Experimental Analysis of Authentication-Based Privacy Leakage in 5G SA

Hamza Ahmed Farah[1], Azza H. Ahmed[1,2], and Thomas Dreibholz[2]

[1] Oslo Metropolitan University, Oslo, Norway
hamza.a.m.farah@gmail.com
azzahass@oslomet.no
[2] Simula Metropolitan Centre for Digital Engineering, Oslo, Norway
dreibh@simula.no

Abstract. 5G Standalone (SA) networks introduce the Subscriber Concealed Identifier (SUCI) and the 5G Authentication and Key Agreement (5G-AKA) protocol to eliminate the plaintext identity exposure of earlier generations. However, privacy leakage can persist through observable protocol behaviour rather than exposed identifiers. This paper presents an experimental evaluation of two such attack classes — the *SUCI-catcher* and the *SQN inference attack* — implemented as a transparent N2-interface proxy on a fully virtualised, 3GPP Release 16-compliant testbed built on OpenAirInterface, UERANSIM, and a custom NGAP proxy. The SUCI-catcher exploits the binary outcome of the 5G-AKA challenge-response exchange as a presence oracle, successfully recording SUCIs from six subscribers and confirming target presence without recovering any keying material, weakening location privacy and unlinkability in a controlled virtualised environment. The SQN inference attack recovers subscriber sequence-number bits through controlled synchronisation failures, accurately inferring 8 bits of *SQN* verified against ground-truth core network state, and demonstrating that SQN changes between observations track subscriber activity, weakening undetectability in a controlled virtualised environment. Both attacks share a structural dependency on abusing core-network authentication-vector generation. Operator-level rate-limiting and anomaly detection are identified as near-term mitigations; long-term protection requires protocol-level changes to 5G-AKA.

Keywords: 5G Networks · Privacy Attacks · Authentication Security · Subscriber Tracking · Protocol Vulnerability · Experimental Security Analysis

1 Introduction

Fifth-generation mobile networks have become critical infrastructure for mobile broadband, IoT, and industrial services worldwide [17,16]. Because these networks carry sensitive subscriber data, privacy is a fundamental requirement addressed in 3GPP Release 15 [11]. Earlier generations exposed the permanent *International Mobile Subscriber Identity* (IMSI) in plaintext, enabling IMSI-catcher attacks [10]. 5G Stand-Alone (SA) addresses this with the *Subscriber*

Concealed Identifier (SUCI), which encrypts the permanent identifier, and the improved *5G Authentication and Key Agreement* (5G-AKA) protocol [11,4,14].

Despite these protections, privacy in 5G depends not only on cryptographic strength but also on protocol behaviour. Observable message exchanges — registration requests, authentication challenges, and synchronisation failures — can expose subscriber information without revealing the permanent identifier. Two attack classes exploit this property.

The *SUCI-catcher* [8,6,3] conventionally operates as a rogue base station. In this work we implement an equivalent capability via an N2-interface proxy — a transparent man-in-the-middle between a legitimate gNB and the Access and Mobility Management Function (AMF) — achieving the same SUCI substitution without radio hardware, making the implementation faster and more reproducible. A known SUCI is injected into a live Registration Request; the UE's binary response (Authentication Response or MAC Failure) confirms or denies subscriber presence, weakening location privacy and unlinkability.

The *SQN inference attack* [5,7] recovers bits of SQN_{UE} by XOR-cancelling the anonymity key across two AUTS resynchronisation responses collected under the same RAND. Because the SQN encodes cumulative authentication events, inferred SQN changes between observations reveal subscriber activity, weakening undetectability. This attack was theorised for 5G SA [5] but had not been experimentally validated prior to this work.

Both attacks have been individually studied [8,5,13], but no prior work provides a joint evaluation within the same 5G SA testbed [3,9]. This paper addresses that gap using a fully virtualised, 3GPP Release 16-based environment: OpenAirInterface (OAI) core network, UERANSIM, and a custom N2 proxy.

Our contributions.

- *First experimental validation* of the SQN inference attack in a 5G SA environment, with ground-truth verification against the OAI core.
- Experimental confirmation of the SUCI-catcher covering SUCI recording, presence confirmation, and multi-subscriber probing.
- Comparative analysis of both attacks against the 5G privacy model — unlinkability, undetectability, and behavioural privacy.
- Identification of practical operator-level and protocol-level mitigations.

Paper organization. First, Sections 2–4 cover background, related work, and the threat model. In the following, Section 5 describes the testbed and implementation. After that, Section 6 presents results. Finally, Sections 7–8 discuss countermeasures, limitations, and conclude.

2 Background

This section introduces the 5G Standalone architecture, the subscriber identity mechanisms, and the 5G-AKA authentication protocol to the extent required to understand the two attacks evaluated in this paper.

2.1 5G Standalone Architecture

5G is defined by the 3GPP under the ITU IMT-2020 framework, with the first New Radio (NR) specification completed in Release 15 (2019) [17]. Early deployments operate as *Non-Standalone* (NSA), anchoring the 5G radio to existing 4G LTE infrastructure. The focus of this paper is *5G Standalone* (SA), in which a native 5G Core (5GC) independently manages both the control plane and the user plane alongside 5G NR.

The 5GC adopts a *Service-Based Architecture* (SBA) in which modular *Network Functions* (NFs) communicate via REST APIs over HTTP/2, as standardised in the Common API Framework (CAPIF) [11]. Control- and user-plane traffic are separated (CUPS): the *User Plane Function* (UPF) forwards data packets, while the control plane is handled by the following NFs relevant to authentication:

- **AMF** (Access and Mobility Management Function) — terminates Non-Access Stratum (NAS) signalling from the UE and initiates authentication procedures.
- **AUSF** (Authentication Server Function) — validates UE credentials, derives serving-network anchor keys, and interfaces with the home network.
- **UDM/ARPF** (Unified Data Management / Authentication Credential Repository and Processing Function) — holds the home-network private key for SUCI decryption and stores long-term subscriber credentials mirrored on the USIM.
- **SEAF** (Security Anchor Function) — acts as the security-context holder during authentication; holds the root anchor key for visited networks.

