derives K_{AUSF}, and then creates a 5G HE AV from $RAND$, $AUTN$, $XRES*$, and K_{AUSF}, where $AUTN = SQN \oplus AK||AMF||MAC$, $AK = f_5(BK, RAND)$, $MAC = f_1(BK, SQN||RAND||AMF)$, $CK = f_3(BK, RAND)$, $IK = f_4(BK, RAND)$, $XRES = f_2(BK, RAND)$, $XRES* = KDF(CK||IK, SNN||RAND||XRES)$, $K_{AUSF} = KDF(CK||IK, SNN||SQN \oplus AK)$, and $BK = KDF(K, x \cdot y \cdot G||SNN)$;
6. The UDM sends the 5G HE AV to the *AUSF* together with $SUPI$. When an $AKMA$ subscription is used, the UDM also sends $AKMA$ to the *AUSF*;

7. The $AUSF$ stores the $XRES*$ temporarily together with the received $SUPI$;
8. The $AUSF$ generates a 5G AV from the 5G HE AV received from the UDM/ARPF by computing $HXRES*$ from $XRES*$, computing K_{SEAF} from K_{AUSF}, replacing $XRES*$ with $HXRES*$, and replacing K_{AUSF} with K_{SEAF} in the 5G HE AV;
9. The ASUF creates a 5G SE AV by adding $SUPI$ to the 5G AV, then sends the 5G SE AV (i.e., $RAND$, $AUTN$, $HXRES*$, K_{SEAF}, and $SUPI$) together with $RAND_{SN}$ to the $SEAF$;
10. The $SEAF$ stores $HXRES*$, computes MAC_{SN}, and then sends $RAND_{SN}$, $RAND$, $AUTN$, $ngKSI$, $ABBA$, and MAC_{SN} to the UE, where MAC_{SN} is a MAC of the $SEAF$ and $MAC_{SN} = HMAC(K_{SEAF}, RAND_{SN}||RAND||AUTN||ngKSI||ABBA)$;
11. In the UE, the ME forwards $RAND$ and $AUTN$ to the USIM. Upon receipt of $RAND$ and $AUTN$, the USIM first computes $BK = KDF(K, x \cdot y \cdot G||SNN)$ and the anonymity key $AK = f_5(BK, RAND)$ and retrieves the sequence number $SQN = (SQN \oplus AK) \oplus AK$. Next, the USIM computes $XMAC = f_1(BK, SQN\ ||RAND||AMF)$ and compares this with MAC which is included in $AUTN$. Then, the USIM verifies that the received SQN is in the correct range. If $XMAC$ is the same as MAC and SQN is in the correct range, then the USIM computes a response $RES = f_2(BK, RAND)$, $CK = f_3(BK, RAND)$, and $IK = f_4(BK, RAND)$, and then returns RES, CK, and IK to the ME. The ME then computes $RES* = KDF(CK||IK, SNN||RAND||RES)$, K_{AUSF}, and K_{SEAF}. Finally, the ME verifies MAC_{SN} using K_{SEAF}. If the verification fails, then the ME aborts;
12. The UE computes $MAC_{UE,2}$, and then sends $RES*$ and $MAC_{UE,2}$ to the $SEAF$, where $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}||RES*)$ is another MAC of the UE;
13. The $SEAF$ verifies $MAC_{UE,2}$. If the verification fails, then the $SEAF$ aborts. Otherwise, the $SEAF$ computes $HRES* = SHA256(RAND||RES*)$ and compares this with $HXRES*$. If they coincide, then the $SEAF$ considers the authentication as successful from the serving network point of view. If not, then the $SEAF$ considers the authentication as unsuccessful;
14. The $SEAF$ sends the received $RES*$ to the $AUSF$;
15. The $AUSF$ compares the received $RES*$ with the stored $XRES*$. If $RES*$ and $XRES*$ are equal, then the $AUSF$ considers the authentication as successful from the home network point of view. Then, the $AUSF$ informs the UDM about the authentication result;
16. The $AUSF$ indicates to the $SEAF$ whether the authentication was successful or not from the home network point of view (i.e., $Result$).

In step 11, if $XMAC$ and MAC are different, then the UE directly discards the first received message of the UE without responding to a "MAC failure" indication, so the HN will initiate a new authentication procedure towards the UE when the HN does not receive an authentication response message or a synchronization failure message within a certain period of time.

In step 11, if SQN is not in the correct range, then the USIM computes $AUTS = SQN_{UE} \oplus AK^*||MAC - S$, and then sends $AUTS$ with a "synchronization failure" indication to the ME, where $AK^* = f_5^*(BK, RAND)$ and $MAC - S = f_1^*(BK,\ SQN_{UE}||RAND||AMF_0)$. Then, the ME computes $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}\ ||Syncf||AUTS)$, and then sends $AUTS$ and $MAC_{UE,2}$ with a "synchronization failure" indication to the $SEAF$, where $Syncf$ = "Synchronization failure". Further, the $SEAF$ verifies $MAC_{UE,2}$; if the verification fails then the $SEAF$ aborts, otherwise the $SEAF$ sends $RAND$ and $AUTS$ with a "synchronization failure" indication to the $AUSF$. Finally, the ASUF sends $RAND$ and $AUTS$ with a "synchronization failure" indication to the UDM/ARPF;

Note that the fields not specifically explained in the above steps are the same as Figure 1. Compared with the latest version of the 5G AKA protocol, the main improvements of our proposed 5G-IPAKA protocol are as follows:

- **Replace the pre-shared key between the *UE* and the *HN* with a derivation key of the pre-shared key.** In detail, K is replaced with $BK = KDF(K, x \cdot y \cdot G||SNN)$ on the *UE* and the *HN*;
- **Add the challenge-response mechanism for the *SN*.** Firstly, $RAND_{SN}$ is added to the first send message of the *SEAF* as a challenge and is added to the second received message of the *SEAF* as a response. Then, $RAND_{SN}$ is added to the second send message of the *SEAF* as a challenge and is added to the third received message of the *SEAF* as a response (i.e., $RAND_{SN}$ in $MAC_{UE,2}$);
- **Add the mutual authentication and key confirmation between the *UE* and the *SN*.** Firstly, K_{SEAF} and $SUPI$ are moved to the second sent message of the $AUSF$ from the last sent message of the $AUSF$. Then, the *UE* and the *SN* perform a mutual authentication and key confirmation process based on MAC_{SN} and $MAC_{UE,2}$, which are generated by using K_{SEAF};
- **Replace the *MAC* failure procedure with the timeout mechanism on the *HN*.** If $XMAC$ in the received $AUTN$ and MAC calculated locally by the *UE* are different, then the *UE* directly discards the first received message of the *UE* without responding to a "*MAC* failure" indication, so the *HN* will initiate a new authentication procedure towards the *UE* when the *HN* does not receive an authentication response message or a synchronization failure message within a certain period of time.

