Strong Privacy-Preserving Universally Composable AKA Protocol with Seamless Handover Support for Mobile Virtual Network Operator

Rabiah Alnashwan*
Imam Mohammad Ibn Saud Islamic University
ralnashwan@imamu.edu.sa
The University of Sheffield
ralnashwan1@sheffield.ac.uk

Yang Yang*†
National University of Singapore
y.yang@u.nus.edu

Yilu Dong
The Pennsylvania State University
yiludong@psu.edu

Prosanta Gope†
The University of Sheffield
p.gope@sheffield.ac.uk

Behzad Abdolmaleki
The University of Sheffield
behzad.abdolmaleki@sheffield.ac.uk

Syed Rafiul Hussain
The Pennsylvania State University
hussain1@psu.edu

ABSTRACT

Consumers seeking a new mobile plan have many choices in the present mobile landscape. The Mobile Virtual Network Operator (MVNO) has recently gained considerable attention among these options. MVNOs offer various benefits, making them an appealing choice for a majority of consumers. These advantages encompass flexibility, access to cutting-edge technologies, enhanced coverage, superior customer service, and substantial cost savings. Even though MVNO offers several advantages, it also creates critical security and privacy concerns for the customer simultaneously. For instance, in the existing solution, MVNO needs to hand over all the sensitive details, including the users' identities and master secret keys of their customers, to a mobile operator (MNO) to validate the customers while offering any services. This allows MNOs to have unrestricted access to the MVNO subscribers' location and mobile data, including voice calls, SMS, and Internet, which the MNOs frequently sell to third parties (e.g., advertisement companies and surveillance agencies) for more profit. Although critical for mass users, such privacy loss has been historically ignored due to the lack of practical and privacy-preserving solutions for registration and handover procedures in cellular networks. In this paper, we propose a universally composable authentication and handover scheme with strong user privacy support, where each MVNO user can validate a mobile operator (MNO) and vice-versa without compromising user anonymity and unlinkability support. Here, we anticipate that our proposed solution will most likely be deployed by the MVNO(s) to ensure enhanced privacy support to their customer(s).

*Both authors contributed equally to this research.
†Corresponding Author.

ACM CCS, October 14-18, 2024, Salt Lake City, U.S.A.

ACM ISBN 979-8-4007-0636-3/24/10
https://doi.org/10.1145/3658644.3690331

CCS CONCEPTS

• **Security and privacy** → *Network security*; Formal security models; **Pseudonymity, anonymity and untraceability**; *Digital signatures*; **Authentication**.

KEYWORDS

5G, MVNO, Privacy-Preserving Authentication

ACM Reference Format:
Rabiah Alnashwan, Yang Yang, Yilu Dong, Prosanta Gope, Behzad Abdolmaleki, and Syed Rafiul Hussain. 2024. Strong Privacy-Preserving Universally Composable AKA Protocol with Seamless Handover Support for Mobile Virtual Network Operator. In *Proceedings of ACM Conference on Computer and Communications Security (ACM CCS).* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3658644.3690331

1 INTRODUCTION

Mobile Virtual Network Operators (MVNOs) are making significant strides in telecommunications by capitalizing on existing wireless network infrastructure [47, 63]. Through acquiring network capacity from Mobile Network Operators (MNOs) like Vodafone and T-Mobile, MVNOs such as Virgin Mobile and Google Fi have swiftly carved out a niche in the market, employing a business model focused on delivering cost savings to consumers. This strategic framework not only allows MVNOs to present more budget-friendly pricing options but also positions them as compelling alternatives to traditional MNOs. Now, with the advent of 5G, the fifth generation of wireless technology, MVNOs are empowered with faster speeds, lower latency, increased capacity, and the ability to support a wide range of connected devices. This has improved their existing services and opened up opportunities to explore new business models and expand their offerings [45, 56, 60]. This may include enhanced IoT connectivity, immersive multimedia experiences, and other applications that benefit from the capabilities of 5G. However, the inherent vulnerabilities in cellular network technology, such as the lack of authentication of broadcast messages[41, 48, 55, 61], missing confidentiality and integrity protections for pre-authenticated and even for some post-authenticated messages [17, 40, 42, 46, 52, 59, 68] compromise the desired security and privacy requirements of the

MVNO users. Therefore, MNOs and MVNOs must work together to implement appropriate security measures and ensure the privacy and protection of user data.

The operational foundation of an MVNO (e.g., Virgin Mobile in the UK) hinges on the infrastructure of another operator (O2 in the UK) through a well-established partnership. Consequently, a user registered with an MVNO effectively utilizes the network infrastructure provided by MNO and seamlessly accesses MNO's infrastructure within this collaborative arrangement while maintaining registration with MVNO, the only operator they trust. This cooperation between MVNOs and established MNOs underscores the pivotal role of secure and privacy-preserving *Authentication* and *Handovers* in MVNO environments [1] and emphasizes the need to address the following key concerns in this dynamic domain.

P1: Preventing identity exposure: During both the Authentication and Key Agreement (AKA) and the handover procedures, users engage with MNO infrastructure (i.e., base stations and core network), which are considered third parties from the MVNO users' perspective. Consequently, users reveal their identities (e.g., Subscription Permanent Identifier or SUPI and other temporary identifiers) to these third parties, i.e., MNOs, for authentication and handover purposes. Thus, they break entirely the anonymity guarantees. This allows the third-party MNO to access users' footprints and link their activities, making them susceptible to tracking and fingerprinting attacks [13, 32, 39]. Moreover, the current 5G AKA protocol has been shown to be vulnerable to linkability attacks despite attempts to enhance user privacy. Recent research has demonstrated that an attacker can still link 5G AKA sessions and trace users' movements by exploiting protocol weaknesses, particularly in the failure messages [19]. This vulnerability persists even with the introduction of concealed long-term identifiers, highlighting a critical gap in user privacy protection within the existing 5G infrastructure that extends to MVNO scenarios.

Unfortunately, this issue poses a significant privacy concern as MNOs consistently sell and expose the mobile location browsing data and call metadata of millions of users. This behavior by MNOs leaves users with no means to address these issues due to the lack of industry regulation and the dominance of third-party data collectors. Hence, it becomes imperative to minimize the disclosure of user information or, ideally, eliminate any exposure. The current literature lacks a fully anonymous, unlinkable, and footprint-free protocol despite these privacy concerns. Such a protocol is essential to enable users to receive the required services without compromising their privacy.

P2: Privacy-preserving mutual authentication: MVNO users must mutually authenticate the MNOs and their base stations to confirm the authenticity of the network providers and prevent fake base station attacks without exposing their footprints. However, ensuring the authenticity of a third-party provider (MNO) with whom the user is not registered, and vice versa, presents a challenge that has been overlooked in existing literature. Addressing these aspects becomes paramount to establishing a comprehensive security framework in the evolving 5G and MVNO interactions landscape.

P3: Privacy-preserving and secure revocation: An additional challenge arises when considering the need for user revocation in these networks. User revocation, the process of terminating/deactivating a user's access to network services, is important for managing resources and security. Implementing revocation while preserving anonymity is particularly challenging in the complex MNO-MVNO environment, as the current 5G protocols [37] require exposing user identities to MNOs during revocation, thereby compromising the privacy we aim to preserve. This introduces a new dimension to the privacy problem in MNO-MVNO networks that must be addressed alongside authentication and handover privacy concerns.

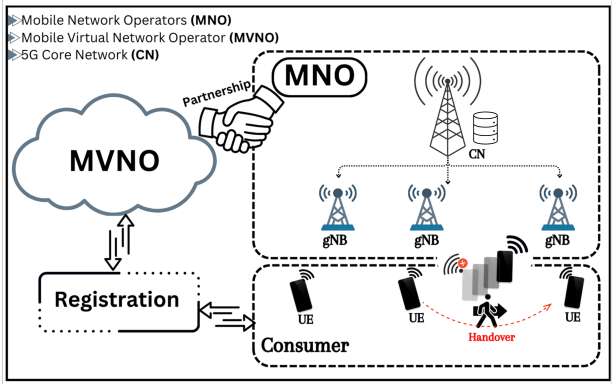


Figure 1: Proposed System Model

P4: Universal Composability: Third, a crucial property, especially when deployed in large systems like 5G and MVNO, is the ability to securely compose protocols to obtain more complex ones, as achieved through the Universal Composability (UC) framework [27]. Relying on the UC framework enables arbitrary and secure protocol composition in a modular manner (e.g., compositions of AKA and handover), ensuring that the security guarantees of individual components extend to the entire system. This capability is vital for 5G systems where multiple cryptographic protocols must harmonize to provide robust security while preserving user privacy. Ongoing research and development efforts in academia and industry are dedicated to enhancing their composability and versatility to meet the ever-evolving demands of the digital age [6, 7, 28, 29, 36]. However, in the context of MVNOs, this security property has not been studied thus far, and the construction of UC-secure MVNOs remains unknown. In the context of MVNOs, which operate in dynamic and complex network environments, universally composable security offers several advantages, including (a) *comprehensive security analysis*, i.e., identifying and mitigating threats by considering various possible interactions with other components and protocols, (b) *interoperability*, i.e, ensuring security guarantees even in the presence of the interactions with MNOs and users, (c) *flexibility and adaptability*, in other words, allowing MVNOs to adapt and evolve their security measures without sacrificing the overall security posture even when the mobile network environment is subject to changes and advancements, and (d) *formal verification* which allows for mathematical proofs of security properties.

In the pursuit of addressing the above concerns and enhancing security and privacy within 5G-enabled MVNO environments, we make the following contributions.

- *We design privacy-preserving AKA and revocation mechanisms* for an MVNO environment that supports a new notion of practical ZKP that aligns with the revised definition of revocation. This

[1] Refer to Figure 1 for a clearer view of the scenario

definition maintains user privacy by utilizing non-identifying yet unique attributes to facilitate revocation, ensuring legal compliance without disclosing user identities. We then present a novel approach (using a list of hash of identity of the user and designing ZKP for that) to design two security protocols that integrate ZKP with Universal Composability (UC) security. To the best of our knowledge, this is the *first* solution to offer a universal solution to address the challenges linked to secure and privacy-preserving schemes within the MVNO environment.
- We develop a *secure privacy-preserving handover protocol* that provides seamless user handover without central entity intervention, reducing overhead on the core network;
- We introduce a *new* notion of user privacy, i.e., *"Comprehensive Privacy"* in the context of an MVNO environment. This concept guarantees the anonymity of the user within the network, ensuring that the user's identity remains protected not only from the core network but also from all base stations controlled by MNO. Besides, our proposed scheme ensures security against linkability attacks and fake base station attacks.
- We perform a rigorous formal analysis of the security and privacy properties of our proposed schemes based on a comprehensive *Universal Composability* framework.
- We implemented and evaluated our proposed schemes in an open-source 5G testbed. Our schemes achieve the desired security and privacy guarantees with only 0.19s overhead compared to conventional AKA. We also compared our approach with existing works aiming to enhance 5G AKA. Our approach can provide stronger security properties while increasing less overhead. We release the source code of our schemes and the existing approaches to which we compare at [12].

Finally, we give our notation used in this paper in Table 1.

2 PRELIMINARIES

This section first provides a brief primer on MVNOs and then presents and defines the security of the cryptographic primitives that are fundamental to the construction of the proposed scheme.

2.1 Mobile Virtual Network Operator

A Mobile Virtual Network Operator (MVNO) is a wireless communication service provider that operates without owning the physical infrastructure of the wireless network it utilizes to deliver services to its customers. Instead of deploying its own network infrastructure, an MVNO leases network capacity from a Mobile Network Operator (MNO). The MNO, which owns the network infrastructure, provides the necessary services, and the MVNO essentially acts as a reseller, offering mobile services under its brand. MVNOs are categorized into three types based on their dependence on host carriers: *skinny*, *light*, and *thick*. Skinny MVNOs heavily rely on base carriers, with limited control over network elements, focusing on branding and marketing. Light MVNOs strike a balance between dependence and autonomy, offering additional services for differentiation. Thick MVNOs have the highest autonomy, controlling essential network elements, allowing flexibility in services, and sometimes investing in their infrastructure. These classifications illustrate the varying levels of control and differentiation within the mobile telecommunications ecosystem. Among the three categories, Thick MVNOs are the least common. Therefore, this study focuses on MVNO types primarily relying on the MNO infrastructure for their services, such as Skinny and Light MVNOs.

Table 1: Notation used in our proposed scheme.