The radio access side is provided by the *next-generation base station* (gNB), which splits into a *Centralised Unit* (CU) handling non-real-time protocols and a *Distributed Unit* (DU) for physical-layer and real-time functions. *User Equipment* (UE) consists of *Mobile Equipment* (ME) and a tamper-proof *Universal Subscriber Identity Module* (USIM) stored on a *Universal Integrated Circuit Card* (UICC). The USIM stores the long-term secret key and performs authentication functions, acting as the ultimate trust anchor for the subscriber.

2.2 Subscriber Identity and SUCI

Every 5G subscriber is identified by a *Subscriber Unique Permanent Identifier* (SUPI), equivalent to the legacy IMSI. To prevent the plaintext-IMSI exposure that plagued earlier generations, the USIM conceals the SUPI before any over-the-air transmission, using the *Subscriber Concealed Identifier* (SUCI) [11].

The SUCI is a structured field composed of: *SUPI Type*, *Home Network ID* consisting of Mobile Country Code (MCC) and Mobile Network Code (MNC), *Routing Indicator*, *Protection Scheme ID*, *Home Network Public Key ID*, and a *Scheme Output*. When a non-null protection scheme is active, the scheme output is produced using the *Elliptic Curve Integrated Encryption Scheme* (ECIES): the UE generates an ephemeral key pair, encrypts the MSIN (the subscriber-specific portion of the SUPI) with the home network's public key, and appends a MAC

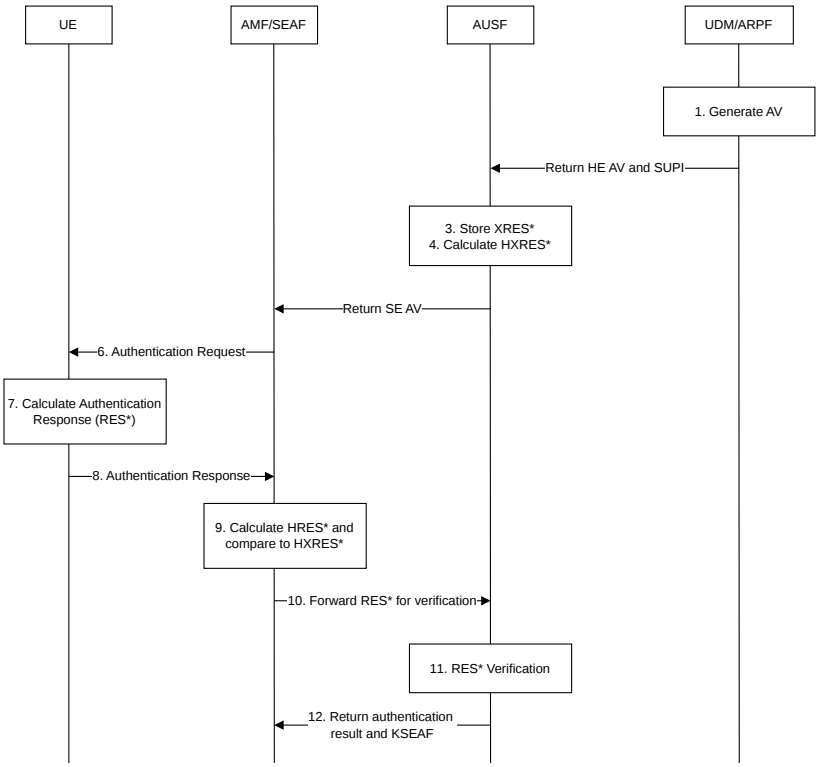

Fig. 1. 5G-AKA protocol overview (based on 3GPP TS 33.501 [11]).

tag. Only the home network's UDM can reverse this using its private key via the *Subscription Identifier De-concealment Function* (SIDF).

The SUCI therefore changes with every registration — providing pseudonymity — but its static fields (Home Network ID, Routing Indicator) remain constant across registrations. Although the scheme outputs are unique, the decrypted content remains identical, a property exploited by the SUCI-catcher attack described in Subsection 5.3.

2.3 5G-AKA Authentication Protocol

The gNB connects to the AMF via the *N2 interface*, carrying NGAP (Next-Generation Application Protocol) signalling using the Stream Control Transmission Protocol (SCTP) over IP [18]. Key N2 messages include the *Initial UE Message* (delivers the NAS Registration Request, containing the SUCI, from the UE to the AMF) and *Downlink / Uplink NAS Transport* (carries subsequent NAS exchanges). NAS itself is divided into 5G Mobility Management (5GMM), which handles authentication and registration, and 5G Session Management (5GSM).

3GPP defines three UE authentication methods: 5G-AKA, EAP-AKA′, and EAP-TLS [11]. This paper focuses on 5G-AKA, the primary method for 5G SA, which involves four entities: the UE, the SEAF, the AUSF, and the UDM/ARPF. The protocol, illustrated in Figure 1, proceeds as follows:

1. **HE AV generation.** The UDM/ARPF generates a 5G Home Environment Authentication Vector (HE AV) comprising a random challenge ($RAND$), an authentication token ($AUTN$), an expected response ($XRES^*$), and a home-network anchor key (K_{AUSF}).
2. **SUCI decoding.** The SIDF at the UDM decrypts the SUCI, revealing the SUPI, which is included in the response to the AUSF.
3. **5G AV derivation.** The AUSF transforms the HE AV into a Serving-network AV by deriving K_{SEAF} from K_{AUSF} and $HXRES^*$ from $XRES^*$.
4. **Authentication Request.** The SEAF forwards $RAND$ and $AUTN$ to the UE inside a NAS Authentication Request message.
5. **UE challenge response.** The USIM verifies the freshness of $AUTN$. If valid, it computes a response (RES) and the ME derives RES^*, K_{SEAF}, and K_{AUSF} locally. The UE returns RES^* in an Authentication Response.
6. **Verification.** The SEAF hashes RES^* and compares it with $HXRES^*$; the AUSF compares RES^* with $XRES^*$. Agreement indicates successful mutual authentication.
7. **Activation.** The AUSF delivers K_{SEAF} to the SEAF, which derives K_{AMF}. The AMF initiates a NAS Security Mode Command, establishing an encrypted and integrity-protected NAS channel.

