ZK-eSIM: A Privacy-Centric Zero-Knowledge Approach for eSIM Provisioning

Liza Ahmad*
University of Sheffield
Sheffield, United Kingdom
lahmad2@sheffield.ac.uk

Quan Shi*
National University of Singapore
Singapore, Singapore
shi.quan@u.nus.edu

Joshua Haworth†
University of Sheffield
Sheffield, United Kingdom
jhaworth1@sheffield.ac.uk

Yilu Dong†
Pennsylvania State University
University Park, PA, United States
yiludong@psu.edu

Prosanta Gope✉
University of Sheffield
Sheffield, United Kingdom
p.gope@sheffield.ac.uk

Behzad Abdolmaleki
University of Sheffield
Sheffield, United Kingdom
behzad.abdolmaleki@sheffield.ac.uk

Syed Rafiul Hussain
Pennsylvania State University
University Park, PA, United States
hussain1@psu.edu

Abstract

GSMA Remote SIM Provisioning (RSP) enables over-the-air delivery of eSIM profiles, but it exposes long-lived identifiers during profile ordering and download. In particular, stable device identifiers (e.g., EID), profile identifiers, and long-lived certificate material enable mobile operators and profile-delivery infrastructure to link provisioning events to the same eUICC and, when combined with account records, to the same subscriber. This undermines subscriber anonymity and enables cross-session tracking. We present *ZK-eSIM*, a privacy-preserving redesign that achieves subscriber anonymity and provisioning-session unlinkability while retaining accountable traceability by exception. ZK-eSIM (i) replaces direct disclosure of device identifiers with a zero-knowledge proof of device validity and eligibility; (ii) enforces session unlinkability through short-lived, one-time pseudonymous credentials and per-session identifiers to prevent cross-session tracking; and (iii) provides privacy-preserving accountable traceability through a jointly authorised escrow mechanism, so that no single entity can unilaterally deanonymise a user. We formalise a multi-entity, honest-but-curious threat model and prove subscriber anonymity and the unlinkability of provisioning sessions under standard cryptographic assumptions. We implement a Java Card applet on a test eUICC to evaluate performance on commodity hardware with a modified LPA and SM-DP+ server. Our experiments quantify end-to-end cryptographic overhead relative to conventional RSP, confirming that *ZK-eSIM* adds only practical overhead, closing a critical privacy gap while preserving deployability within existing GSMA roles and interfaces.

CCS Concepts

• **Security and privacy → Pseudonymity, anonymity and untraceability**; **Privacy-preserving protocols**.

Keywords

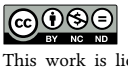
eSIM, privacy, zero-knowledge proofs (ZKP), unlinkability

ACM Reference Format:
Liza Ahmad, Quan Shi, Joshua Haworth, Yilu Dong, Prosanta Gope✉, Behzad Abdolmaleki, and Syed Rafiul Hussain. 2026. ZK-eSIM: A Privacy-Centric Zero-Knowledge Approach for eSIM Provisioning. In *Proceedings of the 2026 ACM SIGSAC Conference on Computer and Communications Security (CCS '26), November 15–19, 2026, The Hague, Netherlands.* ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3830454.3846617

1 Introduction

The embedded SIM (eSIM) is rapidly replacing removable SIM cards as the default method for devices to connect to mobile networks. Unlike physical SIMs, which require manual swapping, an eSIM profile is provisioned digitally and stored on the embedded Universal Integrated Circuit Card (eUICC), a secure chip built directly into smartphones, wearables, and IoT devices. In today's consumer ecosystem, the profile download process follows a standard Remote SIM Provisioning (RSP) workflow defined by the GSM Association (GSMA) [19, 52]. At a high level, the workflow involves the user device, the mobile operator, a server that prepares and stores the eSIM profile (SM-DP+: the Subscription Manager–Data Preparation+), and the GSMA trust infrastructure (Fig. 1). eSIM adoption is now mainstream: GSMA reports that the number of eSIM-capable smartphones in active use increased from 144 million in 2022 to nearly 600 million in 2024 [21], and major vendors are moving toward eSIM-only devices [4]. At this scale, the privacy and security properties of the eSIM provisioning protocol become an urgent concern. In particular, the current provisioning protocol design

*Both authors contributed equally to this research.
†Valuable contribution to the implementation of ZK-eSIM
✉ Corresponding author.



CCS '26, The Hague, Netherlands

ACM ISBN 979-8-4007-2871-6/2026/11
https://doi.org/10.1145/3830454.3846617

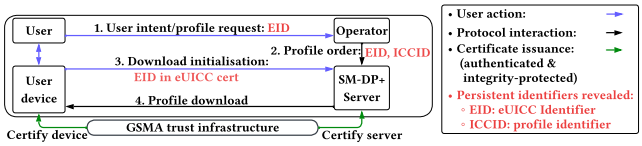


Figure 1: Identifier exposures in the GSMA Consumer Remote SIM Provisioning (RSP) workflow.

may enable tracking of the user's device, location, and data over time, which is at odds with privacy-by-design expectations (e.g., GDPR [34] and GSMA's RSP architectural principles [19]). Reported cases of user data leaks and resulting regulatory fines show that these concerns are not purely theoretical [35, 42].

To comprehend the source of the risk, it is essential to analyse the identification process of user devices during provisioning. Each eSIM-enabled device is equipped with specific, immutable information, including a permanent device identifier referred to as the EID (eUICC Identifier). This information serves as a basis for the provisioning infrastructure to authenticate the device and facilitate the delivery of new profiles. However, due to the reuse of this identifying information across multiple provisioning sessions, various stakeholders within the infrastructure can consistently recognise the same device during its interactions with the system. As a result, different profile downloads, which may be intended to be independent, can be linked back to the same device and associated with the subscriber's identity. The root cause is architectural: conventional RSP operation interlinks *stable identifiers* and exposes them to infrastructure entities (Fig. 1). We identify **three** critical *privacy risks in the conventional RSP protocol*:

•**Identity–EID binding at subscription time.** During profile ordering, the operator and SM-DP+ learn the device's permanent identifier (EID) together with the subscriber's account data, binding the device to a real person and linking future provisioning events for that EID to the same subscriber [20].

• **Linking across multiple downloads.** The current eSIM provisioning protocol reuses the same device identifier for each profile request or download, allowing the operator or profile server to link otherwise separate profiles to the same device, even when installed for different purposes. Over time, this enables a combined view of the user's activity and, in shared backends serving multiple operators, can extend across operators [32].

• **Persistent certificate identifiers.** Even if the EID is hidden during provisioning, long-lived certificate data exchanged during authentication can act as a persistent fingerprint, allowing infrastructure parties to recognise the device across sessions [2, 20].

The above risks are amplified by contemporary provisioning practices, as profile switching is no longer a corner case. The most visible example is the rapid growth of travel eSIM usage (up by 85% in 2025) [41]. Travellers increasingly create short-lived, data-only travel profiles rather than relying on carrier roaming, which remains significantly more expensive in many regions, such as China and India [47, 50]. While the plan is *temporary*, the device's EID is not: it persists across all installations and profile switches. As a result, reseller platforms and SM-DP+ backends operating shared infrastructures can expose stable mappings between the device EID, the issued profile identifier, and the Integrated Circuit Card Identifier (ICCID), across otherwise independent provisioning events [32].

GSMA standards and prior analyses [2, 19, 20] focus on delivery security, confidentiality, and the authenticity of profile downloads, rather than on privacy. The key challenge is therefore not secure transport, but privacy by default with accountability by exception. Our privacy goal is *anonymity and unlinkability during provisioning*: provisioning should not reveal a subscriber's real-world identity or permanent device identifiers, and distinct provisioning sessions should not be linkable to the same device or subscriber. This gap directly motivates our three research questions:

RQ1: *Subscriber anonymity. Can a legitimate eUICC obtain a profile without disclosing any device identifiers (e.g., EID) to the SM-DP+ or the Mobile Network Operator (MNO) during provisioning?*

RQ2: *Provisioning-session unlinkability. Can provisioning sessions remain unlinkable to each other, even across different operators and multi-tenant SM-DP+ platforms, and resist certificate-chain tracking?*

RQ3: *Accountable traceability. Can we provide accountable traceability alongside the anonymity and unlinkability guarantees of RQ1–RQ2, such that on-demand deanonymisation requires joint authorisation and no single entity can unilaterally compromise privacy?*

To answer **RQ1–RQ3**, we present **ZK-eSIM**, a privacy-preserving redesign of the RSP workflow that preserves the existing GSMA roles and interfaces while removing stable identifier exposure from provisioning transcripts (Fig. 2). Our **contributions** are:

• **Anonymous provisioning without persistent device identifiers.** We design ZK-eSIM, an end-to-end provisioning workflow that uses fresh, per-session pseudonyms and zero-knowledge proofs of eUICC legitimacy. The MNO and SM-DP+ learn only session-local handles, not persistent identifiers or long-lived identifying certificate-chain attributes (addressing RQ1).

• **Unlinkability across sessions, operators, and multi-tenant SM-DP+.** ZK-eSIM makes protocol-visible credentials and tokens one-time and eliminates certificate-chain fingerprints by using short-lived, session-scoped authentication material. These mechanisms prevent both application-layer and certificate-based linkage across repeated provisioning events, including those handled by the same SM-DP+ (addressing RQ2).

• **Accountable traceability via joint deanonymisation.** We design a trace mechanism in which identity recovery requires joint authorisation by the MNO and a designated Law Enforcement Authority (LEA), providing accountability *by exception* while routine provisioning remains anonymous and unlinkable (addressing RQ3).

• **Evidence for RQ1–RQ3: formalisation and implementation.** We formalise the above privacy and traceability properties in a multi-entity threat model and prove ZK-eSIM's anonymity, provisioning-session unlinkability, and joint-traceability guarantees under standard cryptographic assumptions (Sec. 8). We implement and evaluate a prototype on commodity smartphones (including a Java Card applet[1]) and *a test eUICC alongside a GSMA-compliant SM-DP+ Server* to demonstrate that ZK-eSIM is practical in real mobile environments (Sec. 9, addressing RQ1–RQ3).

[1]Implementation overview: https://github.com/NaivEPoi/ZK-eSim

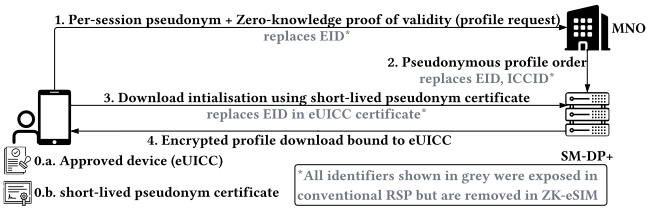


Figure 2: ZK-eSIM provisioning overview. (0.a) An approved eUICC obtains (0.b) a short-lived pseudonym certificate, (1) authenticates to the MNO with a per-session pseudonym and ZK proof, (2) triggers a pseudonymous SM-DP+ order, (3) initialises profile download with the short-lived certificate, and (4) receives an eUICC-bound encrypted profile, without exposing persistent identifiers.