Notation	Description
PK_u, SK_u	user's Public/secret keys for Enc/Dec
spk, ssk	Public/secret keys for signature
PK^g_{san}, SK^g_{san}	Public/secret keys for sanitising signatures
$\mathbb{C}_G$	gNB certificates consists of $(\mathbb{C}_{mod}\&\mathbb{C}_{fix})$
σ_G	gNB certificates' signature
pid	user pseudo identity
Enc/Dec	Encryption and decryption
crs	A common reference string
ck	A commitment key

2.2 5G AKA and Handover

5G-AKA. In the existing 5G-AKA protocol, a base station (gNBs) periodically broadcasts its cell information to inform nearby phones, i.e., User Equipment (UEs) of the gNB's presence. Although these broadcast messages allow the UEs to connect to the gNB, 5G-AKA does not provide any mechanism for UEs to authenticate these initial broadcast messages. This leads to the fake base station attacks [40, 41]. After the initial connection between a UE and a gNB, the 5G-AKA protocol provides mutual authentication between the User Equipment (UE) and the Core Network (CN) through a sequence of control-plane messages [37]. Session keys are also derived to maintain secure communication between UE, gNB, and CN. However, the current mutual authentication scheme requires the UE to send its identity (SUPI) over-the-air (OTA), before the shared session key is derived. In this case, an attacker can sniff the OTA packets and learn the sensitive user data, e.g., SUPI (if unencrypted) and temporary identifiers, which allow a Dolev-Yao attacker [34] to track the user. Although the user identity can be sent in an encrypted form (SUCI), the MNO usually needs to decrypt the user identity to authenticate the user and provide the service. The MNO can also decrypt user plane traffic, enabling them to monitor and analyze user Internet activity. This allows the MNO to gain insights into the types of websites visited, the applications used, and overall user behaviour online. Nonetheless, the inherent flaws in the 5G AKA protocol, such as lack of observational equivalence in certain sub-protocols, lead to the linkability/traceability ttacks [13, 40].

Handover. Handover is essential in cellular communication to provide seamless connections [9]. Since the base stations are stationary, mobile devices need to switch the connected base stations frequently. Here, we discuss *three distinguished types of handovers*: intra-cell handover, inter-cell handover, and inter-MNO handover.

Intra-cell handover. In intra-cell handover, the UE needs to disconnect the current radio connection and establish a new connection (usually in a different frequency) within the same cell. During this type of handover, an attacker can use a fake base station to impersonate the cell-to-be-connected and try to get the user's identity.

Inter-cell handover. In inter-cell handover, the UE seamlessly moves from one base station (source) to another (target). It needs to re-establish the radio connection with the target base station.

In this case, UE is also vulnerable to fake base station attacks. In addition, in inter-cell handover, the source base station sends the UE context (e.g., UE's and base station's identities, UE security context, and PDU session information) to the target gNB either through direct communication, i.e., using Xn interface between two gNBs or through the MNO's CN, i.e, with N2 interface between gNBs and CN. As a result, the infrastructure owners (MNOs), is able to track user movements from the handover requests.

Inter-MNO handover. Inter-MNO handover is specific to the MVNO settings. An MVNO user can use the base station infrastructure from several MNOs to provide better radio coverage. To provide seamless handover, an interface between AMFs of different MNOs must be implemented to transfer the UE context. The MNOs have information about when the user enters and leaves its network and also which MNO it moves to. However, the MNOs usually are not motivated to implement this interface. Hence, UE must de-register from the current MNO and register with the new MNO, which causes an interruption of the current service.

2.3 Sanitizable signatures

Sanitizable Signature is a signature scheme where signing capabilities can be delegated to another party: the sanitiser. The accuracy of the sanitiser capabilities can be managed through a couple of deterministic functions *ADM* and *MOD*. The former indicates specific parts of which a sanitiser can modify a message. The latter function specifies what modification the sanitiser has made, where $m^* = MOD(m)$ and $ADM(m^*, m) \rightarrow \{0, 1\}$. A SanSig is a tuple of algorithms: SanSig = {Kgen, Sign, Sanit, Verify, Proof, Judge}

- $\mathsf{Kgen} : (pk_{sig}, sk_{sig}) \leftarrow_{\$} \mathsf{Kgen}(1^{\lambda})$, $(pk_{san}, sk_{san}) \leftarrow_{\$} \mathsf{Kgen}_{san}(1^{\lambda})$
- $\mathsf{Sign} : \sigma \leftarrow_{\$} \mathsf{Sign}(m, sk_{sig}, pk_{san}, ADM)$
- $\mathsf{Sanit} : (m^*, \sigma^*) \leftarrow_{\$} \mathsf{Sanit}(m, MOD, \sigma, pk_{sig}, sk_{san})$
- $\mathsf{Verify} : b \leftarrow \mathsf{Verify}(m, \sigma, pk_{sig}, pk_{san})$

Sanitizable Signature Unforgeability: We say that SanSig supports Existential Unforgeability under Chosen Message Attack (EUFCMA-Secure) if the advantage of $Adv_{SanSig}(\mathcal{A})$ is negligible, where:

$$Adv^{EUC-CMA}_{SanSig}(\mathcal{A}) = Pr[Exp^{EUC-CMA}_{SanSig}(\mathcal{A}) = 1]$$

2.4 Universal Composability

The Universal Composability (UC) framework was introduced by Canetti in [27]. In the UC framework, one analyzes the protocol's security under real-world and ideal-world paradigms. More precisely, in this setting, the real-world execution of a protocol is compared with an ideal-world interaction with the primitive it implements. Then, a composition theorem in this model states that the security of the UC-secure protocols remains if it is arbitrarily composed with other UC-secure protocols or the protocol itself. Additionally, the UC-secure property guarantees security in practical applications where individual instances of protocols are run in parallel, such as the Internet. The entities in the UC framework in both ideal-word and real-word executions are modelled as PPT (probabilistic polynomial time) interactive Turing machines that send and receive messages through their output and input tapes, respectively. In the ideal world execution, dummy parties (possibly controlled by an ideal-word adversary Sim, also called simulator) communicate directly with the ideal functionality $\mathcal{F}$. The ideal functionality can be viewed as a trusted party that creates the primitives to implement the protocol. Correspondingly, in the real-world execution, parties (possibly corrupted by a real-world adversary $\mathcal{A}$) communicate with each other as a protocol Π that realizes the ideal functionality. Both the ideal and real executions are controlled by the environment $\mathcal{Z}$, an entity that sends inputs and receives the outputs of $\mathcal{A}$, the individual parties, and Sim. Finally, after seeing the ideal or real protocol execution, $\mathcal{Z}$ returns a bit, which is considered the execution output. Then, the rationale behind this framework lies in showing that the environment $\mathcal{Z}$ can not efficiently distinguish between the ideal and real executions, therefore meaning that the real-world protocol is as secure as the ideal-world (the ideal functionality).

Besides the two aforementioned models (real-world and ideal-world) of computation, the UC framework considers the hybrid world, where the executions are similar to the real world but with the additional assumption that the parties are allowed to access an auxiliary ideal functionality $\mathcal{G}$. More precisely, in this case, instead of honest parties interacting directly with the ideal functionality, the adversary passes all the messages from and to the ideal functionality. Also, the transmission channels are considered to be ideally authenticated, meaning that the adversary is not able to modify the messages but is only able to read them. Unlike information transferred between parties, which can be read by the adversary, the information transferred between parties and the ideal functionality is split into a public and private header. The private header carries some information like as the private inputs of parties and it cannot be read by the adversary. The public header carries only some information that can be viewed publicly, such as receiver, sender, type of message, and session identifiers. Let denote the output of the environment $\mathcal{Z}$ that shows the execution of a protocol Π in a real-world model and a hybrid model, respectively, as $\textsc{Ideal}^{\mathcal{F}}_{\mathsf{Sim}}$ and $\textsc{Hybrid}^{\mathcal{G}}_{\Pi,\mathcal{A}}$. Then the UC security is formally defined as:

Definition 1. *A n-party ($n \in \mathbb{N}$) protocol Π UC-realizes an ideal functionality $\mathcal{F}$ in the hybrid model if, for every PPT adversary $\mathcal{A}$, there exists a simulator* Sim *such that for all environments $\mathcal{Z}$,*

$$\textit{Ideal}^{\mathcal{F}}_{\mathsf{Sim}} \approx_{\lambda} \textit{Hybrid}^{\mathcal{G}}_{\Pi,\mathcal{A}}.$$

The protocol Π is statistically secure if the above definition holds for all unbounded $\mathcal{Z}$. In thesis-sec:UC, we define the ideal functionality for a commitment scheme and provide its corresponding hybrid functionality to prove the UC security of the scheme.

2.5 Commitment Scheme

Definition 2. *A commitment scheme* $\Pi = (\mathsf{Kgen}, \mathsf{Com}, \mathsf{Decom})$, *is defined by the following three algorithms:*

- $\mathsf{ck} \leftarrow \mathsf{Kgen}(\lambda)$*: given a security parameter λ, generates a public parameter* ck *of the scheme that implicitly passed as input to the other algorithms.*

- $(c, \delta) \leftarrow \mathsf{Com}_{\mathsf{ck}}(m, r)$*: given the public parameter* ck*, a message* m *from message space* M*, and a randomness* r *from randomness space* R*, outputs a commitment* c *together with an opening information* δ[2].
- $m/\perp \leftarrow \mathsf{Decom}(\mathsf{ck}, c, m, \delta)$*: given given the public parameter* ck*, a commitment* c*, the message* m *and an opening information* δ*, outputs* m *or* $\perp$ *if the opening verification fails.*

Such a scheme must satisfy both *hiding* property (meaning that the commit phase does not disclose any information about the committed message m), and *binding* property (meaning that the decommit phase (opening phase) can successfully open to only one value). The aforementioned properties may be achieved in a perfect, statistical or computational according to the power of the adversary against those properties. Besides, some additional strong properties are demanded in some systems, like the UC-secure commitment scheme. The first is *extractability*, which states that given a trapdoor, one (i.e., the simulator Sim in the UC model) can recover the committed value m. The second one is *equivocability*, which means that given a trapdoor, one (i.e., the simulator Sim in the UC model) can open a commitment to any message $m' \neq m$.

2.6 Non-Interactive Zero-Knowledge

Zero-knowledge proofs and in particular, non-interactive zero-knowledge proofs (NIZKs), is a protocol between a prover and a verifier that allows the prover to convince the verifier of the validity of a statement without disclosing any more additional information. Let REL be a relation generator, such that $\mathsf{REL}(1^\lambda)$ outputs a polynomial time decidable binary relation $\mathcal{R} = \{(x, w)\}$. Here, x and w are, respectively, the statement and the witness. Let $\mathcal{L}_\mathcal{R} = \{x : \exists w, (x, w) \in \mathcal{R}\}$ be an NP-language. NIZK proofs in the CRS model consist of the four algorithms $(\mathsf{K}_{\mathsf{crs}}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ where $\mathsf{K}_{\mathsf{crs}}$, P, V, and Sim are common reference strings (CRS) generator, prover, verifier, and the simulator, respectively.

Definition 3. *A NIZK system* Ψ *for any relation generator* REE *consists of four PPT algorithms:*

- $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{crs}}(\lambda)$*: A* probabilistic *algorithm that, given the security parameter* λ *outputs a* CRS trapdoor td *and a CRS* crs*. Otherwise, it outputs* $\perp$*.*
- $\pi \leftarrow \mathsf{P}(\mathsf{crs}, x, w)$*: A probabilistic algorithm that, given* (crs, x, w)*, outputs an argument* π *if* $(x, w) \in \mathcal{R}$*. Otherwise, it outputs* $\perp$*.*
- $0/1 \leftarrow \mathsf{V}(\mathsf{crs}, x, \pi)$*: a probabilistic algorithm that, given* (crs, x, π)*, returns either* 0 *(reject) or* 1 *(accept).*
- $\pi \leftarrow \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, x)$*:] a probabilistic algorithm that, given* $(\mathsf{crs}, \mathsf{td}, x)$ *outputs an argument* π*. Otherwise, it outputs* $\perp$*.*

A NIZK must provide the following properties:

(1) **Completeness.** For any λ, and $(x, w) \in \mathcal{R}$,

$$\Pr\left[(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{crs}}(\lambda) : \mathsf{V}(\mathsf{crs}, x, \mathsf{P}(\mathsf{crs}, x, w)) = 1 \quad \right] = 1 \ .$$

(2) **Statistical Zero-Knowledge.** For any computationally unbounded adversary $\mathcal{A}$, $|\varepsilon_0^{zk} - \varepsilon_1^{zk}| \approx_\lambda 0$, where $\varepsilon_b^{zk} :=$

$$\Pr\left[(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{crs}}(\lambda), b \leftarrow \{0,1\} : \mathcal{A}^{\mathsf{O}_b(\cdot,\cdot)}(\mathsf{crs}) = 1 \quad \right].$$

The oracle $\mathsf{O}_0(x, w)$ returns $\perp$ (reject) if $(x, w) \notin \mathcal{R}$, and otherwise it returns $\mathsf{P}(\mathsf{crs}, x, w)$. Similarly, $\mathsf{O}_1(x, w)$ returns $\perp$ (reject) if $(x, w) \notin \mathcal{R}$, and otherwise it returns $\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, x)$.

(3) **Computational Soundness.** For any non-uniform PPT $\mathcal{A}$,

$$\Pr\left[\begin{matrix}(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}_{\mathsf{crs}}(\lambda); (x, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) : \\ \mathsf{V}(\mathsf{crs}, x, \pi) = 1 \wedge \neg(\exists w : (x, w) \in \mathcal{R})\end{matrix}\right] \approx_\lambda 0 \ .$$

[2]**The opening information will be used in the decommit phase to prove that the commitment** $\vec{c}$ **contains a valid message** m**.**

3 SYSTEM AND THREAT MODEL

This section presents the system model and the threat model we consider in this work.