If the AUSF or SEAF verification of RES^* fails, the SEAF either rejects the authentication or requests a fresh SUCI via an Identity Request. If the USIM's verification of $AUTN$ fails, the UE sends an *Authentication Failure* whose cause distinguishes two cases relevant to the attacks in this paper: a *MAC failure* (challenge bound to a different subscriber, no AUTS parameter) and a *Synchronization Failure* (stale SQN, accompanied by the AUTS resynchronization parameter described below).

2.4 Sequence Number Mechanism

The AKA *Sequence Number* (SQN) is a 48-bit counter used for replay protection [5]. It comprises a 43-bit sequence counter (SEQ) and a 5-bit serving-network index (IND). The home network maintains SQN_{HN} and increments it after each successful authentication; the UE maintains SQN_{UE}, updated to match SQN_{HN} upon every successful challenge response.

The SQN is not transmitted in plaintext. Instead, $AUTN$ includes a concealed value:

$$CONC = SQN_{HN} \oplus AK, \tag{1}$$

where the *Anonymity Key* $AK = f_5(R, K_{\text{IMSI}})$ is derived from the challenge random value R and the subscriber's long-term shared key. Upon receiving $AUTN$, the UE recomputes AK from the same R and recovers $SQN_{HN} = CONC \oplus AK$.

The UE then checks freshness: authentication proceeds only if $SQN_{HN} > SQN_{UE}$ and $SQN_{HN} \leq SQN_{UE} + \Delta$, where Δ is an operator-configured acceptance window. If this check fails, the UE sends an Authentication Failure with cause `Sync Failure`, including an $AUTS$ resynchronization parameter:

$$AUTS = \langle\, SQN_{UE} \oplus AK,\ \ MAC^* \,\rangle, \tag{2}$$

which allows the home network to re-synchronise SQN_{HN} to SQN_{UE}.

The critical observation for the SQN inference attack is that, when the same R is reused in two replayed challenges, the anonymity key AK is identical in both AUTS responses. XOR-ing the two $CONC$ values therefore cancels AK:

$$CONC_1 \oplus CONC_2 = (SQN_{UE} \oplus AK) \oplus (SQN'_{UE} \oplus AK) = SQN_{UE} \oplus SQN'_{UE}, \tag{3}$$

leaking the difference between two SQN states and enabling bit-by-bit inference of SQN_{UE}, as detailed in Section 3.

3 Related Work

The *IMSI-catcher* is a well-documented attack spanning 2G, 3G, and 4G networks [10]. The attack deploys a rogue base station that attracts nearby UEs by broadcasting a stronger signal than the legitimate network. Once the UE associates with the rogue station, the attacker sends an Identity Request or awaits a Registration Request; because earlier generations transmit the IMSI in plaintext, the UE transmits with the IMSI directly. This enables persistent tracking and linkability of a Person of Interest (POI) without any cryptographic capability on the attacker's side.

Hussain et al. [12] demonstrated a systematic, model-based approach to uncovering pre-authentication vulnerabilities in 4G LTE with LTEInspector, identifying several protocol-level weaknesses that persist even with partial encryption. 5G SA was designed to close the IMSI exposure vector by replacing plaintext IMSI transmission with SUCI concealment, motivating the evolution to the SUCI-catcher class of attacks described below.

The SUCI-catcher exploits a logical vulnerability in 5G-AKA: initial NAS signalling messages are transmitted before the NAS Security Mode Command establishes encryption and integrity protection. Although SUCI prevents direct SUPI exposure, it does not prevent an adversary from *linking* a SUCI to a specific subscriber through the authentication protocol's response behaviour [8].

The attack operates in two phases. During the *discovery phase*, the adversary obtains a SUCI associated with the target subscriber, either by passively sniffing a prior Registration Request or by computing a SUCI from a known SUPI using the home network's public key. During the *attack phase*, the adversary operates a rogue gNB or an N2 proxy, intercepts a live Registration Request, replaces the SUCI with the previously recorded one, and forwards it to the core network. The network issues an authentication challenge bound to the target subscriber. The UE's response reveals its identity: an *Authentication Response* confirms that the UE owns the injected SUCI; an *Authentication Failure* with MAC cause indicates it does not.

Chlosta et al. [8] introduced the foundational SUCI-catcher, proving experimentally that SUCI concealment alone does not guarantee unlinkability, anonymity, or undetectability. The paper also characterises the *Reset & Sync* mechanism and evaluates a multi-POI probing scenario in a lab environment. Cabrera and Gallego [6] extended the attack to commercial deployments in Germany, the UK, and Spain via the SUCI-Cracker variant, which leverages a 4G discovery phase to obtain subscriber-specific authentication material for use in the 5G probing phase. Barraud et al. [3] implemented the attack in an open-source 5G testbed and shifted focus to *detection*, proposing rule-based detection of registration-request spikes and ML-based anomaly classification. A complementary protocol fix, 5G-AKA′, was proposed by Wang et al. [20]: it applies ECIES to encrypt portions of the authentication challenge, rendering replayed challenges indistinguishable to the UE and removing the observable response difference the SUCI-catcher relies upon.

Collectively, the existing literature shows that SUCI-catcher attacks are technically feasible and applicable to commercial 5G SA deployments. However, prior work focuses primarily on the attack mechanism, protocol behaviour, and detection; less attention has been paid to systematically evaluating which 5G privacy properties are violated in a controlled experimental environment.

The SQN inference attack exploits the synchronization-failure mechanism to recover bits of the subscriber's sequence counter. When the same RAND is reused across two replayed challenges, AK is identical in both AUTS responses; XOR-ing the two concealed values cancels AK, exposing the XOR of two SQN states (Equation 3).