2 Background and Related Work

GSMA Consumer RSP enables a subscriber device to download and install operator profiles on an eUICC, a tamper-resistant secure element integrated into the device. The eUICC stores one or more operator profiles and protects the long-term keys and subscription material contained in those profiles. The MNO or mobile virtual network operator (MVNO) authorises profile issuance, while the SM-DP+ prepares, stores, and delivers the protected profile package to the target eUICC, which installs the profile [19, 20]. The RSP trust model is anchored in the GSMA public-key infrastructure: the eUICC presents a certificate chain rooted in the GSMA trust hierarchy, while the SM-DP+ authenticates as an authorised profile-delivery server. During profile download, these certificates enable the SM-DP+ to verify that it is interacting with an eligible eUICC, allow the device/eUICC to authenticate the delivery server, and bind the protected profile package to the intended target [19, 20]. Several identifiers are central to this workflow. The EID identifies the eUICC. The ICCID identifies an individual profile. The Matching ID provided to the device indicates how to retrieve a pending profile. Finally, the eUICC and SM-DP+ certificates support mutual authentication and profile-package protection under the GSMA trust infrastructure. Fig. 3 shows two consumer RSP provisioning sessions for the same device with different operators. The first session corresponds to a profile ordered from MNO/MVNO 1, while the second models a later, independent order from MNO/MVNO 2, for example, when the user purchases a travel eSIM while abroad. We first describe the functional flow; Sec. 3 then analyses the privacy risks from identifier reuse. The device obtains the eUICC identifier ① and provides it during subscription or profile purchase together with the requested profile type ②. The operator then requests profile preparation for the target eUICC ④. The SM-DP+ reserves a profile identifier ⑤, returns the corresponding profile handle and matching information ⑥, and finalises the order after operator confirmation ⑦–⑧. The operator then delivers activation material to the device ⑨. Using the activation material, the device first obtains eUICC information and fresh challenge material from the eUICC ⑩, then contacts the SM-DP+ to initiate authentication for the download session ⑪, and verifies the SM-DP+ authentication response ⑫–⑬. The eUICC then authenticates itself to the SM-DP+ using signed session data and its certificate chain ⑭. Once the SM-DP+ verifies the eUICC and confirms that the pending profile matches

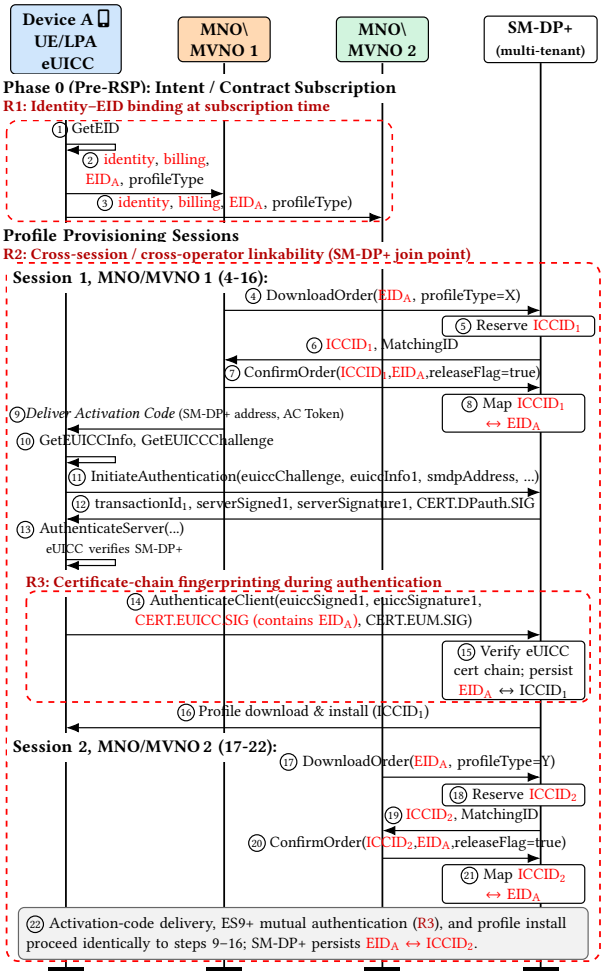


Figure 3: Privacy Risks in Conventional RSP (§3 R1–R3)

the authorised order ⑮, it delivers the protected profile package for installation on the eUICC ⑯. The later provisioning session with MNO/MVNO 2 follows the same pattern (③ and ⑰–㉒), resulting in the provisioning of a new profile on the same eUICC.

In practice, the SM-DP+ may be operated by the MNO itself (e.g., Vodafone Idea [23]) or outsourced to a multi-tenant provider serving multiple operators (e.g., Workz [38]). These deployment choices do not change the provisioning flow in Fig. 3, but they affect which domains observe provisioning state. Existing GSMA specifications [19, 20] focus on correctness, authentication, and secure profile transport, while leaving subscriber anonymity and unlinkability outside the main security goals. A formal analysis by Ahmed *et al.* [2] confirms this security–privacy split: RSP satisfies delivery guarantees, but does not provide anonymity or unlinkability for subscribers. Accountability-oriented proposals, such as the SIM Profile Transparency Protocol proposed by Ahmed *et al.* [3], help detect profile misissuance, but still depend on stable identifiers and therefore do not eliminate linkability at the provisioning layer. Moreover, the problem is not confined to provisioning. Rao *et al.* [40] show that higher-layer functionality built on top of RSP, including device authentication and network access procedures, inherits the same

identifiers and consequently propagates provisioning-time privacy deficits into the operational lifetime of the subscription.

ZK-eSIM composes established cryptographic techniques. Anonymous credentials support unlinkable proofs of eligibility [10, 12], blind signatures enable privacy-preserving credential issuance [13], and pseudonym certificates and identity escrow provide session-based authentication and accountable anonymity [9, 11]. However, these frameworks typically address individual authentication tasks rather than privacy across the complete, stateful RSP workflow, and they do not consider integration with the GSMA trust and operational infrastructure. Existing cellular privacy work instead focuses mainly on protecting subscriber identifiers during 5G network authentication [5, 26], rather than profile provisioning. ZK-eSIM's contribution is therefore the RSP-specific adaptation and end-to-end composition of established techniques to provide privacy across the GSMA Consumer RSP workflow.

3 Privacy Risks in Conventional RSP

Fig. 3 highlights three privacy risks arising from the repeated exposure of stable identifiers throughout the conventional RSP workflow: identity–EID binding at subscription time (**R1**); cross-session and cross-operator linkability during provisioning (**R2**); and certificate-chain fingerprinting during authentication (**R3**). We consider a multi-tenant SM-DP+ deployment, where the same SM-DP+ backend serves multiple MNO/MVNO tenants. The main privacy issue is not just that these identifiers exist, but that they can be reused to connect events that a user would reasonably expect to remain separate.

• **R1:** The earliest risk occurs before profile download begins. During subscription, the operator receives an EID together with Know Your Customer (KYC), account, and billing data, allowing it to associate a real-world subscriber with a specific eUICC. In Phase 0, the Local Profile Assistant (LPA) retrieves the EID via GetEID ① and forwards it during contract subscription with each operator ②, ③. MNO/MVNO 1 or MNO/MVNO 2 therefore learns that EID_A belongs to the subscriber purchasing the profile. This binding has standalone privacy implications. Once an EID is associated with an identity, it can be correlated with other operator-side identifiers and records, including IMEI, IMSI/SUPI, billing data, and account metadata. In multi-device settings, the same process can also reveal a subscriber's device portfolio: for example, if a phone and a smartwatch are provisioned under the same account, their distinct EIDs may become linked to the same user. Such records also create an insider-abuse surface: an adversary with access to operator systems could enumerate the EIDs associated with a target and use this information to facilitate fraudulent profile replacement or SIM-swap-style account takeover [27].

•**R2:** The primary privacy risk arises when the same long-lived EID is reused across otherwise independent profile downloads. Each download creates a fresh ICCID, but the backend observes the same device identifier: the ICCID identifies the profile being issued, whereas the EID identifies the eUICC itself. In the first provisioning session, MNO/MVNO 1 sends DownloadOrder to the SM-DP+ ④. The SM-DP+ reserves ICCID_1 for that order ⑤ and returns ICCID_1 together with a MatchingID ⑥, which later acts as a handle for retrieving the reserved profile. After receiving ConfirmOrder ⑦, the SM-DP+ can persist the mapping $\mathsf{ICCID}_1 \leftrightarrow \mathsf{EID}_\mathrm{A}$ ⑧. The activation code delivered to the device ⑨ then tells the LPA which SM-DP+ to contact and which reserved profile to claim. The second session through MNO/MVNO 2 repeats this pattern with EID_A reappearing in ⑰–㉒, allowing the SM-DP+ to persist a second mapping $\mathsf{ICCID}_2 \leftrightarrow \mathsf{EID}_\mathrm{A}$. The SM-DP+ can therefore infer that both profile downloads correspond to the same eUICC, even though they involve different profiles and different operators. This is already a privacy risk even without knowing the subscriber's civil identity: the backend can link purchases, destinations, provisioning times, profile types, and operator choices to the same device. In a multi-tenant SM-DP+ deployment, the risk becomes cross-operator: a single backend may observe provisioning requests from multiple operators and join them using the EID. When combined with **R1**, the same mechanism becomes subscriber-level tracking by the operator, because the linked device activity can be associated with a real-world user.

• **R3:** A natural mitigation would be to minimise or hide the EID in the profile-ordering flow. However, the authentication phase can reintroduce stable device-linked material. After the device receives the activation code in the first provisioning session, it obtains local eUICC information and a challenge ⑩ and invokes InitiateAuthentication with the SM-DP+ ⑪. The SM-DP+ responds with transaction-specific authentication data and its certificate material ⑫, and AuthenticateServer lets the eUICC verify that it is communicating with the legitimate SM-DP+ ⑬. The reciprocal step is more privacy-sensitive. In AuthenticateClient, the eUICC proves itself to the SM-DP+ by presenting signed authentication data and its certificate chain ⑭. This chain includes CERT.EUICC.SIG, containing stable eUICC identity material associated with EID_A. The SM-DP+ then verifies the chain and can persist the resulting association between the eUICC identity and the issued profile, here $\mathsf{EID}_\mathrm{A} \leftrightarrow \mathsf{ICCID}_1$ ⑮, before the profile is downloaded and installed ⑯. In the second provisioning session, the same activation-code delivery, mutual authentication, certificate-chain exposure, and profile installation are compressed into ㉒; the privacy consequence is identical, because the SM-DP+ can again associate the same eUICC identity with ICCID_2. R3 is not redundant: it explains why R2 cannot be removed by application-layer identifier minimisation alone. If the SM-DP+ can still recognise the same eUICC during mutual authentication, then hiding the EID in DownloadOrder does not provide unlinkability. Because **R1** may already have tied EID_A to a subscriber's real identity, the certificate material in **R3** can carry user-identifying meaning into every authenticated provisioning session [2, 20].

The severity of the privacy risks depends on who operates the SM-DP+. In a *primary* deployment, where the home MNO runs the SM-DP+ end-to-end, identifier reuse remains within the operator's existing administrative boundary; compared with physical SIM issuance, the additional privacy exposure is mainly to insider misuse or data breaches. In a *non-primary* deployment, by contrast, the SM-DP+ is a third-party or multi-tenant service distinct from the home MNO, creating a new observation point that can correlate provisioning events across operators without user visibility. In our work, we focus on the non-primary deployment model where ZK-eSIM is more valuable.

This non-primary setting is already common in practice. Motallebighomi et al. [32] conduct a large-scale empirical analysis of

travel eSIM products and reseller platforms. They find that "travel eSIM" connectivity is frequently implemented via home-routed roaming, exporting user traffic and associated metadata to third-country infrastructures, and further show that reseller dashboards can expose sensitive provisioning state, including device identifiers and EID-to-ICCID mappings on commercial SM-DP+ backends [32]. Current deployment practices amplify this risk. In practice, SM-DP+ platforms are often multi-tenant services serving hundreds of operators; for example, Workz advertises a cloud platform used by over 100 operators worldwide [38]. This consolidation turns the SM-DP+ into a high-leverage observation point: a single backend can observe when the same device provisions profiles from multiple operators and at multiple times, enabling cross-operator linkage, coarse travel-history inference, and user association over time. This is a serious threat if provisioning logs are monetised, subpoenaed, or leaked [32]. The threat surface is especially misaligned with the user's intent in adopting travel eSIMs. Users increasingly buy travel eSIMs to reduce dependence on home-carrier roaming arrangements and to obtain "local-like" connectivity on demand. Yet the provisioning layer reuses identifiers precisely where users expect ephemerality. The EU "Roam Like at Home" regime does not cover all countries (e.g., Switzerland), pushing subscribers into expensive day passes, and UK roaming passes (£2–£6/day) often exceed the cost of a travel eSIM (e.g., £7.21 for 5GB in Italy) [15, 48]. The market conditions that drive short-lived travel profile adoption also intensify repeated provisioning events, making **R2** a practical and recurring privacy problem.

Limitations of Strawman Solutions. We consider two alternatives that might appear to address these risks. One could attempt to address these risks at the hardware layer by rotating or randomising the EID itself (e.g., via virtual identifiers). However, the EID is a factory-programmed, device-lifetime identifier embedded in the eUICC; modifying it would require a hardware-rooted redesign that is difficult to deploy at ecosystem scale. A user might attempt to achieve unlinkability by provisioning multiple profiles in advance and rotating among them across sessions. This is infeasible: eUICCs support only a limited number of profiles, and each profile remains bound to the same EID during download, reintroducing linkability. ZK-eSIM therefore keeps the EID and profile-storage model intact and shifts privacy protection entirely to the protocol layer.

4 System Model, Threat Model, and Design Goals

We first define the system entities, followed by adversarial capabilities, and security and privacy goals of ZK-eSIM.

• **System Model**: *ZK-eSIM* consists of five principal entities: UE, MNO, SM-DP+, Pseudonym Certificate Authority (PCA), and LEA. It also relies on the existing GSMA PKI as external trust infrastructure. Below, we describe the roles and capabilities of each entity.

- User Equipment (UE): consists of the LPA and a tamper-resistant eUICC. The LPA mediates profile lifecycle operations, while the eUICC stores profiles and long-term secrets and executes security-sensitive cryptographic operations. We assume the standard GSMA hardware protections and model the LPA–eUICC interface as a local authenticated channel preserving command integrity and response authenticity.

- Mobile Network Operator (MNO): provides the subscription service, performs eligibility and subscriber-registration checks, and authorises profile orders with the SM-DP+. Our model also covers MVNO deployments, where the MVNO assumes the operator role while using a host MNO's access infrastructure. End-to-end encryption keeps subscriber data inaccessible to the host MNO.