3.1 Proposed System model

The system model is outlined into three main layers: the *MVNO*, *MNO,* and *Consumer* layers, as illustrated in Figure 1. The MVNO is critical in generating essential security parameters, encompassing keys and IDs, for its users, namely UE. The MVNO typically does not own any mobile network infrastructure and rents the services leveraging MNO infrastructure such as Virgin, Lebara, and Tesco. The second layer of the system model comprises two essential participants— the Core Network (CN) and the base station (gNB). Both entities in this layer share mutual trust, usually owned by a traditional mobile network operator (MNO), meaning they own the infrastructure they use, such as O2 and Vodafone. However, it is important to note that despite being owned by the same company, no secure communication channel is assumed between entities within this layer. The final layer in the proposed system model is the consumer layer, consisting of MVNO's users or user equipment; these terms are used interchangeably to denote the same entity.

3.2 Threat Model

Our work focuses on scenarios where the network entities, such as CN and gNBs, are controlled by an MNO, and all connected gNBs belong to this same MNO. We specify a threat model that includes a typical protocol-level Dolev-Yao adversary, an adversary against the privacy of the UE, and an adversary against the UC model. For this, we separate each adversary into one of *three types* and define our threat model as follows:

• $[\mathcal{A}_1]$ The Type 1 (Dolev-Yao) adversary consists of the typical Dolev-Yao model [34], which is capable of eavesdropping on the network between the UE and the entities operated by MNO (e.g., gNB, CN). This type of adversary can also use fake base station [41] or machine-in-the-middle (MitM) relay [55] to add, drop or modify messages between UE and MNO by adhering to the cryptographic assumptions, i.e., it can decrypt an encrypted message only if she obtains the decryption key.

• $[\mathcal{A}_2]$ The Type 2 (Privacy) adversary is intent on compromising user privacy by attempting to breach the user's anonymity and establish links between the user's activities. This is broadly known as *unlinkability* or *observational equivalence.* In the MVNO context, MNO's gNB and CN Type 2 adversaries as they are interested in users' footprint. Although these adversaries may not be able to compromise the network (MNO) entities. Still, it maliciously intends to collect and sell users' sensitive information to third parties. In

addition, a Dolev-Yao adversary, i.e., $\mathcal{A}_1$ may also attempt to track users' footprint by violating the observational equivalence property.
• [$\mathcal{A}_3$] Finally, we consider a Type 3 (UC-based) adversary, which is more powerful and can run both the $\mathcal{A}_1$ and $\mathcal{A}_2$ internally. Unlike a Dolev-Yao adversary ($\mathcal{A}_1$), which cannot reason about the security and privacy of multiple and unbounded parallel executions of the protocol, the $\mathcal{A}_3$ can see many executions of the protocols when they are composed of other systems. Here, by having many executions in parallel, the adversary can verify observational equivalence properties that may find linkability/traceability attacks [13, 40]. The $\mathcal{A}_3$ adversary can also infer and control/manipulate the internal protocol states of the victim. For instance, MNO's CN can infer the internal protocol states of the UE and can control those states by initiating the AKA or other common procedures at any time. The adversary can operate in a setting with multiple concurrent protocol instances, interacting with all instances simultaneously.

Unlike $\mathcal{A}_1$ adversary, which is limited to symbolic operations and does not break cryptographic primitives, the UC adversary is more powerful, and can have full control over the network and adaptively and momentarily compromise parties at any time, obtaining their entire internal states. For instance, an external adversary may compromise a gNB by exploiting misconfigurations [31], vulnerabilities in implementation [31], dependency weaknesses [64], and potential compromises by malicious user devices [1].

To summarize, the attacker in the universally composable security model retains the Dolev-Yao capabilities and can perform the following actions in the context of MVNO: *(A) Initiating Protocol Sessions*: The adversary can initiate multiple protocol instances and interact with honest parties. *(B) Choosing Instances of Sub-protocols*: The adversary can select instances of sub-protocols used within the larger protocol, potentially choosing those that benefit its malicious goals. *(C) Internal State Manipulation*: The adversary may attempt to manipulate the internal state of the protocol participants, influencing their behaviour. *(D) Choosing Cryptographic Keys*: The adversary might have control over the generation or selection of cryptographic keys used in the protocol. In essence, the universally composable security model captures the full range of potential attacks and adversaries that the 5G-AKA and Handover protocols may face in the context of MVNO. This allows a more comprehensive and realistic framework for analyzing cryptographic protocols compared to the more abstract and idealized Dolev-Yao model. The detailed comparison between the UC and the Dolev-Yao model has been included in **Appendix B**.

4 SECURITY AND PRIVACY REQUIREMENTS FOR MVNO

In the context of MVNOs, AKA and HO require specific security and privacy requirements due to the unique challenges and privacy concerns inherent in the MVNO setting. The proposed scheme aims to meet the following security and privacy requirements:

• **Mutual Authentication Under MVNO settings (MA+)**. In general, mutual authentication (MA) ensures the legitimacy of both the network components and the user equipment (UE), which is crucial. This property protects against attacks such as man-in-the-middle (MitM) and impersonation attacks by verifying that all communication parties are genuine and authorised. Now, MA+ mandates that network entities, such as gNB and CN, authenticate the UE and vice versa before rendering any services. The key distinction between the two lies in the user's registration with a different network provider (MVNO), while the network entities providing services are provided by the MNO. Consequently, network entities must ensure users' authenticity without accessing their information, necessitating a method to ensure the authenticity and genuineness of other participants (UE) with whom they communicate. Our scheme achieves MA+ security to provide a privacy-preserving secure communication environment for all network participants. We provide details of this security experiment in Section 6. We define the authentication goal as follows:

Establishing privacy-preserving mutual authentication between UE *and* gNB *under MA+:* UE can perform the initial authentication of the broadcast message from the gNB. In this way, we can deal with the fake base station attacks. Besides, at the end of the execution of the proposed scheme, both the UE (registered with MVNO) and gNB (operated by MNO) are expected to establish mutual trust between themselves.

Establishing privacy-preserving mutual authentication between UE *and* CN *under MA+:* Even though the UE registers into MVNO, not in CN. However, at the end of the execution of the proposed scheme, both the UE and CN are expected to establish mutual trust between themselves. In this regard, UE does not need to reveal his/her identity. In this context, a weak agreement means that a participant in the protocol has undergone the protocol with its counterpart. Yet, there is no obligation for them to reach a consensus on any data exchanged or secrets established during the session.

• **Comprehnsive Privacy (CP)**. A privacy-preserving approach needs to go beyond mere anonymity, extending to a model where even the core network remains unaware of the user's identity while retaining the ability to validate their legitimacy. Specifically, this holds true even for roaming users undergoing the handover protocol. We define the comprehensive privacy goals as follows:

User Anonymity (UA): Given a UE considers an interaction with AKA/HO session involving CN or gNB or both, no active attacker ($\mathcal{A}_2$) can recover a user's identity ("SUPI must remain secret").

Unlinkabilty (UL): Given two UEs identified as UE_1 and UE_2, and considering an AKA/HO session involving either UE_1 or UE_2, it is impossible for attackers to discern which specific UE (UE_1 or UE_2) it is interacting with.

• **Privacy-preserving Revocation (PR)**. It is imperative to guarantee that only authorized users access network resources. However, achieving a balance between user revocation and privacy is widely recognized as a challenge in existing literature [9]. In the context of MVNO, it is critical to ensure user anonymity and confidentiality while empowering the network to revoke specific users when necessary.

• **Universal Composability (UC)**. Universally composable security equips MVNOs to counteract security threats effectively in dynamic and interconnected mobile network environments. The proposed scheme is expected to provide a strong security guarantee by ensuring that the proposed scheme remains secure even when composed of arbitrary other protocols in a more extensive system.

5 OUR PROPOSED SCHEME

In this section, we introduce our proposed scheme, a structured scheme that unfolds in three phases: Registration, Initial Authentication, and Handover. During the registration phase, there are two sub-categories, MNO registration and MVNO registration, based on the entity responsible for the registration process. When a User Equipment (UE) seeks to join the network, the MVNO facilitates the secure transmission of essential information, including IDs and keys, to the registering participant through a secure channel. However, in the case of gNB and CN, it is the responsibility of the MNO to provide them with the keys and certificates. In the Initial Authentication phase, all users can be authenticated anonymously to the network using Zero-Knowledge Proofs (ZKP), as illustrated in Figure 2. Simultaneously, the authentication of all gNBs to users is achieved through SanSig certificates generated by the MNO during the registration phase. The continuity of secure services for roaming users is ensured through a user re-authentication through the execution of the Handover phase, as depicted in Figure 4. This iterative handover protocol allows roaming users to seamlessly and securely receive services from the network.

5.1 System Registration

In the initial phase of the proposed scheme, crucial parameters are defined, involving entity registration and key generation. User registration is entrusted to the MVNO, while the responsibility for gNB and CN registration rests with the MNO. In a collaborative effort facilitated by the partnership between the MVNO and MNO, essential information is exchanged between the two entities. As a result, this phase can be subdivided into three main components based on the assigned responsible party and the sharing process between them. This segmentation establishes a clear delineation of roles within the collaborative framework, enhancing the efficiency and clarity of the proposed system.

(1) *MNO* Setup: The MNO takes on the responsibility of generating key pairs for both the CN and gNBs, ensuring their secure distribution. When the CN sends a setup request to the MNO, the latter generates all the necessary keys for the CN. This encompasses pairs of public/secret keys for digital signature ($ssk_{\text{CN}}, spk_{\text{CN}}$) and SanSig ($pk^{\text{CN}}_{sig}, sk^{\text{CN}}_{sig}$) algorithms. The latter is responsible for signing certificates for gNBs within the network.

Similarly, when a gNB in the network seeks registration with the MNO, it initiates the process by sending a registration request to the MNO, enclosing pertinent information. Upon receipt of the registration request, the MNO undertakes the authentication of the gNB, and subsequently generates an identity for this base station (id_{gNB}), along with the associated pairs of keys ($ssk_{\text{gNB}}, spk_{\text{gNB}}$), ($SK^{\text{gNB}}_{san}, PK^{\text{gNB}}_{san}$), and a certificate ($\mathbb{C}_G, \sigma_G$) using the SanSig algorithm ($\sigma_G \leftarrow_{\$} \text{SanSig.Sign}(\mathbb{C}_G, sk^{CN}_{sig}, pk^{\text{gNB}}_{san}, ADM(\mathbb{C}^G_{mod}))$). This certificate encompasses id_{gNB}, the location information of gNB, and a certificate expiration period (EXP), denoting the certificate's validity duration. This comprehensive process ensures secure and authenticated registration of gNB within the network.

(2) *MVNO* Setup: At the beginning, the MVNO creates a common reference string (crs), its trapdoor td and commitment key (ck) for the Zero-Knowledge Proof (ZKP) algorithm. This proactive step ensures that all registered users have access to essential information. After that, the MVNO generates all key pairs and identities for users and securely distributes these credentials. To use the network services provided by the MVNO, each user must first register on the network. The registration process begins when a User Equipment (UE) selects the MVNO plan and provides necessary information to the MVNO. Upon receipt of the registration request, the MVNO creates a pseudo-identity (pid) for the user, which aligns with the specified ZKP range. Next, the MVNO hashes the user identity (H_{pid}) and stores it in a designated list (List). Finally, the MVNO transmits the pid, H_{pid}, along with the common reference string (crs), and commitment key (ck) to the registered user, thereby completing this secure registration process.

(3) *MVNO − MNO* information exchange: Due to the collaboration between the MVNO and MNO, crucial information is exchanged to ensure the successful and seamless execution of our proposed scheme. On the MVNO side, this involves sharing the pre-generated crs and the List (i.e. a list of hashed identities of the registered users), which may undergo updates throughout their partnership. Simultaneously, the MNO contributes by sharing essential information with the MVNO, including the public keys of CN and gNBs. This exchange ensures that users can verify the authenticity of the network, establishing a robust foundation for secure interactions within the system.

5.2 Initial authentication (MVNO-AKA)

Each registered user who wants to join the network must execute the initial authentication phase, as shown in Figure 2. During the execution of this protocol, the CN generates credentials for new users, which will be used in the subsequent Handover protocols:

Step 1: gNB → UE. $\mathbf{M_1}$:$[\mathbb{C}^*_G, \sigma^*_G]$:- In the first step, gNB utilizes the SanSig.Sanit(.) algorithm to sanitize their certificate to include their identity (id_{gNB}) and a timestamp to ensure message integrity and to prevent the known MITM and DOS attacks. The updated/sanitized certificate with its signature (σ) is sent to the UE via M_1.