Borgaonkar et al. [5] introduced and cryptographically analysed this vulnerability, validated it on a 4G environment, and noted that the attack is theoretically applicable to 5G SA under 3GPP Release 16 but did not provide experimental validation in 5G SA, leaving a gap between standard-level analysis and practical confirmation. Cheng and Shen [7] presented a concrete tracking scenario in an srsLTE testbed, demonstrating that inferred SQN changes can reveal subscriber presence and behaviour patterns, and proposed a countermeasure requiring the UE to generate a fresh RAND in the AUTS. Whereas the SUCI-catcher has been validated in real 5G SA deployments, the SQN inference attack has not been experimentally confirmed in a 5G SA environment prior to this work.

Several works analyse 5G-AKA through formal methods. Basin et al. [4] performed the first comprehensive formal verification using the Tamarin prover, confirming stronger identity-privacy guarantees than 4G-AKA while revealing that certain privacy properties depend on assumptions not stated in the standard. Koutsos [14] formally analysed unlinkability and anonymity, showing that residual linkability remains possible when an adversary can trigger observable protocol-state differences — precisely the mechanism exploited by both attacks in this paper. Hussain et al. [13] uncovered eleven design-level weaknesses in 5G control-plane protocols with 5GReasoner, several of which expose privacy-relevant information through side-channel responses akin to the synchronization-failure leakage studied here. Most recently, Eleftherakis et al. [9] systematised knowledge across all known 5G-Advanced privacy and security attacks, confirming that authentication-layer behavioural leakage remains an open problem in Release 18 and beyond.

Three gaps motivate this paper:

1. The SQN inference attack has been theorised but never experimentally validated in a 3GPP-compliant 5G SA environment.
2. No prior work provides a *joint* experimental evaluation of the SUCI-catcher and SQN inference attack in the same 5G SA testbed, preventing a direct comparison of preconditions, scalability, and privacy impact.
3. Existing implementations provide limited engineering detail on translating attack descriptions into working prototypes against standards-compliant 5G components, reducing reproducibility.

This paper addresses all three gaps through controlled experimentation in a fully virtualised 5G SA testbed, as described in Section 5.

4 Threat Model

4.1 Attacker Position

We consider an adversary positioned on the *N2 interface* between the gNB and the AMF, implemented as a transparent NGAP proxy. Figure 2 illustrates the architecture: The gNB is configured to direct its N2 connection to the proxy, which forwards all traffic to the legitimate AMF. The proxy operates exclusively on the N2 interface; all other core-network interfaces (N11, N4) and the user-plane interface (N3) remain unmodified.

This proxy position is equivalent to a *man-in-the-middle* on the N2 interface and is representative of a scenario in which an adversary has compromised the backhaul link between the base station and the core network [13,4]. Notably, the attacker does *not* require physical radio access or a rogue base station: the N2 proxy intercepts and manipulates NGAP signalling at the software level, making the attack more accessible than a rogue-gNB deployment.

4.2 Attacker Capabilities

The attacker is an *active* adversary with the following capabilities on the N2 interface:

- **Intercept** — read all NGAP messages, including NAS payloads encapsulated in Initial UE Message and Downlink/Uplink NAS Transport messages.
- **Modify** — alter NAS content before forwarding; specifically, replace the SUCI field in a Registration Request.
- **Replay** — retransmit captured NGAP messages, including Authentication Requests, to induce controlled UE responses (Authentication Response or Sync Failure). Also replay of Initial UE Message to collect Authentication Request for both attacks.
- **Delay** — hold messages temporarily to serialise or coordinate attack steps.

The attacker does *not* possess the subscriber's SUPI or long-term key, the home network's private key, or the ability to decrypt NAS traffic (which is protected only after the Security Mode Command — but all exploitable signals

occur before this point). For the SUCI-catcher discovery phase, the attacker requires a previously captured SUCI or knowledge of the target's SUPI and the public home-network key. For the SQN inference attack, the attacker requires the ability to trigger repeated authentication interactions with the same UE.

4.3 Privacy Assets at Risk

3GPP TS 33.501 [11] defines five subscriber privacy properties that 5G SA is designed to enforce. Table 1 maps these properties to the two attacks.

Table 1. 5G privacy properties violated by each attack. ✓ = violated; – = not violated.

Privacy Property	SUCI-Catcher	SQN Inference
Identity privacy	–	–
Location privacy	✓	✓
Unlinkability	✓	✓
Undetectability	✓	✓
Behavioural privacy	–	✓

Identity privacy is not violated: neither attack recovers the SUPI. **Location privacy** is violated by both: confirming a subscriber is active at a gNB reveals cell-level location. **Unlinkability** is violated by both: the SUCI-catcher links a known SUCI to an active session; SQN tracking correlates successive observations. **Undetectability** is the primary goal of both attacks. **Behavioural privacy** is additionally broken by SQN inference: the SQN increments on each authentication event, so the inferred value between two observations reveals subscriber activity volume.

5 Testbed and Attack Implementation

5.1 Testbed Architecture

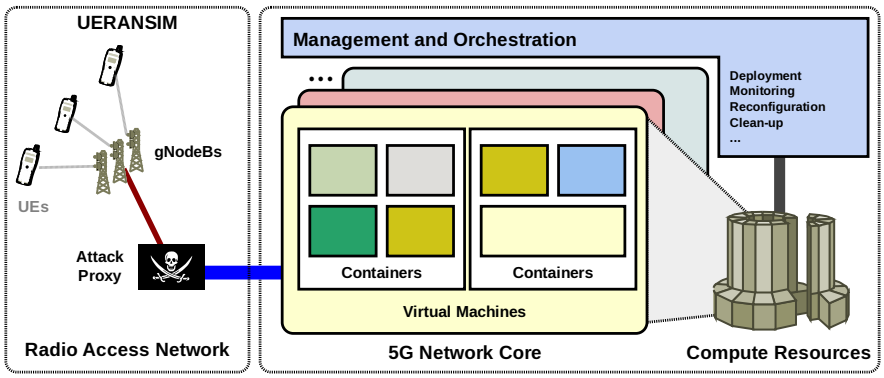

Fig. 2. The 5G testbed setup used for the attack evaluation.