- Subscription Manager–Data Preparation+ (SM-DP+) : prepares, stores, and delivers operator profiles to authorised eUICCs. It receives profile orders from the MNO and performs authenticated and confidential profile delivery. We consider both dedicated single-tenant and shared multi-tenant SM-DP+ platforms.

- Pseudonym Certificate Authority (PCA): is a new entity introduced to address **R3** (Sec. 3). It issues short-lived, per-session pseudonym certificates to eligible eUICCs. The PCA operates as an independent sub-CA under a GSMA Certificate Issuer (CI), allowing each pseudonym certificate to chain to a trust anchor already recognised by UEs and SM-DP+ servers. The certificates contain only the metadata required for validation, such as their validity period.

- Law Enforcement Authority (LEA): is an external governance entity involved only in de-pseudonymisation under lawful process. It cannot unilaterally deanonymise a subscriber; de-pseudonymisation requires joint cooperation with an MNO under applicable legal policies, reflecting statutory lawful-interception obligations on service providers [24]. For clarity, we model the LEA as a single logical entity. Alternatively, this role may be instantiated by a threshold of independent authorities [14, 45].

- Communication and Setup Assumptions: All inter-entity communications use TLS: device-facing channels are server-authenticated, while the MNO–SM-DP+ channel is mutually authenticated. All entities additionally have read-only access to a public CRS generated either through an auditable multi-party setup with at least one honest contributor or through a trusted setup whose trapdoor is deleted [7, 30].

Deployment Models and Trust Assumptions. ZK-eSIM operates among mutually untrusted parties; each entity follows the protocol correctly but may exploit its view. Additionally, ZK-eSIM relies on non-collusion, a practical assumption driven by strict data protection regulations (e.g., GDPR [34]) across multi-operator architectures. This aligns with existing privacy-preserving cellular systems (e.g., PGPP [43], LOCA [29], PGUS [51], LPG [46]). In non-primary deployments (which we adopt in ZK-eSIM), the MNO and SM-DP+ are operated by different organisations, and the PCA may be operated independently, either by the GSMA or by a GSMA-authorised sub-CA [18, 38]. Keeping these records separate reduces data-protection risk and supports GSMA requirements. In primary deployments where an MNO operates its own SM-DP+, we consider them as colluding. **Appendix C** provides an analysis of the privacy degradation in this case and in all the remaining pairwise and full-collusion cases.

• **Threat Model**: Broadly, we consider two classes of adversaries.

- $\mathcal{A}_1$ (Network-level adversary): We adopt the standard Dolev-Yao adversary [16], which fully controls the communication channels between entities. The adversary can intercept, delay, replay, modify, or inject messages. The cryptographic keys and the GSMA trust anchors are assumed to remain uncompromised. eUICC-resident

long-term keys remain hardware-protected; physical extraction and related hardware attacks are out of scope. This adversary captures risks such as: Man-in-the-middle attacks, passive surveillance, replay attacks, and rogue base stations. Certain threats are out of scope, such as denial-of-service attacks and global adversaries that can correlate cross-domain traffic.

- $\mathcal{A}_2$ *(Honest-but-curious infrastructure adversary):* We model the MNO, SM-DP+, and PCA as honest-but-curious entities. Each executes its prescribed protocol operations correctly but may analyse its entire legitimate view to deanonymise a subscriber, link multiple provisioning sessions, or derive a stable device identifier. These entities act independently under the non-collusion assumption defined above. We next specify the legitimate view of each entity and the functions for which it is trusted:

∘**MNO:** observes the subscriber identity, EID, and billing information obtained through the independent onboarding process, together with the provisioning requests, profile orders, and settlement records that it handles. It is trusted to perform eligibility and subscriber-registration checks, authorise profile orders, execute its prescribed provisioning operations, and participate correctly in authorised de-pseudonymisation. However, because of its honest but curious nature, an MNO may attempt to correlate its onboarding records with provisioning sessions or to link multiple sessions belonging to the same subscriber.

∘**SM-DP+:** observes profile orders, session-specific authorisation and authentication material, and its profile-delivery and settlement records. It is trusted to prepare and deliver the correct profile, verify the MNO's authorisation material, and enforce one-time token use, but not to refrain from attempting to correlate different provisioning sessions. While conforming to protocol behaviour, the SM-DP+ may attempt to track users or misreport transactions (e.g., inflate the number of provisioned profiles). Settlement integrity is enforced through signed, single-use provisioning tokens and cryptographically bound settlement records, preventing both over-billing and under-reporting.

∘**PCA:** observes each pseudonym-certificate request, the accompanying eligibility proof, and the short-lived certificate it issues. It is trusted to validate requests, issue certificates only to eligible eUICCs, and protect its signing key, but not to refrain from attempting to link repeated certificate requests.

Under the assumptions above, server-authenticated TLS neutralises $\mathcal{A}_1$ at the channel level; the privacy violations we study therefore arise under $\mathcal{A}_2$ and depend on *endpoint-visible* stable identifiers that enable linkability.

• **Design Goals of ZK-eSIM**: ZK-eSIM preserves RSP's delivery security while ensuring that stable device identifiers are not exposed during provisioning, sessions are unlinkable by default, and identity recovery is possible only via jointly authorised tracing. Our design goals (DG1–DG6) are derived from adversarial threats identified in Sec. 4. We categorise them into two groups: *baseline security goals* (DG1, DG3), which are retained from RSP, and *novel privacy and accountability goals* (DG2, DG4–DG6).

- DG1: Secure Communication. For each communicating pair (e.g., UE and SM-DP+), the protocol ensures confidentiality, integrity, entity authentication, and replay protection against a Dolev-Yao adversary controlling the network.

Table 1: Conventional Consumer RSP vs. ZK-eSIM

Design Goal	Conventional RSP	ZK-eSIM
DG1: Secure Communication	✓	✓
DG2: Unlinkable Mutual Authentication	✗	✓
DG3: Profile Confidentiality & Integrity	✓	✓
DG4: Subscriber Anonymity (at Provisioning)	✗	✓
DG5: Provisioning-Session Unlinkability	✗	✓
DG6: Privacy-Centric Accountable Traceability (LEA-mediated)	✗	✓

- DG2: Unlinkable Mutual Authentication. Mutual authentication must verify endpoints without revealing persistent device identifiers or enabling linkage across sessions. In ZK-eSIM, the UE proves eligibility in zero knowledge and authenticates using short-lived pseudonym certificates that chain to the GSMA CI. Thus, authentication remains verifiable under the GSMA PKI hierarchy while unlinkable to the subscriber's long-term identity.

- DG3: Profile Confidentiality and Integrity. The eSIM profile must be delivered unmodified and confidentially to the eUICC. ZK-eSIM maintains the guarantee by protecting against honest-but-curious infrastructure through authenticated key exchange, encryption, and end-to-end integrity checks. Each profile package is cryptographically bound to the target via an eUICC-held key, so installation on any other device is invalid.

- DG4: Subscriber Anonymity (at provisioning). No party, internal (MNO, SM-DP+) or external (network adversary), should learn a subscriber's real-world identity or permanent device identifiers during provisioning. ZK-eSIM achieves this by replacing stable identifiers with ephemeral pseudonyms and unlinkable proofs.

- DG5: Provisioning-Session Unlinkability. Distinct provisioning sessions must not be linkable to the same subscriber, even when initiated on the same device or across different operators. This prevents tracking and behavioural profiling by MNOs or hosted SM-DP+ platforms. ZK-eSIM enforces unlinkability via Pseudorandom Function (PRF)-generated pseudonyms and ZKPs, across all phases from profile request to installation.

- DG6: Accountable Traceability. Strong anonymity and unlinkability must coexist with lawful accountability. ZK-eSIM ensures that deanonymisation requires collaboration between the LEA and an MNO (holding the relevant provisioning record) under legally authorised conditions. No single entity has the unilateral power to deanonymise, ensuring both regulatory compliance and protection against unjustified surveillance.

Table 1 illustrates that conventional GSMA Consumer RSP meets DG1 and DG3 but falls short on DG2 and DG4–DG6. This gap motivates our work, ZK-eSIM, which is designed to close these specific shortcomings.

5 High-level Overview of ZK-eSIM

This section provides a high-level overview of the *ZK-eSIM* workflow (Fig. 4). ZK-eSIM redesigns the GSMA Consumer RSP flow so that, during provisioning, no non-colluding provisioning party can link a provisioning session to both the subscriber record and the eUICC's long-term identifier. The core idea is to replace cleartext stable identifiers with an unlinkable eligibility credential, a fresh per-session pseudonym certificate, a challenge-derived pseudonym and escrowed EID with a ZKP of correct formation, and one-time

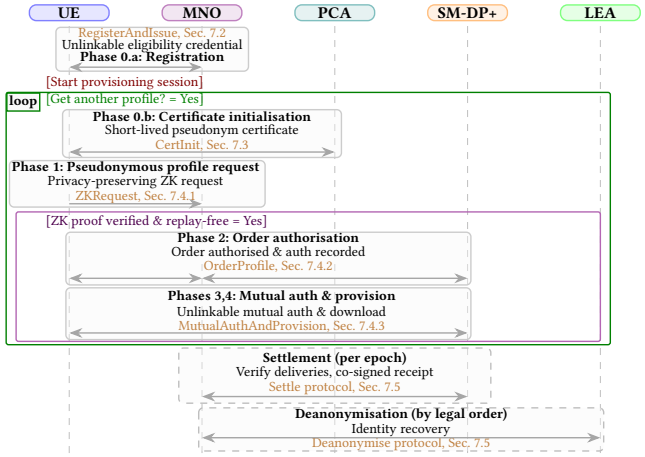


Figure 4: High-level phase view of the ZK-eSIM protocol

authorisation material bound to the session context. Lawful subscriber resolution remains possible, but requires joint LEA–MNO cooperation. The phases of ZK-eSIM are:

Phase 0.a - Registration and Eligibility : The user enrols once with the MNO and obtains an unlinkable eligibility credential bound to the same eUICC that will later generate per-session keys.

Phase 0.b – Certificate Initialisation: Before each provisioning session, the eUICC derives a fresh ephemeral key pair and obtains from the PCA a short-lived pseudonym certificate binding the corresponding public key to a bounded validity window.

Phase 1 – Pseudonymous Profile Request: The UE sends to the MNO a privacy-preserving request containing a fresh challenge-derived per-session pseudonym, an escrow ciphertext of the EID, a pseudonym certificate, and a ZKP that the pseudonym, escrow, and previously issued eligibility credential are bound to the same eUICC and are well formed. The MNO verifies the proof, checks challenge freshness, and rejects replays without learning a stable identifier.

Phases 2–4 - Order authorisation, mutual authentication, and provisioning : Upon successful verification, the MNO computes a hashed pseudonym and places an order at the SM-DP+ using only that hashed pseudonym. It also returns to the UE an authorisation bundle consisting of an MNO-issued authorisation credential and a single-use token, both bound to the authorised hashed pseudonym and to the hash of the presented pseudonym certificate. The UE then initiates a fresh provisioning session with the SM-DP+. The SM-DP+ verifies that the hashed pseudonym was authorised, recomputes the certificate hash, checks the authorisation credential and the one-time token, and enforces one-time token use. Only after these checks do the parties complete mutual authentication and an authenticated key exchange, after which the SM-DP+ delivers the encrypted profile for installation on the eUICC. If the user requests another profile, the loop repeats from Phase 0.b with fresh session credentials.

Supporting Procedure - Settlement: Operational reconciliation runs per epoch between the MNO and the SM-DP+. Each side maintains a log: the MNO over issued tokens, the SM-DP+ over redeemed tokens. The SM-DP+ proves that each redeemed token corresponds to a prior authorisation; both parties co-sign a settlement receipt. No persistent identifier appears in settlement.

Supporting Procedure - Deanonymisation by Exception: Under a valid legal order, the LEA requests the in-scope escrow ciphertext from the MNO, decrypts it to recover the EID, and returns the result to the MNO for subscriber resolution. Thus, escrow decryption and subscriber resolution are separated, and subscriber deanonymisation requires joint LEA–MNO cooperation.

Phase 0.a establishes eligibility once, decoupled from provisioning. For each new profile, Phases 0.b–4 use a fresh certificate, pseudonym, and one-time authorisation bundle. Under the paper's threat model, this yields provisioning-session unlinkability for the MNO and SM-DP+ views while preserving operational settlement and lawful traceability by exception. Collectively, these phases realise design goals DG1–DG6 (Sec. 4).