5. Formal Verification of the 5G-IPAKA Protocol

To simplify the formal verification of the 5G-IPAKA protocol, we assume the following:

Assumption 1. *The parties of the 5G-IPAKA protocol shown in Figure 2 are simplified as the* UE, *the* SN, *and the* HN*;*

Assumption 2. *There is a session key between the* SN *and the* HN, *and it is secure;*

Assumption 3. *Here, ngKSI and ABBA do not affect the security of the 5G AKA protocol, so they are ignored.*

According to these assumptions, the 5G-IPAKA protocol shown in Figure 2 can be summarized into two cases as follows:

Case I: The verification of $AUTN$ succeeds and the authentication is successful. The steps of this case are as follows:

1. $UE \rightarrow SN: SUCI;$
2. $SN \rightarrow HN: \{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}};$
3. $HN \rightarrow SN: \{RAND_{SN}||K_{SEAF}||SUPI||RAND||AUTN||HXRES*\}_{K_{SN,HN}};$
4. $SN \rightarrow UE: RAND_{SN}||RAND||AUTN||MAC_{SN};$
5. $UE \rightarrow SN: RES*||MAC_{UE,2};$
6. $SN \rightarrow HN: \{RES*\}_{K_{SN,HN}};$
7. $HN \rightarrow SN: \{Result\}_{K_{SN,HN}}.$

Case II: The verification of $AUTN$ fails and it is a synchronization failure. The steps of this case are as follows:

1. $UE \rightarrow SN: SUCI;$
2. $SN \rightarrow HN: \{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}};$
3. $HN \rightarrow SN: \{RAND_{SN}||K_{SEAF}||SUPI||RAND||AUTN||HXRES*\}_{K_{SN,HN}};$
4. $SN \rightarrow UE: RAND_{SN}||RAND||AUTN||MAC_{SN};$
5. $UE \rightarrow SN: Syncf||AUTS||MAC_{UE,2};$
6. $SN \rightarrow HN: \{Syncf||RAND||AUTS\}_{K_{SN,HN}}.$

In the above cases, K on the *UE* and the *HN* is replaced with BK, where $BK = KDF(K, x \cdot y \cdot G||SNN)$. $K_{SN,HN}$ denotes the session key between the *SN* and the *HN*.

The strand space model [28–30] is a well-studied formal analysis method for security protocols. In [28], the authors studied the case of mixed protocols, where principals use

secret material in more than one protocol. In such cases, the two protocols can potentially interact, forming vulnerabilities that are not present in either protocol alone.

As mentioned above, there are two cases in the 5G-IPAKA protocol, so there may be interactions between these cases, forming vulnerabilities that do not exist in any single case. Therefore, we use the mixed strand space model [28] to analyze the security of our proposed 5G-IPAKA protocol as follows.

Definition 1. *A regular strand space $\sum_I$ is a space for case I of the 5G-IPAKA protocol if $\sum_I$ is the union of three kinds of strands: (1) Initiator strands $s \in Init_I[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, RES*]$ with trace: $< +SUCI, -RAND_{SN}||RAND||AUTN||MAC_{SN}, +RES*||MAC_{UE,2} >$. The principal associated with this strand is UE. XMAC computed locally is equal to $MAC \subset AUTN$ and $SQN \subset AUTN$ is in the correct range (i.e., $SQN_{UE} < SQN$). (2) Responder strands $r \in Resp_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, H_1, H_2, H_3, Result, K_{SEAF}, SUPI]$ with trace: $< -SUCI, +\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}}, -\{RAND_{SN}||K_{SEAF}||SUPI||RAND||H_1||H_2\}_{K_{SN,HN}}, +RAND_{SN}||RAND||H_1||MAC_{SN}, -H_3||MAC_{UE,2}, +\{H_3\}_{K_{SN,HN}}, -\{Result\}_{K_{SN,HN}} >$. The principal associated with this strand is SN. H_1, H_2 and H_3 are three messages that are not inspected by SN, where $H_2 = SHA256(RAND||H_3)$. (3) Server strands $t \in Serv_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$ with trace: $< -\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}}, +\{RAND_{SN}||K_{SEAF}||SUPI||RAND||AUTN||HXRES*\}_{K_{SN,HN}}, -\{RES*\}_{K_{SN,HN}}, +\{Result\}_{K_{SN,HN}} >$. The principal associated with this strand is HN.*

Definition 2. *A regular strand space $\sum_{II}$ is a space for case II of the 5G-IPAKA protocol if $\sum_{II}$ is the union of three kinds of strands: (1) Initiator strands $s \in Init_{II}[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, Syncf, AUTS]$ with trace: $< +SUCI, -RAND_{SN}||RAND||AUTN||MAC_{SN}, +Syncf||AUTS||MAC_{UE,2} >$. The principal associated with this strand is UE. XMAC computed locally is equal to $MAC \subset AUTN$, but $SQN \subset AUTN$ is not in the correct range (i.e., $SQN_{UE} \geq SQN$). (2) Responder strands $r \in Resp_{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, H_1, H_2, Syncf, H_4]$ with trace: $< -SUCI, +\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}}, -\{RAND_{SN}||K_{SEAF}||SUPI||RAND||H_1||H_2\}_{K_{SN,HN}}, +RAND_{SN}||RAND||H_1||MAC_{SN}, -Syncf||H_4||MAC_{UE,2}, +\{Syncf||RAND||H_4\}_{K_{SN,HN}} >$. The principal associated with this strand is SN. H_1, H_2, and H_4 are three messages that are not inspected by SN. (3) Server strands $t \in Serv_{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$ with trace: $< -\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}}, +\{RAND_{SN}||K_{SEAF}||SUPI||RAND||AUTN||HXRES*\}_{K_{SN,HN}}, -\{Syncf||RAND||AUTS\}_{K_{SN,HN}} >$. The principal associated with this strand is HN.*

Definition 3. *An infiltrated strand space Σ,$\mathcal{P}$ is a space for the 5G-IPAKA protocol if $\Sigma = \Sigma_I \cup \Sigma_{II} \cup \mathcal{P}$, where penetrator strands $p \in \mathcal{P}$ [28–30].*