Figure 2 illustrates our testbed setup: The OpenAirInterface[3] (OAI) 5G-SA core is instantiated as containers in a virtual machine setup within a data centre. The components of the core basically use the upstream OAI implementation, with some debugging, monitoring, and maintenance tools added into the containers. A detailed description of this setup, including Git repository links of the implementation, is provided in [2]. Since the 5G-SA core is left unmodified, to mimic a "normal" 5G network provider, and just works as defined in the standard, we omit a detailed description here.

Connected to the core is the RAN simulation setup using RANSIM. Since we do not need the actual RAN communication, the RAN part is also fully virtualised as container in another virtual machine. Particularly, the NGAP communication session between simulated gNB and 5G-SA core over SCTP is intercepted by the attack proxy.

5.2 N2 Proxy Design

The proxy runs on a `Python:3.11-slim` container, using mainly the library `pycrate` [15] with SCTP to read and send NGAP messages. This is based on architecture presented in related work, but no implementation could be found. The proxy maintains two concurrent SCTP associations — one towards the gNB and one towards the AMF — and decodes all NGAP PDUs before forwarding. Core capabilities include:

- **NGAP parsing** — full decoding to extract NAS-PDUs from Initial UE Message and NAS Transport containers.
- **NAS field access** — extraction and replacement of individual NAS information elements, including the Mobile Identity field carrying the SUCI.
- **Message injection and replay** — holding, duplicating, or re-injecting captured NGAP messages in a controlled sequence.
- **Session state tracking** — per-UE context (SUCI, AMF-UE-NGAP-ID, RAN-UE-NGAP-ID) to correctly route modified messages.

A key implementation constraints are the NAS *T3560 timer* and UE fail attempt counter: consecutive invalid authentication challenges or timeout to complete registration cause the UE to stop responding [1]. Both attacks are required to stay within the time window given by the timeout setting. The SUCI-catcher therefore implements a *Reset & Sync* step — injecting one valid Authentication Request between rounds — to maintain the UE session.

5.3 SUCI-Catcher Implementation

Phase 1: Discovery. The proxy passively monitors uplink N2 traffic. Each Initial UE Message carries a NAS-PDU whose Mobile Identity field contains the SUCI. The proxy extracts and stores each observed SUCI with a timestamp and session identifiers, requiring no active interference.

Phase 2: Attack. When a new UE registers, the proxy intercepts the Initial UE Message, replaces the Mobile Identity with a stored target SUCI, re-encodes

[3] OpenAirInterface (OAI): `https://openairinterface.org`.

the NGAP PDU, and forwards it to the AMF. The AMF generates an Authentication Request bound to the target subscriber and sends it to the UE. The outcome is binary:

- **Authentication Response** — the challenge was valid for this UE; the target subscriber is present.
- **Authentication Failure (MAC cause)** — the challenge was not directed at this UE; the injected SUCI belongs to a different subscriber.

After each probe, a valid Authentication Request is injected to resynchronise the UE before the next round.

5.4 SQN Inference Implementation

The attack recovers the j-th bit of SQN_{UE} through XOR-cancellation of two controlled Sync Failure responses:

1. **Vector collection.** Trigger 2^j Registration Requests; the CN increments SQN_{HN} by 1 per round, producing 2^j fresh Authentication Requests.
2. **First valid exchange.** Forward $(RAND_0, AUTN_0)$ to the UE. The UE accepts and updates SQN_{UE} to SQN_{HN}.
3. **Forced failure.** Inject an Authentication Request with a fabricated RAND. The UE rejects it naturally via MAC failure, resetting its stored authentication state.
4. **Baseline Sync Failure.** Replay $(RAND_0, AUTN_0)$. The UE returns:
$$AUTS_0 = \langle\ SQN_{UE} \oplus AK_0,\ MAC_0^*\ \rangle$$
5. **Offset exchange.** Forward the stored Auth. Request with $SQN_{HN} + 2^j$; the UE accepts and updates SQN_{UE} to $SQN_{HN} + 2^j$.
6. **Forced failure.** Re-trigger authentication failure to prevent UE from accepting old data.
7. **Second Sync Failure.** Replay $(RAND_0, AUTN_0)$ again:
$$AUTS_1 = \langle\ (SQN_{UE} + 2^j) \oplus AK_0,\ MAC_1^*\ \rangle$$
8. **Bit recovery.** XOR the two $CONC^*$ values; AK_0 cancels, yielding:
$$CONC_0^* \oplus CONC_1^* = SQN_{UE} \oplus (SQN_{UE} + 2^j) \tag{4}$$
Carry propagation reveals whether bit j is 0 or 1.

Recovery cost grows as 2^n+ Authentication Requests for the first n bits, scaling exponentially with bit depth and bounded in practice by NAS timers and CN rate limits.

5.5 Experimental Scenarios

Scenarios 0–3 cover the SUCI-catcher, from passive recording through multi-target probing. Scenarios 4–5 cover the SQN inference attack, from bit recovery through activity tracking. Scenario 6 characterises the practical boundaries of both attacks under NAS timer and CN rate-limiting constraints. A summary of the scenarios is provided in Table 2.

Table 2. Experimental scenarios and their attack mapping.

No.	Scenario	Attack	Goal
0	Baseline Authentication	None	Verify testbed compliance
1	SUCI Recording	SUCI-catcher	Discovery feasibility
2	Single SUCI Replacement	SUCI-catcher	Presence confirmation
3	Multi-SUCI Probing	SUCI-catcher	Probing scalability
4	SQN Inference Attack	SQN inference	LSB recovery
5	Activity Tracking via SQN	SQN inference	Behavioural leakage
6	Scalability and Limits	Both	Attack boundaries

6 Experimental Results

All seven scenarios executed successfully on the testbed described in Subsection 5.5. Table 3 summarises each scenario's primary outcome and privacy finding; the subsections below detail the key evidence.

Table 3. Summary of scenario outcomes.

