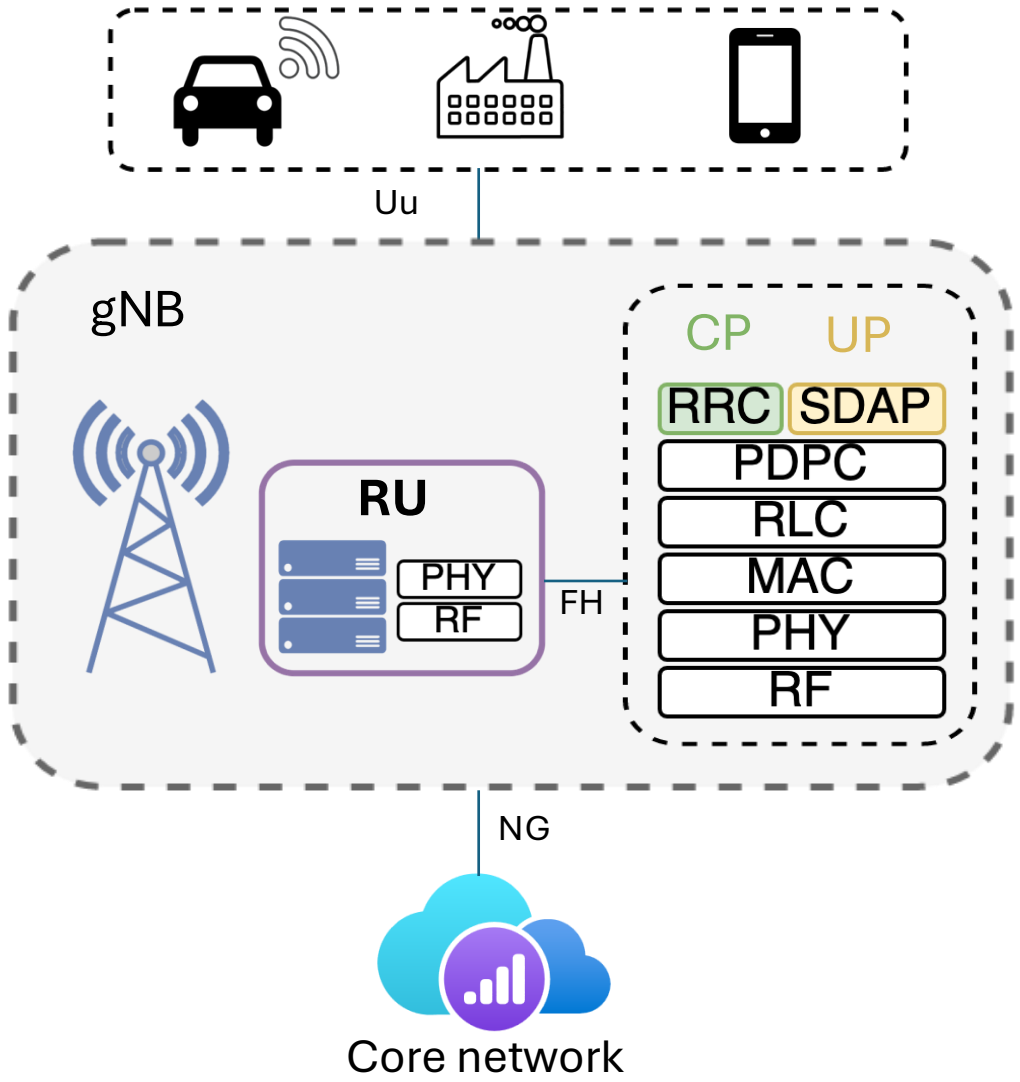
ORANClaw: Shredding E2 Nodes in O-RAN via Structure-aware MiTM Fuzzing

Geovani Benita Maldonado [1]
Matheus Eduardo Garbelini [2]
Sudipta Chattopadhyay [3]
Jianying Zhou [1]

[1] Singapore University of Technology and Design (SUTD)
[2] Nanyang Technological University (NTU)
[3] University of Missouri-Kansas City (UMKC)

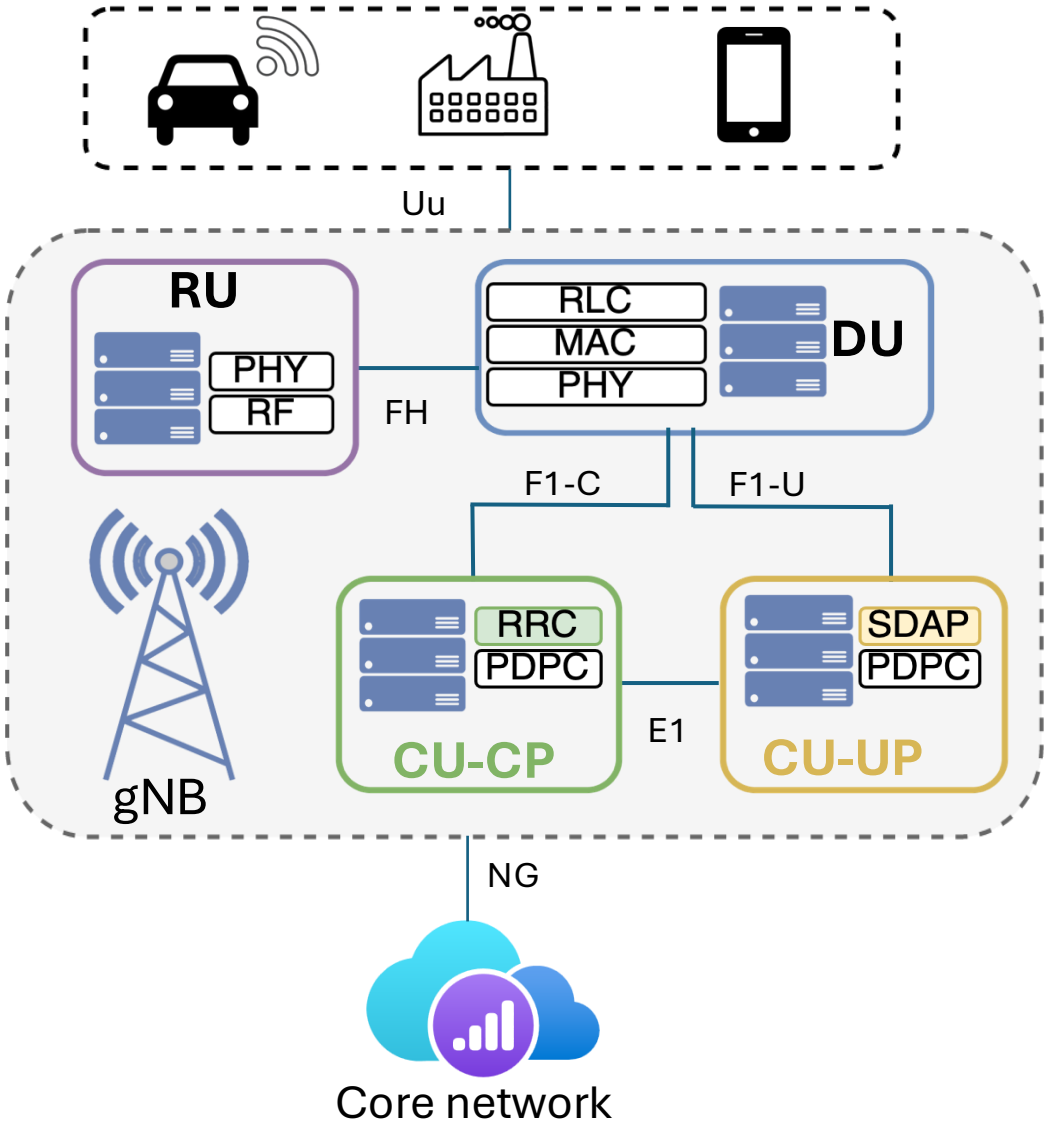
ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec)
Saarbrücken, Germany June 30 - July 3, 2026

Traditional 5G RAN architecture



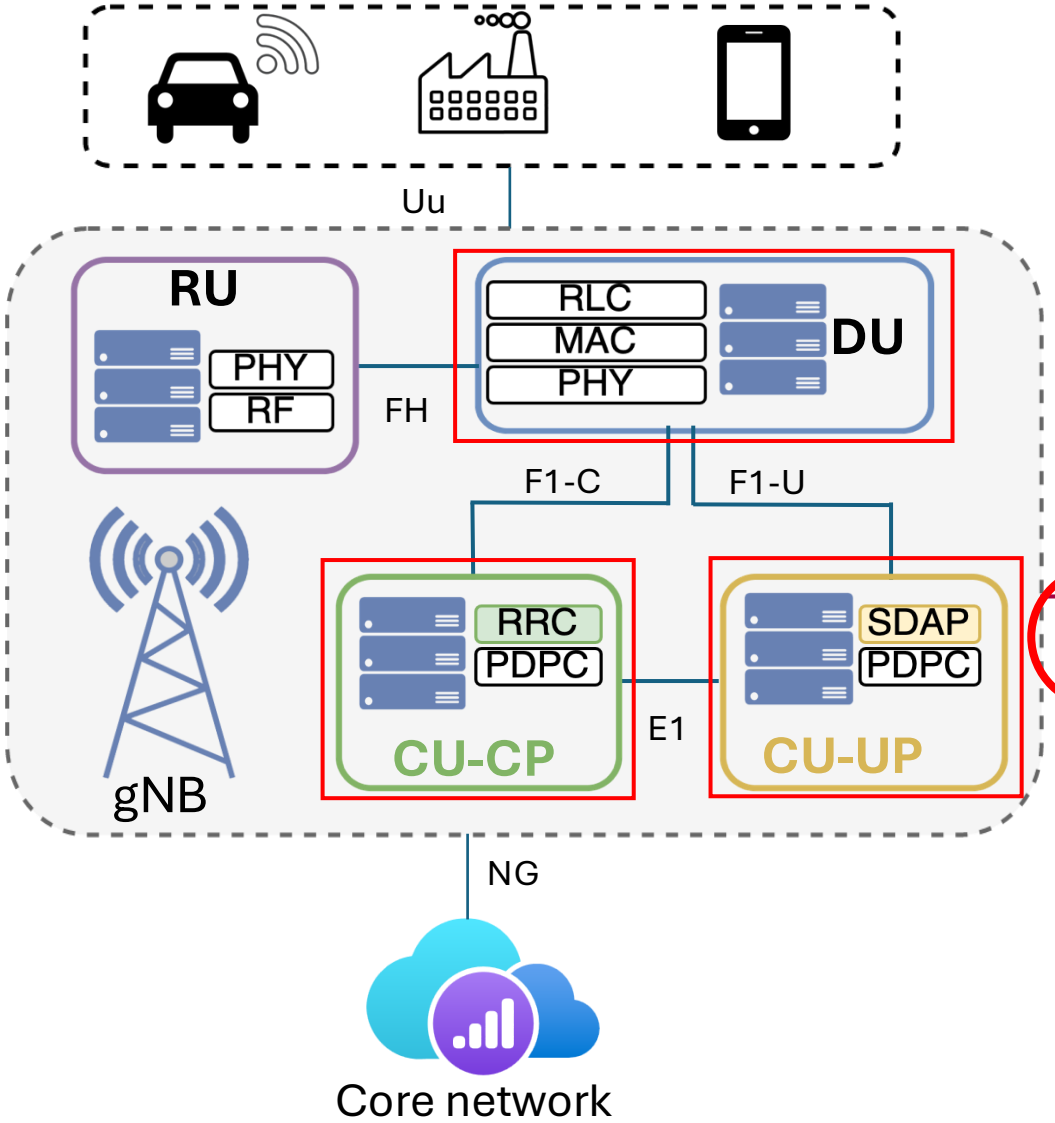
- UEs connect via **Uu** (NR air interface) to the gNB
- Proprietary vendor stack
- Single gNB hosts full stack: PHY → RLC → PDCP → RRC/SDAP
- CP and UP tightly co-located within one node