Theorem 1. *Suppose (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing an initiator strand $s \in Init_I[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, RES*]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique server strand $t \in Serv_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$ and a unique responder strand $r \in Resp_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$.*

Proof of Theorem 1. Since $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2. Because $MAC = f_1(BK, SQN||RAND||AMF)$ and $RAND$ uniquely originate in $\sum$, $MAC \subset AUTN \subset term(< s, 2 >)$ must uniquely originate on a server strand t according to Definitions 1 to 3. If t is a server strand of Definition 1, then $t \in \text{Serv}_\text{I}[UE, SN, HN, SUCI, SNN, RAND'_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assump-

tion 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{SN} = HMAC(K_{SEAF}, RAND_{SN}||RAND\ ||AUTN)$, $MAC_{SN} \subset term(< s, 2 >)$ must originate on a responder strand r. According to Assumption 2, $\{RAND_{SN}||K_{SEAF}||SUPI''||RAND||AUTN||H''_2\}_{K_{SN,HN}} = term(< r, 3 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $RAND'_{SN} = RAND_{SN}$. According to Assumption 2, $\{RES*\}_{K_{SN,HN}} = term(< t, 3 >)$ must originate on a responder strand $r' \in \text{Resp}_{\text{I}}[UE''', SN, HN, SUCI''', SNN, RAND'''_{SN}, RAND, H'''_1, HXRES*, RES*, Result, K'''_{SEAF}, SUPI''']$, where $SUPI''' \subset SUCI'''$. Similarly, $\{RAND'''_{SN}||K'''_{SEAF}||SUPI'''||RAND||H'''_1||HXRES*\}_{K_{SN,HN}} = term(< r', 3 >)$ must originate on a server strand t''. Since $RAND$ uniquely originates in $\sum$, $t'' = t$, so $RAND'''_{SN} = RAND_{SN}$, $K'''_{SEAF} = K_{SEAF}$, $H'''_1 = AUTN$, $SUPI''' = SUPI$ and $UE''' = UE$. Similarly, $\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}} = term(< t, 1 >)$ must originate on a responder strand r''. Since $RAND_{SN}$ uniquely originates in $\sum$, $r'' = r' = r$, then $SUCI''' = SUCI$.

If t is a server strand of Definition 2, then $t \in \text{Serv}_{\text{II}}[UE, SN, HN, SUPI, SUCI, SNN, RAND'_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS']$, where $SQN'_{UE} \subset AUTS'$. Since $BK \notin \mathcal{K}_P$, $MAC - S' = f_1^*(BK, SQN'_{UE}||RAND||AMF_0) \subset AUTS' \subset term(< t, 3 >)$ must originate on an initiator strand $s' \in \text{Init}_{\text{II}}[UE, SN, HN, SUCI, RAND''_{SN}, RAND, AUTN'', Syncf, AUTS']$, so x originates on $term(< s', 1 >)$. According to Assumption 1, x originates on $term(< s, 1 >)$. Since x uniquely originates in $\sum$, $s' = s$. However, $s' \in \text{Init}_{\text{II}}$ and $s \in \text{Init}_{\text{I}}$, $s' \neq s$. Hence, t is not a server strand of Definition 2. □

Theorem 2. *Suppose (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing an initiator strand $s \in Init_{II}[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, Syncf, AUTS]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique server strand $t \in Serv_{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$ and a unique responder strand $r \in Resp_{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$.*

Proof of Theorem 2. Since $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2. Because $MAC = f_1(BK, SQN||RAND||AMF)$ and $RAND$ uniquely originate in $\sum$, $MAC \subset AUTN \subset term(< s, 2 >)$ must uniquely originate on a server strand t according to Definitions 1–3.

If t is a server strand of Definition 1, then $t \in \text{Serv}_{\text{I}}[UE, SN, HN, SUCI, SNN, RAND'_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$. Since $BK \notin \mathcal{K}_P$, $CK = f_3(BK, RAND) \notin \mathcal{K}_P$ and $IK = f_4(BK, RAND) \notin \mathcal{K}_P$, so $CK||IK \notin \mathcal{K}_P$. Hence, $RES* \subset term(< t, 3 >)$ must originate on an initiator strand $s' \in \text{Init}_{\text{I}}[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN'', RES*]$, so x originates on $term(< s', 1 >)$. According to Assumption 1, x originates on $term(< s, 1 >)$. Since x uniquely originates in $\sum$, $s' = s$. However, $s' \in \text{Init}_{\text{I}}$ and $s \in \text{Init}_{\text{II}}$, $s' \neq s$. Hence, t is not a server strand of Definition 1.

If t is a server strand of Definition 2, then $t \in \text{Serv}_{\text{II}}[UE, SN, HN, SUPI, SUCI, SNN, RAND'_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS']$, where $SQN'_{UE} \subset AUTS'$. Since $BK \notin \mathcal{K}_P$, $MAC - S' = f_1^*(BK, SQN'_{UE}||RAND||AMF_0) \subset AUTS' \subset term(< t, 3 >)$ must originate on an initiator strand s'. Since x uniquely originates in $\sum$, $s' = s$, so $SQN'_{UE} = SQN_{UE}$ and $AUTS' = AUTS$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assumption 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{SN} = HMAC(K_{SEAF}, RAND_{SN}||RAND\ ||AUTN)$, $MAC_{SN} \subset term(< s, 2 >)$ must originate on a responder strand r. According to Assumption 2, $\{RAND_{SN}||K_{SEAF}||SUPI''||RAND||AUTN||H''_2\}_{K_{SN,HN}} = term(< r, 3 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $RAND'_{SN} = RAND_{SN}$. According to Assumption 2, $\{Syncf||RAND||AUTS\}_{K_{SN,HN}} = term(< t, 3 >)$ must originate on a responder strand $r' \in \text{Resp}_{\text{II}}[UE''', SN, HN, SUPI''', SUCI''', SNN, RAND'''_{SN}, K'''_{SEAF}, RAND, H'''_1, H'''_2, Syncf, AUTS]$, where $SUPI''' \subset SUCI'''$. Similarly, $\{RAND'''_{SN}||K'''_{SEAF}||SUPI'''||RAND||H'''_1||H'''_2\}_{K_{SN,HN}} = term(< r', 3 >)$ must originate on a server strand t''. Since $RAND$ uniquely originates in $\sum$, $t'' = t$, so $RAND'''_{SN} = RAND_{SN}$,

$K'''_{SEAF} = K_{SEAF}$, $H'''_1 = AUTN$, $H'''_2 = HXRES*$, $SUPI''' = SUPI$ and $UE''' = UE$. Similarly, $\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}} = term(< t,1 >)$ must originate on a responder strand r''. Since $RAND_{SN}$ uniquely originates in $\sum$, $r'' = r' = r$, so $SUCI''' = SUCI$. □

According to Theorems 1 and 2, UE successfully authenticates HN and SN, and injection agreement [28–30] can be established.