No.	Primary outcome	Privacy finding
0	Baseline confirmed	5G-AKA / SUCI compliant
1	6/6 SUCIs recorded	Discovery phase feasible
2	Presence confirmed	Location privacy broken
3	POI found (probe 3)	Unlinkability broken
4	SEQ = 172 recovered	SQN inference feasible
5	ΔSEQ = 3 observed	Behavioural leakage confirmed
6	Incomplete at 11 bits	Bounded by NAS timers

6.1 Baseline Compliance and SUCI Recording (Scenarios 0–1)

Scenario 0. Without proxy interference, the complete 5G-AKA message sequence was observed in Wireshark, shown in Figure 3. The Registration Request carried a correctly formed SUCI using ECIES scheme profile A (scheme 1), with the MSIN concealed under an ephemeral elliptic-curve key. The Authentication Request appeared in plaintext. UERANSIM confirmed the initial SQN as `0x40` (decimal 64) and a PDU session was established successfully.

Scenario 1. With the proxy in passive recording mode, six UEs registered sequentially, each with a distinct SUPI. The proxy extracted every SUCI from the corresponding *InitialUEMessage* and stored it in JSON format without modifying any message or disrupting any authentication exchange. All six UEs authenticated and established PDU sessions identically to Scenario 0. Table 4 shows the recorded SUCI structure; the ECC ephemeral key, ciphertext, and MAC differ per registration, confirming ECIES freshness.

6.2 SUCI-Catcher Attack (Scenarios 2–3)

Scenario 2. A single stored SUCI (UE 031, index 0) was injected into a live Registration Request while the genuine UE was present. The AMF generated an

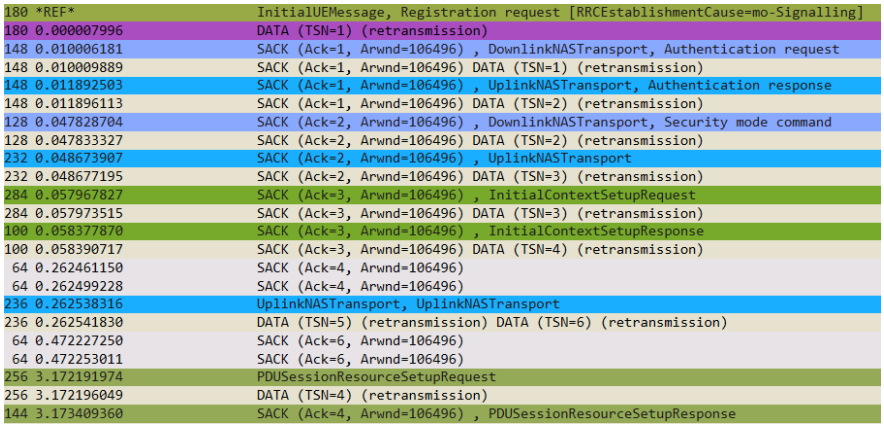

Fig. 3. Wireshark screenshot of UE registration and authentication

Table 4. SUCI components recorded in Scenario 1 (all: PLMN = 20895, routing = 0210, scheme = 1).

UE	ECCEphemPK (prefix)	CipherText	MAC (prefix)
031	`676130f3...`	`588798faae`	`21652595...`
032	`26cb957d...`	`04d3034d74`	`6324e035...`
033	`a196dc1d...`	`8c9d3eded9`	`f45a4f41...`
034	`6c916921...`	`2c1780887c`	`917ec4f0...`
035	`1e2e9d16...`	`7c22404638`	`3418a2d7...`
036	`d76c3d1b...`	`9b0139fd5c`	`78aaa1ed...`

Authentication Request for the target subscriber, and the UE responded with a valid *Authentication Response*, confirming its presence. Proxy log:

```
Forwarding InitialUEMessage with SUCI 0
Replaced SUCI with: {...}
Authentication Response - RAN-UE-NGAP-ID: 1
UE successfully found 0
```

A single probe was sufficient to confirm subscriber presence, violating location privacy and undetectability.

Scenario 3. Six SUCIs were probed sequentially; the matching SUCI was at index 4. Each non-matching SUCI produced an *Authentication Failure* (`MAC_FAILURE`), while the matching SUCI produced an *Authentication Response*. After two consecutive failures the Reset & Sync step was triggered automatically. Proxy log trace:

```
Forwarding InitialUEMessage with SUCI 0  -> MAC FAILURE
Forwarding InitialUEMessage with SUCI 1  -> MAC FAILURE
Reset & Sync
Forwarding InitialUEMessage with SUCI 2  -> MAC FAILURE
```

Table 5. SQN bit-inference results (Scenario 4).

Probe j	Bit pos.	Delta ($CONC_0^* \oplus CONC_j^*$)	Bit
0	6	0x000000000040	0
1	7	0x000000000080	0
2	8	0x000000000700	1
3	9	0x000000000600	1
4	10	0x000000000400	0
5	11	0x000000001800	1
6	12	0x000000001000	0
7	13	0x000000006000	1

```
Forwarding InitialUEMessage with SUCI 3  -> MAC FAILURE
Reset & Sync
Forwarding InitialUEMessage with SUCI 4  -> AUTH RESPONSE
UE successfully found 4
```

The UE eventually deregistered due to NAS timer T3510 expiry, but the subscriber match was recorded beforehand. The UE remained connected through modified session and the subscriber match was recorded. The binary SUCI-probing oracle is reliable: non-matching and matching SUCIs produce unambiguously different protocol responses, and unlinkability is broken because a connection attempt can be tied to a known stored SUCI.

6.3 SQN Inference Attack (Scenarios 4–6)

Scenario 4. The OAI core was pre-loaded with SQN = 0x2B00, SEQ = 172 (the number of past AKA sessions). The proxy replayed 257 Registration Requests to obtain 257 authentication vectors, then executed the AUTS-based bit-recovery procedure (Subsection 5.4). Nine AUTS values were collected — one baseline plus one per bit position ($j = 0, \ldots, 7$, targeting SQN bits 6–13) — and XOR-cancellation yielded the delta values in Table 5.

The recovered bit string `00110101` (bits 6–13) corresponds to an inferred SEQ of 172 and can be extend to a SQN value of 0x2B00 with the IND, matching the ground-truth OAI database value exactly:

```
Inference results
Bit string: 00110101
Inferred SEQ: 172 (amount of past AKA sessions)
Inferred SQN_HN: 11008  (0x000000002b00)
```

After bit inferrence, the UE recovered and completed registration. Later *Authentication Reject* messages from leftover registration attempts were blocked by the proxy and did not reach the UE.