6 Preliminaries for ZK-eSIM

To keep the paper self-contained, we recall the cryptographic primitives used in our construction. Let $\lambda \in \mathbb{N}$ denote the security parameter and 1^λ for its unary encoding. For a finite set S, $x \leftarrow\!\$\ S$ denotes a uniform sample from S. For a (probabilistic) algorithm A, we write $y \leftarrow A(x; r)$ for the result of running A on input x using randomness r; when r is omitted, probabilities are taken over all internal randomness. We use $\{0,1\}^*$ for bitstrings, and $a \leftarrow b$ to denote assignment. All algorithms are probabilistic polynomial-time (PPT) unless stated otherwise. A function $\mathsf{negl}(\lambda)$ is *negligible* if for every $k \in \mathbb{N}$ there exists λ_0 such that for all $\lambda \geq \lambda_0$, $\mathsf{negl}(\lambda) \leq \lambda^{-k}$. Additional preliminary information is available in the **extended version** [1].

6.1 Non-Interactive Zero-Knowledge

A non-interactive zero-knowledge argument (NIZK) allows a prover P to convince a verifier V of a statement x without leaking anything beyond its validity, using a single message and CRS [8]. We consider an NP-relation $R \subseteq \{0,1\}^* \times \{0,1\}^*$ where $(x, w) \in R$ means witness w proves statement x; the corresponding language is $L_R := \{\, x \mid \exists w : (x, w) \in R \,\}$. A NIZK for R consists of PPT algorithms $(\mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify})$ and simulator $S = (S_1, S_2)$ with the following interfaces: $\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda)$ outputs a common reference string; $\pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$ generates a proof for statement x with witness w (output $\perp$ if $(x, w) \notin R$); $b \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$ verifies proof π for statement x, returning $b \in \{0,1\}$; $(\mathsf{crs}, \mathsf{td}) \leftarrow S_1(1^\lambda)$ generates a simulated CRS (computationally indistinguishable from real setup) with trapdoor td; $\pi \leftarrow S_2(\mathsf{td}, x)$ generates a simulated proof using the trapdoor. The NIZK satisfies completeness, zero-knowledge, and knowledge soundness.

6.2 ZK-eSIM: Definitions

ZK-eSIM is a primitive that enables a user holding an eligible eUICC to be authorised by the MNO and obtain an eSIM profile from SM-DP+ such that the correctness of authorisation and provisioning is verifiable, while the issuance and use phases remain unlinkable and the user's identifying information is hidden. We formalise *ZK-eSIM* as a privacy-preserving provisioning scheme that achieves anonymity at provisioning, multi-session unlinkability, and accountable traceability. The system is executed by the entities

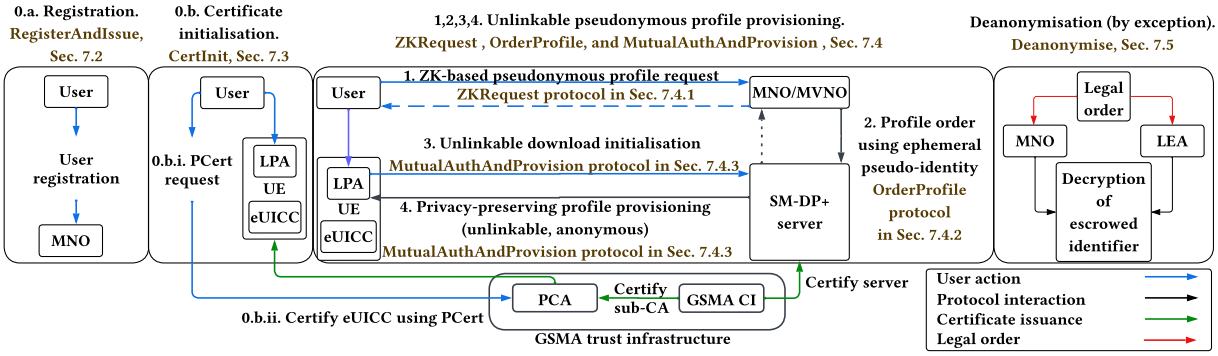


Figure 5: System architecture and protocol phases of ZK-eSIM.

defined in Sec. 4. Let pp denote the *public* parameters output by Setup and shared by all algorithms.

Definition 1 (ZK-eSIM). *A ZK-eSIM scheme is a tuple of PPT algorithms and protocols* $\Pi = (\text{Setup}, \text{RegisterAndIssue}, \text{CertInit}, \text{ZKRequest}, \text{OrderProfile}, \text{MutualAuthAndProvision}, \text{Settle}, \text{Deanonymise})$, *with* pp *as implicit input. Correctness requires that, for any honest execution with fresh session randomness, U obtains the intended profile, the corresponding one-time token is recorded exactly once, and the MNO and SM-DP+ obtain consistent logs and receipts.*

- $(\text{pp}, L_{\text{auth}}, L_{\text{spent}}, sk_b) \leftarrow \text{Setup}(1^\lambda)$: the setup algorithm outputs pp, initialises empty *per-entity* logs $L_{\text{auth}}, L_{\text{spent}} = \emptyset$ (kept private by their respective owners MNO and SM-DP+), and samples a per-card binding secret sk_b *kept on eUICC and never disclosed.*
- $(\sigma_{\text{EID}}, C_{\text{EID}}, r_b) \leftarrow \text{RegisterAndIssue}(\text{pp};\ \text{EID}, sk_b)$: the one-time registration protocol (Fig. 6) between U and MNO takes (EID, sk_b), then returns to U an unlinkable eligibility credential σ_{EID}, commitment C_{EID} and randomness r_b. sk_b, r_b are local to U.
- $(\text{PCert}_U, sk_U, r_{\text{seed}}) \leftarrow \text{CertInit}(\text{pp};\ \sigma_{\text{EID}}, \text{EID}, sk_b)$: the certificate initialisation protocol (Fig. 7) between U and PCA samples a fresh session key pair (sk_U, pk_U), issues to U a short-lived pseudonym certificate PCert_U on pk_U, and returns fresh session randomness r_{seed}. The values sk_b, sk_U, EID are used only locally.
- $(x, \text{PCert}_U, \pi_{\text{req}}, \text{nonce}_{\text{MNO}}) \leftarrow \text{ZKRequest}(\text{pp}; \sigma_{\text{EID}}, sk_b, \text{EID}, r_{\text{seed}}, \text{PCert}_U)$: the request protocol (Fig. 8) run between U and MNO takes $(\sigma_{\text{EID}}, sk_b, \text{EID}, r_{\text{seed}}, \text{PCert}_U)$ as input (where U uses sk_b, r_{seed} locally, never transmitted), then outputs the statement $x = (pk_{\text{MNO}}, pk_{\text{LEA}}, pk_U, \text{nonce}, \text{pid}, \text{EncEid}), \text{PCert}_U, \pi_{\text{req}}$, and $\text{nonce}_{\text{MNO}}$ (stored by MNO), attesting correctness in zero knowledge and same-eUICC binding without revealing EID.
- $(T_i, \sigma_{\text{cred}}, \text{Hpid}, \pi_{\text{inc}}, \text{root}_{\text{auth}}, \sigma^{\text{root}}_{\text{MNO}}) \leftarrow \text{OrderProfile}(\text{pp}; x, \pi_{\text{req}}, \text{PCert}_U, \text{nonce}_{\text{MNO}}, L_{\text{auth}})$: the order-side algorithm (Fig. 8) run by MNO takes x as input, accepts if the certificate chain and the request proof verify and the nonce matches, then appends the hashed pseudonym Hpid to L_{auth} (which is stored privately by MNO), places a DownloadOrder for Hpid at SM-DP+, and returns to U the authorisation credential σ_{cred}, the one-time token T_i, and the inclusion proof π_{inc} together with the accumulator root $\text{root}_{\text{auth}}$ and its MNO signature $\sigma^{\text{root}}_{\text{MNO}}$.
- $\text{status} \leftarrow \text{MutualAuthAndProvision}(\text{pp};\ \text{PCert}_U, sk_U, \text{Hpid}, T_i, \sigma_{\text{cred}}, \pi_{\text{inc}}, \text{root}_{\text{auth}}, \sigma^{\text{root}}_{\text{MNO}})$: the download-side protocol (Fig. 9) between U and SM-DP+ takes the MNO-issued artifacts together with (PCert_U, sk_U), performs mutual authentication, verifies $\text{Hpid} \in L_{\text{auth}}$ and the bindings of σ_{cred} and T_i, enforces short-lived certificate use and no-double-spend by inserting T_i into L_{spent} on accept, delivers and installs the profile on the eUICC, and outputs a single acceptance flag status $\in \{\text{OK}, \perp\}$.
- $\text{SR}_\tau \leftarrow \text{Settle}(\text{pp};\ \tau, L_{\text{spent}}, L_{\text{auth}}, \text{TokSet}_\tau)$: the settlement protocol for epoch τ between MNO and SM-DP+ reconciles tokens in TokSet_τ against the spent ledger $L_{\text{spent}}[\tau]$ and the authorisation ledger $L_{\text{auth}}[\tau]$, computes a tariff-based settlement amount, and outputs a cross-signed settlement receipt SR_τ.
- $(\text{EID}', \pi)/\perp \leftarrow \text{Deanonymise}(\text{pp}; \text{EncEid}, \text{warrant}, sk_{\text{LEA}})$: the accountable de-anonymisation protocol is executed between LEA (holding sk_{LEA}) and MNO (holding EncEid). Given a valid warrant defining a lawful scope S, LEA decrypts the escrowed identity and returns (EID', π) to MNO, where π proves correct decryption. MNO verifies π and resolves EID′ to the subscriber's KYC record, or outputs $\perp$ if any policy or verification check fails.

ZK-eSIM achieves provisioning-session unlinkability and unforgeability; **refer to Sec. 8 for formal security analysis.**

7 ZK-eSIM Construction

We present the end-to-end construction (Fig. 5), using the cryptographic interfaces of Sec. 6.2. The remainder of this section consists of: 1. **System Setup**, 2. **Registration**, 3. **Certificate Initialisation**, 4. **Unlinkable Pseudonymous Profile Provisioning**, and 5. **Supporting Procedures**. Complete pseudocode appears in the **extended version [1]**.

1. System Setup: $\text{Setup}(1^\lambda)$ initialises the public parameters, cryptographic interfaces and operational anchors. It also initialises per-entity logs, and seals a per-card binding secret sk_b inside each eUICC. The MNO creates an append-only authorisation log: L_{auth} indexed by hashed subscriber pseudonyms Hpid. SM-DP$^+$ initialises L_{spent}, an append-only record of one-time tokens used during provisioning to prevent reuse. Both logs are commitment- backed: each owner signs the current log digest, and short inclusion proofs can be verified against the signed root.

2. Registration (Phase 0.a): Eligibility and Accountability Setup (UE ↔ MNO) : The purpose of this phase (Fig. 6) is not to hide the user's identity from the issuer at registration; rather,

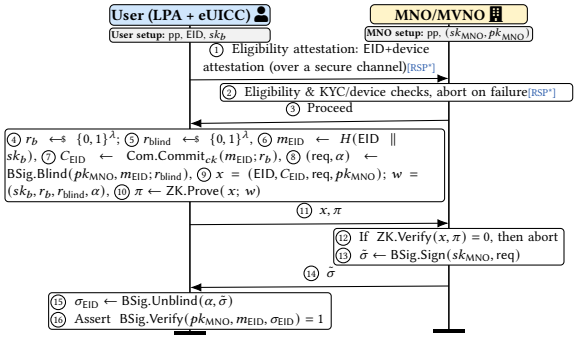


Figure 6: *RegisterAndIssue* Protocol: User registration with MNO/MVNO over a secure channel to obtain anonymous credential σ_{EID}. [RSP*] denotes steps that follow the RSP flow with modified semantics. Unmarked steps are new.

it is to issue a blind, device-bound eligibility credential that can later be used pseudonymously. The flow proceeds as follows: over a secure channel, the user presents EID for eligibility/KYC and the MNO authorises continuation. Next, the eUICC samples fresh randomness r_b and r_{blind}, derives a device-bound digest m_{EID} from (EID, sk_b), computes the commitment C_{EID} using r_b, and constructs a blinded signing request req and (locally) derives the unblinding factor α for later use. Here sk_b is the device-sealed bootstrap secret (unique to and never leaving the secure element), used only to bind issuance to this eUICC via m_{EID}. The zero-knowledge step attests that the instance satisfies the relation $\mathcal{R}_{issue}$: the prover knows sk_b, r_b, r_{blind} (and a corresponding α) such that (i) m_{EID} is correctly derived from EID and sk_b; (ii) C_{EID} is a valid commitment to that m_{EID} under randomness r_b; and (iii) req is a well-formed blinded request under pk_{MNO} for the same m_{EID}, while revealing none of these values. The statement x and ZKP π are then transmitted; the MNO verifies π and, if valid, signs the blinded request and returns $\tilde{\sigma}$. Unblinding with α yields σ_{EID} (a blind-signed, device-bound token that can be instantiated as an anonymous credential [28] (e.g., BBS+)), which the eUICC verifies under pk_{MNO}, completing issuance of (σ_{EID}, C_{EID}).