Theorem 3. *Suppose (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing a server strand $t \in Serv_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique initiator strand $s \in Init_I[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, RES*]$ and a unique responder strand $r \in Resp_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$.*

Proof of Theorem 3. Since $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2. Because $CK = f_3(BK, RAND)$ and $IK = f_4(BK, RAND)$, $CK \notin \mathcal{K}_P$, and $IK \notin \mathcal{K}_P$, so $CK||IK \notin \mathcal{K}_P$. Hence, $RES* = KDF(CK||IK, SNN||RAND||RES) \subset term(< t,3 >)$ must originate on an unique initiator strand $s \in \text{Init}_\text{I}[UE, SN, HN, SUCI, RAND'_{SN}, RAND, AUTN', RES*]$ according to Assumption 3, where $SQN' \subset AUTN'$. Similarly, $MAC' \subset AUTN' \subset term(< s,2 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $SQN' = SQN$ and $AUTN' = AUTN$. According to Assumption 2, $\{RES*\}_{K_{SN,HN}} = term(< t,3 >)$ must originate on a responder strand $r \in \text{Resp}_\text{I}[UE'', SN, HN, SUCI'', SNN, RAND''_{SN}, RAND, H''_1, HXRES*, RES*, Result, K''_{SEAF}, SUPI'']$, where $SUPI'' \subset SUCI''$. Similarly, $\{RAND''_{SN}||K''_{SEAF}||SUPI''||RAND||H''_1||HXRES*\}_{K_{SN,HN}} = term(< r,3 >)$ must originate on a server strand t''. Since $RAND$ uniquely originates in $\sum$, $t'' = t$, so $RAND''_{SN} = RAND_{SN}$, $K''_{SEAF} = K_{SEAF}$, $H''_1 = AUTN$, $SUPI'' = SUPI$ and $UE'' = UE$. Similarly, $\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}} = term(< t,1 >)$ must originate on a responder strand r'. Since $RAND_{SN}$ uniquely originates in $\sum$, $r' = r$, so $SUCI'' = SUCI$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assumption 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}||RES*)$, $MAC_{UE,2} \subset term(< r,5 >)$ must originate on an initiator strand s'. Since x uniquely originates in $\sum$, $s' = s$, so $RAND'_{SN} = RAND_{SN}$. □

Theorem 4. Suppose (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing a server strand $t \in \text{Serv}_\text{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique initiator strand $s \in \text{Init}_\text{II}[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, Syncf, AUTS]$ and a unique responder strand $r \in \text{Resp}_\text{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$.

Proof of Theorem 4. *Since* $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2, so $MAC - S = f_1^*(BK, SQN_{UE}||RAND||AMF_0) \subset AUTS \subset term(< t,3 >)$ must originate on a unique initiator strand $s \in \text{Init}_\text{II}[UE, SN, HN, SUCI, RAND'_{SN}, RAND, AUTN', Syncf, AUTS]$ according to Assumption 3, where $SQN' \subset AUTN'$. Similarly, $MAC' \subset AUTN' \subset term(< s,2 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $SQN' = SQN$ and $AUTN' = AUTN$. According to Assumption 2, $\{Syncf||RAND||AUTS\}_{K_{SN,HN}} = term(< t,3 >)$ must originate on a responder strand $r \in \text{Resp}_\text{II}[UE'', SN, HN, SUPI'', SUCI'', SNN, RAND''_{SN}, K''_{SEAF}, RAND, H''_1, H''_2, Syncf, AUTS]$, where $SUPI'' \subset SUCI''$. Similarly, $\{RAND''_{SN}||K''_{SEAF}||SUPI''||RAND||H''_1||H''_2\}_{K_{SN,HN}} = term(< r,3 >)$ must originate on a server strand t''. Since $RAND$ uniquely originates in $\sum$, $t'' = t$, so $RAND''_{SN} = RAND_{SN}$, $K''_{SEAF} = K_{SEAF}$, $SUPI'' = SUPI$, $UE'' = UE$, $H''_1 = AUTN$ and $H''_2 = HXRES*$. Similarly, $\{RAND_{SN}||SUCI||SNN\}_{K_{SN,HN}} = term(< t,1 >)$ must originate on a responder strand r'. Since

$RAND_{SN}$ uniquely originates in $\sum$, $r' = r$, so $SUCI'' = SUCI$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assumption 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}||Syncf||AUTS)$, $MAC_{UE,2} \subset term(< r, 5 >)$ must originate on an initiator strand s'. Since x uniquely originates in $\sum$, $s' = s$, so $RAND'_{SN} = RAND_{SN}$. □

According to Theorems 3 and 4, HN successfully authenticates UE and SN, and the injection agreement [28–30] can be established.

Theorem 5. *Suppose (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing a response strand $r \in Resp_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, H_1, H_2, H_3, Result, K_{SEAF}, SUPI]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique server strand $t \in Serv_I[UE, SN, HN, SUCI, SNN, RAND_{SN}, RAND, AUTN, HXRES*, RES*, Result, K_{SEAF}, SUPI]$, and a unique initiator strand $s \in Init_I[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, RES*]$.*

Proof of Theorem 5. Through Assumptions 2 and 3, $K_{SN,HN} \notin \mathcal{K}_P$ and $RAND$ uniquely originates in $\sum$, so $\{RAND_{SN}||K_{SEAF}||SUPI||RAND||H_1||H_2\}_{K_{SN,HN}} = term(< r, 3 >)$ must uniquely originate on a server strand t according to Definitions 1 to 3.