Scenario 5. Two SQN measurements were taken around three legitimate authentication sessions. The first inference yielded SEQ = 1; after the four sessions the inference was repeated:

```
Bit string: 10000000   Inferred SEQ: 1
     [3 legitimate authentication events]
Bit string: 10100000   Inferred SEQ: 5
```

The inferred SEQ increased by 4 (one more than the three explicitly simulated sessions; the extra increment stems from an additional Authentication Request at the end of the first inference round needed to connect UE). The change in inferred SEQ directly reflects UE activity, demonstrating that an adversary can estimate the number of authentication events a subscriber performs between observations without recovering the absolute SQN value — a violation of behavioural privacy and undetectability.

Scenario 6. Extending inference to 11 bits requires 2049 authentication vectors. The collection phase took significantly longer, and the NAS timer T3510 expired before the baseline challenge could be delivered. Multiple spontaneous UE re-registrations interspersed themselves during collection, injecting new NGAP sessions that disrupted the controlled message sequence and caused the collection counter to reset:

```
Setup: collected auth vector 2048/2049
Initial UE Message - RAN-UE-NGAP-ID: 2
Setup: collected auth vector 2008/2049
```

In some runs, a short sequence of replayed Registration Requests were rejected with `ILLEGAL_UE`, terminating the UE's connection to the network. This appears to stem from unexpected CN behavior, with the likelihood increasing as the number of Authentication Requests grows.

Bit recovery did not complete reliably, confirming that the practical limit is approximately 8 bits under 3GPP NAS timer constraints. The exponential cost ($2^{n+1}-2$ vectors for n bits) eventually exhausts the session window, establishing a measurable boundary on the attack's reach without protocol-level changes.

7 Discussion

7.1 Privacy Impact

The experimental results confirm that, within the controlled testbed environment, residual privacy leakage in 5G SA arises from observable differences in protocol behaviour rather than failures of SUCI concealment itself; their applicability to operational deployments is subject to the limitations discussed in Subsection 7.3.

The SUCI-catcher exploits the binary outcome of the 5G-AKA exchange as: the UE's distinguishable response to a correctly or incorrectly targeted challenge was shown to reveal whether the subscriber is present in the cell, violating location privacy and unlinkability without recovering the SUPI.

The SQN inference attack exploits the AUTS resynchronisation mechanism. XOR-cancellation of two AUTS values collected under the same RAND recovers individual bits of SQN_{UE}, demonstrating the feasibility of behavioural tracking — changes in the inferred SQN between observations reveal the subscriber's authentication activity — and cross-session linkability via a distinctive SQN fingerprint. This violates undetectability and unlinkability (Table 1).

The two attacks differ in network footprint. SUCI probing scales linearly with the candidate population and produces low, inconspicuous traffic. SQN inference requires $2^n + 2$ requests per n bits, generating an anomalous burst detectable by the core network [19], and is practically bounded at $\approx$8 bits by NAS timer constraints in this implementation.

7.2 Countermeasures

The observed attack behaviour suggests both near-term operational mitigations and longer-term protocol-level remedies.

Rate limiting authentication-vector generation. Both attacks depend on the core network servicing a large number of authentication requests from the same gNB or for the same SUCI. Operators can enforce per-gNB and per-SUCI rate limits on AUSF/UDM vector generation to cap the number of vectors an attacker can collect within a session window, directly constraining the SQN inference bit depth and slowing SUCI probing.

Anomaly detection on authentication patterns. The SQN inference attack produces a distinctive trace: hundreds of Registration Requests replayed with the same SUCI, followed by alternating Authentication Responses and Synchronisation Failures. Core-network intrusion detection can flag repeated Sync Failures for a single subscriber or abnormally high registration rates from a single gNB, consistent with the detection approach of Tucker et al. [19].

Binding SUCI freshness to SQN state. The SUCI-catcher succeeds because a recorded SUCI remains valid across multiple authentication rounds. Incorporating a UE-side freshness indicator — such as a counter derived from SQN_{UE} — into the SUCI construction, and verifying it at the home network, would invalidate replayed SUCIs. This is consistent with the direction proposed by Wang et al. [20], whose privacy-preserving AKA protocol addresses linkability in the authentication exchange.

Randomised AUTS concealment. Replacing the deterministic f_5 anonymity-key derivation with a randomised function would prevent XOR-cancellation across AUTS values, blocking SQN inference [5].

7.3 Limitations

The evaluation is conducted on a software-simulated testbed and is subject to several limitations. First, the testbed models a single-cell environment; real deployments involve inter-cell mobility and multiple AMF instances, which may affect attack timing and session stability. Second, UERANSIM initially required a modification to suppress AUTN verification for the SQN inference attack; a hardware USIM would enforce this check and may behave differently under out-of-range challenges. The fabricated RAND forced-failure step described in Subsection 5.4 removes this dependency with negligible impact on timing and traffic overhead; however, the results presented are based on the modified UERANSIM. Third, the observed NAS timer constraints represent UERANSIM's default configuration. Operator deployments may use different timer values that expand or shrink the practical attack window. Fourth, the N2 proxy design already assumes a privileged placement, and operator-side rate limiting on authentication vector generation would further constrain practical attack reach.

7.4 Ethical Considerations

All experiments were conducted in a fully isolated, software-defined testbed with no connection to any live operator network. No real subscriber credentials, SIM cards, or user data were used at any point. The SUCI and SQN values processed are synthetic, generated by UERANSIM against a locally-deployed OAI core. The attacks described pose no risk to real subscribers or network infrastructure.

8 Conclusion

This paper experimentally evaluated two residual privacy attacks against 5G Standalone authentication: the SUCI-catcher and the SQN inference attack. Using a transparent N2-interface proxy on a 3GPP-compliant open-source testbed, the study showed that SUCI concealment prevents direct identity disclosure but does not eliminate indirect privacy leakage through observable authentication behaviour.