3. Certificate Initialisation (Phase 0.b): Pseudonymous Key Binding (UE ↔ PCA) : This phase (Fig. 7) converts the eligibility credential into a fresh, certified pseudonymous certificate. The eUICC derives a fresh, hardware-bound public key and proves to the PCA, in zero knowledge, that this key belongs to the same eligibility-checked device. To preserve multi-session unlinkability, this phase is executed freshly for each provisioning session. The eUICC generates a seed r_{seed}, derives an ephemeral keypair (sk_U, pk_U) from sk_b and r_{seed}, and produces a ZKP π_{bind} for the relation $\mathcal{R}_{bind}$: the prover knows $sk_b, r_{seed}, EID, \sigma_{EID}$ such that (i) pk_U is correctly derived within the same eUICC anchored by sk_b, (ii) a hash-based binding m_{EID} is consistent with (EID, sk_b), and (iii) the eligibility credential σ_{EID} verifies under pk_{MNO}, all without disclosing these values. The user then transmits x and π_{bind} to the PCA; upon successful verification, the PCA issues and returns a time-bounded certificate $PCert_U$ over pk_U.

4. Unlinkable Pseudonymous Profile Provisioning (Phases 1,2,3,4) : This section composes Phases 1–4 into an unlinkable provisioning workflow. Our goals are to: (i) prove eligibility in zero

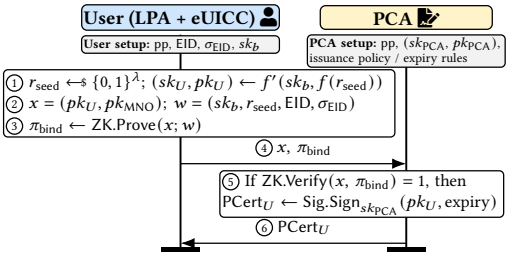


Figure 7: *CertInit* Protocol: Pseudonym Certificate Initialisation.

knowledge; (ii) let the operator place an order without disclosing the long-lived EID at order time; (iii) bind download authorisation to a fresh session and a one-time release token; and (iv) deliver and install the profile without introducing new cross-session linkers. We follow the GSMA Activation Code control flow with modified semantics: the operator-generated Matching ID / activation token is instantiated with the per-session pseudonym Hpid. This preserves the control-flow shape while replacing the long-lived device identifier with a fresh, session-scoped value.

- *Phase 1: ZK-based pseudonymous profile request (U ↔ MNO)*: This phase (Fig. 8) turns an eligibility-certified, hardware-rooted device into a pseudonymous requester over server-authenticated TLS. The MNO samples a fresh challenge nonce and sends it, and stores it as $nonce_{MNO}$. Inside the eUICC, the device derives a per-session pseudonym pid from sk_b and nonce, and in parallel creates an escrow ciphertext EncEid of the same EID using LEA pk_{LEA} and fresh randomness r. It then forms the statement x and witness w and produces a ZKP π_{req} for the relation $\mathcal{R}_{req}$, asserting knowledge of $(EID, sk_b, r_{seed}, \sigma_{EID}, r)$ such that: (i) the certified key pk_U corresponds to the same eUICC; (ii) EID is bound to sk_b; (iii) the eligibility credential σ_{EID} verifies under pk_{MNO}; (iv) pid is correctly derived from $(sk_b, nonce)$ and EID; and (v) EncEid encrypts that same EID under pk_{LEA}, without revealing any of these values. The user sends $(x, PCert_U, \pi_{req})$ and submits it to the MNO over TLS. Here sk_b is the device-sealed bootstrap secret, never exported from the secure element. The MNO learns only a pseudonym and a verifiable proof of hardware continuity and eligibility: unlinkability across sessions follows from the fresh nonce, while accountability is preserved via the stored mapping Hpid/EncEid and escrow to pk_{LEA}; thus an eligible device can authenticate pseudonymously yet remain trace-ready. To keep the core protocol minimal, we treat profile selection and payment as a separate step carried over the same authenticated channel. Concretely, the client sends a context-bound, non-replayable, anonymous proof of payment attesting that the correct tariff was paid to the MNO while revealing nothing about the payer's identity. A mature body of literature provides suitable instantiations of this primitive [6, 13, 33]. Upon verifying the proof of payment, the MNO treats the user's request as an authorised order and proceeds with provisioning. This assumption is orthogonal to the ZK request proof and preserves privacy and accountability.

- *Phase 2: Pseudonymous profile order (UE ↔ MNO ↔ SM-DP+)* In this phase (Fig. 8), the MNO receives $(x, PCert_U, \pi_{req})$ from the user over server-authenticated TLS. It first binds the request to its own challenge and recovers the certified user key by checking nonce = $nonce_{MNO}$, verifying the chain and expiry of $PCert_U$ under

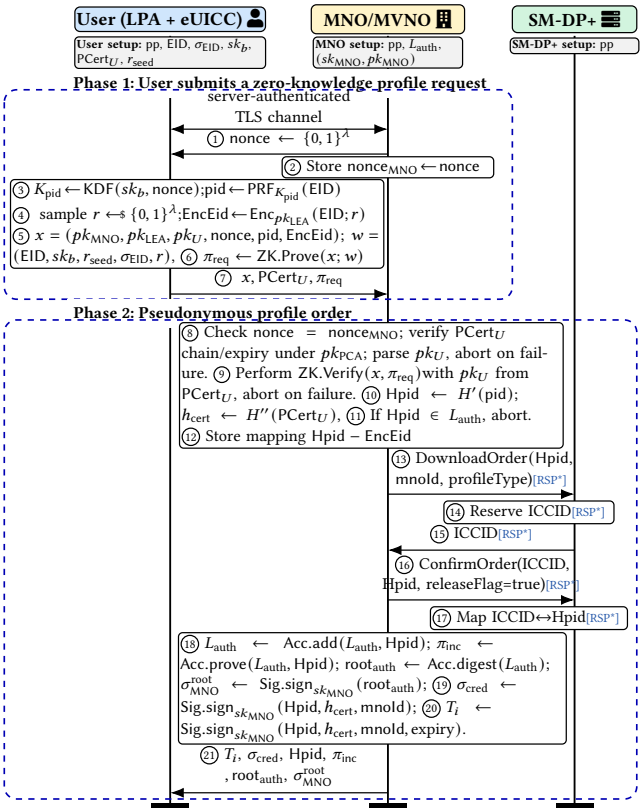


Figure 8: *ZKRequest* and *OrderProfile* Protocols: Profile Download Initiation. [RSP*] denotes steps that follow the RSP flow with modified semantics. Unmarked steps are new.

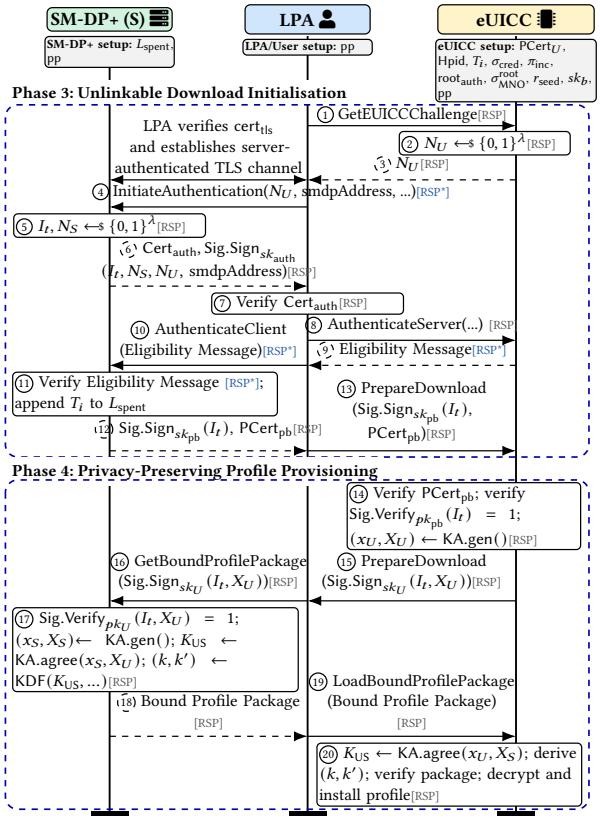


Figure 9: *MutualAuthAndProvision* Protocol: Mutual Authentication and Profile Download. [RSP] denotes unmodified GSMA Consumer RSP steps; [RSP*] denotes steps that follow the same RSP flow with modified semantics. Unmarked steps are new.

pk_{PCA}, and parsing pk_U; it aborts on failure. The MNO then verifies π_{req} against x using the pk_U recovered from PCert_U, thereby attesting same-eUICC linkage, eligibility under the MNO's policy, pseudonym correctness, and escrow well-formedness; invalid proofs are rejected. Next, it computes the hashed pseudonym Hpid and the certificate hash h_{cert}, aborts if $\text{Hpid} \in L_{\text{auth}}$, and stores the mapping ($\text{Hpid} \mapsto \text{EncEid}$). The MNO then places a DownloadOrder with the SM-DP+ over an authenticated backend channel. The SM-DP+ reserves a fresh *ICCID* and returns it to the MNO, after which the MNO finalises the order with ConfirmOrder. The SM-DP+ then records the internal binding $\textit{ICCID} \leftrightarrow \text{Hpid}$. After confirmation, the MNO commits the authorisation decision by updating the accumulator L_{auth}, producing the inclusion proof π_{inc}, computing the digest $\text{root}_{\text{auth}}$, and signing it as $\sigma^{\text{root}}_{\text{MNO}}$. It also issues credentials bound to $(\text{Hpid}, h_{\text{cert}}, \text{mnoId})$, namely σ_{cred}, and derives a one-time token T_i with expiry. These values, together with $(\text{Hpid}, \pi_{\text{inc}}, \text{root}_{\text{auth}}, \sigma^{\text{root}}_{\text{MNO}})$, are then returned to the user in the final response. ZKRequest and OrderProfile thus convert a ZK-authenticated, pseudonymous request into a minimally identifying yet enforceable authorisation: replay is prevented by checking $\text{Hpid} \notin L_{\text{auth}}$, the authorisation state is committed via the accumulator root and MNO signature, and subsequent steps can proceed using T_i and σ_{cred} without revealing EID, while accountability and trace-readiness remain intact through the established escrow.

- *Phases 3 and 4: Unlinkable Download Initialisation and Privacy-Preserving Profile Provisioning (LPA/UE/eUICC ↔ SM-DP+)* In these phases (Fig. 9), Phase 3 covers server authentication, eligibility validation, and download preparation, while Phase 4 begins with the eUICC's verification of the profile-signer material and continues with the authenticated key establishment and protected profile loading. The LPA first invokes GetEUICCChallenge, causing the eUICC to sample a fresh nonce N_U and return it. The LPA then verifies cert_{tls} and establishes a server-authenticated TLS channel to the SM-DP+, before sending InitiateAuthentication. The SM-DP+ samples a fresh transcript identifier I_t and server nonce N_S, and returns $\text{Cert}_{\text{auth}}$ together with a signature under sk_{auth}. After the LPA verifies $\text{Cert}_{\text{auth}}$, it relays the server-authentication material to the eUICC through AuthenticateServer, thereby binding the session to the fresh challenge pair (N_U, N_S) and transcript I_t. The eUICC then sends an eligibility message = $(\text{PCert}_U, \sigma_{\text{cred}}, T_i, \text{Hpid}, \pi_{\text{inc}}, \text{root}_{\text{auth}}, \sigma^{\text{root}}_{\text{MNO}}, \text{Sig.sign}_{sk_U}(I_t, N_S, N_U, T_i, \text{Hpid}, \text{PCert}_U, \text{smdpAddress}))$ via the LPA, which relays it to the SM-DP+ using AuthenticateClient. The SM-DP+ verifies this eligibility message and atomically appends T_i to the append-only set L_{spent}, rejecting reuse. It next returns $(\text{Sig.sign}_{sk_{\text{pb}}}(I_t), \text{Cert}_{\text{pb}})$ through the LPA. At the start of Phase 4, the eUICC verifies Cert_{pb} and the signature on I_t, thereby authenticating the profile-signing credential for the current transcript, and generates an ephemeral key pair (x_U, X_U). It then signs (I_t, X_U)

under sk_U and returns this authenticated key-share contribution through PrepareDownload and GetBoundProfilePackage. The SM-DP+ verifies the signature under pk_U, generates its own ephemeral pair (x_S, X_S), computes K_{US}, and derives (k, k'). Finally, the SM-DP+ sends the *BoundProfilePackage*=$(C, t_P, \mathsf{mnoId}, t_{\mathsf{mno}}, \mathsf{iccid}, t_{\mathsf{iccid}}, X_S, \mathsf{Sig.sign}_{sk_{\mathsf{pb}}}(I_t, X_S, X_U, \mathsf{smdpAddress}))$ via the LPA. Using X_S carried in that package, the eUICC recomputes K_{US}, derives the same (k, k'), verifies the package, and decrypts and installs the profile. Together, these phases convert the Phase 2 pseudonymous authorisation into a one-time, session-bound profile download: the server is authenticated, client eligibility is validated and spent exactly once, the download keys are established with authenticated eUICC participation, and the resulting profile is delivered under confidentiality and integrity protection.