If t is a server strand of Definition 1, then $t \in \text{Serv}_\text{I}[UE, SN, HN, SUCI', SNN, RAND_{SN}, RAND, AUTN', (HXRES*)', (RES*)', Result, K_{SEAF}, SUPI]$, where $SUPI \subset SUCI'$, $x' \subset AUTN'$, $x' \subset (HXRES*)'$, $x' \subset (RES*)'$ and K_{SEAF} is generated for $SUPI$. Similarly, $\{RAND_{SN}||SUCI'||SNN\}_{K_{SN,HN}} = term(< t, 1 >)$ must originate on a responder strand r'. Since $RAND_{SN}$ uniquely originates in $\sum$, $r' = r$, so $SUCI' = SUCI$ and $x' = x$ according to Assumption 1. Hence, $AUTN' = AUTN$, $(HXRES*)' = HXRES*$ and $(RES*)' = RES*$. Since $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2. Because $CK = f_3(BK, RAND)$ and $IK = f_4(BK, RAND)$, $CK \notin \mathcal{K}_P$ and $IK \notin \mathcal{K}_P$, so $CK||IK \notin \mathcal{K}_P$. Hence, $RES* = KDF(CK||IK, SNN\ ||RAND||RES) \subset term(< t, 3 >)$ must originate on a unique initiator strand $s \in \text{Init}_\text{I}[UE, SN, HN, SUCI, RAND''_{SN}, RAND, AUTN'', RES*]$ according to Assumption 3, where $SQN'' \subset AUTN''$. Similarly, $MAC'' \subset AUTN'' \subset term(< s, 2 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $SQN'' = SQN$ and $AUTN'' = AUTN$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assumption 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}||RES*)$, $MAC_{UE,2} \subset term(< r, 5 >)$ must originate on an initiator strand s'. Since x uniquely originates in $\sum$, $s' = s$, so $RAND''_{SN} = RAND_{SN}$.

If t is a server strand of Definition 2, then $t \in \text{Serv}_\text{II}[UE, SN, HN, SUPI, SUCI', SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN', (HXRES*)', Syncf, AUTS']$, where $SUPI \subset SUCI'$, $x' \subset AUTN'$, $x' \subset (HXRES*)'$ and $x' \subset AUTS'$. Similarly, $\{Syncf||RAND||AUTS'\}_{K_{SN,HN}} = term(< t, 3 >)$ must originate on a responder strand $r' \in \text{Resp}_\text{II}[UE'', SN, HN, SUPI'', SUCI'', SNN, RAND''_{SN}, K''_{SEAF}, RAND, H''_1, H''_2, Syncf, AUTS']$. Similarly, $\{RAND''_{SN}||K''_{SEAF}||SUPI''||RAND||H''_1||H''_2\}_{K_{SN,HN}} = term(< r', 3 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $RAND''_{SN} = RAND_{SN}$ and $RAND_{SN}$ originates on $term(< r', 2 >)$. According to Assumptions 1 and 3, $RAND_{SN}$ originates on $term(< r, 2 >)$. Since $RAND_{SN}$ uniquely originates in $\sum$, $r' = r$. However, $r' \in \text{Resp}_\text{II}$ and $r \in \text{Resp}_\text{I}$, $r' \neq r$. Hence, t is not a server strand of Definition 2. □

Theorem 6. *Suppose: (1) $\sum$ is a space for the 5G-IPAKA protocol, and $\mathcal{C}$ is a bundle containing a response strand $r \in \text{Resp}_\text{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, H_1, H_2, Syncf, H_4]$; (2) $K \notin \mathcal{K}_P$ and $K_{SN,HN} \notin \mathcal{K}_P$; (3) x, $RAND$, $RAND_{SN}$ uniquely originates in $\sum$. Then, $\mathcal{C}$ contains a unique server strand $t \in \text{Serv}_\text{II}[UE, SN, HN, SUPI, SUCI, SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN, HXRES*, Syncf, AUTS]$ and a unique initiator strand $s \in \text{Init}_\text{II}[UE, SN, HN, SUCI, RAND_{SN}, RAND, AUTN, Syncf, AUTS]$.*

Proof of Theorem 6. According to Assumptions 2 and 3, $K_{SN,HN} \notin \mathcal{K}_P$ and $RAND$ uniquely originate in $\sum$, so $\{RAND_{SN}||K_{SEAF}||SUPI||RAND||H_1||H_2\}_{K_{SN,HN}} = term(< r,3 >)$ must uniquely originate on a server strand t according to Definitions 5–7.

If t is a server strand of Definition 1, then $t \in \text{Serv}_\text{I}[UE, SN, HN, SUCI', SNN, RAND_{SN}, RAND, AUTN', (HXRES*)', (RES*)', Result, K_{SEAF}, SUPI]$, where $SUPI \subset SUCI'$, $x' \subset AUTN'$, $x' \subset (HXRES*)'$, $x' \subset (RES*)'$, and K_{SEAF} is generated for $SUPI$. Similarly, $\{(RES*)'\}_{K_{SN,HN}} = term(< t,3 >)$ must originate on a responder strand $r' \in \text{Resp}_\text{I}[UE'', SN, HN, SUCI'', SNN, RAND''_{SN}, RAND, H''_1, (HXRES*)', (RES*)', Result, K''_{SEAF}, SUPI'']$. Similarly, $\{RAND''_{SN}||K''_{SEAF}||SUPI''||RAND||H''_1||(HXRES*)'\}_{K_{SN,HN}} = term(< r',3 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $RAND''_{SN} = RAND_{SN}$ and $RAND_{SN}$ originate on $term(< r',2 >)$. Through Assumptions 1 and 3, $RAND_{SN}$ originates on $term(< r,2 >)$. Since $RAND_{SN}$ uniquely originates in $\sum$, $r' = r$. However, $r' \in \text{Resp}_\text{I}$ and $r \in \text{Resp}_\text{II}$, $r' \neq r$. Hence, t is not a server strand of Definition 1.

If t is a server strand of Definition 2, then $t \in \text{Serv}_\text{II}[UE, SN, HN, SUPI, SUCI', SNN, RAND_{SN}, K_{SEAF}, RAND, AUTN', (HXRES*)', Syncf, AUTS']$, where $SUPI \subset SUCI'$, $x' \subset AUTN', x' \subset (HXRES*)'$ and $x' \subset AUTS'$. Similarly, $\{RAND_{SN}||SUCI'||SNN\}_{K_{SN,HN}} = term(< t,1 >)$ must originate on a responder strand r'. Since $RAND_{SN}$ uniquely originates in $\sum$, $r' = r$, so $SUCI' = SUCI$ and $x' = x$ according to Assumption 1. Hence, $AUTN' = AUTN$, $(HXRES*)' = HXRES*$ and $AUTS' = AUTS$. Since $BK = KDF(K, x \cdot y \cdot G||SNN)$, $BK \notin \mathcal{K}_P$ according to Assumption 2, $MAC - S = f_1^*(BK, SQN_{UE}||RAND||AMF_0) \subset AUTS \subset term(< t,3 >)$ must originate on a unique initiator strand $s \in \text{Init}_\text{II}[UE, SN, HN, SUCI, RAND''_{SN}, RAND, AUTN'', Syncf, AUTS]$ according to Assumption 3, where $SQN'' \subset AUTN''$. Similarly, $MAC'' \subset AUTN'' \subset term(< s,2 >)$ must originate on a server strand t'. Since $RAND$ uniquely originates in $\sum$, $t' = t$, so $SQN'' = SQN$ and $AUTN'' = AUTN$. According to t, K_{SEAF} is encrypted by $K_{SN,HN}$. According to Assumption 2, $K_{SEAF} \notin \mathcal{K}_P$. Because $MAC_{UE,2} = HMAC(K_{SEAF}, RAND_{SN}||Syncf||AUTS)$, $MAC_{UE,2} \subset term(< r,5 >)$ must originate on an initiator strand s'. Since x uniquely originates in $\sum$, $s' = s$, so $RAND''_{SN} = RAND_{SN}$. □

According to Theorems 5 and 6, SN successfully authenticates UE and HN, and the injection agreement [28–30] can be established.

