Lab Seed: SEC-101 5G-AKA Identity-Privacy Analysis (Optional Advanced Extension)

2,972 words · ~14 min read

---

*Optional advanced-extension lab seed for `vca-sec-101` Cybersecurity Principles. Pulls forward FROM the canonical 5G-AKA prose at `vca-net-301` Ch 8 §8.8.2. Does not re-derive primary content. Companion to `handouts/cross-chapter-wireless-aka-progression.md` (CT-A; reads 5G-AKA as the cellular-track endpoint of the wireless-AKA progression). Anchors: Kurose & Ross, *Computer Networking: A Top-Down Approach*, 9th ed., Pearson, 2021, §8.8.2 (5G-AKA / SUCI / SUPI / IMSI-catcher closure) + §7.4.3 (5G Core / AMF / AUSF / UDM control-plane decomposition); 3GPP TS 33.501 (5G security architecture; SUCI / SUPI / 5G-AKA normative). Title: "5G-AKA Identity-Privacy Analysis - From IMSI-Catchers to SUCI." *

Purpose: an optional advanced-extension lab seed for SEC-101 students who want to walk the 5G-AKA protocol at the *implementation* register. Implementing the SUCI computation chain in Python, analyzing 4G LTE captures that show the IMSI-catcher attack surface 5G was redesigned to close, and tracing the residual attack surface (downgrade attacks, sequence-number-desync replay, AMF-to-home-network trust assumptions) at the SEC-101 CVE-walk catalog. **Pedagogical thesis:** *the SEC-101 CVE-walk skill matures into protocol-design literacy when the student can read 9e §8.8.2 + §7.4.3, understand why the SUCI design closes the IMSI-catcher class structurally, and reproduce the SUCI computation chain end-to-end against a real 5G NAS exchange.* This is the lab where SEC-101's Week-5 cryptographic-primitive vocabulary (AES-GCM, ECDH/X25519, KDF) becomes operational against a contemporary protocol.

This is a **lab seed**, not a lab manifest. The seed is the structural scaffolding (phase arc + deliverable rubric + cross-references); the running lab's pinned tool versions, canned-capture set, and instructor-side answer key are downstream artifacts the SEC-101 instructor team builds out before the lab runs in cohort. The seed is the canonical reference for what the lab *is*; the instructor manifest is the canonical reference for how to run it.

Picks up at `vca-sec-101.html` Lab Manifest (optional advanced-extension entry) and the WIR-101 skill-transfer row. Cross-references `vca-net-301.html` Ch 8 §8.8.2 sample weave (canonical primary 5G-AKA prose) and `handouts/cross-chapter-wireless-aka-progression.md` (CT-A; AKA-progression sidebar context).

At a glance

Property	Value
Course / module	`vca-sec-101` Cybersecurity Principles. Optional advanced-extension lab (post-Week 12 placement; pairs with the historical-CVE capstone Week 13-14 timeline as a parallel-track option)
Bloom register	**L3 Apply** (implement SUCI computation chain) + **L4 Analyze** (residual attack surface; protocol-design rationale)
Wall-time	~6-8 hr indep practice across three phases (A: ~1 hr review; B: ~3-4 hr build; C: ~2-3 hr analysis)
Prerequisites	SEC-101 Week 5 (cryptographic-primitive literacy: AES-GCM, ECDH, KDF) + Week 11 (CVE-record reading) + NET-101 (Wireshark / pcap-reading fluency) + FND-102 (Python fluency at the `cryptography` package register)
Recommended companion	`vca-net-301` Ch 8 §8.8.2 sample weave (canonical primary prose; **must be read first**) + `handouts/cross-chapter-wireless-aka-progression.md` (CT-A AKA-progression sidebar)
Substrate	Python 3.11+ with `cryptography` package (the same package SEC-101 Week 5 uses); 4G LTE NAS captures (canned set; sources below) + 5G NAS captures (canned set + optional OpenAirInterface live trace)
Deliverables	(1) Working SUCI-decoder + encoder Python script with 3GPP test-vector pass; (2) ~3-page residual-attack-class analysis report at SEC-101 CVE-walk catalog
Voice family	SEC-101 lab-walk register (parallels Lab 4 hash-and-crypto-misuse + Lab 8 CVE-record-walk); phase-arc structure parallels the academy's CVE-2026-34971 sandbox-escape multi-lab Phase A/B/C shape

---

Pedagogical arc. Three phases

The lab follows a deliberate three-phase arc: **experience the attack surface** before being told why the protocol exists; **build the protocol's defining computation** to internalise the design; **analyze what is still attackable** to mature the CVE-walk catalog from "read a record" into "read a protocol-design rationale."