Step 2: UE → gNB. $\mathbf{M_2}$:$[\pi_{ZK}, c, PK_u, \tau_2, \sigma]$:- Upon receiving M_1, UE first checks the timestamp and id_{gNB}, and verifies the signature using (SanSig.Verify()). If all verifications hold, then UE computes a pair of asymmetric keys (PK_u, SK_u) for encryption. Next, the UE samples randomly (r), computes a commitment of their H_{pid} and a zero-knowledge proof (π_{ZK}) for the $\mathcal{L}_{\text{zk}} = \{(c, \text{List}) \mid \exists \text{w} := (pid, r) \text{ s.t } c = \text{Com}_{\text{ck}}(pid, r) \wedge H_{pid} \in \text{List}\}$ where List is list of the all hashes of the user identity, $\text{List} = \{H_{pid_1}, \cdots, H_{pid_n}\}$. Finally, the user signs all the previous computations and sends them to gNB.

Step 3:- gNB → CN. $\mathbf{M_3}$: $[\pi_{ZK}, c, PK_u, \tau_3, \sigma^*]$: After receiving the message M_2, gNB first checks the timestamp and verifies the signature σ to check the message integrity. If both verifications hold, gNB signs the content of M_2 using his singing key (ssk_g). Then gNB forwards the generated signature (σ^*) along with M_2 to the CN.

Step 4: CN → UE. $\mathbf{M_4}$: $[\text{Enc}_{PK_u}\{\sigma_U \| \text{UID} \| \tau_4\}]$:- Upon receiving M_3, the CN verifies both (σ^*) and (π_{ZK}) using signature and ZKP verification algorithms, respectively. If the verification holds, the CN computes a universal user ID (UID_i), which will be the

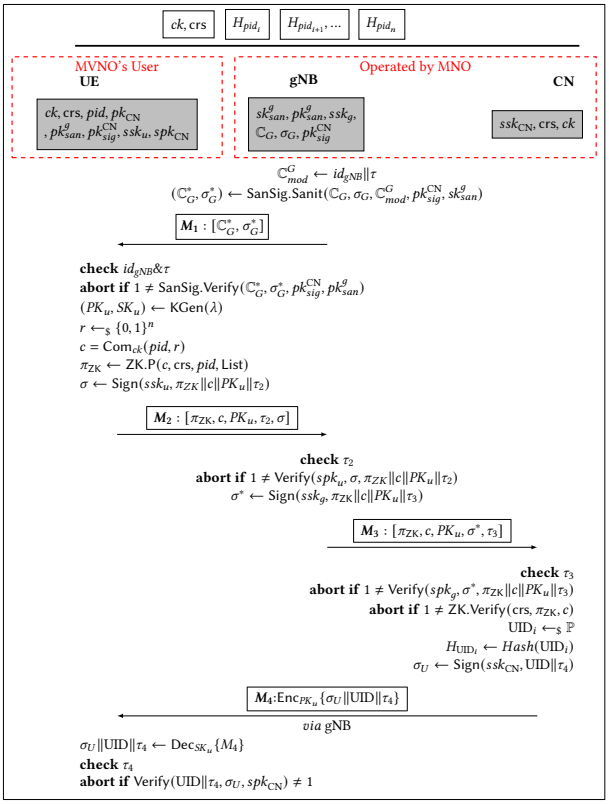


Figure 2: UC-based Initial Authentication protocol for MVNO

user's identifier during HOs. Then CN hash and signs (UID_i) along with a timestamp (τ) using Sign(.) algorithm. CN then stores UID_i and encrypts user certificate UID_i, σ_U and timestamp using (PK_u). Finally, the *CN* sends M_4 to the UE via gNB. Then, the user is responsible for verifying M_4 and the signature.

Remark 1. *If a symmetric session key is needed between the UE and gNB for future communication, the CN can distribute one to both entities within message M_4. This message will encapsulate the encryption of the identical session key twice: once encrypted with the gNB's public key and the other with the UE's public key.*

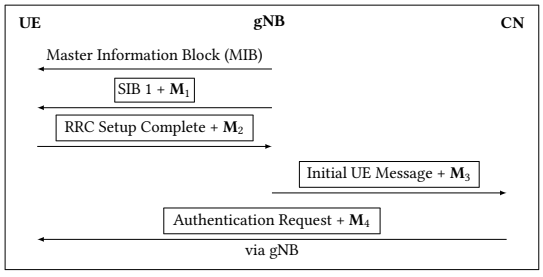


Figure 3: Integration of our Initial Authentication Protocol with 5G-AKA

5.2.1 Integration of the Proposed Initial Authentication Protocol with 5G AKA. This section shows how our proposed initial authentication scheme can be integrated with the conventional 5G-AKA. As shown in Figure 3, our M_1 can be integrated into SystemInformationBlockType1 (SIB1) in the current 5G protocol. By authenticating broadcast messages, the UE can verify the authenticity of the gNB before connecting to it. Since an attacker cannot forge the signature, this method helps prevent fake base station attacks. Then, M_2 is integrated into the RRC Setup Complete message. We extend the Initial UE Message, an NGAP (Next Generation Application Protocol) message between gNB and CN, to transport M_3. The response from CN, M_4, is sent to UE as a response to M_3 inside of the authentication request message. Since our proposed scheme only extends the existing 5G control-plane messages, it is fully compatible with the current 5G authentication scheme. If the UE or the network does not support the new scheme, we can downgrade to the existing 5G-AKA. In our proposed scheme, the UE is not required to send its identity in plaintext over-the-air, and an external attacker cannot forge plaintext 5G control-plane messages to acquire sensitive UE information or perform DoS attacks. Note that this integration does not introduce any new 5G protocol messages and reuses only the extension of existing messages. Hence, our proposed solution will not affect the legacy UEs incapable of supporting our solutions.

5.3 UC Secure MVNO Handover Scheme (MVNO-HO)

Users who complete initial authentication and want to roam between small cells must execute this protocol. As part of this protocol, users will utilize the Zero-Knowledge Proof (ZKP) and the unique identities (UID) generated from the previous protocol to verify their authenticity to the gNBs. Subsequently, the gNBs will cross-check these identities with the list of identities created by the *CN* during the previous protocol. The Handover protocol is described below and illustrated in Figure 4.

Step 1: gNB $\rightarrow$ UE. $\mathbf{M_1}$:$[\mathbb{C}^*_G, \sigma^*_G]$:- This step is identical to the first step of the initial authentication protocol.
Step 2: UE $\rightarrow$ gNB. $\boldsymbol{M_2}$: $[\pi_{ZK}, c, PK_u, \sigma, \tau_u]$:- Upon receiving M_1, UE first checks the timestamp and id_{gNB}, and verifies the signature using (SanSig.Verify()). Assuming both verifications are successful, then UE computes a pair of asymmetric keys (PK_u, SK_u) for encryption. Next, the UE samples randomly (r), computes a commitment of their H_{UID} and a zero-knowledge proof (π_{ZK}) for the $\mathcal{L}_{\text{zk}} = \{(c, \text{List}) \mid \exists \text{w} := (\text{UID}, r)$ s.t $c = \text{Com}_{\text{ck}}(\text{UID}, r) \wedge H_{\text{UID}} \in \text{List}\}$ where List $= \{H_{\text{UID}_1}, \cdots, H_{\text{UID}_n}\}$. Finally, the user signs all the previous computations and sends them to gNB.
Step 3: gNB $\rightarrow$ *UE* $\mathbf{M_3}$:Enc$_{PK_u}\{ACK, \sigma\}$:- Upon receipt of M_2, gNB checks the timestamp and verifies the signature and identity of UE using signature and ZKP verification algorithms, respectively. If both verifications hold, gNB encrypts *ACK* and σ using the user's public key and sends it via M_3 to UE. Finally, after receiving M_3, the user decrypts the message and checks the integrity of σ. Details of this protocol are depicted in Figure 4.

5.3.1 Integration of the Proposed Scheme with 5G-Handover. As shown in Figure 5, our proposed UC secure MVNO handover protocol can also be integrated with 5G Handover between gNBs. Similarly to Section 5.2.1, M_1 is integrated in the SIB1 message. Thus, the UE can verify the identity of gNB before making the handover

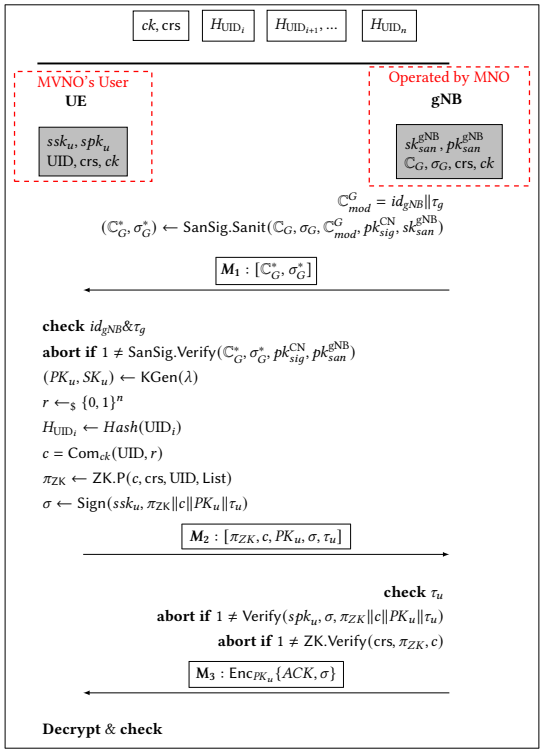


Figure 4: Proposed UC Secure MVNO Handover protocol

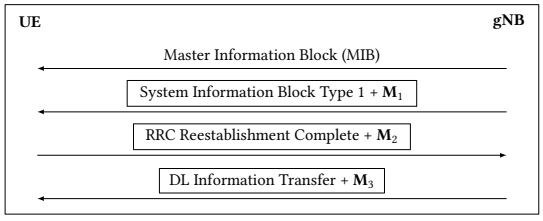


Figure 5: Integration of our MVNO Handover Protocol with 5G Handover Scheme

decision. M_2 is added to the RRC Reestablishment Complete message. Since we don't need communication with CN in this scenario, the gNB directly replies M_2 with M_3 in an extended DL Information Transfer message. After receiving M_3 from gNB, the UE and gNB are mutually authenticated. Our protocol can prevent fake base station attacks on the UE and prevent fake UE attacks from trying to exhaust the resources of the gNB. This design is consistent with our proposed scheme for 5G-AKA, so it is also compatible with the current 5G specifications.

5.4 Privacy-preserving Revocation

The complex structure of MNO-MVNO networks, the critical need for robust data protection, and escalating security threats necessitate a new approach to user revocation. Our solution addresses this challenge by providing comprehensive privacy during revocation, tackling the unique complexities of multi-party MVNO environments. This approach not only enhances user trust and safeguards personal information but also strengthens overall security without compromising anonymity. It enables the removal of compromised, unauthorized, or non-paying users during authentication and handover processes, significantly improving network efficiency by preventing unnecessary resource consumption. However, implementing such a mechanism while ensuring complete user privacy presents a significant challenge. Our scheme overcomes this challenge through an innovative approach implemented during the registration phase. By utilizing the shared List between MNO and MVNO. The procedure involves the MVNO identifying the unique pseudo-ID (*pid*) associated with the revoked user's identity, hashing it to generate H_{pid}, and then transmitting this H_{pid} to the core network (CN) for removal from the public list of authorized UEs (List). Subsequently, the CN retrieves the associated Hashed universal ID (H_{UID}) with H_{pid} to remove it from the handover public list.

5.5 Instantiating the Primitives

In this section, we list all primitives used in each part of the proposed construction and explain how to instantiate each primitive using different hardness assumptions.

Non-interactive commitments are constructed using the following primitives:

- The Pedersen commitment [53] that perfectly hides and computationally binds based on the discrete logarithm (DL) assumption. The UC-secure commitment [4, 5] from DL Assumption.
- The string commitment scheme by Kawachi *et al.*'s [44] based on the SIS assumption [8].
- The commitment scheme by [43] where the hiding property is based on the Learning Parity with Noise (LPN) assumption, a special case of the Learning With Errors (LWE) assumption [54].
- The commitment scheme [67] that is based on Ring-LWE [49] instead of LPN, and they build Σ-protocols from it. Further Σ-protocols based on (Ring-)LWE encryption schemes were presented by Benhamouda *et al.* [16].
- The commitment scheme by Baum *et al.* [14] that relies on the Module-LWE and Module-SIS assumption.

Practical non-interactive zero-knowledge protocols are constructed using the following primitives:

- Depending on the computation cost of the prover, verifier, and the communication complexity, there are several practical transparent zero-knowledge argument schemes based on the discrete logarithm assumption. Spartan [58] with compact proof proof size (tens of KBs) and based on bilinear pairing. The Bulletproofs [25] and Supersonic [26] are based on discrete-log and group of unknown order with the smallest proof size (1-2 KBs).
- The are several practical post-quantum transparent zero-knowledge argument schemes such as Orien [66], Brakedown [38], Aurora [15], and Ligero [10]. Orion has the fastest prover among all schemes. This is slightly faster than Brakedown and is 20 times faster than Ligero and 142 times faster than Aurora because of the linear prover time. The proof size of Orion is significantly smaller than Brakedown and Ligero. But Aurora has the most compact proof size.