6. Discussion

6.1. Security of the 5G-IPAKA Protocol

According to the above formal verification of the 5G-IPAKA protocol, mutual authentication between the *UE* and the *SN*, mutual authentication between the *UE* and the *SN*, and mutual authentication between the *SN* and the *HN* are established. Additionally, an injection agreement [28–30] among the *UE*, the *SN*, and the *HN* is established. Therefore, the 5G-IPAKA protocol is secure in the mixed strand space model.

Because K is replaced with $BK = KDF(K, x \cdot y \cdot G||SNN)$ on the *UE* and the *HN*, $AUTN$ must contain the challenge of the *UE* (i.e., x), which is included in $SUCI$ generated by the *UE*. Hence, the *UE* can find out whether $SUCI$ is a replayed message.

According to the above formal verification of the 5G-IPAKA protocol, mutual authentication between the *UE* and the *SN* is established. In addition, an injection agreement [28–30] among the *UE*, the *SN*, and the *HN* is established, so K_{SEAF} can reach an agreement among the *UE*, the *SN*, and the *HN*.

Because $AUTN$ contains the challenge of the *UE* (i.e., x), the first received message of the *UE* (including $AUTN$) cannot be a replayed message, preventing the location privacy of the *UE* from being compromised.

Since the received messages of the *SN* contain the challenge of the *SN* (i.e., $RAND_{SN}$), these messages cannot be some replayed messages, preventing DoS attacks against the *SN*. In addition, the *UE* directly discards the first received message without responding

to a "*MAC* failure" indication when $XMAC$ messages in the received $AUTN$ and MAC calculated locally by the *UE* are different, defending against attacks based on *MAC* failure.

Because K is replaced with $BK = KDF(K, x \cdot y \cdot G||SNN)$, and both K_{AUSF} and K_{SEAF} are generated based on BK, this provides perfect forward secrecy (PFS) based on the Diffie–Hellman exchange.

Hence, our proposed 5G-IPAKA protocol can overcome the above shortcomings in the latest version of the 5G AKA protocol.

A comparative analysis between the 5G-IPAKA protocol and the recently improved 5G AKA protocols [23,24,26,27] regarding the shortcomings of the latest version of the 5G AKA protocol is shown in Table 1.

Table 1. Comparative analysis between the 5G-IPAKA protocol and the recently improved 5G AKA protocols [23,24,26,27] regarding the shortcomings of the latest version of the 5G AKA protocol.

Shortcomings	5G AKA	[23]	[24]	[26]	[27]	5G-IPAKA
$SUCI$ can be replayed without being found	Yes	No	Yes	No	No	No
Mutual authentication between the *UE* and the *SN* cannot be established	Yes	Yes	Yes	Yes	Yes	No
K_{SEAF} cannot reach an agreement	Yes	Yes	Yes	Yes	Yes	No
The location privacy of the *UE* can be compromised	Yes	No	No	No	No	No
DoS attacks against the *SN* can be formed	Yes	Yes	Yes	Yes	Yes	No
Attacks based on *MAC* failure can be performed	Yes	Yes	No	No	No	No
Perfect forward secrecy cannot be provided	Yes	Yes	Yes	Yes	Yes	No

From Table 1, the recently improved 5G AKA protocols still have some of the shortcomings of the latest version of the 5G AKA protocol, but our proposed 5G-IPAKA overcomes all the shortcomings of the latest version of the 5G AKA protocol.

In [23], the Eph private key and Eph public key of the *UE* (i.e., x and $x \cdot G$), the public–private key pair of the *SN*, and the public–private key pair of the *HN* are used to ensure the security of the channel between the *UE* and the *SN*, the security of channel between the *UE* and the *HN*, and the security of the channel between the *SN* and the *HN*. Since the first received message of the *UE* is encrypted by the Eph public key of the *UE*, this means that the message can only be decrypted by the Eph private key of the *UE*, so it cannot be a replayed message, preventing the location privacy of the *UE* being compromised. In addition, the *UE* can find out whether $SUCI$ is a replayed message. However, the other parts fully inherit the 5G AKA protocol, so the other shortcomings of the 5G AKA protocol still exist in the protocol [23].

In [24], both the synchronization failure and the *MAC* failure are constructed as the format of $RES*$, making it impossible to distinguish them so as to prevent the location privacy of the *UE* being compromised and prevent attacks based on *MAC* failure. However, the other parts fully inherit the 5G AKA protocol, so the other shortcomings of the 5G AKA protocol still exist in the protocol of [24].

In [26], $SUCI$ is included in $AUTH_{SEAF}$ in the second received message of the *UE*, so the *UE* can find out whether $SUCI$ is a replayed message, where $AUTH_{SEAF}$ is an authentication token of the *SEAF*. Additionally, the protocol from [26] removes the synchronization failure procedure and the *MAC* failure procedure, preventing the location privacy of the *UE* from being compromised and defending against attacks based on *MAC* failure. Similarly, MAC_{ARPF} is also included in $AUTH_{SEAF}$ from the second received message of the *UE*, but it does not contain $SEAF_{ID}$, where MAC_{ARPF} is a *MAC* of the ARPF and $SEAF_{ID}$ is the identity of the *SEAF* (i.e., SNN mentioned above). This means that the *UE* cannot authenticate the *SN* being authenticated by the *HN*, meaning that mutual authentication between the *UE* and the *SN* cannot be established and K_{SEAF} cannot reach an agreement. In addition, $RAND_{SEAF}$ is included in the $RAND'_{UE}$ of the second received message of the *SEAF*, $HXRES*$ of the third received message of the *SEAF*, and $RES*$ of the fourth received message of the *SEAF*, although the *SEAF* does not verify these fields, so DoS attacks against the *SN* can be formed, where $RAND'_{UE}$ is calculated based on $RAND_{UE}$ and $RAND_{SEAF}$

(i.e., the challenges of the *UE* and the *SEAF*, respectively). Because K_{AUSF} and K_{SEAF} can be calculated when K is leaked, PFS cannot be provided.