Phase A: Experience (the IMSI-catcher attack surface), ~1 hr

Goal. Before the student touches any 5G-AKA cryptography, they walk a 4G LTE capture that shows the *clear-text IMSI in NAS Identity-Response messages*, the attack surface that drove the 5G redesign. The phase reuses the NET-101 Wireshark workflow from prior coursework; **no new tooling**.

Activities.

1. Open the canned 4G LTE NAS attach trace (Wireshark dissector applies natively). Locate the `Identity Request` and `Identity Response` NAS messages.
2. Observe the IMSI transmitted in the clear inside the `Identity Response`. Annotate the lab notebook with the byte-level position of the IMSI in the message.
3. Read 9e §8.8.2's framing of the IMSI-catcher attack class: *any party operating a fake base station can trigger Identity-Request frames against nearby UEs and harvest IMSIs.* The catch, the UE has no way to authenticate the base station before the Identity-Response goes out.
4. Cross-reference 9e §8.8.2 with the CT-A AKA-progression sidebar's cellular-side framing: SUCI is the cellular-track endpoint of the AKA-progression arc; IMSI-catcher is the named attack class that drove the redesign.

Deliverable for Phase A. Lab-notebook entry (~300 words) describing what the IMSI-catcher attack class is, what the attack vector is in legacy AKA (4G/3G/2G), and *why* a clear-text long-term identity in the air-interface protocol creates a structural privacy weakness.

Capture sources. Canned 4G LTE NAS attach traces from the academy lab repo (instructor team curates from public-domain sources: srsRAN test traces, Sysmocom NITB demo captures, or comparable public lab manifests). **No live 4G capture required**, the lab is laptop-only per SEC-101's equipment row.

Phase B: Build (the SUCI computation chain), ~3-4 hr

Goal. Given 9e §8.8.2 + 3GPP TS 33.501 normative reference, implement the SUCI computation chain end-to-end: ECIES key agreement (X25519, Profile A) + KDF + AES-GCM encryption of the SUPI → SUCI. Reuse SEC-101 Week 5's `cryptography` package fluency.

Activities.

1. Read 9e §8.8.2 + 3GPP TS 33.501 §6.12 (SUCI calculation procedure). Identify the four cryptographic primitives the procedure composes: (a) X25519 (or secp256r1, Profile B) ECDH key agreement; (b) HKDF for the AES-GCM key derivation; (c) AES-GCM for the SUPI encryption; (d) the ANSI X9.63 KDF as the 3GPP-normative KDF alternative. Default to Profile A (X25519) for the lab.
2. Implement `compute_suci(supi, home_network_public_key) -> suci` in Python using the `cryptography` package:
   - Generate ephemeral ECDH keypair.
   - Derive shared secret via X25519 against the home network's public key.
   - Run HKDF (or ANSI X9.63 KDF) over the shared secret to derive the AES-GCM key + IV.
   - Encrypt the SUPI with AES-GCM.
   - Concatenate ephemeral public key || ciphertext || GCM tag → SUCI bytes.
3. Implement the inverse: `decode_suci(suci, home_network_private_key) -> supi`. The home network is the only party that can run this. It holds the private key matching the public key the UE used.
4. Test the implementation against a known-answer test vector from 3GPP TS 33.501 Annex C (the normative reference includes worked examples). The implementation must round-trip a SUPI through `compute_suci` → `decode_suci` and recover the original SUPI byte-exact.
5. Optional advanced step: replace the canned home-network keypair with a real 5G testbed exchange (OpenAirInterface 5G core demo). Capture the NAS Registration-Request, extract the SUCI bytes, and use the testbed's home-network private key to decode it. (This step requires NET-301-track lab substrate; SEC-101 students who lack OAI access can skip and complete the lab using canned vectors.)

Deliverable for Phase B. Python script (`suci_codec.py` or equivalent) that:

- Implements `compute_suci` and `decode_suci` against the X25519 (Profile A) primitive set per 3GPP TS 33.501 Annex C.

- Passes the 3GPP TS 33.501 Annex C known-answer test vectors.
- Includes inline comments at each cryptographic-primitive call naming which 3GPP TS 33.501 section the call implements (cite-discipline practice; matches SEC-101 CVE-walk catalog's source-citation expectation).