Post-Provisioning: ZK-eSIM's privacy claim is intentionally scoped to *provisioning*. Once a profile is installed, ordinary user-initiated lifecycle operations such as enable, disable, and delete remain unchanged and are executed locally between the LPA and the eUICC [20]. These operations use profile-local state (e.g., ICCID) and therefore do not re-expose the device-global EID to the SM-DP+ or require reconstruction of an EID–ICCID mapping. Backend-triggered post-installation management can likewise avoid re-exposing EID. The backend need only address the installed profile via an associated profile-scoped alias recorded at provisioning. Delivery can then proceed by *device pull*, in which the LPA periodically retrieves pending notifications indexed by that profile handle. ZK-eSIM operates at the provisioning layer: it privately installs the eSIM profile and credentials before 5G-AKA begins, after which the UE runs the standard 5G-AKA procedure unchanged, with no additional identifiers, messages, or modifications to 3GPP core functions.

5. Supporting Procedures: Settlement and Accountable Deanonymisation We describe the two operational procedures that complete the design: settlement by one-time tokens and accountable deanonymisation. *For settlement*, the MNO issues each authorised profile order with a single-use token bound to the corresponding profile identifier and authenticated by the MNO. When the SM-DP+ provisions the profile, it records the token as redeemed. At the end of each epoch, both parties commit to their respective views: the SM-DP+ commits to the set of redeemed tokens, while the MNO commits to the set of authorised profile identifiers. The SM-DP+ returns the redeemed-token set together with evidence that each token was included in its committed view. The MNO accepts a redeemed token only if it is validly authenticated by the MNO, appears in the SM-DP+'s committed redemption log, and corresponds to a profile identifier that the MNO had authorised for that epoch. The final settlement count therefore includes only tokens that are both redeemed and authorised. The parties then cross-sign a settlement receipt binding the epoch, the accepted count, the payable amount, and the two committed views, yielding an auditable record without revealing any additional subscriber information. *For accountable deanonymisation*, the eUICC encrypts its EID to the LEA's public key during the request protocol, and the resulting escrow ciphertext is stored by the MNO. Under a valid and lawfully scoped warrant, the LEA obtains the in-scope escrow from the MNO and decrypts it to recover the EID for the specific provisioning session. The MNO can then resolve that EID to the corresponding KYC record for the lawful process. This preserves compatibility with mandatory subscriber registration: the MNO learns the subscriber's identity during Phase 0.a (KYC), which lies outside provisioning-layer anonymity. ZK-eSIM removes only protocol-level cross-session linkability, while requiring LEA cooperation under a scoped legal order to deanonymise a specific provisioning session.

8 Security Analysis

We show that ZK-eSIM satisfies two main security properties: *provisioning-session unlinkability* and *unforgeability*. Unlinkability means that infrastructure entities cannot distinguish whether two accepted provisioning sessions originate from the same eligible eUICC or from two different eligible eUICCs. Unforgeability means that an adversary cannot obtain accepted order-side or download-side provisioning state without legitimate credentials, fresh challenges, and MNO-issued authorisation material. Formal game-based definitions are given in **Appendix** A, and the proofs are given in **Appendix** B.

Theorem 1 (Unlinkability). *If the NIZK system satisfies zero-knowledge, commitments are hiding,* KDF *and* PRF *are secure, H' and H'' are modelled as random oracles, encryption provides IND-CPA security, and pseudonym certificates satisfy ephemeral indistinguishability, then ZK-eSIM achieves provisioning-session unlinkability for $\mathcal{E} \in \{\mathsf{MNO}, \mathsf{SMDP}^+, \mathsf{PCA}\}$. For every PPT adversary $\mathcal{A}$:*

$$\mathsf{Adv}^{\mathsf{UNLINK}\text{-}\mathcal{E}}_{\Pi,\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{ZK}} + \mathsf{Adv}^{\mathsf{KDF}} + \mathsf{Adv}^{\mathsf{PRF}} + \mathsf{Adv}^{\mathsf{Cert}} + \mathsf{Adv}^{\mathsf{IND\text{-}CPA}} + \frac{2q_H}{2^\lambda},$$

where q_H is the total number of random-oracle queries to H' and H''.

Sketch of Proof. We proceed by hybrids, first replacing each $\pi_{\mathsf{req},s}$ with a simulated proof using zero knowledge. We then replace the KDF-derived key and the resulting PRF pseudonym by independent uniform values, using KDF and PRF security. Next, we program H' and H'' at the fresh challenge inputs with uniform outputs; a prior-query event occurs with probability at most $2q_H/2^\lambda$. Ephemeral certificate indistinguishability allows us to replace $\mathsf{PCert}_{U,s}$ by an identity-independent certificate, and IND-CPA security allows EncEid_s to encrypt 0^λ. The resulting view is independent of b, so the adversary succeeds with probability 1/2; the SMDP^+ and PCA cases follow analogously from their respective views.

Theorem 2 (MNO-Side Unforgeability). *Under NIZK knowledge soundness, anonymous credential unforgeability, collision resistance of H and H', and freshness of MNO challenges, ZK-eSIM satisfies MNO-side unforgeability. For every PPT adversary $\mathcal{A}$:*

$$\mathsf{Adv}^{\mathsf{UF\text{-}MNO}}_{\Pi,\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{Sound\text{-}ZK}} + \mathsf{Adv}^{\mathsf{UF\text{-}Cred}} + \mathsf{Adv}^{\mathsf{CR}}_{H} + \mathsf{Adv}^{\mathsf{CR}}_{H'} + \frac{q_N}{2^\lambda},$$

where q_N denotes the number of MNO challenge queries.

Theorem 3 (SM-DP+-Side Unforgeability). *Under EUF-CMA security of the MNO-issued authorisation credential σ_{cred} and one-time token T_i, collision resistance of H' and H'', and correctness of the stateful spent-token check, ZK-eSIM satisfies SM-DP+-side unforgeability. For every PPT adversary $\mathcal{A}$:*

$$\mathsf{Adv}^{\mathsf{UF\text{-}SMDP}^+}_{\Pi,\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{\sigma_{\mathsf{cred}}} + \mathsf{Adv}^{\mathsf{EUF\text{-}CMA}}_{T_i} +$$

$$\mathrm{Adv}^{\mathrm{CR}}_{H'} + \mathrm{Adv}^{\mathrm{CR}}_{H''} + \mathrm{Adv}^{\mathrm{DS}}.$$

Here $\mathrm{Adv}^{\mathrm{DS}}$ *denotes the probability that the stateful spent-token mechanism accepts a token already recorded in* L_{spent}.

9 Implementation and Evaluation

We evaluate ZK-eSIM in two ways: first, by quantifying runtime costs relative to the commercial RSP; second, by conducting a deployment feasibility test on an **open-source GSMA-compliant testbed.** We provide a deployment feasibility evaluation demonstrating that our scheme can be deployed with minimal adjustments to current standards. To facilitate evaluation, we implement a custom Java Card applet packaged inside an eSIM profile that runs the cryptographic operations on a **test eUICC** (sysmoEUICC1-C2T [49]). We also modify a GSMA-compliant open-source SM-DP+ server (osmo-smdpp [37]) and Local Profile Assistant (lpac [17]).

9.1 Testbed Setup and GSMA Integration

To evaluate the performance of the proposed ZK-eSIM protocol against the commercial RSP, we establish an experimental testbed illustrated in Fig. 10. To preserve compatibility with current RSP architectures, we extend GSMA-compliant network entities, i.e., the SM-DP+, and the LPA as discussed in §9.2. Cryptographic operations are performed with widely used libraries on each network entity, with the JCMathLib [31] as a mathematical library in the Java Card applet. On the eUICC, all operations are performed using the Java Card and JavaCardX Security libraries [36] (with JCMathLib where applicable), and cryptographic operations on the SM-DP+ and MNO servers are performed using Python's cryptographic hazmat library [39]. Server-side operations have been performed on a PC running Ubuntu 22.04, with the following specifications: RYZEN 7 8-core CPU, and 64GB RAM.

To complete the ZK-eSIM implementation, we deploy an additional PCA server. In a real-world deployment, the PCA would operate as a sub-CA within the existing GSMA PKI and could be operated by a provider such as Cybertrust, DigiCert, or SEALSQ [22]. In our prototype, we instantiate this role using a dedicated server responsible for certificate issuance and authorisation. The primary role of the PCA in ZK-eSIM is to generate pseudonym certificates for users that can provide a valid proof of eligibility. The provided pseudonym certificate is valid only for a single session of the ZK-eSIM protocol and expires upon completion of profile initialisation.

9.2 Implementation and Deployability

Implementing ZK-eSIM requires extending existing open-source implementations by updating the ASN.1 encodings to add six request messages and their corresponding response messages, introducing four supporting type definitions comprising 17 fields, and extending EuiccSigned1 to carry an eligibility bundle. The changes to the ASN.1 encodings have minimal impact on the GSMA message flow between network entities (SM-DP+, LPA, MNO, and eUICC). We add ZK-eSIM as an alternative message flow within the network entities, following the same messages with the modifications specified in §7.

ASN.1 Modifications. We extend the standard GSMA RSP protocol by modifying the ASN.1 level definitions to include a new optional message type (EligibilityData) within the EuiccSigned1. The EligibilityData definition contains: the hashed pseudonym ID (Hpid), the MNO signed credential (σ_{cred}), the one-time authorisation token (T_i), the accumulator root ($\mathrm{root}_{\mathrm{auth}}$), the signature over the accumulator root ($\sigma^{\mathrm{root}}_{\mathrm{MNO}}$), and the serialised proof of accumulator inclusion (π_{inc}). This enables Phase 2 of ZK-eSIM (pseudonymous profile ordering) with minimal modifications to existing protocol messages. Additionally, due to the lack of a standardised defined communication interface between the eUICC and MNO/PCA servers, we introduce 6 new Request and Response message pairs to facilitate our privacy-preserving zero-knowledge proofs and certificate initialisations. Modifications to the open-source network entities required adding ~628 LoC (Lines of Code) at the SM-DP+ to enable the alternative workflow using our proposed scheme for encoding and decoding ZK-eSIM messages, allowing ZK-eSIM to be built on top of the existing infrastructure and providing a secondary approach to RSP. Furthermore, ~6.7K LoC are used for the implementation of our custom Java Card Applet (including the JCMathLib and cryptographic functions); however, it should be noted that commercial eSIM profiles may already contain functional applets, which can reduce the number of LoC in a real-world deployment.

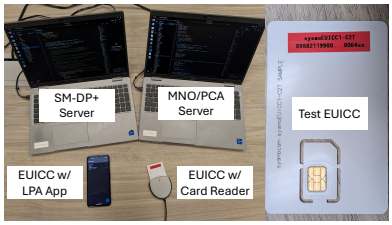


Figure 10: ZK-eSIM Implementation and Evaluation Testbed

Backward Compatibility. To enable backward compatibility, we develop ZK-eSIM as an alternative privacy-enhanced RSP protocol, which can be implemented alongside current deployments. SM-DP+ providers and MNOs can offer ZK-eSIM as an optional workflow that provides improved privacy over the commercial methods for those who wish to remain anonymous during eSIM profile provisioning, while leaving the commercial offering for users less concerned with privacy. This is enabled by the fact that all encoding-based changes are introduced as optional values within the existing ASN.1 format and require only minimal modifications to existing functions to enable the correct processing of new information elements. The main point at which ZK-eSIM deviates from existing deployments is the pseudonym certificate, which requires changes at the PCA (or may require a separate service offering) to allow the ZK-eSIM deployment.