In [27], the time synchronization among the *UE*, the *SN*, and the *HN* is maintained. T_{UE} is included in $SUCI$, so $SUCI$ cannot be a replayed message, where T_{UE} is a timestamp of the *UE*. Additionally, the protocol of [27] also removes the synchronization failure procedure and the *MAC* failure procedure, preventing the location privacy of the *UE* from being compromised and defending against attacks based on *MAC* failure. MAC_{SN} is included in the first received message of the *UE*, but it does not contain SNN. This means that the *UE* cannot authenticate the *SN* being authenticated by the *HN*, meaning that mutual authentication between the *UE* and the *SN* cannot be established and K_{SEAF} cannot reach an agreement. For the received messages, the *SN* does not verify T_{UE} and T_{HN} (i.e., a timestamp of the *HN*), but only verifies whether RES is equal to $XRES$ in phase 1 of the protocol from [27], meaning that DoS attacks against the *SN* can be formed. Similar to [26], PFS cannot be provided.

Therefore, our proposed 5G-IPAKA protocol is better than these recently improved 5G AKA protocols in overcoming the shortcomings of the latest version of the 5G AKA protocol.

6.2. Performance of the 5G-IPAKA Protocol

A comparative analysis between the 5G-IPAKA protocol and the recently improved 5G AKA protocols [23,24,26,27] regarding the number of messages, the amount of calculation, and backward compatibility is shown in Table 2.

Table 2. A comparative analysis between the 5G-IPAKA protocol and the recently improved 5G AKA protocols [23,24,26,27] regarding the number of messages, the amount of calculation, and backward compatibility.

Protocols	The Number of Messages	The Amount of Calculation	Backward Compatibility
5G AKA	11	1*ECDH*+1*ED*+12*F*+2*XOR*	-
[23]	11	4*PED*+1*ED*+10*F*+2*XOR*	No
[24]	11	2*PED*+1*ECDH*+1*ED*+13*F*+1*XOR*	No
[26]	9	1*ECDH*+1*ED*+12*F*	No
[27]	7	1*ED*+15*F*+1*LRCS*+6*XOR*	No
5G-IPAKA	9	1*ECDH*+1*ED*+16*F*+2*XOR*	Yes

In Table 2, the number of messages represents the number of messages among the *UE*, the *SN*, and the *HN*. *ECDH* denotes the generation and verification of an elliptic curve Diffie–Hellman (ECDH) exchange. *PED* denotes the generation and verification of a public key encryption and decryption process. *ED* denotes the generation and verification of a symmetric key encryption and decryption process. *F* denotes the generation and verification of a key function, key derivation function, *MAC* function, or a hash function, which are grouped into one category because they require the same amount of calculation [27]. *LRCS* denotes the left circular shift and the right circular shift. *XOR* denotes the generation and verification of an XOR value.

From Table 2, the number of messages in the 5G-IPAKA protocol is less than the 5G AKA protocol, although the amount of calculation is slightly higher than the 5G AKA protocol. The number of messages in the 5G-IPAKA protocol is less than the protocols in [23,24], and the amount of calculation is also lower than the protocols in [23,24] because they introduce multiple public key encryption and decryption processes. The number of messages in the 5G-IPAKA protocol is the same as the protocol in [26], although the amount of calculation is slightly higher than the protocol in [26]. The number of messages in the 5G-IPAKA protocol is more than in the protocol in [27], and the amount of calculation is also higher than the protocol in [27]. However, the protocol in [27] introduces a timestamp mechanism and must maintain the time synchronization among the *UE*, the *SN*, and the *HN*, which is difficult. Hence, our proposed 5G-IPAKA protocol is efficient.

Additionally, the protocols in [23,24,26,27] destroy the structure of the messages instead of adding fields to the messages or extending fields in the messages, so they are not backward-compatible. Our proposed 5G-IPAKA protocol only extends K and adds some fields to the messages among the *UE*, the *SN*, and the *HN*, so it is forward compatible.

7. Conclusions

In this paper, according to the analysis of the latest version of the 5G AKA protocol, we point out seven shortcomings of this protocol, including that $SUCI$ can be replayed without being found, mutual authentication between the *UE* and the *SN* cannot be established, K_{SEAF} cannot reach an agreement, the location privacy of the *UE* can be compromised, DoS attacks against the *SN* can be formed, attacks based on *MAC* failure can be performed, and PFS cannot be provided.

To overcome these shortcomings, we propose a 5G-IPAKA protocol. Compared with the latest version of the 5G AKA protocol, the main improvements of the 5G-IPAKA protocol include that the pre-shared key between the *UE* and the *HN* is replaced with a derivation key of the pre-shared key, the challenge-response mechanism for the *SN* is added, the mutual authentication and key confirmation between the *UE* and the *SN* is added, and the *MAC* failure procedure is replaced with a timeout mechanism on the *HN*.

Accordingly, we summarize the 5G-IPAKA protocol into two cases, and then use the mixed strand space model for mixed protocols to formally analyze the security of the 5G-IPAKA protocol. As a result, mutual authentication and injection among the *UE*, the *SN*, and the *HN* are established. Therefore, the 5G-IPAKA protocol is secure in the mixed strand space model.

Based on the further discussion and comparative analysis, the 5G-IPAKA protocol can overcome the above shortcomings of the latest version of the 5G AKA protocol, and is better than the recently improved 5G AKA protocols in overcoming these shortcomings. In addition, the 5G-IPAKA protocol is efficient and backward-compatible.

Recently, some authors also point out that the protection mechanism of SQN can be defeated due to its use of XOR in the 5G AKA protocol. This paper does not consider this security problem, and we will further study this security problem in the future.