Reading-the-Phase note. This is where SEC-101's "don't roll your own crypto" professional advice (Week 5) coexists with "implement crypto for educational purposes." The lab-notebook reflection should address the distinction: *the student is not deploying production crypto; they are reading a protocol's spec and reproducing it for understanding.* The cryptographic primitives invoked are all from the `cryptography` package's audited implementations; the lab work is the protocol-level composition, not the primitive-level implementation. The same discipline applies in PEN-101 (engagement reporting), RE-101 (vulnerability research), and ADV-101 (CVE-reproduction).

Phase C: Extend (analyze residual attack surface), ~2-3 hr

Goal. Apply SEC-101's CVE-walk catalog to *read 5G-AKA's residual attack surface as protocol-design literature.* The student identifies what 5G-AKA *did not* close (the attack classes that remain even in the post-IMSI-catcher era) and writes the analysis at the same register a CVE record uses.

Activities.

1. Read 9e §7.4.3 on the 5G Core control plane (AMF / AUSF / UDM split). Identify the trust assumption 5G-AKA depends on: *the home network's UDM/AUSF is honest, and the visited network forwards AKA challenges faithfully without altering them.* What attacks remain if the visited network is malicious? What attacks remain if the UDM/AUSF is compromised?
2. Survey the public literature on 5G-AKA residual attacks. The student should locate at least three named attack classes:
   - **Downgrade attacks.** Forced fallback to legacy AKA (4G / 3G / 2G) where the IMSI-catcher class is still applicable. Cross-reference 3GPP downgrade-protection countermeasures.
   - **Sequence-number desync replay.** The SQN tracking at home-network UDM is the replay-protection mechanism; what happens when SQN tracking is desynchronised (legitimately, e.g., USIM swap; or maliciously)? Cross-reference Borgaonkar et al., "New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols" (PETS 2019) for the canonical academic treatment.

    - **AMF-to-home-network trust attacks.** The visited-network AMF mediates between UE and home-network AUSF; what if the AMF is compromised or operates in a hostile-policy jurisdiction? Cross-reference 3GPP discussion on lawful-intercept interfaces and the residual privacy concerns they create.
3. Optional advanced extension: 5G-AKA-prime variant. Read 3GPP TS 33.402 for the WiFi-to-5G-Core integration story; analyze the AKA-prime variant's threat model relative to baseline 5G-AKA. (Cross-reference CT-A AKA-progression handout's mention of 5G-AKA-prime as a forward-pointer.)

Deliverable for Phase C. ~3-page analysis report at SEC-101 CVE-walk catalog documenting:

- The attack class 5G-AKA closes (IMSI-catcher, with the SUCI design as the structural response).
- At least three residual attack classes 5G-AKA does NOT close, each with: (a) attack description, (b) trust assumption being violated, (c) at least one cited primary source (academic paper, 3GPP working-group discussion, or CVE record).
- A "what would we do differently" section (200-400 words; mirrors the SEC-101 capstone Lab 9 reflection structure) proposing what the next-generation cellular AKA standard (post-5G-AKA) might address that 5G-AKA does not.

Reading-the-Phase note. This is where the lab matures from "implement a protocol" to "read a protocol-design rationale." Phase C is the SEC-101 CVE-walk catalog applied to *a protocol that is still in active deployment*; the residual attack surface is where future CVEs will land. Students who complete Phase C have walked the full arc, from "what the protocol exists to fix" (Phase A) through "how the fix works" (Phase B) to "what the fix does not address" (Phase C). That arc IS the practitioner's protocol-literacy register.

Deliverable rubric (Bloom L3-L4)

The lab produces two artifacts, scored separately. **Both must pass for lab-completion credit.** Optional advanced extension; not graded against the SEC-101 final-grade rubric.

Artifact 1. `suci_codec.py` (or equivalent): Phase B

Tier 1. Does it work. The script must:

- Implement `compute_suci(supi, home_network_public_key) -> suci` and `decode_suci(suci, home_network_private_key) -> supi`.
- Round-trip a SUPI through both functions byte-exact.
- Pass at least one 3GPP TS 33.501 Annex C known-answer test vector (Profile A, X25519).

Scripts that fail Tier 1 do not pass the artifact; no further scoring is performed.

Tier 2. Quality scoring (once Tier 1 passes).