Practical sanitizable signatures are constructed using the following primitives: Sanitizable signatures offer diverse construction methods, such as employing chameleon hashes and a standard digital signature scheme, as demonstrated in [11]. Another approach involves using two types of signatures: the conventional digital signature and a group signature, as outlined in [22]. Additionally,

the creation of a sanitizable signature can involve utilizing two conventional digital signature schemes, such as RSA-based signature and ECDSA, as depicted in [21, 23, 24]. It's important to note that the performance and security of the SanSig may vary based on the underlying construction and primitives used.

6 SECURITY ANALYSIS

In this section, we provide the security proof of our construction. We first define $\mathcal{F}_{mvno}$ as the ideal functionality of our proposed scheme below. Note that besides the session identifier *sid*, the functionality now takes another unique "identifier" *cid*, which may be used if a sender sends to the same receiver multiple times within a session. We assume that the combination of *sid* and *cid* is globally unique.

$\mathcal{F}_{mvno}$ parameterized by a message space $\mathcal{M}$ and interact with adversary **Sim** and parties $P_1, P_2, ..., P_n$ as follows:

- Upon receiving $(\text{message}_1, sid, cid, P_i, P_j, M_1)$ from P_i, it proceeds as follows: if a tuple $(sid, cid, \cdots)$ with the same (sid, cid) was previously recorded, do nothing. Otherwise, record $(sid, cid, P_i, P_j, M_1)$ and send $(\text{message}_1, sid, cid, P_i, P_j)$ to P_j and Sim.
- Upon receiving $(\text{message}_k, sid, cid, P_i, P_j, M_2)$ from P_i for $k \in \{2, 3\}$, it proceeds as follows: if a tuple $(sid, cid, \cdots)$ with the same (sid, cid) was previously recorded, record $(sid, cid, P_i, P_j, M_k)$ and send $(\text{receipt}, sid, cid, P_i, P_j)$ to P_j and Sim. Otherwise, do nothing.
- Upon receiving $(\text{message}_4, sid, cid, P_i, P_j, M_4)$ from P_i, it proceeds as follows: if a tuple $(sid, cid, \cdots)$ with the same (sid, cid) was previously recorded, do nothing. Otherwise, record $(sid, cid, P_i, P_j, M_4)$ and send $(\text{message}_4, sid, cid, P_i, P_j)$ to P_j and Sim.
- Upon receiving $(\text{corrupt}, sid, cid)$ from the adversary, send M to the adversary if there is already an entry (sid, cid, P_i, P_j, M). Change the record to $(sid, cid, P_i, P_j, M^*)$, if the adversary provides some M^* and $(\text{receipt}, sid, cid, P_i, P_j)$ has not yet been written on P_j 's output tape.

6.1 Formal security analysis

In this section, we analyse the security of the proposed scheme. In this regard, we consider formal security analysis that primarily revolves around the Universal Composability (UC) framework [27], which is inherently designed to offer strong, composable security guarantees that extend to the real and ideal models of protocol execution. This framework provides a robust structure for analysing the security properties of cryptographic protocols in a way that is preserved even when these protocols are composed of others in complex systems. While tools like ProVerif [18] and Tamarin [50] are indeed powerful for automating the verification of security properties, particularly in cryptographic protocols, their use is primarily tailored towards protocols that need to be verified against specific properties such as secrecy and authentication in symbolic models. Our manual analysis allows for detailed, nuanced handling of the specifics of the UC model, which are often only directly supported by these automated tools with considerable customisation and significant manual efforts. Thus, while ProVerif [18] and Tamarin [50] are invaluable in many contexts, the security validation in our paper is most appropriately addressed through rigorous manual proofs within the UC framework. This ensures accurate and context-specific security assurance that aligns with our protocol's innovative aspects.

We consider a sequence of *hybrid* games between the real and ideal worlds. This is a general approach that one can follow to prove the security of a commitment scheme in the UC model. The game starts with the real game, adversary A interacts with real parties, and ends with the ideal game. In the ideal game, we build SIM that interfaces between adversary A and ideal functionality F_{mvno}.

$Game_0$: This is the original real game that corresponds to the real world in the model. This game executes the real protocol between committer P_i and receiver P_j. The environment $\mathcal{Z}$ chooses the input for the honest committer P_i, and $\mathcal{Z}$ receives the output of the honest committer. In our framework, there is an adversary $\mathcal{A}$ that aims to attack the real protocol in the real world by corrupting some parties P and listening to all flows from parties. In that case, $\mathcal{A}$ can read the corrupted party's current inner state and fully control it. In our security game, environment $\mathcal{Z}$ can control adversary $\mathcal{A}$ and see all communication messages from all parties and also all of $\mathcal{A}$'s interactions with other parties.

$Game_1$: We consider that the adversary $\mathcal{A}$ controls UE. In the setup phase of this game, simulator **Sim** chooses the *crs*, its trapdoor td and utilises the SanSig.Sanit() algorithm to sanitise the certificate to include their identity and a timestamp. After that, simulator **Sim** sends a certificate with its signature σ_G^* to the UE via M_1. Upon receiving message M_2 from UE, simulator **Sim** first check the τ_2 and verify the Message using zero-knowledge proof. After that, simulator **Sim** randomly generates a UID_i and computes the hash value H_{UID_i}. Finally, simulator **Sim** encrypt message M_4 and send it to the User.

Lemma 1. *If* Π *= (KGen, SanSig, Enc, Dec, Sign, Verify), the labelled signature is EUF-CMA secure, the labelled ZKP holds commitment sound and binding, the output of* $\mathcal{Z}$ *in* $Game_0$ *and* $Game_1$ *is computationally indistinguishable.*

Proof. In $Game_1$, we consider two possible scenarios and split our proof into two cases. We first observe that SIM reveals verified results after some party P_i^1 open commitment to message M_1. We assume that **bad** defines the case that sender P_i^1 successfully generates a valid H_{pid}, which means it can generate a valid π_{ZK}. The bad happens with a negligible probability because the commitment generated by the ZKP algorithm has the sound and binding property. In the second scenario, we assume that **bad** defines the case that sender P_i^2 successfully generates a valid σ. The bad happens with a negligible probability because the signature scheme is EUF-CMA secure. Hence, from the proof above, the bad cases happen only with a negligible probability, and two games $Game_0$ and $Game_1$ are computationally indistinguishable in a view of $\mathcal{Z}$.

$Game_2$: In this game, we consider that the adversary $\mathcal{A}$ controlled gNB and CN, which means the UE doesn't trust them. Upon the simulator **Sim** receiving message M_1 from the gNB, the simulator **Sim** first check the $id_{gNB}\&\tau$. After that, the simulator **Sim** verifies the sanitizable signature using SanSig.Verify. After generate the secret key SK_{Sim} and public key SK_{Sim}, the simulator **Sim** generate $r \leftarrow_{\$} \{0, 1\}$. Hence the simulator **Sim** don't know the pid, it generates the commitment c from $c = \text{Com}_{ck}(0, r)$. And than the simulator **Sim** generates the proof using $\pi_{ZK} \leftarrow \text{ZP.SIM}(\text{td}, c, crs, 0)$. Finally, the simulator **Sim** sends message M_2 to the gNB.

Lemma 2. *If* Π *= (KGen, ZK.P, Sign, SanSig, Verify), the labelled signature is EUF-CMA secure, the labelled ZKP holds zero-Knowledge property and the commitment has hiding property, the output of* $\mathcal{Z}$ *in* $Game_1$ *and* $Game_2$ *is computationally indistinguishable.*

Proof. In $Game_2$, we can observed that after P_i open commitment to message M, simulator SIM reveals verified results V_σ. Suppose that **bad** defines the case that receiver P_j successfully get value H_{pid}. The **bad** happens with a negligible probability due to the zero-knowledge property and the hiding property of the commitment. Therefore, the proof generated by the simulator **Sim** is indistinguishable from the real proof. Also, due to the unlinkability and the unforgeability of the signature mechanism, the output of the signature for both simulator **Sim**'s zero-knowledge proof and real proof are indistinguishable.

Hence, from the proof above, the bad case happens only with a negligible probability and two games $Game_1$ and $Game_2$ are computationally indistinguishable in a view of $\mathcal{Z}$.

$Game_3$: This game corresponds to the ideal world in the CRS model. In an ideal world, there exists an ideal function F_{mvno} and an honest task. Parties in the ideal world simply pass inputs from environment Z to the ideal world function and vice versa. In an ideal world, an ideal honest party interaction has only environment Z and ideal functionality. In this game, the ideal world adversary **Sim** proceeds following functions:

- *Initialisation step:* Sim chooses the crs, its td, and ck.
- *Simulating the communication with* $\mathcal{Z}$*:* Every input value that Sim receives from $\mathcal{Z}$ is written on $\mathcal{A}$'s input tape (as if coming from $\mathcal{Z}$) and vice versa.
- *Simulating the the first round when sender* P_i *is honest*: Upon receiving the receipt message (receipt, sid, cid, P_i, P_j) from F_{mvno}, Sim computes M_k like a honest party and sends (message$_1$, sid, cid, M_k) to P_j.
- *Simulating the second round when sender* P_i *is honest*: Upon receiving the receipt message (receipt, sid, cid, P_i, P_j) from F_{mvno}, Sim computes M_2 by $\mathsf{Com}_{\mathsf{ck}}(0, r)$ for randomly chosen r, and run the ZK simulator to compute the proof π (with using the trapdoor td) and sends (message$_2$, sid, cid, M_2) to P_j.
- *Simulating the the round* $k \in \{3, 4\}$ *when sender* P_i *is honest*: Upon receiving the receipt message (receipt, sid, cid, P_i, P_j) from F_{mvno}, Sim computes M_1 like a honest party and sends (message$_1$, sid, cid, M_1) to P_j.
- *Simulating adaptive corruption of Pi after the round k*: When P_i is corrupted, Sim can immediately read ideal P_i's inner state and obtain M . Then, Sim produces M_k as in the case of the round $k + 1$ when P_i is honest and outputs (sid, cid, M_k) to the P_j.
- *Simulating the commit phase when committer* $\hat{P}_i$ *is corrupted and the receiver* P_j *is honest*: After receiving (message$_k$, sid, cid, M_k) from $\hat{P}_i$ controlled by $\mathcal{A}$ in the round k, Sim runs the extractor of ZK and compute M', and sends (message, sid, cid, P_j, M') to F_{mvno}.
- *Simulating adaptive corruption of* P_j *after the round k but before the verifying phase*: When P_j is corrupted, Sim simply outputs (sid, cid, M_k).

By the construction of the above functions, $Game_3$ is identical to the $Game_2$.

Table 2: Comparison with the state-of-the-art.

Features / Schemes	SMCT	MA+	CP		UC	PR
			AN	UL		
Conventional-5G[37]	5G	✗	✓	✗	✗	✗
$5G-AKA'$ [65]	5G	✗	✓	✓	✗	✗
PGPP [57]	MVNO	✗	✓	✓	✗	✗
AAKA [69]	4G. 5G	✗	✓	✓	✗	✗
Thick Model and SM-DP+	MVNO*	✓	✓	✗	✗	✗
MVNO-AKA (Ours)	MVNO*	✓	✓	✓	✓	✓
MVNO-HO (Ours)	MVNO*	✓	✓	✓	✓	✓

SMCT: Supported Mobile Communication Type;
MA+: MA within MVNO environment; **AN**:Anonymity; **UL**: Unlinkability
CP: Comprehnsive Privacy, **UC**: Universal Compositions Security;
PR: Privacy-preserving Revocation, **MVNO***: Applicable for MVNO, 4G, 5G and next-generation mobile communication
Thick Model and SM-DP+: Thick model and an SM-DP+ that rotates IMSIs.

7 COMPARISON WITH EXISTING WORK

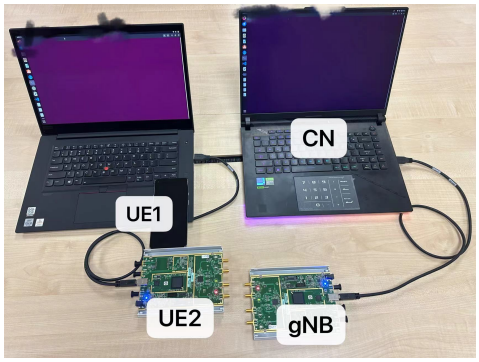
In this section, we explore state-of-the-art closely aligned with this research and qualitatively compare them with our proposed approach. Considering the privacy aspect, Wang et al. [65] focus on addressing only linkability attacks within the current 5G AKA protocol. However, this protocol does not provide complete privacy protection, such as preventing identity exposures, privacy-preserving mutual authentication, and universal composability within the MVNO environment. Moreover, Wang et al. do not consider secure revocation mechanism which is crucial for the privacy of MVNO users. Additionally, Schmitt and Raghavan [57] have proposed a refactor-based approach, named PGPP, to safeguard user identity and location privacy. The authors introduce a logical entity, termed the PGPP Gateway (PGPP-GW), situated interstitially between the User Plane Function (UPF) and the public Internet. This configuration serves to decouple authentication from connectivity credentials, thereby providing a mechanism for authentication while concurrently preserving user privacy. However, while PGPP-GW facilitates privacy protection, it does not provide a concrete security solution considering the protocol-level challenges. Instead, they suggested major infrastructural changes in 5G settings.