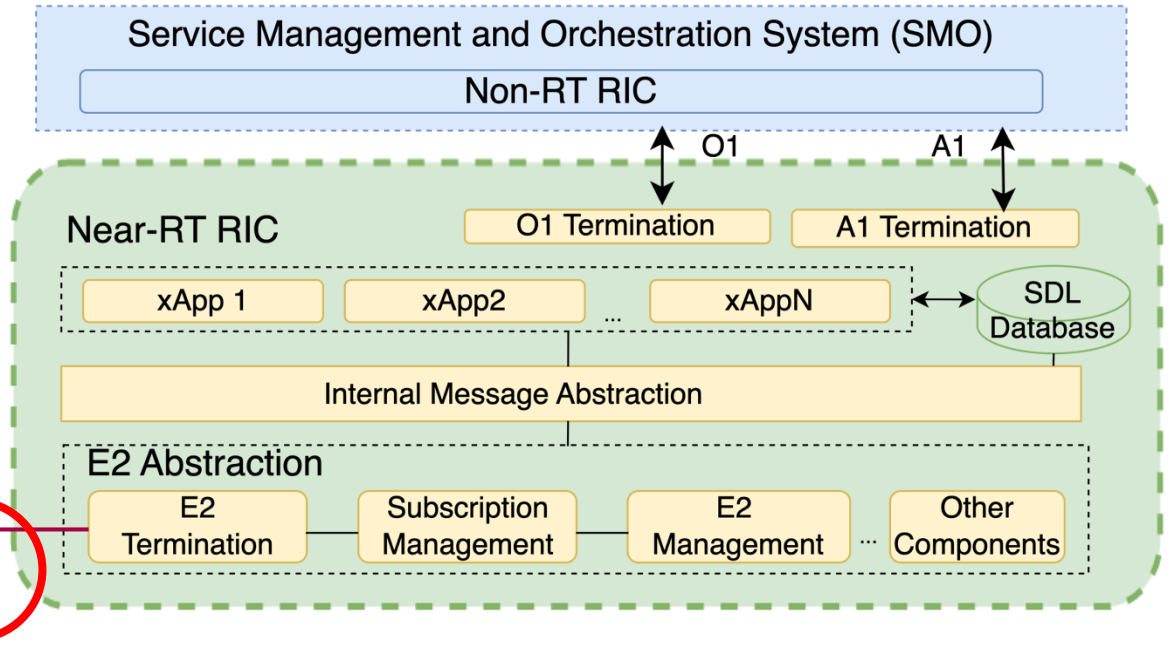
O-RAN Recap



- Monolithic gNB disaggregated into **RU, DU, CU** functional units
- Each unit runs on commodity hardware from **multiple vendors**
- Improved interoperability
- All inter-unit interfaces are open and standardized

O-RAN Recap

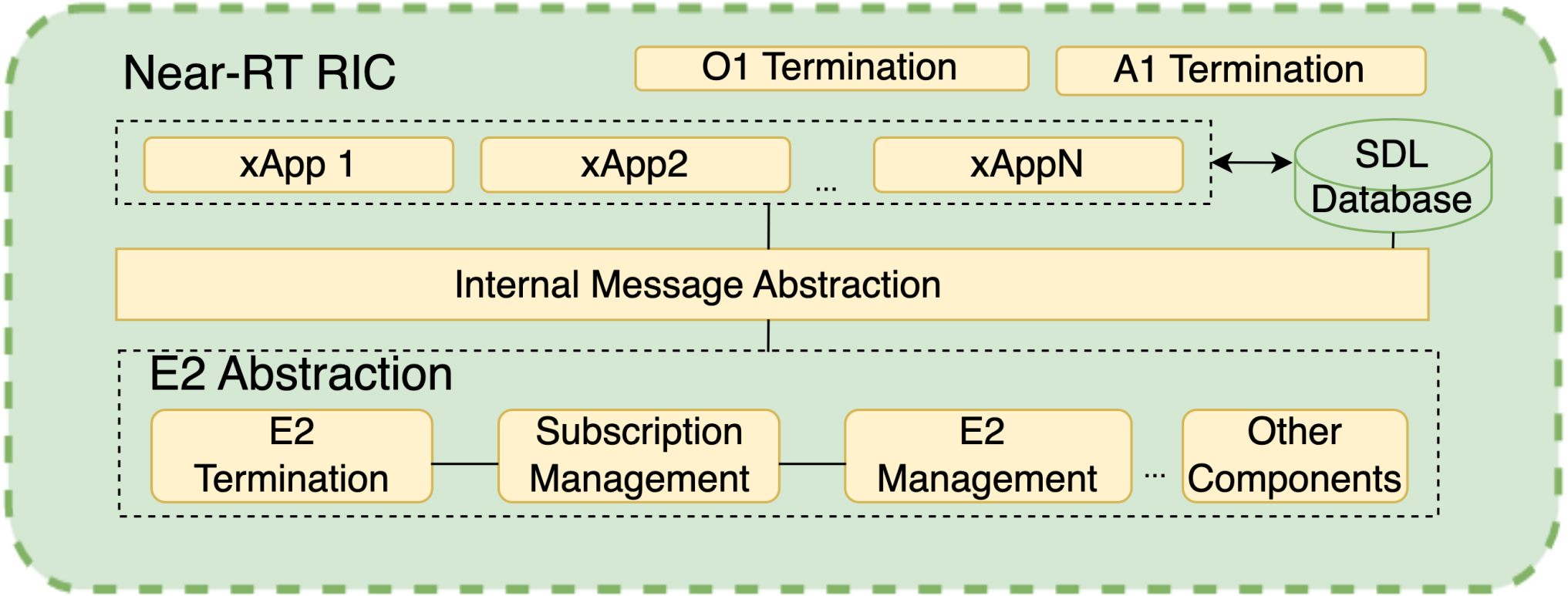




- RIC decouples RAN optimization
- Promotes using hardware from different vendors and software from third parties
- Near-RT RIC optimizes RAN operations
- E2 Nodes, E2 Interface

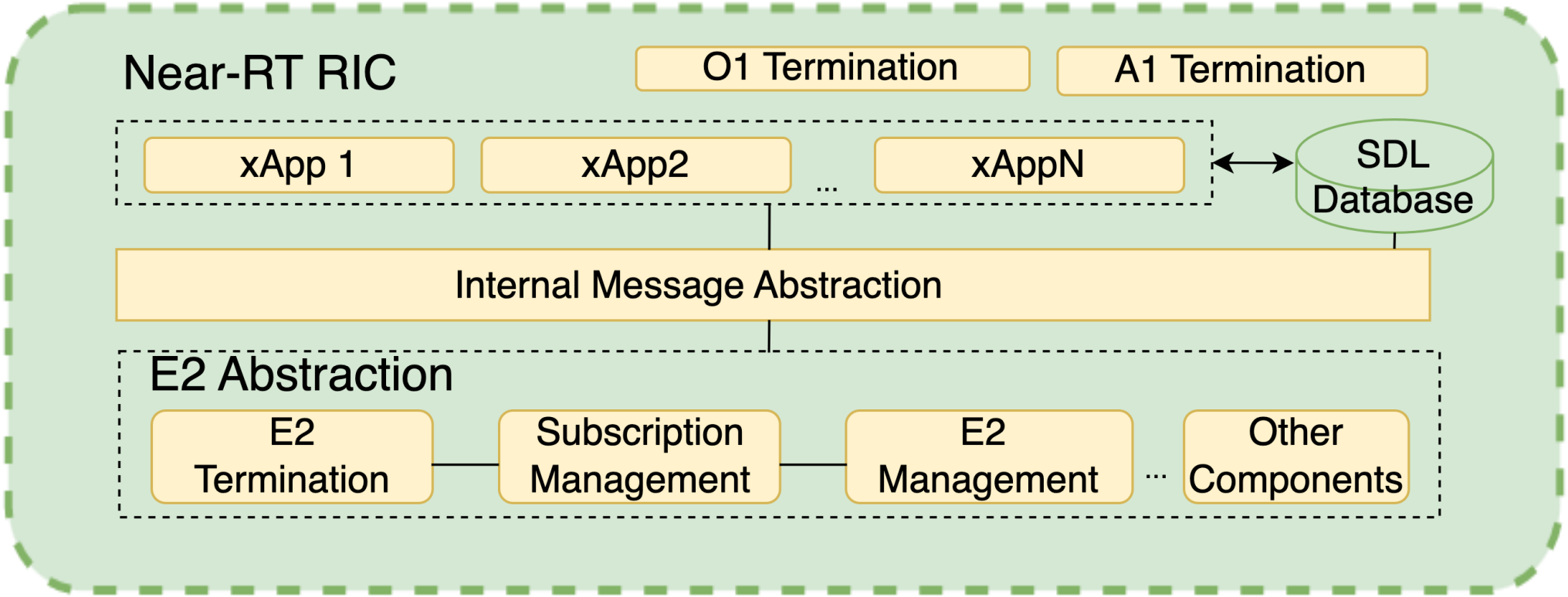
RAN Intelligent Controller (RIC) Architecture