- **Cryptographic-primitive sourcing (50%).** Are all four primitives sourced from the `cryptography` package's audited implementations rather than student-implemented? (No "rolling your own crypto" violations.)
- **Cite-discipline (30%).** Does each cryptographic-primitive call have an inline comment naming the 3GPP TS 33.501 section it implements? (Mirrors the SEC-101 CVE-walk source-citation expectation.)
- **Code-quality (20%).** Functions named clearly; no dead code; type hints if Python ≥ 3.11 features are used; no plaintext SUPIs in committed test files (use environment vars or test-vector inputs).

Artifact 2: Phase C analysis report (~3 pages)

Tier 1. Does it work. The report must:

- Correctly identify the IMSI-catcher attack class as the named threat 5G-AKA's SUCI design closes.
- Correctly explain *why* the SUCI design closes the IMSI-catcher class structurally (not just patches it implementation-wise).
- Identify at least three residual attack classes, each with at least one cited primary source.

Reports that fail Tier 1 do not pass the artifact.

Tier 2. Quality scoring.

- **Technical accuracy (40%).** Does the report's account of SUCI computation match 9e §8.8.2 + 3GPP TS 33.501? Are the residual attack classes accurately described?

- **CVE-walk catalog fluency (35%).** Does the report read like the SEC-101 Lab 8 CVE-record-walk register. Primary sources cited; technical detail at the right depth; no hand-waving where precision is achievable?
- **"What would we do differently" reflection (25%).** Does the reflection engage seriously with the post-5G-AKA design question? Does it propose specific architectural moves rather than generic "more crypto"?

Setup / lab substrate

Required. Python 3.11+ with `cryptography` package (`pip install cryptography` at current stable; the version used in SEC-101 Week 5). Wireshark with 4G/5G NAS dissectors enabled (default in modern Wireshark builds; `wireshark` package on Debian / `brew install wireshark` on macOS). Canned capture set from the academy lab repo (instructor team curates).

Optional advanced. OpenAirInterface 5G core testbed access (NET-301 lab substrate; SEC-101 students can skip). srsRAN test environment (RF-301 lab substrate; SEC-101 students can skip).

Reference reading order before lab.

1. KR 9e §8.8.2 (5G-AKA / SUCI / SUPI). Primary; **must be read before Phase A**.
2. KR 9e §7.4.3 (5G Core control plane). Primary; **must be read before Phase C**.
3. `vca-net-301.html` Ch 8 §8.8.2 sample weave. Canonical academy primary prose.
4. `handouts/cross-chapter-wireless-aka-progression.md` (CT-A), AKA-progression context.
5. 3GPP TS 33.501 §6.12 + Annex C. Normative reference for SUCI computation; **required reading for Phase B**.
6. Borgaonkar et al., "New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols" (PETS 2019), Phase C residual-attack-class survey starting point.

Cross-track register notes

Why this lab lives at SEC-101's optional advanced register. SEC-101 is the security-literacy gate; the student arrives with cryptographic-primitive vocabulary (Week 5) and CVE-walk fluency (Weeks 11-12 + Lab 8 + Lab 9 capstone). The 5G-AKA lab seed pulls forward FROM the canonical NET-301 Ch 8 §8.8.2 prose. It does not re-derive the protocol's design rationale; it puts SEC-101's CVE-walk catalog into operation

against a contemporary in-deployment protocol. Students who complete the lab graduate with an explicit *protocol-design literacy* skill that the standard SEC-101 syllabus introduces (CVE-walk + threat-modeling) but does not exercise at this depth.

Why optional, not required. The lab requires NET-301 prose access (the canonical primary 5G-AKA framing lives there) and ~6-8 hours of indep practice on top of SEC-101's existing 116 hr. SEC-101's required lab manifest is already calibrated to course-completion expectations; the 5G-AKA lab seed is forward-stretch material for students who want to walk a contemporary protocol at the implementation set before reaching NET-301. Students on the WIR-101 / RE-101 / ADV-101 path may find more direct value in the existing SEC-101 manifest; students on the NET-301 / advanced-cellular path benefit from the lab as a bridge.

Cross-track register transfer.