Concurrent Work. In an independent and concurrent work, [69] also uses zero-knowledge proof to address the issue of tracking users' digital footprint in the cellular network. Roughly speaking, [69] introduces (AAKA) an AKA protocol that relies on a combination of cryptographic primitives, including Decisional Diffie-Hellman, zero-knowledge proof, BBS signatures, Keyed-Verification Anonymous Credential, and ElGamal Encryption. Despite asserting its compatibility with 5G, including these asymmetric cryptographic elements raises practicality concerns.

The AAKA protocol exhibits several limitations that warrant consideration. Primarily, AAKA is explicitly tailored for complete privacy, Making it hard to revoke a specific user from the network. Additionally, its lack of composability and absence of UC security potentially compromise its robustness when integrated into larger, real-world systems. From a cryptographic perspective, AAKA's reliance on pairing-based settings and the discrete logarithm assumption renders it vulnerable in post-quantum scenarios, as it does not incorporate quantum-secure primitive-based assumptions. Furthermore, While the primary goal of AKAA is to achieve comprehensive privacy, it is important to note that this privacy pertains

exclusively to the AKA protocol, leaving the privacy of roaming users during handovers unaddressed. When MVNO users undergo handover, the MNOs also obtain handover data, e.g., information about the source and destination cells. By secure handover (SH), we mean MNOs will be oblivious to this handover. Now, in order to achieve anonymity, particularly at the MVNO setting, one of the possible approaches could be using a thick model and an SM-DP+ that rotates IMSIs. However, this approach cannot prevent linkability attacks and fake base station attacks. In this regard, each time when the IMSI rotates, the attacker can try to relink the victim UE to the new IMSI by replaying the Authentication Request message for the previous IMSI [13]. Furthermore, even if the IMSI catcher is not realizable due to the use of public-key encryption of SUPI, the attacker can still launch other fake base station attacks. For example, the attacker can use RRC Reject to launch a DoS attack as described in 5GReasoner[40]. While AAKA protocol addresses certain privacy concerns, our proposed protocol offers a more comprehensive and robust solution to the privacy and security challenges in 5G networks. Both approaches improve upon the conventional 5G AKA protocol, which is susceptible to linkability attacks [65]. These attacks exploit the protocol's handling of MAC failures and its predictable challenge-response approach. When a MAC verification fails, the network sends a distinct error message, allowing an attacker to differentiate between a targeted subscriber and others. We achieve unlinkability through a novel mechanism: encrypting all responses from the Core Network using fresh, session-specific public keys for each user. This ensures that even if an adversary forwards a message (e.g., M4) to a group of users, including the intended recipient, none of them can decrypt it. This is because users have already generated new session keys, and without the correct private key, decryption is impossible. Additionally, our protocol's ZKP-based approach guarantees user unlinkability across sessions, as the network never receives user identifiers. As demonstrated in Lemma 2, this comprehensive strategy achieves robust unlinkability throughout the entire protocol, significantly enhancing user privacy and resistance to tracking attacks.

Another key distinguishing feature of our protocol is its resilience against fake base station attacks, an area where previous protocols like AAKA and 5G AKA show limitations. We implement a dual-layered strategy: first, a public key infrastructure for base stations, requiring sanitizable signatures with CN and gNB public keys, dramatically increases the difficulty of simulating legitimate nodes. Second, we maintain comprehensive privacy through ZKP, allowing user authentication without revealing specific information to gNBs. In contrast, conventional 5G-AKA and AAKA remain vulnerable to fake base station attacks due to unauthenticated System Information Block messages [13, 40]. They also struggle with privacy-preserving handovers, as MNOs can track user locations through gNB observations. These systems would require impractical UE re-registration with new IMSIs at each base station connection. Table 2 offers a comparative analysis of related works, including the AAKA protocol, alongside our proposed scheme. This comparison illustrates the comprehensive nature of our solution in addressing various security and privacy challenges in 5G networks.




```
[RRC]   gNB[id_gNB = 8] Generating Message M1 in SIB1 message.
[RRC]   Generating certificate.
[RRC]   Generating Signature using SanSig.Sanit.
[RRC]   Generating Signature succeffully with length: 71
[RRC]   Finish Generating Message M1 in SIB1 message
```

Figure 6: (Top) Testbed Setup, and (Bottom) Log From gNB.

8 EVALUATION

We first provide the details of our testbed setup. We also present the performance analysis of the proposed scheme and compare it with the state-of-the-art protocols presented [30, 35, 37, 65, 70].

8.1 Testbed Setup

As shown in Figure 6, to build the testbed environment, we use an ASUS machine with an i9 core, 5.6GHz CPU, and 16.0 GB RAM, two USRP B210 [3] software-defined radios (SDRs) connected to the computer running an Ubuntu 22.02 desktop OS. We use a popular open-source 5G stack called OpenAirInterface [2] to set up the UE, 5G base station (gNB) and core network. To implement our protocol on the standard 5G protocol, we modify the code of the core network, gNB, and UE stack in OpenAirInterface to support our initial authentication protocol. In this setup (as shown in Figure 6), one USRP connected with a Lenovo laptop acts as the 5G base station, and another USRP connected with the second Lenovo Laptop works as a UE (UE_2). Both machines run Ubuntu 22.02 operating system on a core i7 core machine with a 2.7GHz CPU and 32.0 GB RAM.

In addition, to effectively measure the performance of our proposed scheme on 5G phones, we use a Galaxy Note 9 smartphone (UE_1) running Android 10 mobile operating system and equipped with octa-core processors (1.8GHz Quad-Core ARM Cortex-A55, 2.7GHz Quad-Core Mongoose M3) and 6GB of RAM. The cryptographic operations of our proposed scheme and related works were implemented using OpenSSL 3.0 [51], Java Pairing-Based Cryptography (JPBC) [33] and Java Cryptography Extension (JCE) [62]. Since we cannot modify the existing firmware of the basebands, we implemented the cryptographic algorithms on the application processor. We measured the computation time to demonstrate the feasibility of the proposed cryptographic constructs on the commodity phones. To comprehensively analyse and compare our proposed scheme against existing works, we implemented the cryptographic algorithms/protocols proposed in the compared papers and measured those works' computational and communication costs (wrt. time) in our testbed. For analysing the communication costs, the

communication setup involves considering factors such as propagation and transmission time, message size, and the network's data rate to measure transmission delays across all protocols accurately.

Due to the page limit, we provide only a single log picture of our 5G implementation in Figure 6. This log shows the process of gNB generating message M_1 at the initial authentication protocol. Full implementation details can be found in the Appendix. We also provide the prototype of the implementation through GitHub [12].

8.2 Evaluation Results with Testbed

We have implemented the entire proposed AKA protocol in our 5G testbed. We evaluate it using three metrics. First, we compute the computational cost, i.e., the time required by UE, gNB, and CN to perform the cryptographic operations involved in our proposed protocol. Second, we measure the communication costs, i.e., the number of additional bytes introduced by our approach to the existing 5G-AKA over-the-air messages. Finally, we evaluate the end-to-end latency of the entire AKA protocol. This includes all the computational and communication costs. We also use the same SDR-based testbed to compare the end-to-end latency of our protocol with those of state-of-the-art protocols [37, 57, 65, 69].

Table 3: Time Required for Cryptographic Operations and Communication Overhead.

Compare \ Message	M1	M2	M3	M4
Time Required (ms)	0.416	12.236	4.876	1.679
Message Size (bytes)	95	830	830	256

8.2.1 Time Required for Cryptographic Operations. In this experiment, we report the time required for different entities to run all cryptographic operations for message M_1 to message M_4 by the corresponding entities. In our proposed AKA protocol, the message M_1 is generated and sent from the gNB. The major cryptographic operation involved is the generation of the signature, which takes 0.416ms. The message M_2 is generated by the MVNO user, and it consists of a verification process, zero-knowledge proof, and a singing phase. Note that proof can be generated offline; hence, we have not considered its associated time. So, the time required to generate M_2 is 12.236ms. For message M_3, it needs a verification phase and a signing phase, which is around 4.876ms. The last message M_4 involves the verification and the signing phase, where the verification of the proof takes longer time, around 1.679 ms.

8.2.2 Communication Overhead. Here, we provide the details of each message regarding the overhead. While the message M_1 contains the certificate (24 bytes) and its signature (71 bytes), the SIB1 message of our protocol is a total of 95 bytes. In messages M_2 and M_3, we compute a zero-knowledge proof (304 bytes), a serialised public key (426 bytes), a timestamp (8 bytes) and the signature (71 bytes), which is 830 bytes in total with other supported bytes. In the last message, M_4, we send encrypted data using a public encryption scheme, which is 256 bytes.

8.2.3 ***End-to-End Cost.*** The goal of this experiment is to measure the overall latency of the protocol. To better illustrate the results, we measure the time required by our proposed AKA scheme and other state-of-the-art protocols from the time that UE receives the SystemInformationBlockType1 (SIB1) (i.e., M1 in Figure 3) message to UE sends the authentication response (Next message of M4 in Figure 3). It includes the computational cost (cryptography part), communication cost (transmission part), and all other delays between the UE and gNB. Table 4 provides the results of the state-of-the-art protocols [37, 57, 65, 69] for the end-to-end cost based on SDR-based testbed. It details the overall latency (time) required for the execution of the initial Authentication (AKA), i.e., AKA protocols, measuring the total time needed to perform the protocol at the User Equipment level (T_{UE}) and the system level (T_{Sys}) (i.e., gNB and CN). As we can see, the conventional 5G protocol takes around 1.32 seconds, which is the baseline of the state-of-the-art protocols. The 5G-AKA' takes a similar amount of time; hence, it changed a small part of the protocol. The AAKA scheme takes a longer time ($1.44 + \kappa$ seconds) than ours because it uses zero-knowledge proof with paring operations and ECDH, which have higher computational costs. κ refers to the transmission time needed after UE sends the authentication response. For ours, it takes 1.41 seconds while we are using zero-knowledge proof but we can move the computational overhead out of the real-time phase. Although our proposed scheme exhibits slightly higher latency compared to conventional 5G protocols, it significantly enhances security guarantees relative to existing alternatives. It is important to note that all related works have been compared except for PGPP [57]. Empirical evaluation with PGPP is difficult because PGPP did not make its implementation open-source. Moreover, PGPP requires a dedicated HW/infrastructure, such as a PGPP gateway, which is impossible to implement as the implementation details are missing in their paper. We first introduce the results of the SDR-based testbed.

8.3 Evaluation of Cryptographic Operations on Phone and without Testbed

We also evaluate the time required by a commercia-gradel UE to perform the cryptographic operations involved in different AKA and Handover protocols. For this, we use a Galaxy Note 9 as the UE and implemented only the cryptographic operations (e.g., signature and zero-knowledge proof generation and verification) of the Authentication (AKA) and Handover (HO) protocols without implementing the message flows between UE, gNB, and CN. We compute the time required by Galaxy Note 9 to perform those cryptographic operations during the executions of different AKA and Handover protocols. Note that it is difficult to modify the baseband to run the protocols, so we use the application processor of Galaxy Note 9 to perform the experiment, which is reasonable for the purpose of comparing the relative difference between our protocol and the other state-of-the-art protocols.

Table 5 presents the computation time required for the execution of both the initial authentication (AKA) and Handover (HO) protocols, measuring the total time needed to perform the protocol at the phone level (T). It shows that our proposed initial authentication protocol requires approximately 5.65 μs. On the other hand, our proposed Handover protocol requires approximately 5.45 μs. Compared with the AAKA protocol, our scheme takes less time for the initial authentication phase. Meanwhile, compared with the conventional 5G protocol, even though our proposed protocol takes longer time, our scheme provides higher security guarantees.

Table 4: End-to-End Performance comparison with SDR-based Testbed.

Schemes / Compare	Conventional-5G [37]	5G-AKA' [65]	PGPP [57]	AAKA [69]	Ours Scheme
Protocol Type	5G	5G	MVNO	MNO	MVNO*
End-to-End AKA Cost (s)	1.31	1.33	-	1.44 + κ	1.41

Table 5: Performance comparison of only the cryptographic operations on the application processor of a commercial phone (Galaxy Note 9).

Schemes	Type	Phase	T_{Phone} (ms)
Conventional-5G [37]	5G	IA	2.82
		HO	2.7
		Total (IA+HO)	5.52
5G – AKA' [65]	5G	IA	1.69
		HO	1.99
		Total (IA+HO)	3.68
PGPP [57]	MVNO	IA	-
		HO	NS
		Total (IA+HO)	-
AAKA [69]	MNO	IA	108.95
		HO	NS
		Total (IA+HO)	-
Ours	MVNO*	IA	5.65
		HO	5.45
		Total(IA+HO)	11.11

9 DISCUSSION

• **Lawful traceability**: In general, to ensure comprehensive privacy, we do not allow MNOs to access the user's identity. Thus, in our current solution, the MVNO needs to incorporate MNOs to locate individuals legally. With this aim, we propose a new definition of revocation, discussed in Section 5.4, alongside a practical ZKP, where the MVNO generates the setup for the ZKP. The combination of these elements allows us to maintain user privacy by using non-identifying yet unique attributes to trigger revocation. However, in cases requiring lawful traceability, the MVNO is obligated to share secret information with MNO, including the trapdoor of crs. This approach ensures compliance with legal requirements without compromising user identities.