- Service based architecture.
- Software centric RIC with third party providers

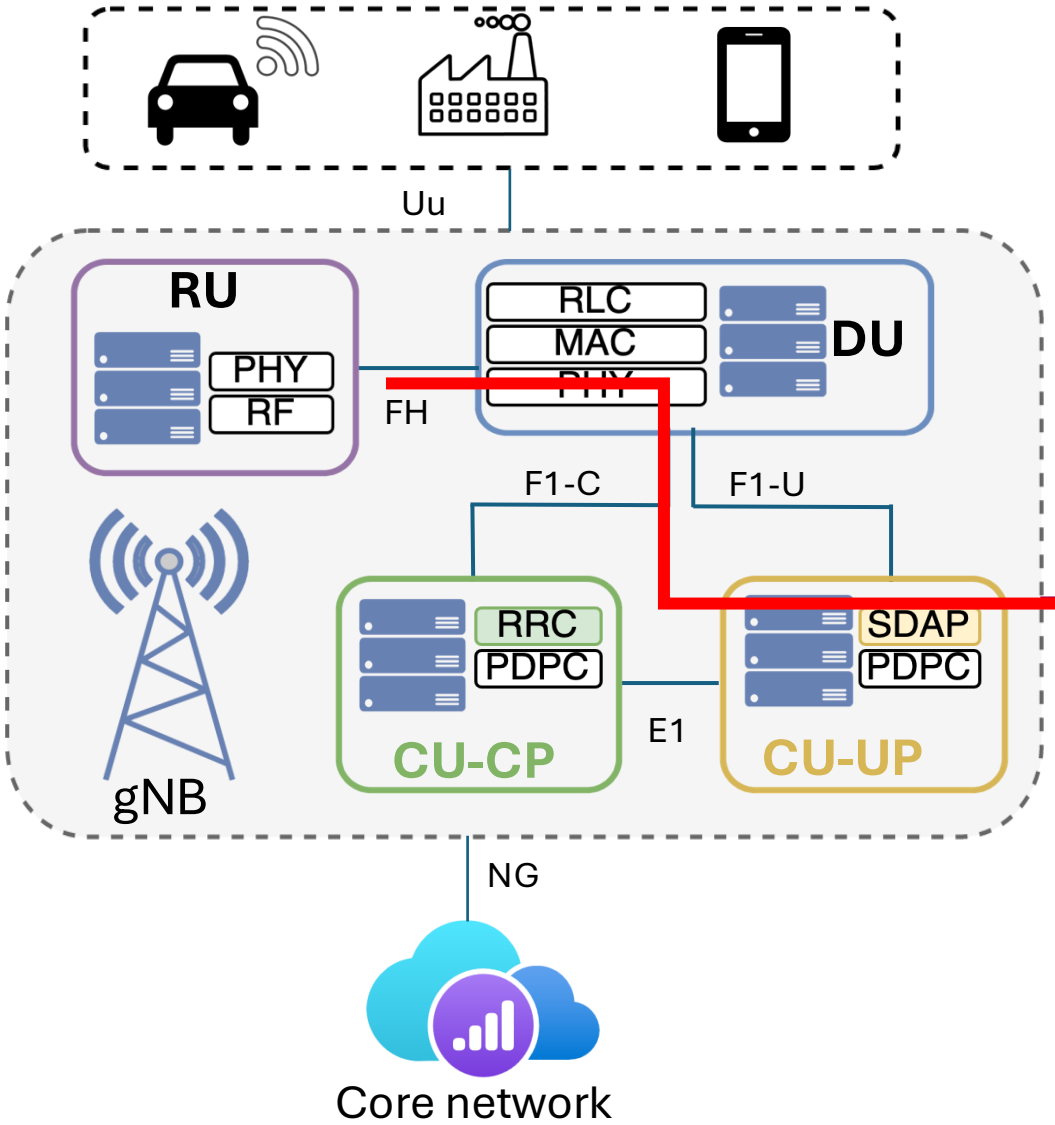


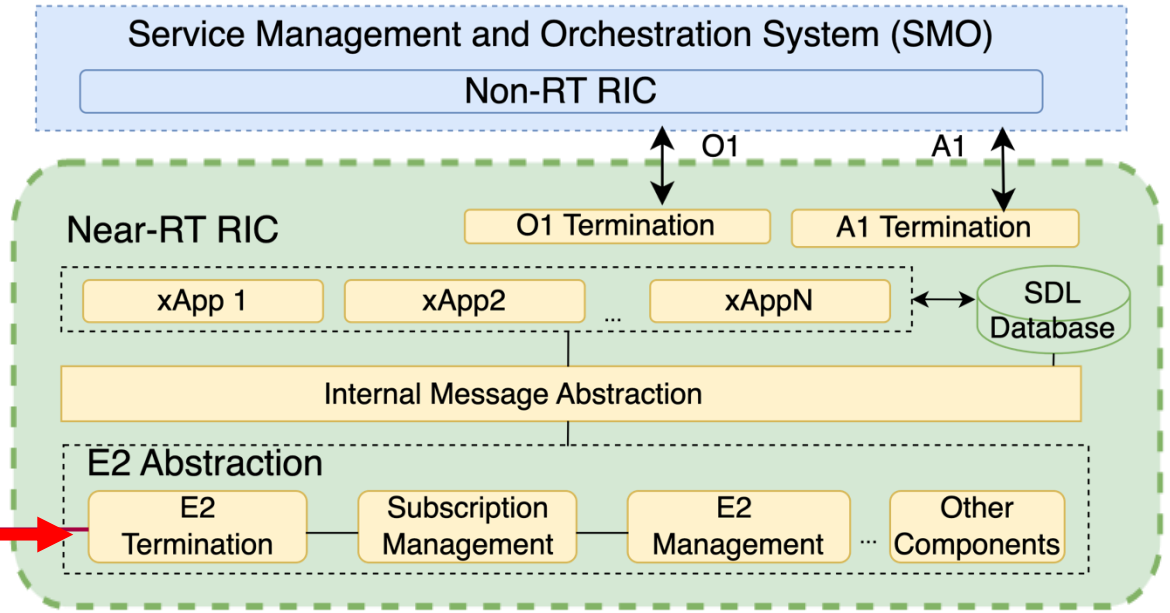
RAN Intelligent Controller (RIC) Architecture

- Service Models, define how an xApp interacts with the E2 nodes.
  - E2SM-KPM, E2SM-RC, E2SM-CCC



RAN Intelligent Controller (RIC) Architecture

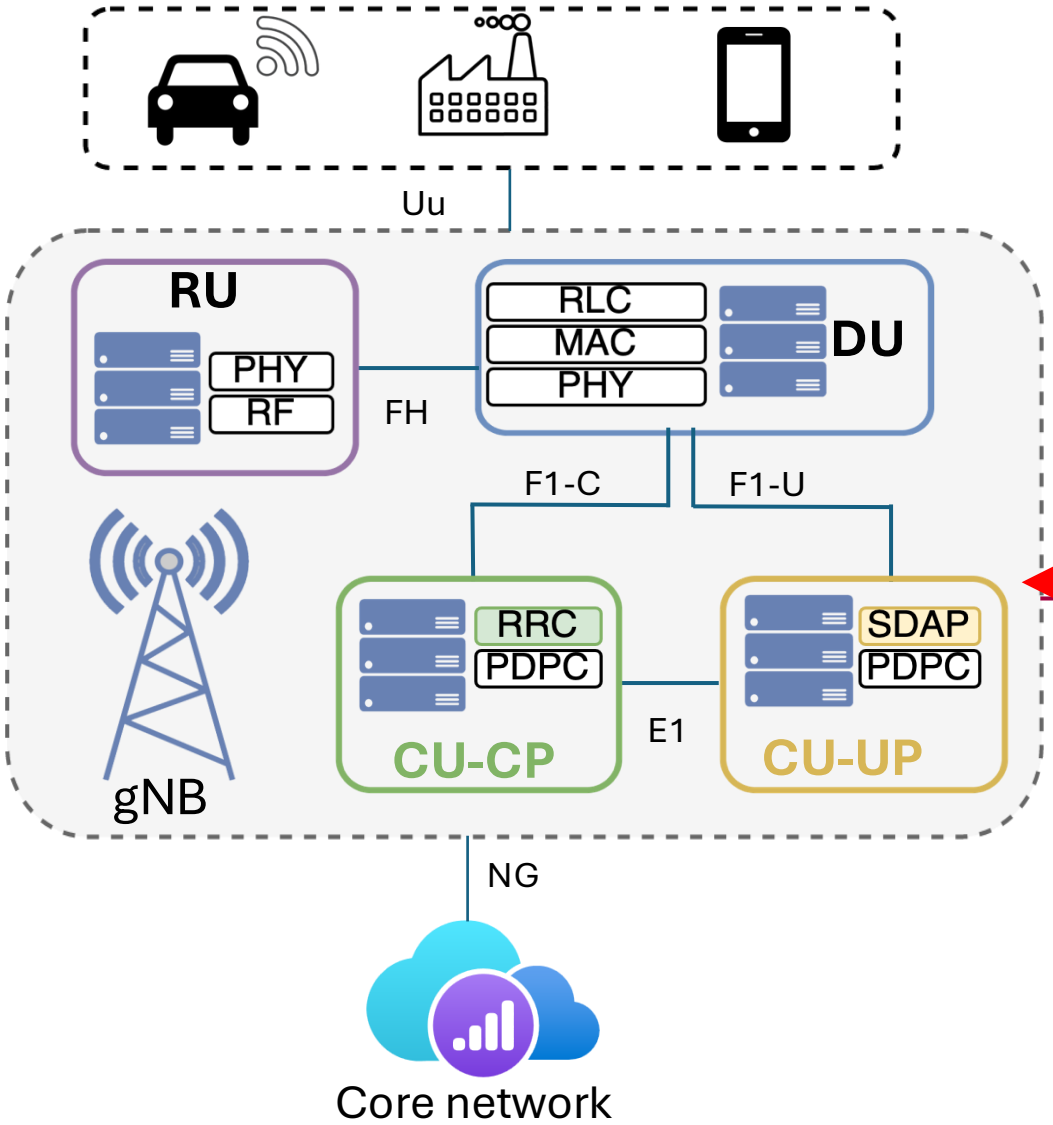


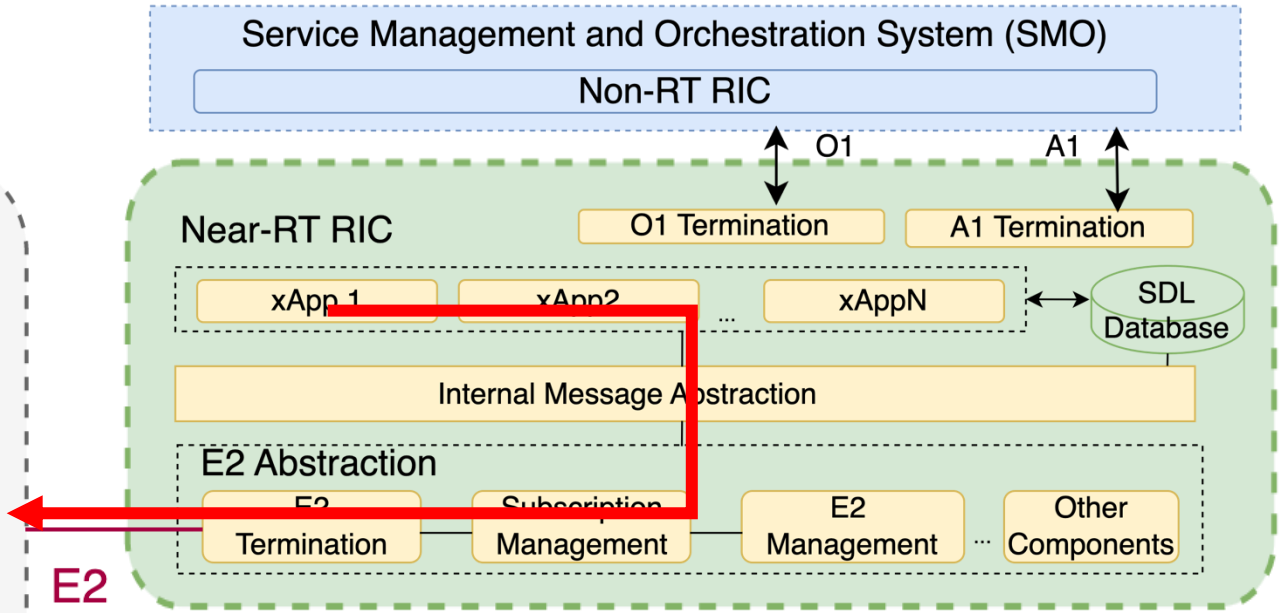


E2

- Real-time network states
- Traffic statistics
- Throughput
- Radio Resource Data