9.3 End-to-End Cost Comparison

End-to-End Cost. Table 2 presents the mean and standard deviation (denoted by ±) for the execution time of each phase in both the commercial and the proposed ZK-eSIM schemes. The reported values capture the full end-to-end computational and communication costs of cryptographic operations and message flows for each protocol phase. All operations are executed on their respective devices in the experimental setup. ECDSA and SHA256 have been used during the certificate initialisation phase to align with the GSMA RSP specification [19]. Additionally, for this evaluation, we use an EC-based

Table 2: End-to-End cost of commercial RSP and the proposed ZK-eSIM averaged over 25 runs

Scheme \ Cost	Runtime Cost (ms)		End-to-End Cost (ms)
Commercial RSP[19]	Registration	N/A	53660
	Certificate Initialisation	7 (±1)	
	Order Profile	1073	
	Profile Download	52580	
ZK-eSIM	Registration	9964 (±918)	98199
	Certificate Initialisation	10448 (± 620)	
	Order Profile	18432 (± 764)	
	Profile Download	59355 (± 955)	

Schnorr proof [44] to allow the user to prove their identity without leaking their EID (simplified in implementation for a Java Card environment), SHA256-based commitments during the certificate initialisation phase, and ECDH [25] for key derivation and agreement. In terms of computational performance, ZK-eSIM exhibits an approximate increase in cryptographic processing time of 83% (from 53.660 seconds vs. 98.199 seconds). One contributing factor to this increase is the registration phase, which is not included in the cost of the commercial RSP because this step occurs prior to protocol initialisation. The cost is included in the ZK-eSIM calculation due to the introduced cryptographic primitives that demonstrate the overhead costs associated with improvements to user privacy when a user registers with a new MNO. Considering only the phases within the commercial RSP, the cost increases by ~ 64%. This increase results from the integration of additional cryptographic primitives, which introduce additional communication overhead costs between the UE and SM-DP+ and MNO servers. The added overhead is accompanied by a significant enhancement in user privacy compared to the commercial RSP, achieved by anonymising user identifiers. Protocol phases containing zero-knowledge proofs contribute the most to the cost, increasing by 10.441 seconds and 17.359 seconds for certificate initialisation and profile ordering, respectively. Both of these phases involve ZKP generation and verification, which incur relatively high runtimes compared to the commercial method, which has no proof computations and offers little to no privacy guarantees. Overall, these results indicate that although ZK-eSIM incurs higher computational costs (as evidenced by the end-to-end execution time) than the commercial RSP, it considerably improves user privacy. This is achieved by integrating ZKPs with additional cryptographic primitives, mitigating inherent user-identifier leakage in RSP. Combined with the analysis in Table 1, we note that ZK-eSIM requires longer runtimes but provides greater privacy than the commercial RSP, justifying the additional cost for those who value privacy.

9.4 Server Scalability of ZK-eSIM

Figure 11 presents the peak memory usage (and standard deviation of memory usage across the session) and cumulative CPU-seconds consumed by each server as the total number of requests served scales towards 1,000,000 concurrent download requests. Peak memory usage remains reasonable for large scale and closely clustered across all three roles, converging to 998 ± 9 MB at the SM-DP+, 1006 ± 7 MB at the MNO, and 1008 ± 8 MB at the PCA. The small increase in memory usage indicates that the ZK-eSIM scheme incurs no significant per-request memory accumulation, since the zero-knowledge proof verification operates on fixed-size cryptographic

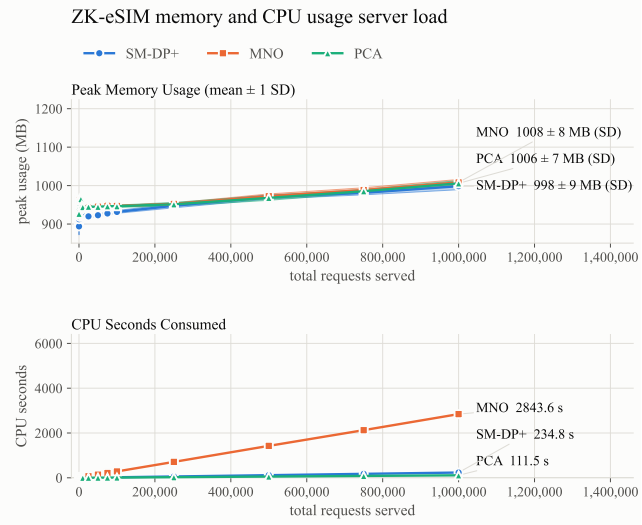


Figure 11: Server performance under load in range from 1 to 1,000,000 concurrent download requests

material. The CPU-seconds consumed reveal the same asymmetry observed in the per-download timings, with the majority of the processing cost concentrated at the MNO. Serving 1,000,000 requests, the MNO consumes 2843.6 s in CPU time on average, an order of magnitude greater than the 234.8 s at the SM-DP+ and the 111.5 s at the PCA. This reflects the MNO's role in performing the EC-based Schnorr identity proof verification, which incurs a higher computational load while ensuring that the network learns no permanent device or subscriber identifiers. Overall, the server load scales linearly with the number of requests served, with memory usage remaining bounded across all roles and CPU consumption remaining minimal at the SM-DP+ and PCA. As with the per-download analysis, the bulk of the cost is produced at the MNO, where the core identity privacy processing occurs, and this additional cost coincides with significant user privacy enhancements delivered through the anonymisation of user identifiers.

10 Discussion

ZK-eSIM primarily focuses on securing provisioning-layer privacy but does not address cross-protocol risks that persist once a profile is installed. Users may still face jurisdictional exposure when traffic is routed through unexpected foreign networks, when silent background communication initiated by embedded profile commands goes unnoticed, when fragile lifecycle operations silently fail, and when private 5G deployments shift control to administrators with weaker oversight [32]. Also, IP address deanonymisation is out of scope for this paper. Extending ZK-eSIM to support routing transparency, IP address anonymisation, constrained profile behaviour, and resilient lifecycle management remains future work.

11 Conclusion

ZK-eSIM replaces the persistent identifiers exposed during GSMA Consumer RSP with ZKPs, single-use pseudonym certificates, and per-session pseudonyms, achieving provable subscriber anonymity and provisioning-session unlinkability while retaining accountable

traceability. A Java Card applet prototype on a test eUICC, integrated with a modified open-source SM-DP+ and LPA, shows that these guarantees are practical within the existing GSMA ecosystem.

Acknowledgements

This work has been supported by the NSF under grants 2145631, 2215017, 2613659, and 2614551, and the Public Wireless Supply Chain Innovation Fund (PWSCIF) Under Federal Award ID Number 51-60-IF007.

Open Science

To support reproducibility, we share our modified SM-DP+ server [37], extended LPA [17], and Java Card applet at https://github.com/NaivEPoi/ZK-eSim.

Ethical Considerations

ZK-eSIM reduces identifier exposure while preserving secure provisioning and accountable traceability. To limit misuse, we require operator-issued eligibility credentials, one-time tokens, replay checks, and short-lived pseudonym certificates. Identity recovery requires a scoped legal process and joint LEA–MNO cooperation. Our evaluation uses test eUICCs and modified open-source RSP components in a controlled testbed. We do not access production infrastructure, disclose carrier secrets, or use subscriber telemetry or human-subject data. Any deployment should undergo operator, standards-body, and jurisdiction-specific legal review.

References

[1] Liza Ahmad, Quan Shi, Joshua Haworth, Yilu Dong, Prosanta Gope, Behzad Abdolmaleki, and Syed Rafiul Hussain. 2026. ZK-eSIM: A Privacy-Centric Zero-Knowledge Approach for eSIM Provisioning. doi:10.48550/ARXIV.2609.07654

[2] Abu Shohel Ahmed, Aleksi Peltonen, Mohit Sethi, and Tuomas Aura. 2024. Security Analysis of the Consumer Remote SIM Provisioning Protocol. *ACM Transactions on Privacy and Security* 27, 3 (Aug. 2024), 1–36. doi:10.1145/3663761

[3] Abu Shohel Ahmed, Mukesh Thakur, Santeri Paavolainen, and Tuomas Aura. 2021. Transparency of SIM profiles for the consumer remote SIM provisioning protocol. *Annals of Telecommunications* 76, 3-4 (April 2021), 187–202. doi:10.1007/s12243-020-00791-2

[4] Apple. 2025. Apple unveils iPhone 17 Pro and iPhone 17 Pro Max. https://www.apple.com/newsroom/2025/09/apple-unveils-iphone-17-pro-and-iphone-17-pro-max/

[5] David Basin, Jannik Dreier, Lucca Hirschi, Saša Radomirovic, Ralf Sasse, and Vincent Stettler. 2018. A Formal Analysis of 5G Authentication. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Toronto Canada, 1383–1396. doi:10.1145/3243734.3243846

[6] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. 2014. Zerocash: Decentralized Anonymous Payments from Bitcoin. In *2014 IEEE Symposium on Security and Privacy*. IEEE, San Jose, CA, 459–474. doi:10.1109/SP.2014.36

[7] Eli Ben-Sasson, Alessandro Chiesa, Matthew Green, Eran Tromer, and Madars Virza. 2015. Secure Sampling of Public Parameters for Succinct Zero Knowledge Proofs. In *2015 IEEE Symposium on Security and Privacy*. IEEE, San Jose, CA, 287–304. doi:10.1109/SP.2015.25

[8] Manuel Blum, Paul Feldman, and Silvio Micali. 1988. Non-interactive zero-knowledge and its applications. In *Proceedings of the twentieth annual ACM symposium on Theory of computing - STOC '88*. ACM Press, Chicago, Illinois, United States, 103–112. doi:10.1145/62212.62222

[9] Benedikt Brecht, Dean Therriault, André Weimerskirch, William Whyte, Virendra Kumar, Thorsten Hehn, and Roy Goudy. 2018. A Security Credential Management System for V2X Communications. doi:10.48550/ARXIV.1802.05323

[10] Jan Camenisch and Anna Lysyanskaya. 2001. An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In *Advances in Cryptology — EUROCRYPT 2001*, Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, and Birgit Pfitzmann (Eds.). Vol. 2045. Springer Berlin Heidelberg, Berlin, Heidelberg, 93–118. doi:10.1007/3-540-44987-6_7

[11] Jan Camenisch and Anna Lysyanskaya. 2001. An Identity Escrow Scheme with Appointed Verifiers. In *Advances in Cryptology — CRYPTO 2001*, Gerhard Goos, Juris Hartmanis, Jan Van Leeuwen, and Joe Kilian (Eds.). Vol. 2139. Springer Berlin Heidelberg, Berlin, Heidelberg, 388–407. doi:10.1007/3-540-44647-8_23

[12] Jan Camenisch and Anna Lysyanskaya. 2004. Signature Schemes and Anonymous Credentials from Bilinear Maps. In *Advances in Cryptology – CRYPTO 2004*, David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Dough Tygar, Moshe Y. Vardi, Gerhard Weikum, and Matt Franklin (Eds.). Vol. 3152. Springer Berlin Heidelberg, Berlin, Heidelberg, 56–72. doi:10.1007/978-3-540-28628-8_4

[13] David Chaum. 1983. Blind Signatures for Untraceable Payments. In *Advances in Cryptology*, David Chaum, Ronald L. Rivest, and Alan T. Sherman (Eds.). Springer US, Boston, MA, 199–203. doi:10.1007/978-1-4757-0602-4_18

[14] Yvo G. Desmedt. 1994. Threshold cryptography. *European Transactions on Telecommunications* 5, 4 (July 1994), 449–458. doi:10.1002/ett.4460050407

[15] Alex Deutsch. 2025. Is an eSIM Cheaper Than Roaming? Here's the Breakdown. https://esim.planet.uk/news/is-an-esim-cheaper-than-roaming-heres-the-breakdown/

[16] D. Dolev and A. Yao. 1983. On the security of public key protocols. *IEEE Transactions on Information Theory* 29, 2 (March 1983), 198–208. doi:10.1109/TIT.1983.1056650

[17] ESTKme Group. 2025. GitHub - estkme-group/lpac at d214738fa0bdb23faf5833d3d798963079a00468. https://github.com/estkme-group/lpac/tree/d214738fa0bdb23faf5833d3d798963079a00468

[18] GSMA. 2019. Security Accreditation Scheme (SAS). https://www.gsma.com/solutions-and-impact/technologies/security/gsma_resources/security-accreditation-scheme-for-uicc/

[19] GSMA. 2023. *RSP Architecture SGP.21 V3.1.* Technical Report.

[20] GSMA. 2023. *SGP.22 RSP Technical Specification v3.0.* Technical Report.

[21] GSMA. 2024. *eSIM Market – China and Beyond.* Technical Report. GSMA. https://www.gsma.com/solutions-and-impact/technologies/esim/wp-content/uploads/2024/07/GSMA-Welcome-and-eSIM-Market-China-and-Beyond.pdf

[22] GSMA. 2026. eSIM Certificates. https://www.gsma.com/solutions-and-impact/technologies/esim/gsma-root-ci/

[23] GSMA. 2026. SAS Accredited Sites. https://www.gsmaservices.com/assurance-services/security-accreditation-scheme-sas/sas-accredited-sites/

[24] Francesco Intoci, Julian Sturm, Daniel Fraunholz, Apostolos Pyrgelis, and Colin Barschel. 2023. P3LI5: Practical and confidEntial Lawful Interception on the 5G core. In *2023 IEEE Conference on Communications and Network Security (CNS)*. 1–9. doi:10.1109/CNS59707.2023.10288872

[25] Neal Koblitz. 1987. Elliptic curve cryptosystems. *Math. Comp.* 48, 177 (1987), 203–209. doi:10.1090/S0025-5718-1987-0866109-5

[26] Adrien Koutsos. 2019. The 5G-AKA Authentication Protocol Privacy. In *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*. 464–479. doi:10.1109/EuroSP.2019.00041

[27] Kevin Lee, Ben Kaiser, Jonathan Mayer, and Arvind Narayanan. 2020. An empirical study of wireless carrier authentication for SIM swaps. (2020).