• **Lawful revocation**: The revocation procedure of our proposed privacy-preserving AKA protocol is crucial due to the protocol's strong emphasis on user anonymity and unlinkability. This anonymity can complicate identifying and revoking a specific user's credentials, which is essential for handling fraudulent activities or compromised user data. Traditionally, revocation mechanisms rely on identifying information to disable an account, but this approach conflicts with the privacy guarantees of our protocol. To address this, we propose a new definition of efficient revocation, explained in Section 5.4, that allows for effectively managing network security and legal compliance without revealing user identities. This mechanism is crucial for handling instances of fraud or compromised data while keeping to stringent data protection laws like GDPR in Europe, which require the ability to disable access for users engaging in illegal activities without breaching privacy regulations.

10 RELATED WORK

Considerable efforts have been dedicated to enhancing the security of 5G networks, particularly in the realm of user authentication and network security during authentication and handover protocols. Here, we highlight some under-specified security and privacy requirements and weaknesses in the current 5G-AKA and HO protocol versions. These issues were previously addressed by [13, 20, 32, 57], which include vulnerabilities such as traceability attacks from active adversaries, identity confusion attacks, lack of perfect forward secrecy, and confidentiality attacks on sequence numbers. In response to these identified vulnerabilities, ongoing efforts have been undertaken to develop effective countermeasures [20, 35, 61, 65, 70]. Addressing user privacy concerns is a pivotal challenge within the 5G-AKA framework, an issue that is thoroughly examined by Braeken [20] and Wang et al.[65]. To tackle the anonymity and unlinkability issues inherent in the current version, they have introduced an improved 5G-AKA protocol. Beyond privacy, the assurance of base station authenticity in user communication is another critical issue in preventing potential fake base station attacks. In this regard, a robust solution has been proposed in [61], presenting an efficient approach to mitigating this particular security challenge. In the context of the 5G Handover (5G-HO) protocol, [35] introduces a region-based handover protocol (ReHand) that ensures user anonymity, perfect forward secrecy, and a fast revocation mechanism. Similarly, [70] have designed a universal HO protocol using chameleon hash functions and blockchains (RUSH). This protocol achieves security features similar to those of the ReHand protocol. However, it is essential to note that both protocols (Rehand and RUSH) rely on the standardized 5G-AKA protocol, which is vulnerable to perfect forward secrecy and only supports weak anonymity [13]. As a result, the security of their proposed protocols is affected by the security and privacy weaknesses of the current version of 5G-AKA. Despite the extensive efforts in this domain, the current works in the field fall short of accomplishing privacy-preserving and secure authentication and handover protocols. Furthermore, none of the prior works investigated or presented the critical requirement of achieving universal composability in the context of security protocols. Additionally, the security and privacy aspects specific to the 5G-enabled MVNO environment remain unexplored in prior research, rendering this work the first, to the best of our knowledge, in this particular area.

11 CONCLUSION AND FUTURE WORK

In recent times, the Mobile Virtual Network Operator (MVNO) has garnered significant interest. MVNOs provide a multitude of advantages that render them an attractive option for the majority of consumers. In this paper, we first shed light on some of the prominent security and privacy issues in the MVNO environment, where an MVNO needs to share its customer information with the MNO for validation. This may cause serious privacy issues for their customers. In order to address all these issues, here we have introduced a universally composable authentication and handover strategy that provides robust user privacy. The scheme allows any MVNO user to verify a mobile operator (MNO) and vice versa while ensuring user anonymity and unlinkability support. Our proposed

solution is expected to be implemented by the MVNO(s) in order to provide improved privacy support to their customer(s). One of the proposed directions for future work would be to consider cross-layer authentication under MVNO settings.

Future Work. This work does not consider privacy leakage through physical layer and cross-layer communications, e.g., interactions between physical and upper layers. We consider it our future work.

ACKNOWLEDGEMENTS

We thank the anonymous reviewers and the shepherd for their feedback and suggestions. We also thank the vendors for cooperating with us during the responsible disclosure. The work of Prosanta Gope was supported by The Royal Society Research Grant under grant RGS\R1\221183. Meanwhile, this work of Syed Rafiul Hussain has been supported by the NSF under grants 2145631, 2215017, and 2226447, the Defense Advanced Research Projects Agency (DARPA) under contract number D22AP00148, and the NSF and Office of the Under Secretary of Defense– Research and Engineering, ITE 2326898, as part of the NSF Convergence Accelerator Track G: Securely Operating Through 5G Infrastructure Program.

REFERENCES

[1] CVE-2021-45462, https://nvd.nist.gov/vuln/detail/CVE-2021-45462
[2] OpenAirInterface. https://openairinterface.org/, accessed: April 16, 2024
[3] USRP B210. https://www.ettus.com/all-products/UB210-KIT/, accessed: April 16, 2024
[4] Abdolmaleki, B., Baghery, K., Lipmaa, H., Siim, J., Zając, M.: Dl-extractable uc-commitment schemes. In: Applied Cryptography and Network Security: 17th International Conference, ACNS 2019, Bogota, Colombia, June 5–7, 2019, Proceedings 17. pp. 385–405. Springer (2019)
[5] Abdolmaleki, B., Khoshakhlagh, H., Slamanig, D.: A framework for uc-secure commitments from publicly computable smooth projective hashing. In: Cryptography and Coding: 17th IMA International Conference, IMACC 2019, Oxford, UK, December 16–18, 2019, Proceedings 17. pp. 1–21. Springer (2019)
[6] Abdolmaleki, B., Ramacher, S., Slamanig, D.: Lift-and-shift: obtaining simulation extractable subversion and updatable snarks generically. In: Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. pp. 1987–2005 (2020)
[7] Abdolmaleki, B., et al.: Universally composable nizks: Circuit-succinct, non-malleable and crs-updatable. Cryptology ePrint Archive (2023)
[8] Ajtai, M.: Generating hard instances of lattice problems. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. pp. 99–108 (1996)
[9] Alnashwan, R., Gope, P., Dowling, B.: Privacy-Aware Secure Region-Based Handover for Small Cell Networks in 5G-Enabled Mobile Communication. IEEE Transactions on Information Forensics and Security **18**, 1898–1913 (2023). https://doi.org/10.1109/TIFS.2023.3256703, conference Name: IEEE Transactions on Information Forensics and Security
[10] Ames, S., et al.: Ligero: Lightweight sublinear arguments without a trusted setup. In: Proceedings of the 2017 acm sigsac conference on computer and communications security. pp. 2087–2104 (2017)
[11] Ateniese, G., Chou, D.H., de Medeiros, B., Tsudik, G.: Sanitizable Signatures. In: di Vimercati, S.d.C., Syverson, P., Gollmann, D. (eds.) Computer Security – ESORICS 2005. pp. 159–177. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2005). https://doi.org/10.1007/11555827-10
[12] Authors.: 5g protocoltype implementation. https://github.com/place.holder.later. update (2024), access Time2024-07-26
[13] 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. pp. 1383–1396. CCS '18, Association for Computing Machinery, New York, NY, USA (Oct 2018). https://doi.org/10.1145/3243734.3243846, http://doi.org/10.1145/3243734.3243846
[14] Baum, C., et al.: More efficient commitments from structured lattice assumptions. In: International Conference on Security and Cryptography for Networks. pp. 368–385. Springer (2018)
[15] Ben-Sasson, E., et al.: Aurora: Transparent succinct arguments for r1cs. In: Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I 38. pp. 103–128. Springer (2019)
[16] Benhamouda, F., et al.: Better zero-knowledge proofs for lattice encryption and their application to group signatures. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 551–572. Springer (2014)
[17] Bitsikas, E., et al.: Ue security reloaded: Developing a 5g standalone user-side security testing framework. In: Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks. pp. 121–132 (2023)
[18] Blanchet, B., et al.: Modeling and verifying security protocols with the applied pi calculus and proverif. Foundations and Trends® in Privacy and Security **1**(1-2), 1–135 (2016)
[19] Borgaonkar, R., Hirschi, L., Park, S., Shaik, A.: New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols. Tech. Rep. 1175 (2018), http://eprint.iacr.org/2018/1175
[20] Braeken, A.: Symmetric key based 5G AKA authentication protocol satisfying anonymity and unlinkability. Computer Networks **181**, 107424 (Nov 2020). https://doi.org/10.1016/j.comnet.2020.107424, https://www.sciencedirect.com/science/article/pii/S1389128620311130
[21] Brzuska, C., Fischlin, M., Lehmann, A., Schröder, D.: Santizable Signatures: How to Partially Delegate Control for Authenticated Data. pp. 117–128 (Jan 2009)
[22] Brzuska, C., Fischlin, M., Lehmann, A., Schröder, D.: Unlinkability of Sanitizable Signatures. pp. 444–461 (May 2010). https://doi.org/10.1007/978-3-642-13013-7-26
[23] Brzuska, C., Pöhls, H.C., Samelin, K.: Non-interactive Public Accountability for Sanitizable Signatures. In: De Capitani di Vimercati, S., Mitchell, C. (eds.) Public Key Infrastructures, Services and Applications. pp. 178–193. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40012-4-12
[24] Brzuska, C., Pöhls, H.C., Samelin, K.: Efficient and Perfectly Unlinkable Sanitizable Signatures without Group Signatures. In: Katsikas, S., Agudo, I. (eds.) Public Key Infrastructures, Services and Applications. pp. 12–30. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2014). https://doi.org/10.1007/978-3-642-53997-8-2
[25] Bünz, B., et al.: Bulletproofs: Short proofs for confidential transactions and more. In: 2018 IEEE symposium on security and privacy (SP). pp. 315–334. IEEE (2018)
[26] Bünz, B., et al.: Transparent snarks from dark compilers. In: Advances in Cryptology–EUROCRYPT 2020: 39th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10–14, 2020, Proceedings, Part I 39. pp. 677–706. Springer (2020)
[27] Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: Proceedings 42nd IEEE Symposium on Foundations of Computer Science. pp. 136–145. IEEE (2001)
[28] Canetti, R.: Universally composable security. Journal of the ACM (JACM) **67**(5), 1–94 (2020)
[29] Canetti, R., et al.: A simpler variant of universally composable security for standard multiparty computation. In: Advances in Cryptology–CRYPTO 2015: 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2015, Proceedings, Part II 35. pp. 3–22. Springer (2015)
[30] Cao, J., Ma, M., Fu, Y., Li, H., Zhang, Y.: CPPHA: Capability-Based Privacy-Protection Handover Authentication Mechanism for SDN-Based 5G HetNets. IEEE Transactions on Dependable and Secure Computing **18**(3), 1182–1195 (May 2021). https://doi.org/10.1109/TDSC.2019.2916593, conference Name: IEEE Transactions on Dependable and Secure Computing
[31] Chlosta, M., Rupprecht, D., Holz, T., Pöpper, C.: Lte security disabled: Misconfiguration in commercial networks. In: Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks. p. 261–266. WiSec '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3317549.3324927, https://doi.org/10.1145/3317549.3324927
[32] Cremers, C., Dehnel-Wild, M.: Component-Based Formal Analysis of 5G-AKA: Channel Assumptions and Session Confusion. San Diego, CA, USA (Feb 2019), https://publications.cispa.saarland/2758/
[33] De Caro, A., Iovino, V.: jPBC: Java pairing based cryptography. Proceedings of the 16th IEEE Symposium on Computers and Communications, ISCC 2011, IEEE (2011), http://gas.dia.unisa.it/projects/jpbc/
[34] Dolev, D., Yao, A.: On the security of public key protocols. IEEE Transactions on Information Theory **29**(2), 198–208 (Mar 1983). https://doi.org/10.1109/TIT.1983.1056650, conference Name: IEEE Transactions on Information Theory
[35] Fan, C.I., Huang, J.J., Zhong, M.Z., Hsu, R.H., Chen, W.T., Lee, J.: ReHand: Secure Region-Based Fast Handover With User Anonymity for Small Cell Networks in Mobile Communications. IEEE Transactions on Information Forensics and Security **15**, 927–942 (2020). https://doi.org/10.1109/TIFS.2019.2931076
[36] Gajek, S., et al.: Universally composable security analysis of tls. In: Provable Security: Second International Conference, ProvSec 2008, Shanghai, China, October 30-November 1, 2008. Proceedings 2. pp. 313–327. Springer (2008)
[37] 3rd Generation Partnership Project (3GPP), E.: Security architecture and procedures for 5G System. Technical Specification version 16.3.0 Release 16, 3GPP (Aug 2020), https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/16.03.00_60/ts_133501v160300p.pdf