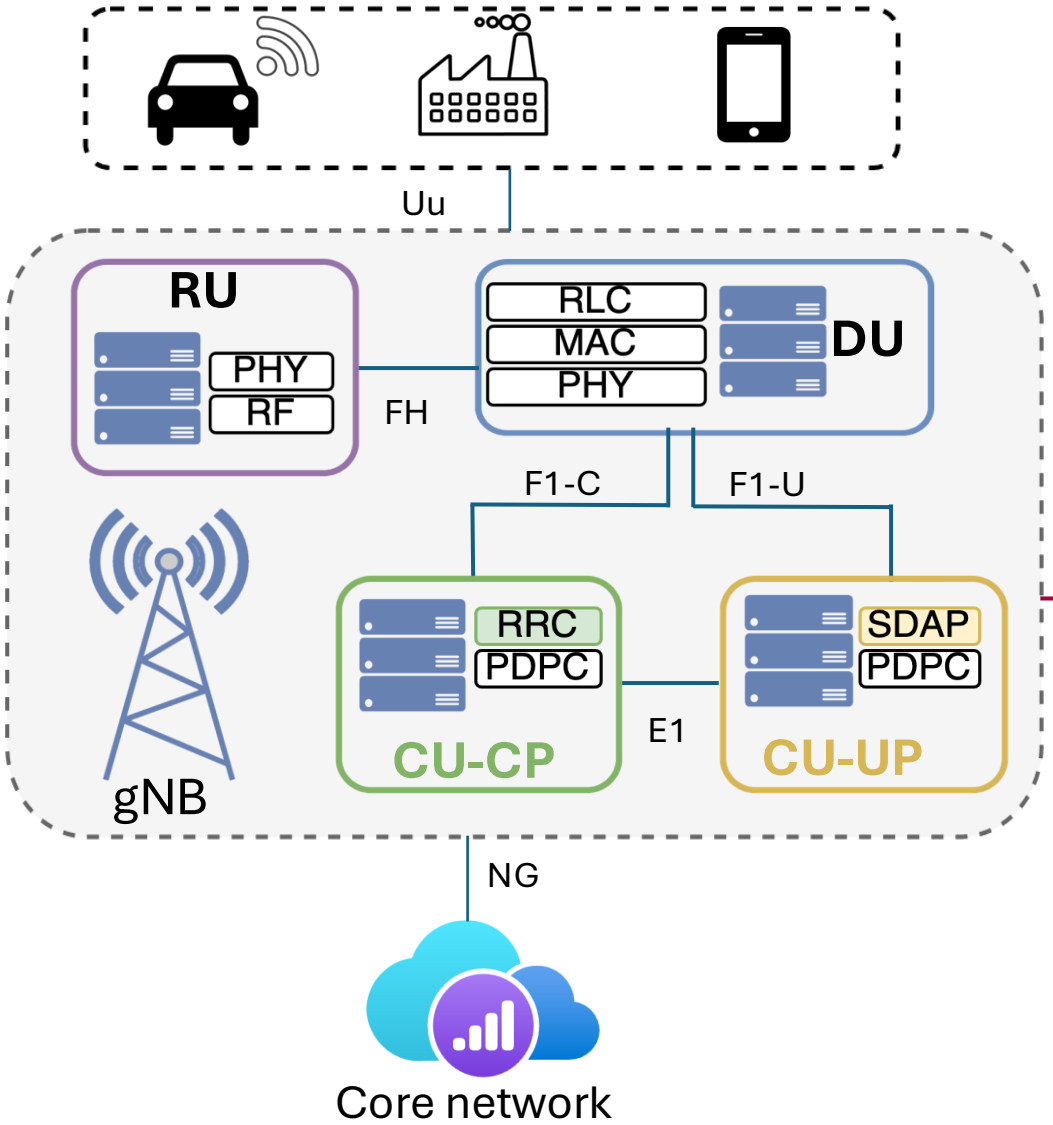
RAN Intelligent Controller (RIC) Architecture

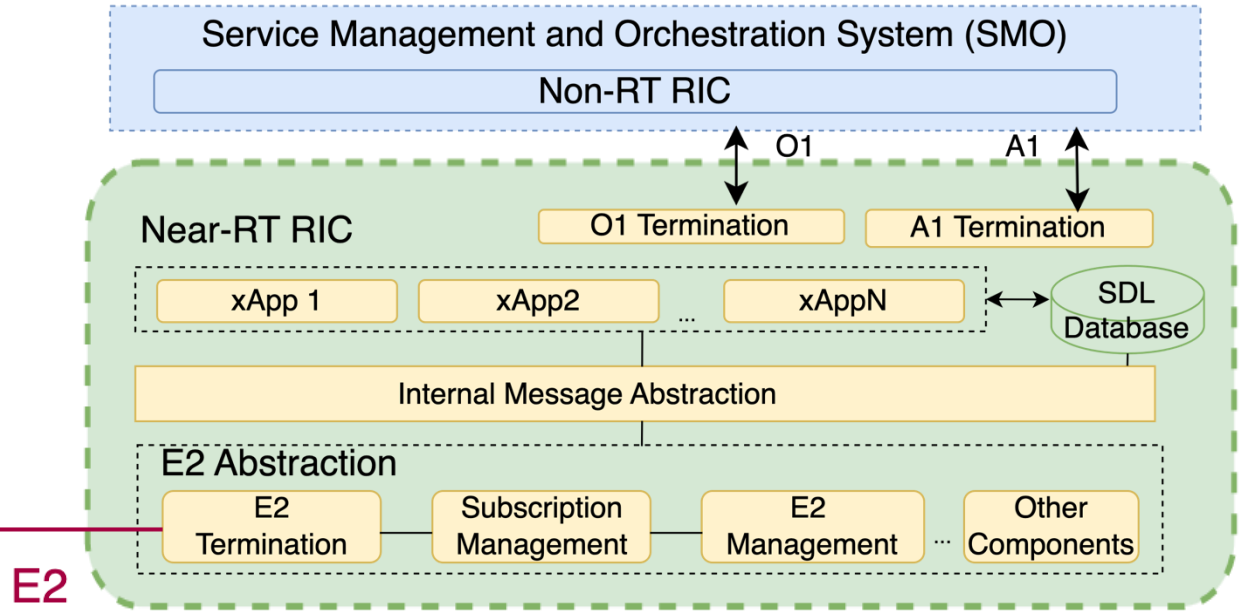




- RAN Control messages
- Policies and configuration commands

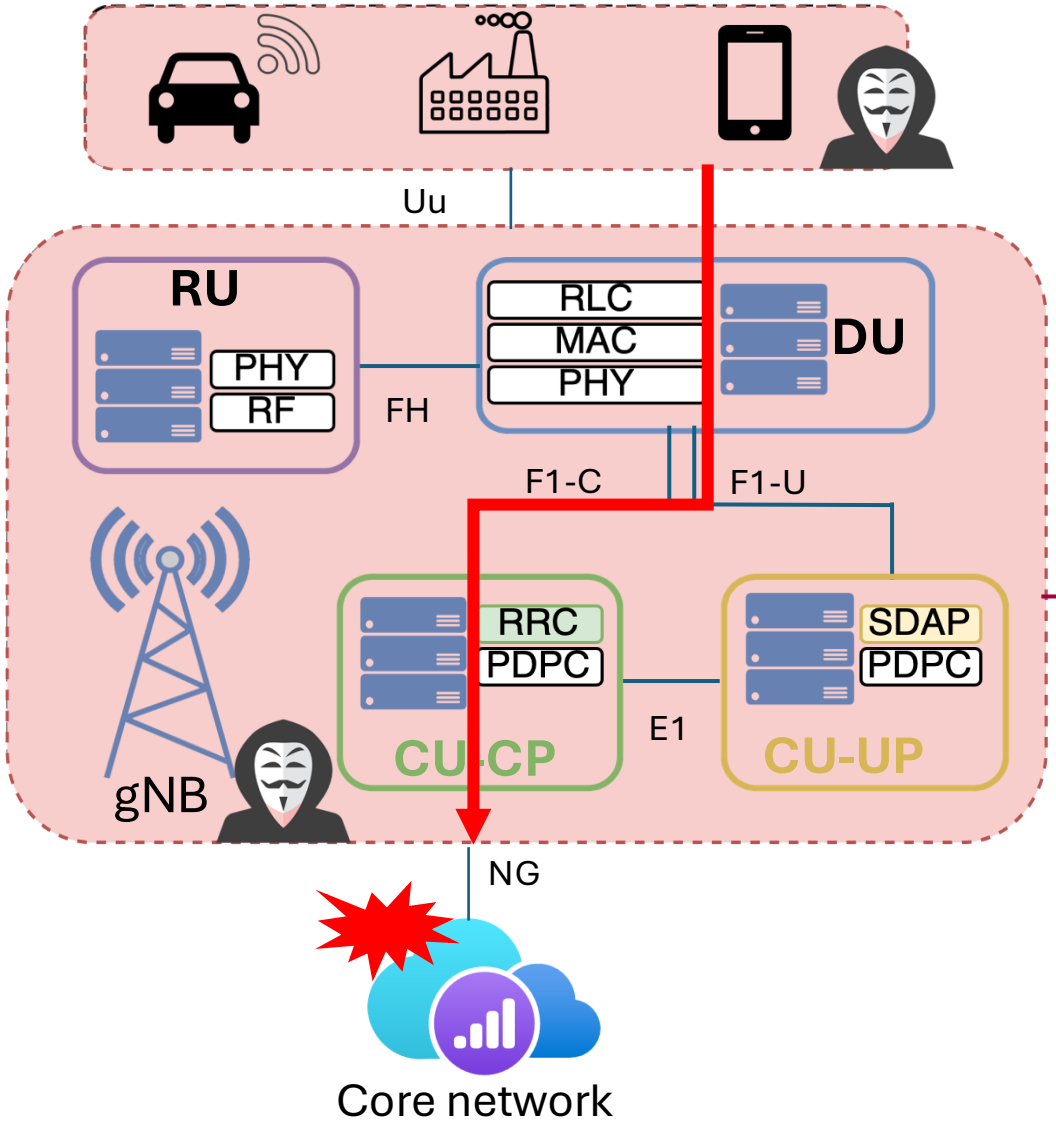
Security concerns within O-RAN architecture