Author Contributions: Methodology, Y.X.; formal analysis, Y.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (No.61741216, 61402367), Shaanxi Science and Technology Co-ordination and Innovation Project (No.2016KTTSGY01-03), National Key Research and Development Program (No. 2018YFC08242-04), and New Star Team Project of Xi'an University of Posts and Telecommunications.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Xu, S.; Gan, Z. Review and trends of 5G security technology. *Radio Commun. Technol.* **2020**, *46*, 133–138.
2. 3GPP TS 33.102: 3G Security. Security Architecture. Available online: https://www.3gpp.org/DynaReport/33102.htm (accessed on 26 January 2022).
3. 3GPP TS 33.401: 3GPP System Architecture Evolution (SAE). Security Architecture. Available online: https://www.3gpp.org/DynaReport/33401.htm (accessed on 26 January 2022).
4. 3GPP TS 33.501: 3GPP System Architecture Evolution (SAE). Security Architecture. Available online: https://www.3gpp.org/DynaReport/33501.htm (accessed on 26 January 2022).
5. Ferrag, M.A.; Maglaras, L.; Argyriou, A.; Kosmano, D.; Janicke, H. Security for 4G and 5G cellular networks: A survey of existing authentication and privacy-preserving schemes. *J. Netw. Comput. Appl.* **2018**, *101*, 55–82. [CrossRef]
6. Jover, R.P.; Marojevic, V. Security and protocol exploit analysis of the 5G specifications. *IEEE Access* **2019**, *7*, 24956–24963. [CrossRef]

7. Ahmad, I.; Shahabuddin, S.; Kumar, T.; Okwuibe, J.; Ylianttila, M. Security for 5G and beyond. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3682–3722. [CrossRef]
8. Khan, R.; Kumar, P.; Jayakody, D.N.K.; Liyanage, M. A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions. *IEEE Commun. Surv. Tutor.* **2019**, *22*, 196–248. [CrossRef]
9. Hussain, S.R.; Echeverria, M.; Karim, I.; Chowdhury, O.; Berino, E. 5GReasoner: A property-directed security and privacy analysis framework for 5G cellular network protocol. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; pp. 669–684.
10. Hussain, S.R.; Echeverria, M.; Chowdhury, O.; Li, N.; Bertino, E. Privacy attacks to the 4G and 5G cellular paging protocols using side channel information. In Proceedings of the 26th Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 24–27 February 2019; pp. 1–15.
11. Khan, H.; Martin, K.M. A survey of subscription privacy on the 5G radio interface-the past, present and future. *J. Inf. Secur. Appl.* **2020**, *53*, 102537. [CrossRef]
12. Dehnel-Wild, M.; Cremers, C. *Security Vulnerability in 5G-AKA Draft*; Department of Computer Science, University of Oxford: Oxford, UK, 2018.
13. Meier, S.; Schmidt, B.; Cremers, C.; Basin, D. The Tamarin prover for the symbolic analysis of security protocols. In Proceedings of the 25th International Conference on Computer Aided Verification, Saint Petersburg, Russia, 13–19 July 2013; pp. 696–701.
14. Basin, D.; Dreier, J.; Hirschi, L.; Radomirovic, S.; Sasse, R.; Stettler, V. A formal analysis of 5G authentication. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; pp. 1383–1396.
15. Liu, F.; Peng, J.; Zuo, M. Toward a secure access to 5G network. In Proceedings of the 2018 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/12th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE), New York, NY, USA, 1–3 August 2018; pp. 1121–1128.
16. Borgaonkar, R.; Hirschi, L.; Park, S.; Shaik, A. New privacy threat on 3G, 4G, and upcoming 5G AKA Protocols. *Proc. Priv. Enhancing Technol.* **2019**, *3*, 108–127. [CrossRef]
17. Cremers, C.; Dehnel-Wild, M. Component-based formal analysis of 5G-AKA: Channel assumptions and session confusion. In Proceedings of the 26th Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, 24–27 February 2019; pp. 1–15.
18. Koutsos, A. The 5G-AKA authentication protocol privacy. In Proceedings of the 2019 IEEE European Symposium on Security and Privacy (EuroS&P), Stockholm, Sweden, 17–19 June 2019; pp. 464–479.
19. Bana, G.; Comon-Lundh, H. Towards unconditional soundness: Computationally complete symbolic attacker. In Proceedings of the First international conference on Principles of Security and Trust (ETAPS), Tallinn, Estonia, 24 March–1 April 2012; pp. 189–208.
20. Bana, G.; Comon-Lundh, H. A computationally complete symbolic attacker for equivalence properties. In Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, Scottsdale, AZ, USA, 3–7 November 2014; pp. 609–620.
21. Braeken, A.; Liyanage, M.; Kumar, P.; Murphy, J. Novel 5G authentication protocol to improve the resistance against active attacks and malicious serving networks. *IEEE Access* **2019**, *7*, 64040–64052. [CrossRef]
22. Gharsallah, I.; Smaoui, S.; Zarai, F. A secure efficient and lightweight authentication protocol for 5G cellular networks: SEL-AKA. In Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC), Tangier, Morocco, 24–28 June 2019; pp. 1311–1316.
23. Hu, X.; Liu, C.; Liu, S.; Cheng, X. A security enhanced 5G authentication scheme for insecure channel. *Trans. Inf. Syst.* **2020**, *103*, 711–713. [CrossRef]
24. Hu, X.; Liu, C.; Liu, S.; Li, J.; Cheng, X. A vulnerability in 5G authentication protocols and its Countermeasure. *IEICE Trans. Inf. Syst.* **2020**, *103*, 1806–1809. [CrossRef]
25. Edris, E.K.K.; Aiash, M.; Loo, J.K. Formal verification and analysis of primary authentication based on 5G-AKA protocol. In Proceedings of the 2020 7th International Conference on Software Defined Systems (SDS), Paris, France, 20–23 April 2020; pp. 256–261.
26. Ouaissa, M.; Ouaissa, M. An improved privacy authentication protocol for 5G mobile networks. In Proceedings of the 2020 International Conference on Advances in Computing, Communication & Materials (ICACCM), Dehradun, India, 21–22 August 2020; pp. 136–143.
27. Parne, B.L.; Gupta, S.; Gandhi, K.; Meena, S. PPSE: Privacy preservation and security efficient AKA protocol for 5G communication networks. In Proceedings of the 2020 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS), New Delhi, India, 14–17 December 2020; pp. 1–6.
28. Fábrega, F.J.T.; Herzog, J.C.; Guttman, J.D. Mixed strand spaces. In Proceedings of the 12th IEEE Computer Security Foundations Workshop, Mordano, Italy, 30 June 1999; pp. 72–82.
29. Fábrega, F.J.T.; Herzog, J.C.; Guttman, J.D. Strand space: Proving security protocols correct. *J. Comput. Secur.* **1999**, *7*, 191–230. [CrossRef]
30. Herzog, J.C. The Diffie-Hellman key-agreement scheme in the strand-space model. In Proceedings of the 16th IEEE Computer Security Foundation Workshop, Pacific Grove, CA, USA, 30 June–2 July 2003; pp. 234–247.