- **WIR-101.** The CT-A AKA-progression sidebar's wir-101 register reads the lab's Phase A as a *historical* progression (KRACK → Dragonblood → IMSI-catcher → SUCI). WIR-101 students who complete this lab arrive at WIR-101 Week 4 with the cellular-side endpoint already walked.
- **NET-301.** NET-301 Ch 8 §8.8.2 carries the canonical primary prose; this lab is one of NET-301's natural advanced-track lab options (the lab's substrate aligns with NET-301's OpenAirInterface 5G testbed). NET-301 students who completed the lab during SEC-101's optional-extension slot arrive at Ch 8 having already implemented the SUCI computation; the Ch 8 prose then operates at the *deeper* register of the 5G-Core control-plane decomposition.
- **RF-301.** The CT-A AKA-progression sidebar's rf-301 register reads each protocol as a signal-side capture-side-channel. RF-301 students who completed this lab during SEC-101 arrive at the RF-301 register with the protocol-level primitives already walked; RF-301 then layers in the signal-side reading.
- **ADV-101.** The lab's Phase C residual-attack-class survey is the SEC-101 register; ADV-101's CVE-reproduction register would be the next step (find a published 5G-AKA CVE; reproduce it against a testbed). The lab is a natural prerequisite for any future 5G-AKA-targeting ADV-101 capstone.

Anchor citations

- **Primary.** Kurose & Ross, *Computer Networking: A Top-Down Approach*, 9th ed., Pearson, 2021. §8.8.2 (5G-AKA / SUCI / SUPI / IMSI-catcher closure, the central 5G-era anchor); §7.4.3 (5G Core / AMF / AUSF / UDM control-plane decomposition, the architectural context for Phase C analysis).
- **Normative.** 3GPP TS 33.501 (5G security architecture; SUCI / SUPI / 5G-AKA normative). 3GPP TS 33.402 (non-3GPP access; 5G-AKA-prime variant; Phase C optional advanced extension only). 3GPP TS 33.501 Annex C (SUCI computation worked examples; Phase B known-answer test vectors).
- **Secondary academic.** Borgaonkar, Hirschi, Park, Shaik, "New Privacy Threat on 3G, 4G, and Upcoming 5G AKA Protocols" (PETS 2019). Basin, Dreier, Hirschi, Radomirović, Sasse, Stettler, "A Formal Analysis of 5G Authentication" (ACM CCS 2018). These are the academic-literature anchors for Phase C residual-attack-class analysis.
- **Cross-references.** `vca-net-301.html` Ch 8 §8.8.2 sample weave (canonical academy primary prose). `handouts/cross-chapter-wireless-aka-progression.md` (CT-A; AKA-progression sidebar). `vca-sec-101.html` WIR-101 skill-transfer row + Lab Manifest optional-advanced entry (the lab's two SEC-101 anchor points).

Cross-references

Picks up at	Section	Purpose
`vca-sec-101.html`	Lab Manifest (optional advanced extension)	Discoverable lab-seed pointer; trailing paragraph after the numbered Lab Manifest list referencing this handout
`vca-sec-101.html`	Per-course skill-transfer roster (WIR-101 row)	"▶ See also" cross-reference appended after the existing CT-A sidebar reference; ties the SEC-101 CVE-walk catalog to the cellular-AKA-implementation set the lab exercises
`vca-net-301.html`	Ch 8 §8.8.2 sample weave	Canonical primary 5G-AKA prose; this lab pulls forward from there. *Not modified by this dispatch. Already live.*
`handouts/cross-chapter-wireless-aka-progression.md`	Companion sidebar	CT-A reads 5G-AKA at the protocol-design-progression register; this lab implements 5G-AKA at the cryptographic-primitive set. The two registers are complementary halves of the same skill.

Pedagogical note, the lab as the SEC-101 "protocol-literacy" gate

The standard SEC-101 syllabus introduces *protocol* literacy as a Week-5 cryptographic-primitive vocabulary lesson and *CVE* literacy as a Week-11 + Lab 8 + Lab 9 capstone discipline. The 5G-AKA lab seed is what *connects* the two: it is the lab where the cryptographic-primitive vocabulary becomes *the means of reading a contemporary protocol's design rationale* and where the CVE-walk catalog matures into *protocol-design literacy.* SEC-101 students who complete the lab arrive at PEN-101 / RE-101 / ADV-101 / NET-301 with a worked example of "I can read a contemporary protocol's spec, reproduce its computation, and analyze its residual attack surface". A skill no single weekly lab in SEC-101's required manifest exercises end-to-end. The lab is forward-stretch by design, and the optional placement is what makes it sustainable for the SEC-101 cohort whose primary focus is the historical-CVE capstone.

Companion to `handouts/cross-chapter-wireless-aka-progression.md`. Optional advanced extension; not part of SEC-101's required lab manifest.

© Virtus Cyber Academy. Generated 2026-05-08.