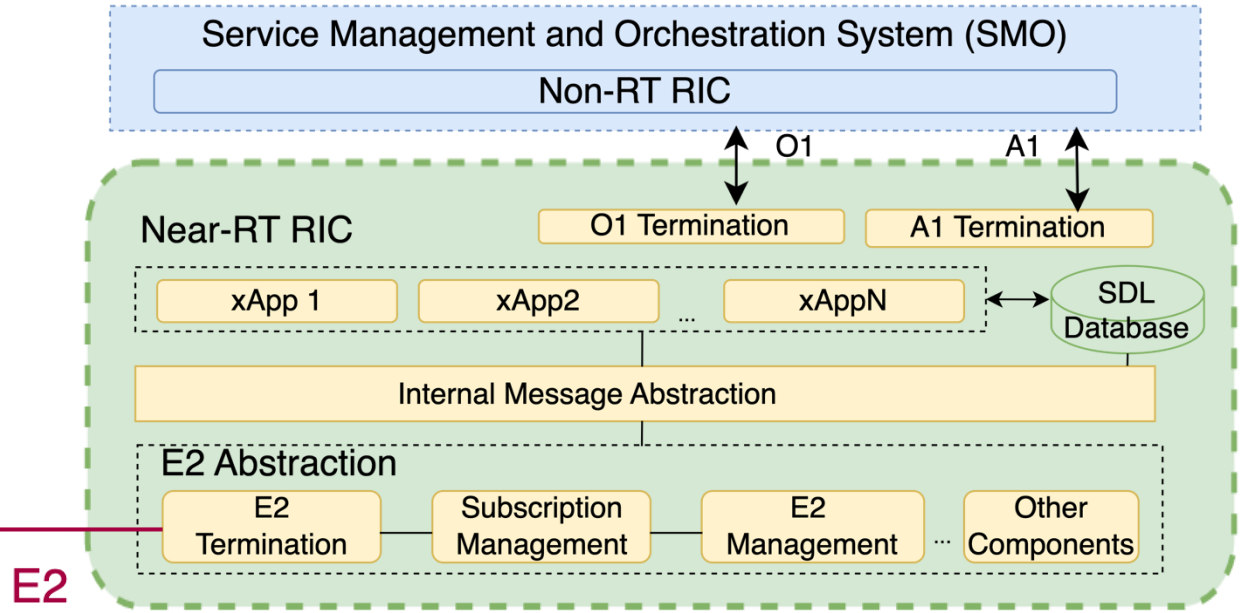




- New components, more complex architecture
- New attack surfaces

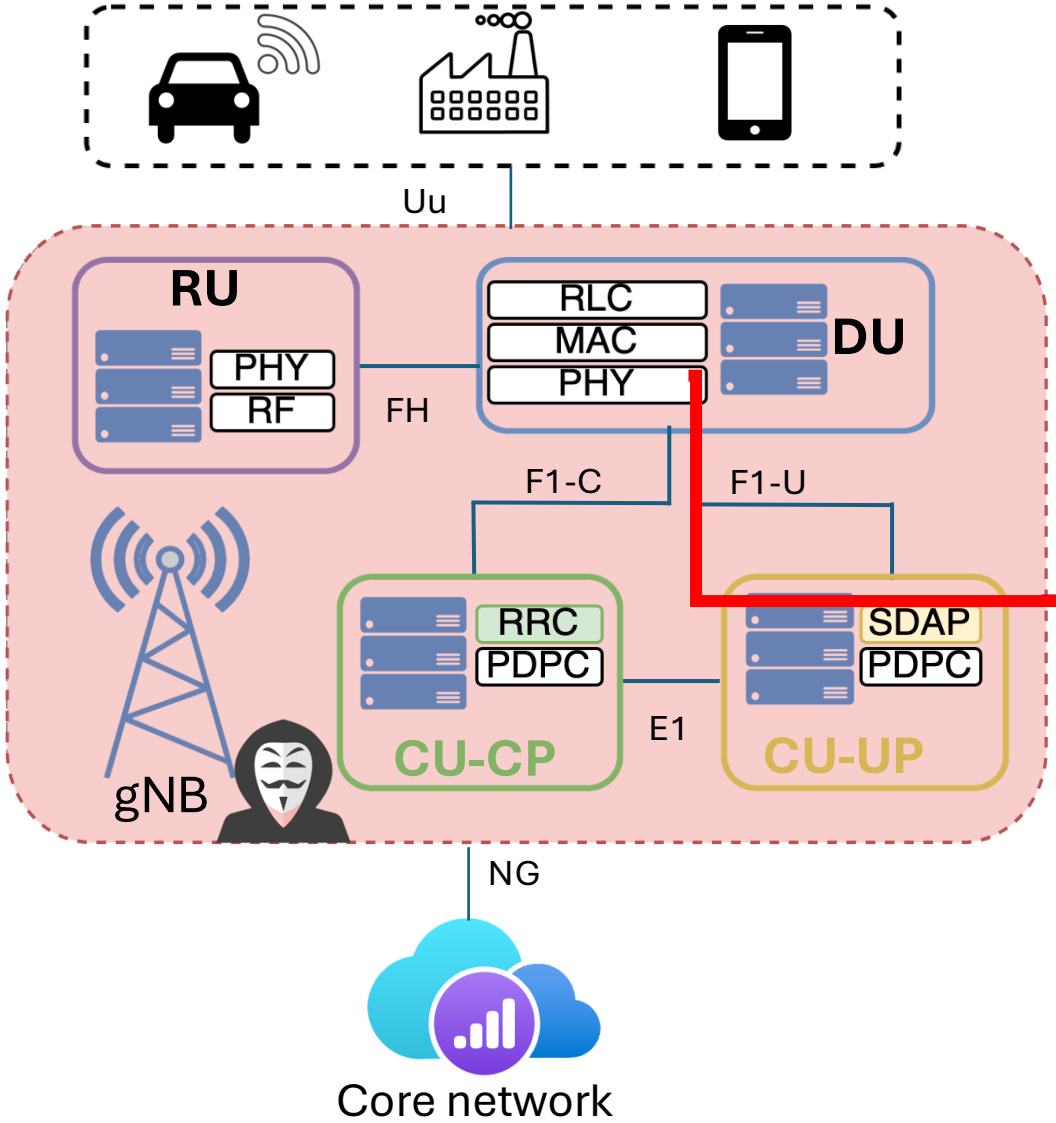
State of the art in O-RAN security

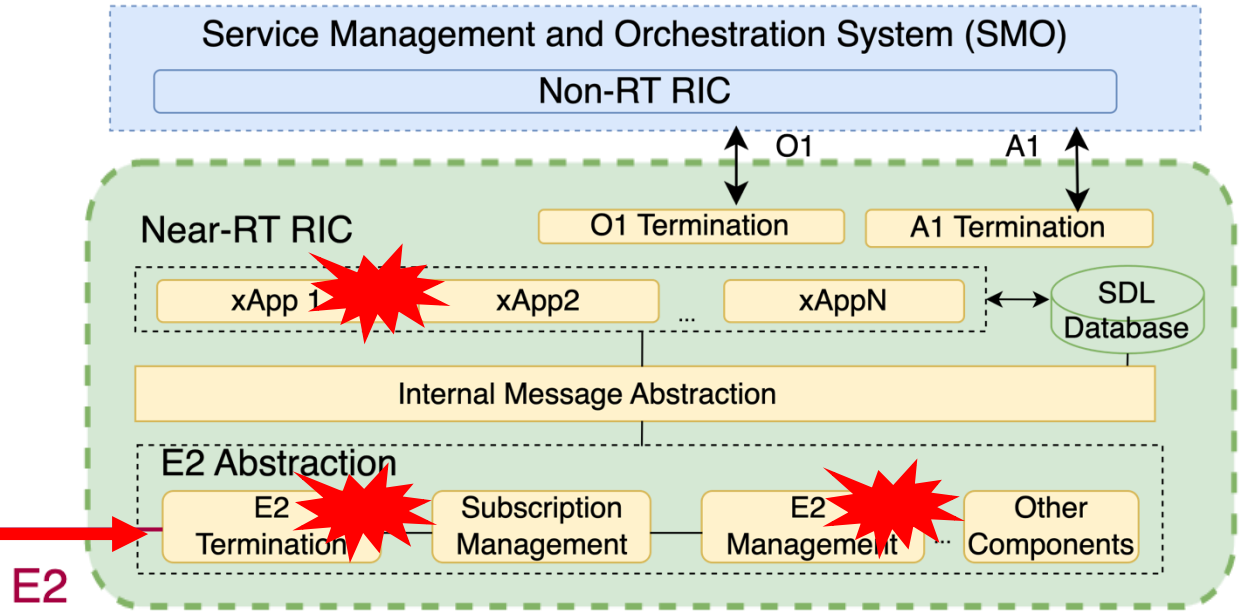




- Malicious gNBs/UEs targeting the core network *(Ransacked, CCS'24)*

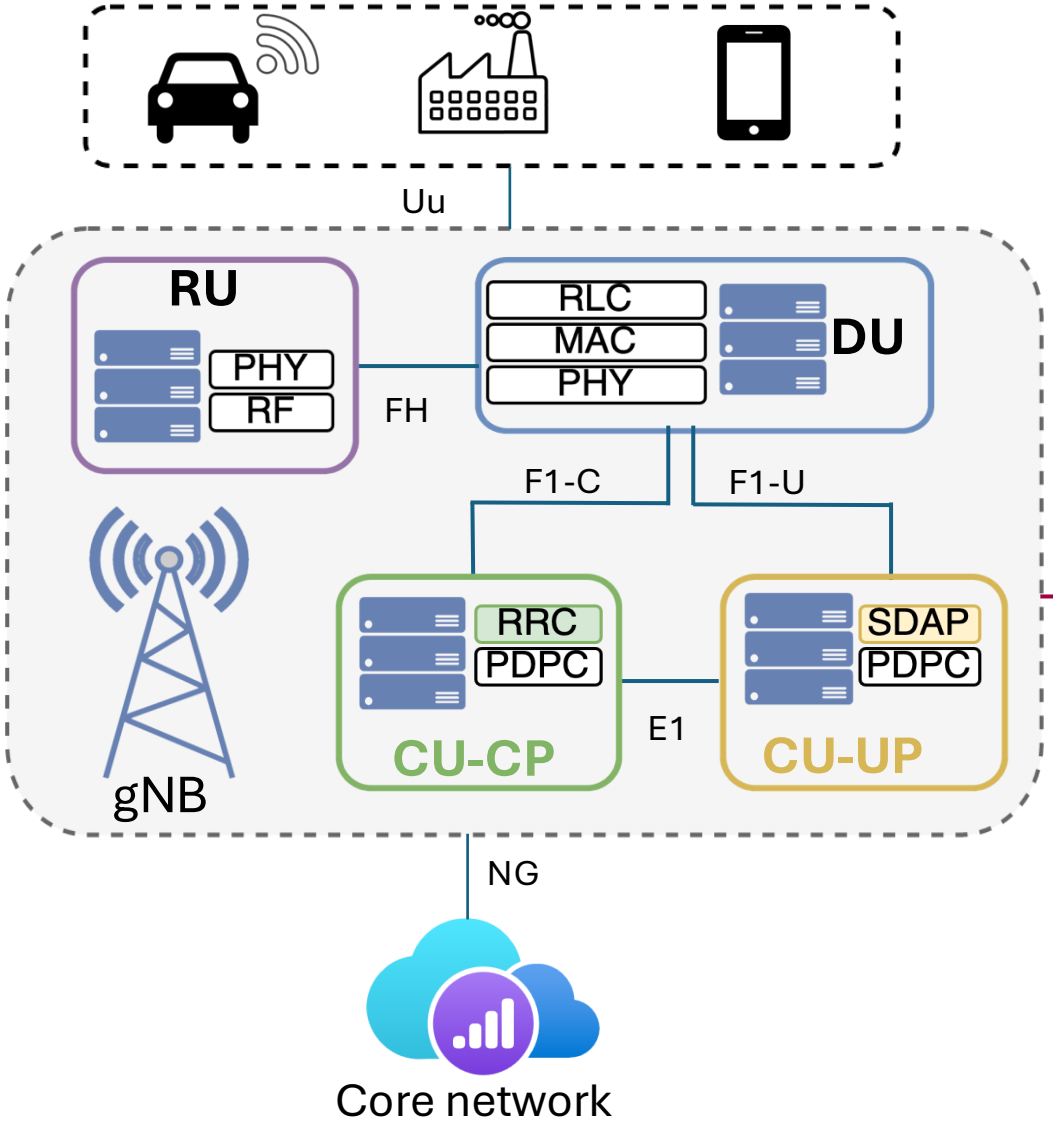