[38] Golovnev, A., et al.: Brakedown: Linear-time and post-quantum snarks for r1cs. Cryptology ePrint Archive (2021)

[39] Hussain, S.R., et al.: Lteinspector: A systematic approach for adversarial testing of 4g lte. In: Network and Distributed Systems Security (NDSS) Symposium (2018)

[40] Hussain, S.R., et al.: 5greasoner: A property-directed security and privacy analysis framework for 5g cellular network protocol. In: Proceedings of the ACM SIGSAC Conference on Computer and Communications Security. pp. 669–684 (2019)

[41] Hussain, S.R., et al.: Insecure connection bootstrapping in cellular networks: the root of all evil. In: Proceedings of the 12th conference on security and privacy in wireless and mobile networks. pp. 1–11 (2019)

[42] Hussain, S.R., et al.: Noncompliance as deviant behavior: An automated black-box noncompliance checker for 4g lte cellular devices. In: Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (2021)

[43] Jain, A., et al.: Commitments and efficient zero-knowledge proofs from learning parity with noise. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 663–680. Springer (2012)

[44] Kawachi, A., et al.: Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: Advances in Cryptology-ASIACRYPT 2008: 14th International Conference on the Theory and Application of Cryptology and Information Security, Melbourne, Australia, December 7-11, 2008. Proceedings 14. pp. 372–389. Springer (2008)

[45] Kim, B.W., et al.: Economic analysis of the introduction of the mvno system and its major implications for optimal policy decisions in korea. Telecommunications Policy **31**(5), 290–304 (2007)

[46] Kim, E., et al.: {BASECOMP}: A comparative analysis for integrity protection in cellular baseband software. In: 32nd USENIX Security Symposium (USENIX Security 23). pp. 3547–3563 (2023)

[47] Li, Y., et al.: Understanding the ecosystem and addressing the fundamental concerns of commercial mvno. IEEE/ACM Transactions on Networking **28**(3), 1364–1377 (2020)

[48] Lotto, A., et al.: Baron: Base-station authentication through core network for mobility management in 5g networks. In: Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks. pp. 133–144 (2023)

[49] Lyubashevsky, V., et al.: On ideal lattices and learning with errors over rings. Journal of the ACM (JACM) **60**(6), 1–35 (2013)

[50] Meier, S., Schmidt, B., Cremers, C., Basin, D.: The tamarin prover for the symbolic analysis of security protocols. In: Computer Aided Verification: 25th International Conference, CAV 2013, Saint Petersburg, Russia, July 13-19, 2013. Proceedings 25. pp. 696–701. Springer (2013)

[51] OpenSSL-Management-Committee: Openssl: Cryptography and ssl/tls toolkit. In: https://www.openssl.org/.

[52] Park, C., et al.: Doltest: In-depth downlink negative testing framework for lte devices. In: 31st USENIX Security Symposium (USENIX Security) (2022)

[53] Pedersen, T.P.: Non-interactive and information-theoretic secure verifiable secret sharing. In: Annual international cryptology conference. pp. 129–140. Springer (1991)

[54] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM) **56**(6), 1–40 (2009)

[55] Rupprecht, D., et al.: On security research towards future mobile network generations. IEEE Communications Surveys & Tutorials **20**(3), 2518–2542 (2018)

[56] Sacoto Cabrera, E.J.o.: Game theoretical analysis of a multi-mno mvno business model in 5g networks. Electronics **9**(6), 933 (2020)

[57] Schmitt, P., Raghavan, B.: Pretty good phone privacy. In: 30th USENIX Security Symposium (USENIX Security 21). pp. 1737–1754 (2021)

[58] Setty, S., Lee, J.: Quarks: Quadruple-efficient transparent zksnarks. Cryptology ePrint Archive (2020)

[59] Shaik, A., et al.: Practical attacks against privacy and availability in 4g/lte mobile communication systems (2016)

[60] Shin, D.H., et al.: A study of mvno diffusion and market structure in the eu, us, hong kong, and singapore. Telematics and Informatics **24**(2), 86–100 (2007)

[61] Singla, A., Behnia, R., Hussain, S.R., Yavuz, A., Bertino, E.: Look Before You Leap: Secure Connection Bootstrapping for 5G Networks to Defend Against Fake Base-Stations. In: Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security. pp. 501–515. ACM, Virtual Event Hong Kong (May 2021). https://doi.org/10.1145/3433210.3453082, https://dl.acm.org/doi/10.1145/3433210.3453082

[62] Technology Network., O.: Java Cryptography Architecture (JCA) Reference Guide (Sep 2022), https://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html

[63] Vallina-Rodriguez, N., et al.: Beyond the radio: Illuminating the higher layers of mobile networks. In: Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services. pp. 375–387 (2015)

[64] Wang, W., Dumont, F., Niu, N., Horton, G.: Detecting software security vulnerabilities via requirements dependency analysis. IEEE Transactions on Software Engineering **48**(5), 1665–1675 (2022). https://doi.org/10.1109/TSE.2020.3030745

[65] Wang, Y., Zhang, Z., Xie, Y.: {Privacy-Preserving} and {Standard-Compatible} {AKA} Protocol for 5G. pp. 3595–3612 (2021), https://www.usenix.org/conference/usenixsecurity21/presentation/wang-yuchen

[66] Xie, T., et al.: Orion: Zero knowledge proof with linear prover time. In: Annual International Cryptology Conference. pp. 299–328. Springer (2022)

[67] Xie, X., et al.: Zero knowledge proofs from ring-lwe. In: Cryptology and Network Security: 12th International Conference, CANS 2013, Paraty, Brazil, November 20-22. 2013. Proceedings 12. pp. 57–73. Springer (2013)

[68] Yu, C., et al.: Secchecker: Inspecting the security implementation of 5g commercial off-the-shelf (cots) mobile devices. Computers & Security p. 103361 (2023)

[69] Yu, H., Du, C., Xiao, Y., Keromytis, A., Wang, C., Gazda, R., Hou, Y., Lou, W.: AAKA: An Anti-Tracking Cellular Authentication Scheme Leveraging Anonymous Credentials (Jan 2024). https://doi.org/10.14722/ndss.2024.24617

[70] Zhang, Y., Deng, R.H., Bertino, E., Zheng, D.: Robust and Universal Seamless Handover Authentication in 5G HetNets. IEEE Transactions on Dependable and Secure Computing **18**(2), 858–874 (Mar 2021). https://doi.org/10.1109/TDSC.2019.2927664

APPENDIX A: IMPLEMENTATION DETAILS

Here, we provide the implementation details. Based on the testbed setup we mentioned in Section 7, all results here are generated in an **openairinterface environment**. We provide the log screenshot using the sequence of the protocol. We first provide the generation of message M1. As shown in Figure 7, gNB generates M1 in the SystemInformationBlockType1 (SIB1) message and broadcasts it.

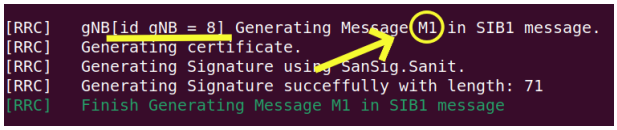
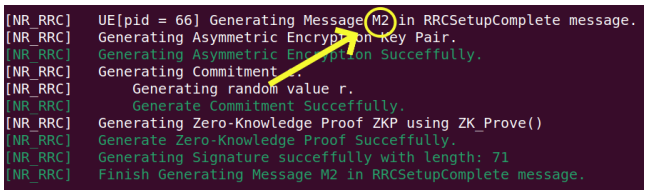


Figure 7: Log From gNB Generating Message 1.

After receiving the broadcast message, UE can authenticate the gNB before connecting to it, which is shown in Figure 8a. After that, UE generates the message M2 in the RRCSetupComplete message, as shown in Figure 8b.

(a) UE Verify M1



(b) UE Generate M2

Figure 8: (a) UE Verify M1, and (a) UE Generate M2.

Upon receiving message M2, gNB first checks the signature from UE and generates the message M3, which is shown in Figure 9. The new message merged into the initial UE message and sent to the core network.

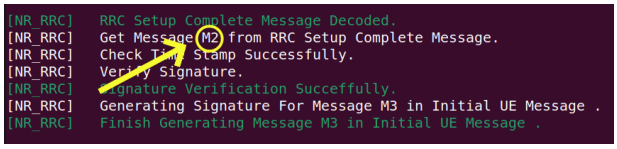


Figure 9: Log From gNB to Generate M3.

As shown in Figure 10, after receiving message M3 from gNB, the core network verifies the gNB signature and the zero-knowledge proof from UE. While the verification is successful, the core network generates the UID, encrypts it using the public key of the UE and sends the message M4 via gNB to UE.

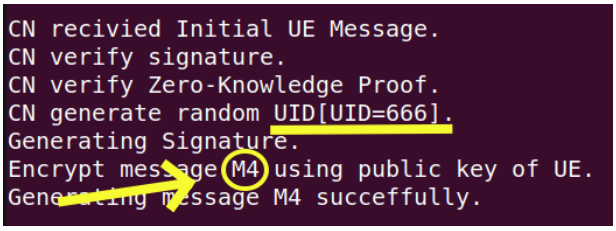


Figure 10: Log From Core Network (CN) to Generate M4.

Figure 11 shows the last step of our protocol. Upon receiving message M4, UE decrypts the message using its own secret key. After verification of the signature of the core network, the UID is retrieved from the core network, and the initial authentication process is finalized.

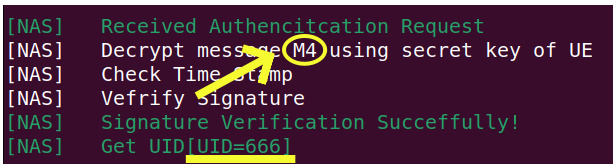


Figure 11: Log From UE to Verify M4.

APPENDIX B: UC VS DOLEV-YAO ADVERSARY MODEL

Here are more detailed reasons for the benefits of the UC model over the Dolev-Yao model:

1. Robust Composability: in the Universal Composability (UC) framework ensures that a security protocol remains secure when combined with other protocols, even in complex, dynamic environments. This feature is crucial because real-world applications often involve multiple interacting protocols, each potentially influencing the others' security properties. The Dolev-Yao model, however, typically examines protocols in isolation. It assumes that the cryptographic primitives are secure and focuses on what happens within a single protocol without external interactions. This isolation can miss critical vulnerabilities that only manifest when protocols operate concurrently or are integrated within larger systems. Thus, while useful for initial protocol design and understanding fundamental vulnerabilities, Dolev-Yao might not adequately predict security failures in more interconnected and realistic scenarios.

2. Adversarial Flexibility: In the UC model, the adversarial capabilities are modeled more comprehensively, including the ability to adapt based on observed interactions, which more closely mirrors potential real-world attacks. The Dolev-Yao model, while useful for basic protocol analysis, operates under more constrained assumptions about adversary capabilities. More precisely, Universal Composability (UC) excels because it simulates adversarial behaviour in a way that mirrors potential real-world tactics. This includes adaptive strategies where the adversary can dynamically adjust their actions based on information gained during protocol execution. It's designed to handle unexpected interactions and coordinated attacks, maintaining security even under such complex

conditions. Dolev-Yao, however, assumes a static adversary bound by initial assumptions about capabilities and unable to adapt to changing circumstances during protocol execution. This model does not account for adversaries learning and adapting, which can leave analyzed protocols vulnerable to more sophisticated, real-world attacks that exploit dynamic or unexpected conditions.

This difference makes UC more suitable for modern cryptographic systems where adaptability and resilience against sophisticated threats are crucial.

3. Ideal vs. Real World Simulation: UC security proofs involve demonstrating that no adversary can distinguish between the ideal functionality (a theoretical, perfectly secure system) and the real protocol implementation. This method provides a high level of assurance that all aspects of protocol security (secrecy, integrity, authentication) are preserved under any operational context, a perspective less emphasized in the Dolev-Yao model.

More precisely, The Dolev-Yao model's approach to adversarial flexibility is limited because it fundamentally views adversaries through a symbolic lens, assuming they can manipulate and intercept communications but cannot break well-established cryptographic primitives. This model does not account for more nuanced, real-world adversarial behaviours such as side-channel attacks, stateful attacks, or dynamic responses to changing protocol states. In practical scenarios, these limitations can hinder the model's ability to fully predict and counter sophisticated or adaptive threats that might exploit specific implementation flaws or emerging vulnerabilities. The UC model, in contrast, allows for a broader range of adversarial capabilities and interactions, reflecting more realistic and complex attack scenarios. However, the Universal Composability (UC) model's strength in adversarial flexibility arises from its comprehensive approach to modelling potential attacker behaviours. Unlike more static models, UC allows for adaptive adversaries who can change their strategies based on observed interactions and outcomes within the protocol execution environment. This dynamic capability is essential for assessing security in realistic scenarios where threats evolve and where interactions between different components can lead to unforeseen vulnerabilities. The UC model's ability to handle such complex, interactive situations makes it particularly robust and suitable for modern cryptographic applications where security needs to be assured even under varied and potentially hostile operational conditions.