The SUCI-catcher enables subscriber presence confirmation through the binary outcome of the 5G-AKA challenge-response exchange, while the SQN inference attack can recover up to 8 bits of the subscriber sequence number through controlled replay and AUTS collection. The results show that SUCI probing is scalable and stealthy, whereas SQN inference is practically limited by NAS timer constraints.

Overall, the findings demonstrate that identity protection and unlinkability are distinct privacy goals in 5G SA. Stronger unlinkability requires protocol-level improvements, such as SUCI freshness binding and randomised AUTS concealment.

Acknowledgments. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. 3GPP: 5G; Non-Access-Stratum (NAS) Protocol for 5G System (5GS); Stage 3. Technical Specification TS 124 501 V15.1.0, ETSI (2018), https://www.etsi.org/deliver/etsi_ts/124500_124599/124501/15.01.00_60/ts_124501v150100p.pdf
2. Ahmed, A.H., Dreibholz, T., Michelinakis, F.I., Čičić, T.: Open 5G Testbed: A Cyber Range Platform for Security Research. In: 2025 Annual Computer Security Applications Conference Workshops (ACSAC Workshops). pp. 579–586. IEEE (2025)
3. Barraud, L., Caccavale, F., Peyrat, J.B., Malouli, W., Capdevielle, V., Khalife, H., Cavalli, A.R.: 5G SUCI Catcher: Attack and Detection. In: 2023 IEEE International Conference on Cloud Computing Technology and Science (CloudCom). pp. 285–290. IEEE (2023). https://doi.org/10.1109/CloudCom59040.2023.00053
4. Basin, D., Dreier, J., Hirschi, L., Radomirović, S., Sasse, R., Stettler, V.: A Formal Analysis of 5G Authentication. In: Proceedings of the 25th ACM Conference on Computer and Communications Security (CCS 2018). pp. 1383–1396. ACM (2018). https://doi.org/10.1145/3243734.3243846

5. Borgaonkar, R., Hirschi, L., Park, S., Shaik, A.: New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols. Proceedings on Privacy Enhancing Technologies (PoPETs) **2019**(3), 108–127 (2019). https://doi.org/10.2478/popets-2019-0039
6. Cabrera, P., Gallego, M.: 5G SUCI Probing in the Wild (2024)
7. Cheng, Y.C., Shen, C.A.: A New Tracking-Attack Scenario Based on the Vulnerability and Privacy Violation of 5G AKA Protocol. IEEE Access **10**, 77679–77687 (2022). https://doi.org/10.1109/ACCESS.2022.3193372
8. Chlosta, M., Rupprecht, D., Pöpper, C., Holz, T.: 5G SUCI-Catchers: Still Catching Them All? In: Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec 2021). pp. 359–364. ACM (2021)
9. Eleftherakis, S., Giustiniano, D., Kourtellis, N.: SoK: Evaluating 5G-Advanced Protocols Against Legacy and Emerging Privacy and Security Attacks. In: Proceedings of the 18th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec 2025). pp. 196–210. ACM (2025). https://doi.org/10.1145/3734477.3734716
10. Eleftherakis, S., Otim, T., Santaromita, G., Zayas, A.D., Giustiniano, D., Kourtellis, N.: Demystifying Privacy in 5G Stand Alone Networks. In: Proceedings of the 30th Annual International Conference on Mobile Computing and Networking (MobiCom 2024). pp. 1330–1345. ACM (2024)
11. ETSI: ETSI TS 133 501 V17.11.0 — Technical Specification: 5G System; Security Architecture and Procedures for 5G System. Technical Specification TS 133 501 V17.11.0, ETSI (2023), https://www.etsi.org/deliver/etsi_ts/133500_133599/133501/17.11.01_60/ts_133501v171101p.pdf
12. Hussain, S.R., Chowdhury, O., Mehnaz, S., Bertino, E.: LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE. In: Proceedings of the 25th Annual Network and Distributed System Security Symposium (NDSS 2018). Internet Society (2018), https://www.ndss-symposium.org/wp-content/uploads/2018/02/ndss2018_02A-3_Hussain_paper.pdf
13. Hussain, S.R., Echeverria, M., Karim, I., Chowdhury, O., Bertino, E.: 5GReasoner: A Property-Directed Security and Privacy Analysis Framework for 5G Cellular Network Protocol. In: Proceedings of the 26th ACM Conference on Computer and Communications Security (CCS 2019). pp. 669–684. ACM (2019). https://doi.org/10.1145/3319535.3354263
14. Koutsos, A.: The 5G-AKA Authentication Protocol Privacy. In: 2019 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 464–479. IEEE (2019)
15. Michau, B.: pycrate (2026), https://github.com/pycrate-org/pycrate
16. Mishra, S., Zanella, A.F., Martínez-Durive, O.E., Madariaga, D., Ziemlicki, C., Fiore, M.: Characterizing 5g Adoption and Its Impact on Network Traffic and Mobile Service Consumption. In: IEEE INFOCOM 2024-IEEE Conference on Computer Communications. pp. 1531–1540. IEEE (2024)
17. Nair, P.: Securing 5G and Evolving Architectures. Addison-Wesley Professional (2021)
18. Silveira, L.B.D., Resende, H.C.D., Both, C.B., Marquez-Barja, J.M., Silvestre, B., Cardoso, K.V.: Tutorial on Communication Between Access Networks and the 5G Core. Computer Networks **216**, 109301 (2022)
19. Tucker, T., Brotman, A., Bhattacharya, A., Kloti, R., Noubir, G.: Detecting IMSI-Catchers by Characterizing Identity Exposing Messages in Cellular Traffic. In: Proceedings of the 2025 Network and Distributed System Security Symposium (NDSS 2025). Internet Society (2025). https://doi.org/10.14722/ndss.2025.241115
20. Wang, Y., Zhang, Z., Xie, Y.: Privacy-Preserving and Standard-Compatible AKA Protocol for 5G. In: Proceedings of the 30th USENIX Security Symposium (USENIX Security 2021). pp. 3595–3612. USENIX Association (2021)