State of the art in O-RAN security

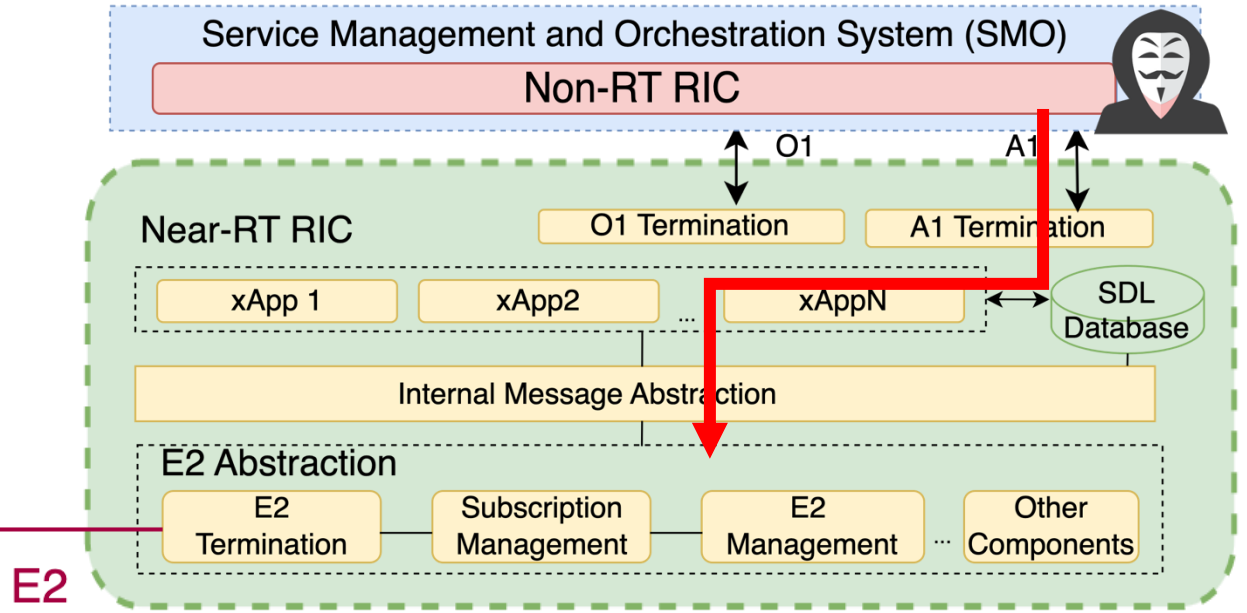




- Malicious gNB targeting the Near-RT RIC *(ORANalyst,* USENIX Sec'24*)*

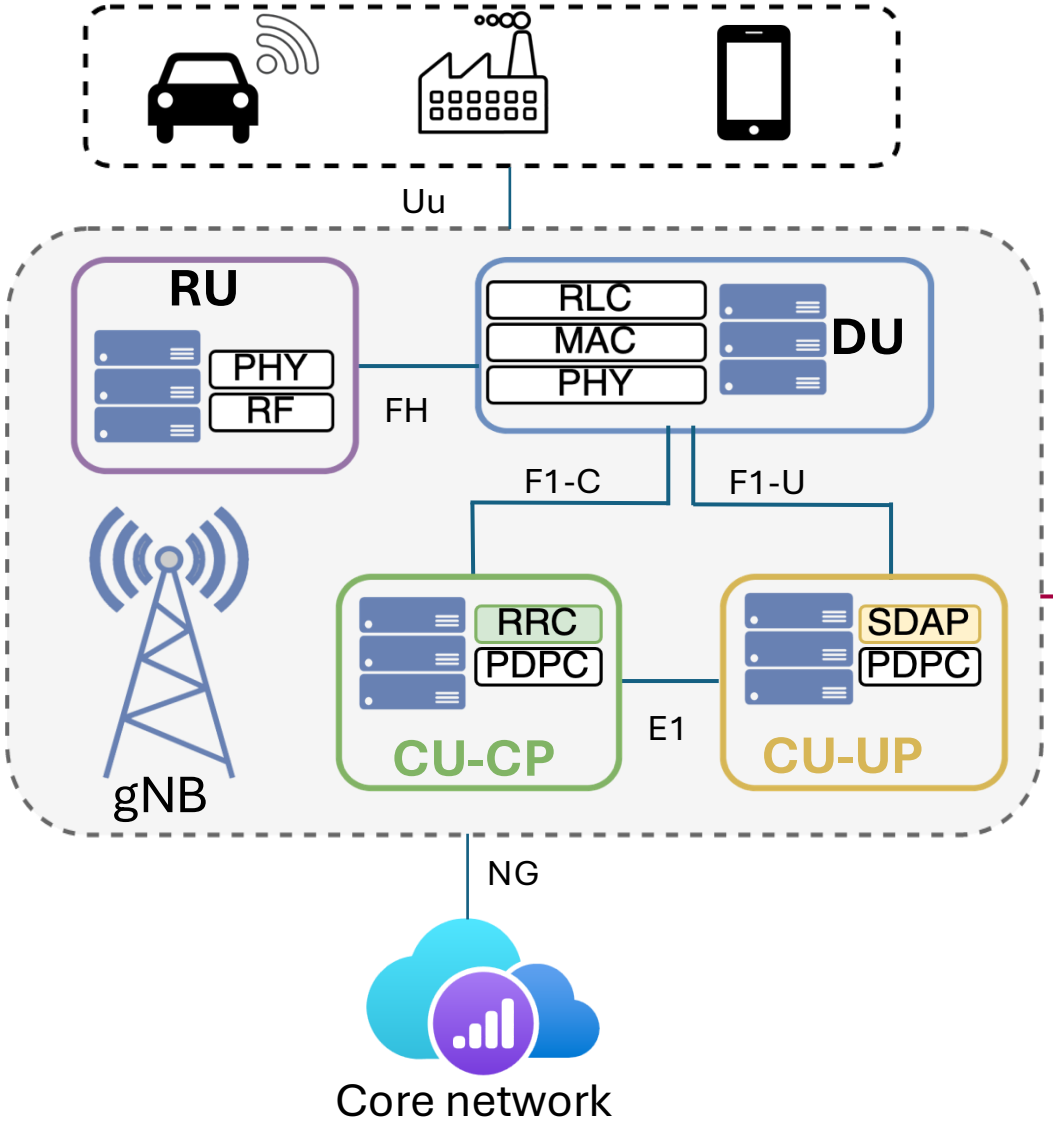
State of the art in O-RAN security

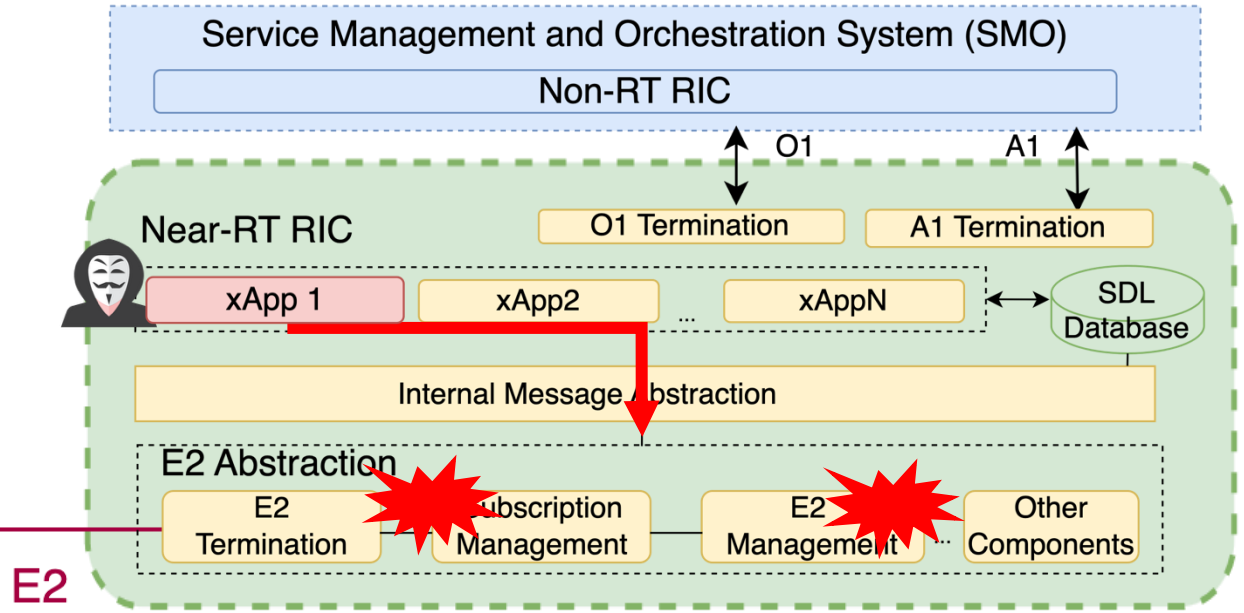




- Malicious Non-RT RIC targeting the Near RT RIC (Sec testing, WiSec'24)
- Policy tampering