[28] Tobias Looker, Vasilis Kalos, Andrew Whitehead, and Mike Lodder. 2026. *The BBS Signature Scheme.* Internet Draft draft-irtf-cfrg-bbs-signatures-10. Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-irtf-cfrg-bbs-signatures/03/

[29] Zhihong Luo, Silvery Fu, Natacha Crooks, Shaddi Hasan, Christian Maciocco, Sylvia Ratnasamy, and Scott Shenker. 2023. LOCA: A Location-Oblivious Cellular Architecture. USENIX Association, Boston, MA, 1621–1646.

[30] Mary Maller, Sean Bowe, Markulf Kohlweiss, and Sarah Meiklejohn. 2019. Sonic: Zero-Knowledge SNARKs from Linear-Size Universal and Updatable Structured Reference Strings. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM, London United Kingdom, 2111–2128. doi:10.1145/3319535.3339817

[31] Vasilios Mavroudis and Petr Svenda. 2020. JCMathLib: Wrapper Cryptographic Library for Transparent and Certifiable JavaCard Applets. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. 89–96. doi:10.1109/EuroSPW51379.2020.00022

[32] Maryam Motallebighomi, Jason Veara, Evangelos Bitsikas, and Aanjhan Ranganathan. 2025. eSIMplicity or eSIMplification? Privacy and Security Risks in the eSIM Ecosystem. In *Proceedings of the 34th USENIX Security Symposium*.

[33] Matteo Nardelli, Francesco De Sclavis, and Michela Iezzi. 2025. A Hitchhiker's Guide to Privacy-Preserving Digital Payment Systems: A Survey on Anonymity, Confidentiality, and Auditability. arXiv:2505.21008. doi:10.48550/arXiv.2505.21008

[34] European Parliament and Council of the European Union. 2016. Regulation (EU) 2016/679 (General Data Protection Regulation). https://eur-lex.europa.eu/eli/reg/2016/679/oj/eng

[35] Gigi Onag. 2025. Privacy watchdog hits SKT with record $97M fine over data breach. https://www.lightreading.com/regulatory-politics/privacy-watchdog-hits-skt-with-record-97m-fine-over-data-breach

[36] Oracle. 2015. javacardx.security (Java Card API, Classic Edition). https://docs.oracle.com/cd/E59935_01/api/javacardx/security/package-summary.html

[37] Osmocom. 2026. osmocom/pysim. https://github.com/osmocom/pysim

Game $\mathsf{UNLINK}^{\mathcal{E}}_{\Pi,\mathcal{A}}(\lambda)$ for $\mathcal{E} \in \{\mathsf{MNO}, \mathsf{SMDP}^+, \mathsf{PCA}\}$

- $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda)$. Initialise $L_{\text{auth}}, L_{\text{spent}} \leftarrow \emptyset$. Let (U_0, EID_0) and (U_1, EID_1) be two eligible registered devices with $\mathsf{EID}_0 \neq \mathsf{EID}_1$.
- $\mathcal{A}$ chooses (U_0, EID_0) and (U_1, EID_1). The challenger samples $\mathsf{b} \xleftarrow{\$} \{0,1\}$ and defines

$$\mathsf{idx}_s(\mathsf{b}) := \begin{cases} 0 & \text{if } \mathsf{b} = 0, \\ s-1 & \text{if } \mathsf{b} = 1, \end{cases} \qquad s \in \{1,2\}.$$

- For each $s \in \{1,2\}$, the challenger runs an accepted provisioning session for $(U_{\mathsf{idx}_s(\mathsf{b})}, \mathsf{EID}_{\mathsf{idx}_s(\mathsf{b})})$ and gives $\mathcal{A}$ the corresponding entity view $\mathsf{View}^{\mathcal{E}}_s$.
- The views are defined as follows, where $\mathsf{Hpid}_s := H'(\mathsf{pid}_s)$ and $h_{cert,s} := H''(\mathsf{PCert}_{U,s})$:

$$\mathsf{View}^{\mathsf{MNO}}_s := (\mathsf{pid}_s, \mathsf{EncEid}_s, \pi_{\text{req},s}, \mathsf{PCert}_{U,s}, h_{cert,s}, \mathsf{Hpid}_s, \sigma_{\text{cred},s}, T_{i,s}, L_{\text{auth}}).$$

$$\mathsf{View}^{\mathsf{SMDP}^+}_s := (\mathsf{PCert}_{U,s}, h_{cert,s}, \mathsf{Hpid}_s, T_{i,s}, \sigma_{\text{cred},s}, \pi_{\text{inc},s}, \mathsf{root}_{\text{auth},s}, \sigma^{\text{root}}_{\text{MNO},s}, I_{t,s}, N_{S,s}, N_{U,s}, L_{\text{spent}}).$$

$$\mathsf{View}^{\mathsf{PCA}}_s := (pk_{U,s}, \pi_{\text{bind},s}, \mathsf{PCert}_{U,s}, \mathsf{expiry}_s).$$

- $\mathcal{A}$ receives $(\mathsf{View}^{\mathcal{E}}_1, \mathsf{View}^{\mathcal{E}}_2)$ and outputs $\mathsf{b}' \in \{0,1\}$.
- The game outputs 1 iff $\mathsf{b}' = \mathsf{b}$.

Figure 12: Provisioning-session unlinkability game.

[38] Alex Passett. 2024. Workz and its Cloud eSIM Platform Surpass 100 Partner Telcos. https://www.iotevolutionworld.com/iot/articles/458797-workz-its-cloud-esim-platform-surpass-100-partner.htm

[39] Python Cryptographic Authority (pyca). 2026. Primitives — Cryptography 48.0.0.dev1 documentation. https://cryptography.io/en/latest/hazmat/primitives/

[40] Siddharth Prakash Rao and Alexandros Bakas. 2023. Authenticating Mobile Users to Public Internet Commodity Services Using SIM Technology. In *Proceedings of the 16th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. ACM, Guildford United Kingdom, 151–162. doi:10.1145/3558482.3590181

[41] Juniper Research. 2025. Travel eSIMs Surge as Roaming Alternative - Up 85% in 2025 Mobile Operators Will Play Growing Role in the Market. https://www.juniperresearch.com/press/travel-esims-surge-as-roaming-alternative/

[42] Mike Robuck. 2024. FCC fines T-Mobile US for data breaches. https://www.mobileworldlive.com/t-mobile-us/fcc-fines-t-mobile-us-for-data-breaches/

[43] Paul Schmitt and Barath Raghavan. 2021. Pretty Good Phone Privacy. In *30th USENIX Security Symposium (USENIX Security 21)*. https://www.usenix.org/conference/usenixsecurity21/presentation/schmitt

[44] C. P. Schnorr. 1990. Efficient Identification and Signatures for Smart Cards. In *Advances in Cryptology — CRYPTO' 89 Proceedings*, Gilles Brassard (Ed.). Vol. 435. Springer New York, New York, NY, 239–252. doi:10.1007/0-387-34805-0_22

[45] Adi Shamir. 1979. How to share a secret. *Commun. ACM* 22, 11 (Nov. 1979), 612–613. doi:10.1145/359168.359176

[46] Quan Shi, Liying Wang, Prosanta Gope, Qi Liang, Haowen Wang, Qirui Liu, Chenren Xu, Shangguang Wang, Qing Li, and Biplab Sikdar. 2026. LPG: Raise Your Location Privacy Game in Direct-to-Cell LEO Satellite Networks. In *35th USENIX Security Symposium (USENIX Security 26)*. https://www.usenix.org/conference/usenixsecurity26/presentation/shi-quan

[47] Stephen Treffinger. 2025. If you're not using an eSIM when you travel, you're getting ripped off. https://www.theguardian.com/lifeandstyle/2025/sep/26/how-does-esim-work

[48] verbraucherzentrale. 2026. International roaming: Avoid hidden costs with Wi-Fi and mobile networks. https://www.verbraucherzentrale.de/wissen/digitale-welt/mobilfunk-und-festnetz/internationales-roaming-bei-wlan-und-mobilfunk-kostenfallen-vermeiden-27954

[49] Harald Welte. 2026. *sysmocom - systems for mobile communications GmbH*. Technical Report. sysmocom. https://www.sysmocom.de/manuals/sysmoeuicc-manual.pdf

[50] Travel and Tour World. 2025. New Travel ESIM Options Are Unlocking Affordable, Global Connectivity For Travelers In China, India, And Southeast Asia. https://www.travelandtourworld.com/news/article/new-travel-esim-options-are-unlocking-affordable-global-connectivity-for-travelers-in-china-india-and-southeast-asia/

[51] Yang Yang, Quan Shi, Prosanta Gope, Behzad Abdolmaleki, and Biplab Sikdar. 2025. PGUS: Pretty Good User Security for Thick MVNOs with a Novel Sanitizable Blind Signature. In *2025 IEEE Symposium on Security and Privacy (SP)*. doi:10.1109/SP61157.2025.00144

[52] Hang Yuan, Artiom Baloian, Jan Janak, and Henning Schulzrinne. 2024. eSIM Technology in IoT Architecture. doi:10.48550/ARXIV.2401.04302

A Game-based Definition

We define the security of ZK-eSIM through two families of games: provisioning-session unlinkability and unforgeability. The former captures whether an infrastructure entity can link two accepted sessions to the same eUICC. The latter captures whether an adversary can make an honest MNO or SM-DP+ accept an unauthorised request.

Definition 2 (Provisioning-Session Unlinkability). *For a PPT adversary $\mathcal{A}$ and an entity $\mathcal{E} \in \{\mathsf{MNO}, \mathsf{SMDP}^+, \mathsf{PCA}\}$, define*

$$\mathsf{Adv}^{\mathsf{UNLINK}\text{-}\mathcal{E}}_{\Pi,\mathcal{A}}(\lambda) := \left| \Pr[\mathsf{UNLINK}^{\mathcal{E}}_{\Pi,\mathcal{A}}(\lambda) = 1] - \tfrac{1}{2} \right|.$$

We say that Π *provides* provisioning-session unlinkability *if this advantage is negligible in λ for every PPT $\mathcal{A}$ and every $\mathcal{E} \in \{\mathsf{MNO}, \mathsf{SMDP}^+, \mathsf{PCA}\}$.*

Definition 3 (Unforgeability). *Let $\mathsf{UF}^{\mathsf{MNO}}_{\Pi,\mathcal{A}}(\lambda)$ and $\mathsf{UF}^{\mathsf{SMDP}^+}_{\Pi,\mathcal{A}}(\lambda)$ denote the order-side and download-side unforgeability games. Define*

$$\mathsf{Adv}^{\mathsf{UF}\text{-}\mathsf{MNO}}_{\Pi,\mathcal{A}}(\lambda) := \Pr[\mathsf{UF}^{\mathsf{MNO}}_{\Pi,\mathcal{A}}(\lambda) = 1],$$

and

$$\mathsf{Adv}^{\mathsf{UF}\text{-}\mathsf{SMDP}^+}_{\Pi,\mathcal{A}}(\lambda) := \Pr[\mathsf{UF}^{\mathsf{SMDP}^+}_{\Pi,\mathcal{A}}(\lambda) = 1].$$

We say that Π *is* unforgeable *if both advantages are negligible in λ for every PPT adversary $\mathcal{A}$.*

Unlinkability (Fig. 12). The challenge bit determines whether two accepted provisioning sessions come from the same registered eUICC or from two different registered eUICCs. The adversary observes only the selected infrastructure entity's view. The protocol is unlinkable if no PPT adversary can distinguish these two cases with non-negligible advantage. Timing and packet-size metadata are excluded from the formal view, consistent with the threat model.
Unforgeability. The adversary may obtain honest registrations, MNO challenges, authorisations, and provisioning transcripts through the four learning oracles. For the MNO side, a successful forgery is an accepted request that is not backed by a registered eligibility credential, a fresh MNO challenge, and a valid proof binding the same eUICC to the credential, pseudonym, escrow ciphertext, and pseudonym certificate. For the SM-DP+ side, a successful forgery is an accepted download that is not backed by MNO-issued authorisation material bound to the same $(\mathsf{Hpid}, h_{cert})$ tuple, or that reuses a token already recorded in L_{spent}. Therefore, a fresh Hpid generated by a legitimate registered device with a fresh MNO challenge is not a forgery. We provide the definition in the **extended version** [1].

B Security Proof

This information is presented in the **extended version** of this manuscript [1].

C Collusion Analysis

This information is presented in the **extended version** of this manuscript [1].