State of the art in O-RAN security





- Malicious xApps targeting the Near RT RIC components (O-RAN WG11 T-NEAR-RT-01/02 specs)

Overlook in O-RAN E2 interface specification

7.3 Security mechanism for the E2 interface

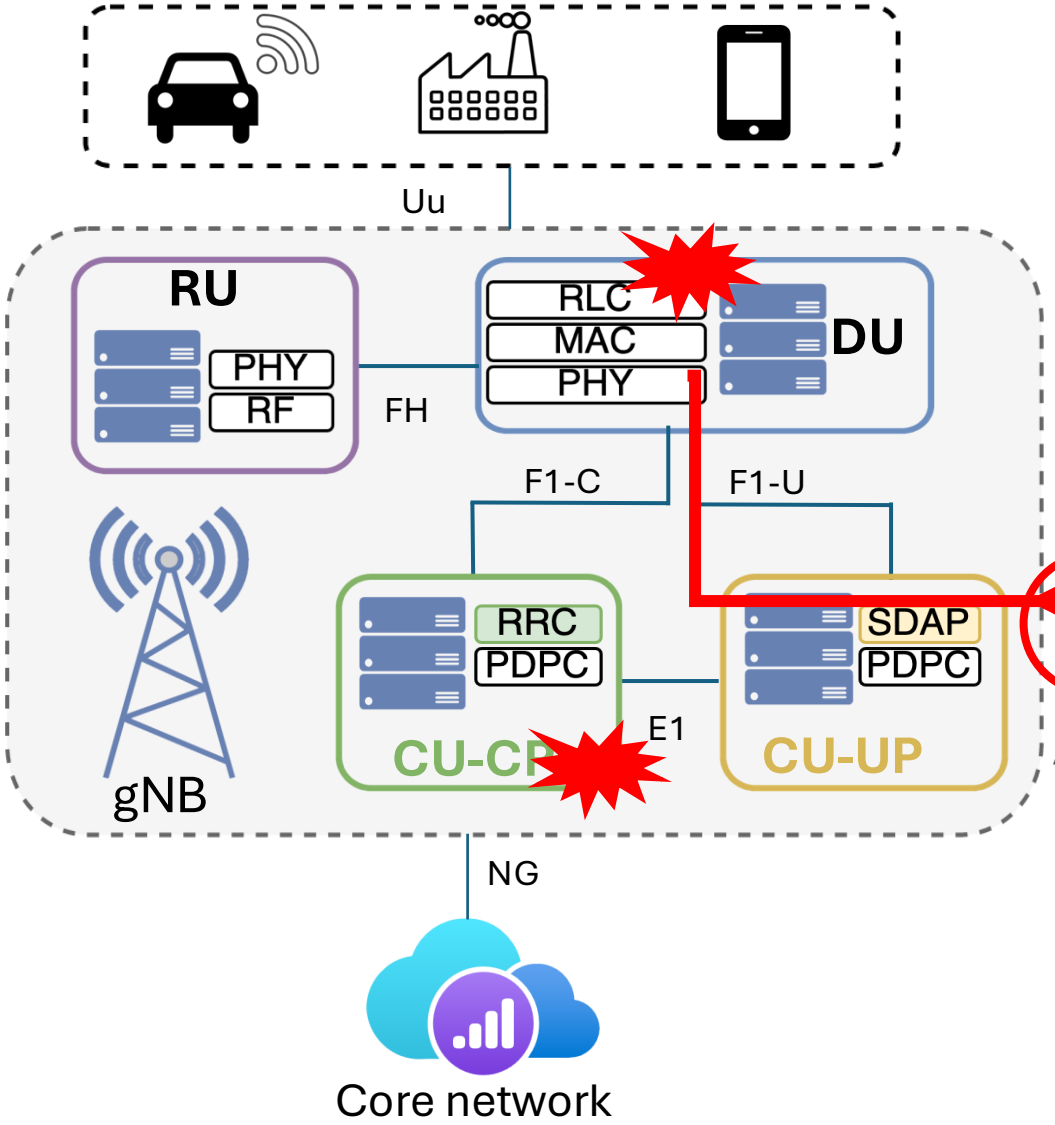
In order to protect the traffic on the E2 interface, IPsec ESP implementation shall be supported according to IETF RFC 4303 [20] as profiled by 3GPP TS 33.210 [21]. For IPsec implementation, tunnel mode is mandatory to support while transport mode is optional. The multiple IKE Security Associations (SAs), multiple IPsec SAs and multiple IPsec SAs per IPsec tunnel (e.g. for rekeying) shall be supported.

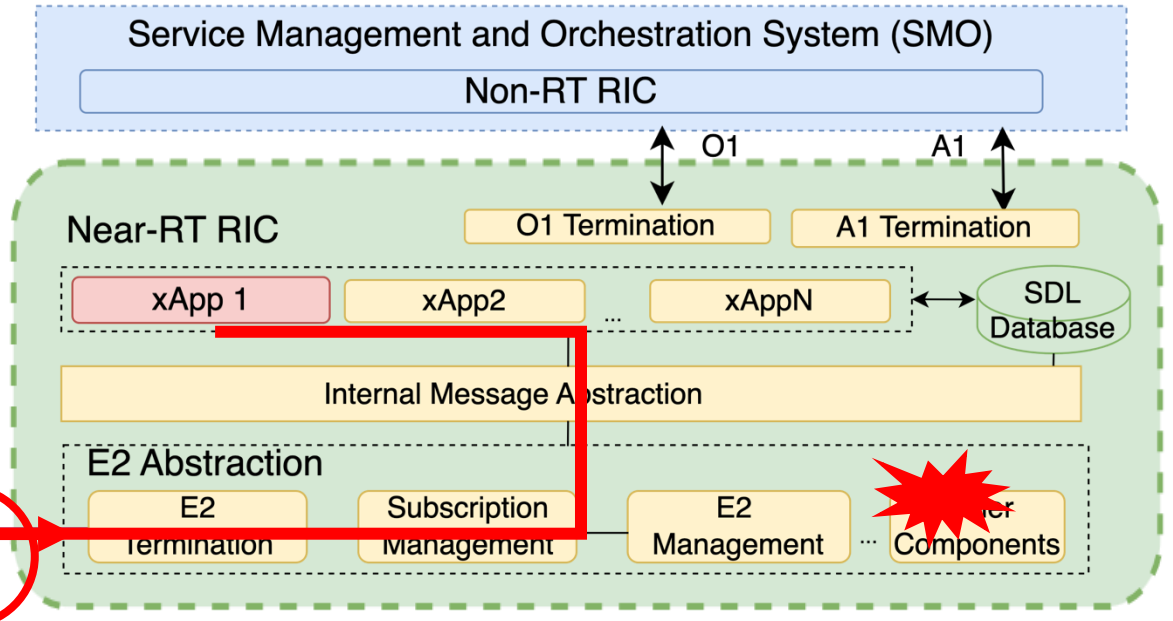
IKEv2 certificate-based authentication implementation shall be supported according to 3GPP TS 33.310 [22]. The certificates shall be supported according to the profile described by 3GPP TS 33.310 [22]. IKEv2 shall be supported conforming to the IKEv2 profile described in 3GPP TS 33.310 [22].

- IPsec is not MANDATORY!

Is it possible to systematically discover vulnerabilities **in gNB components** through malicious or buggy xApps, despite their **indirect communication paths**, by leveraging a man-in-the-middle (MiTM) on the E2 interface?

ORANClaw: Motivation and Threat model

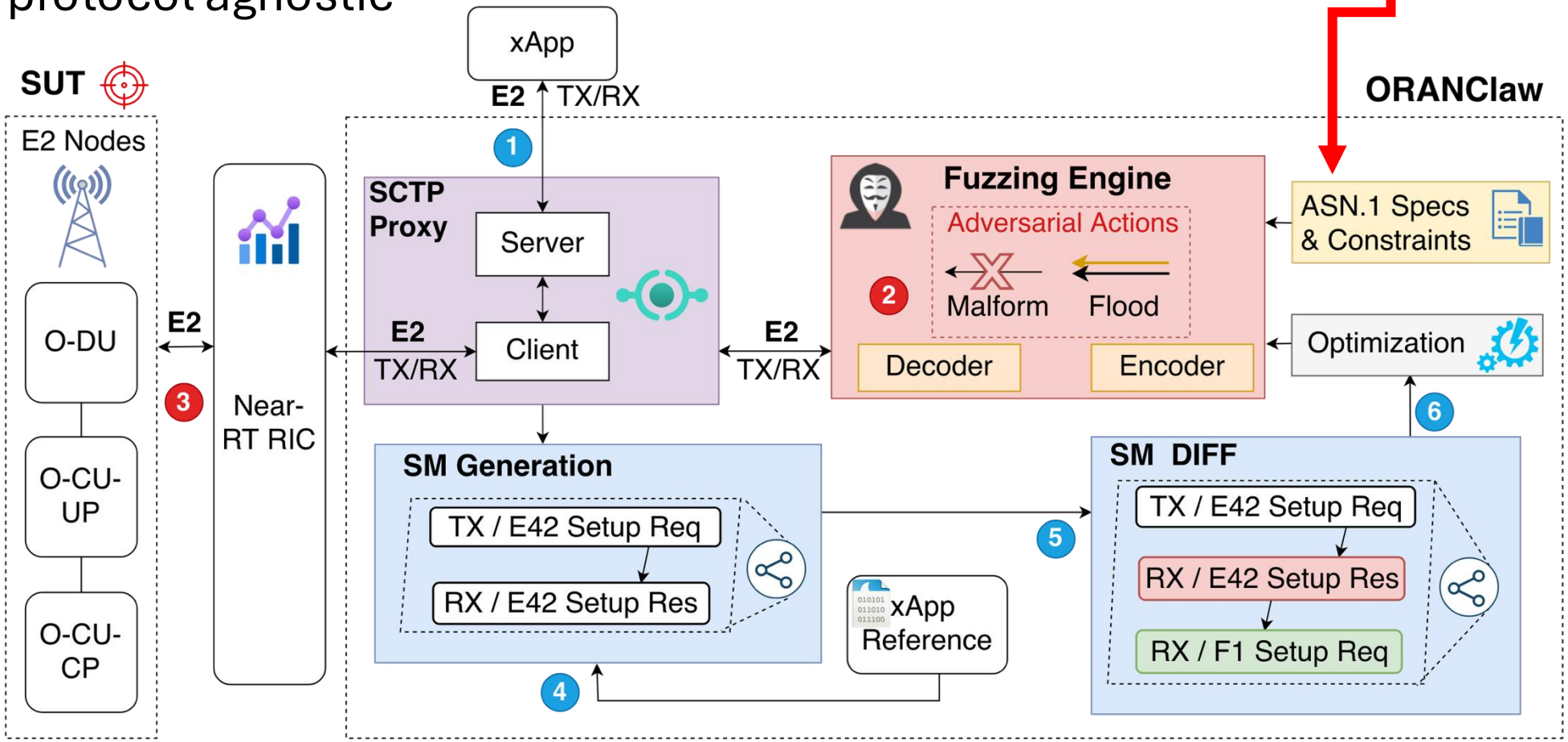




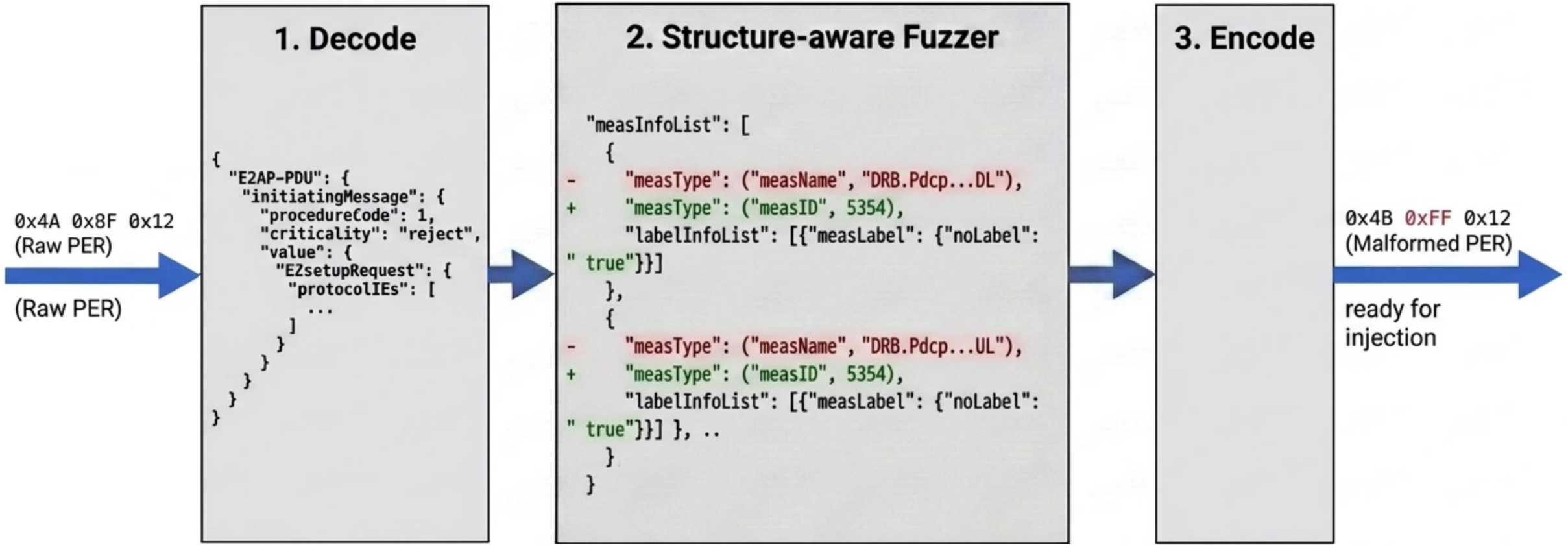
- Malicious/buggy xApps targeting the gNB components
- Systematically manipulate the E2 interface packets
- Bidirectional attacks gNB ⇄ Near RT RIC

ORANClaw: Design

- E2 Application Protocol (E2AP) and E2 Service Model (E2SM) protocol agnostic

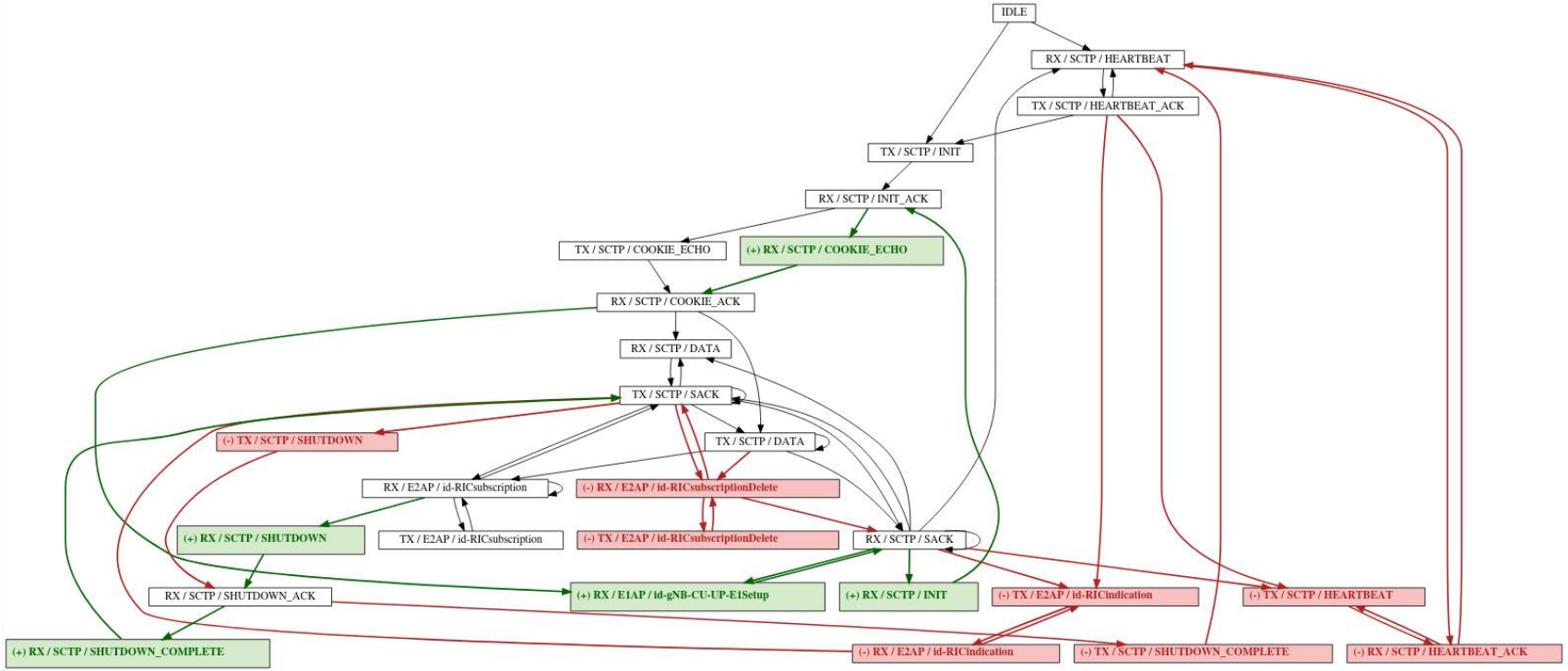


ORANClaw: Fuzzing engine



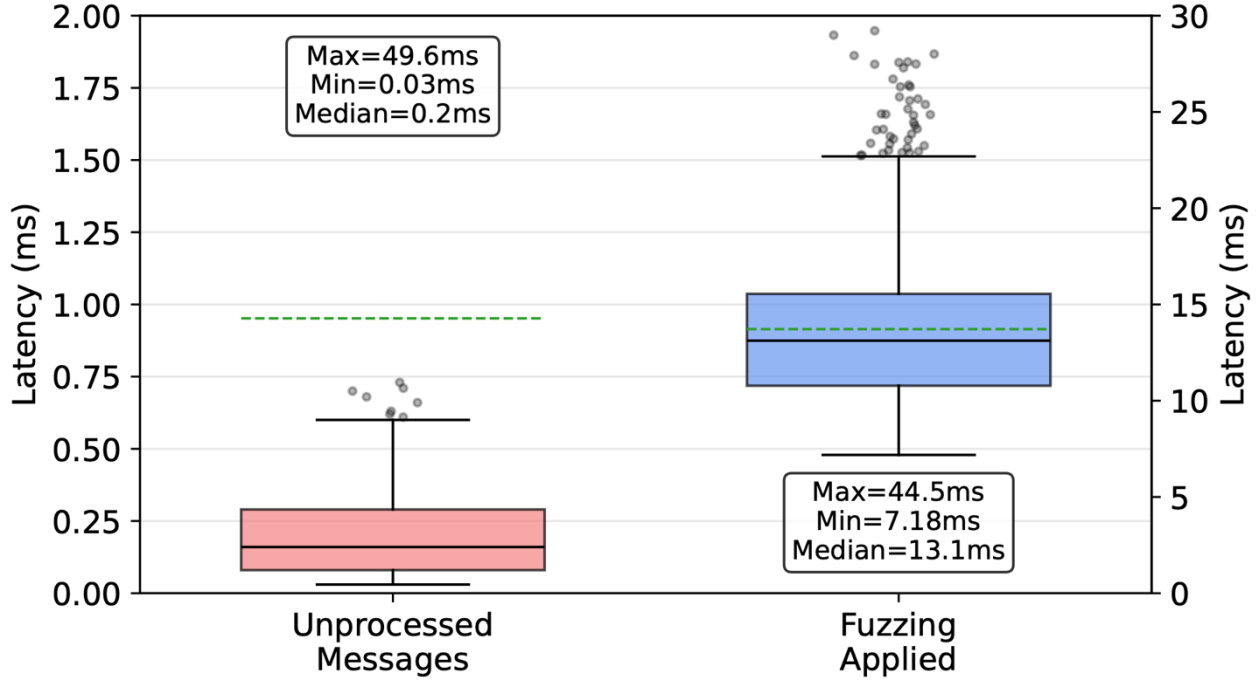
ORANClaw: Optimization

- During fuzzing, ORANClaw generates an on-the-fly differential model
- By highlighting added or removed states and transitions, ORANClaw identifies when a mutated packet triggers novel or anomalous network behaviour, providing feedback to optimize a genetic algorithm.



ORANClaw: Overhead

- To measure ORANClaw's E2 interception overhead, we track packet entry/exit times at the E2 bridge over 24 hours with processing enabled and disabled.
- ORANClaw adds around 13ms of overhead. This delay is low enough to satisfy the timing requirements (10ms to 1s) of theE2 interface



ORANClaw: Evaluation setup

- Due to O-RAN SC Near-RT RIC's complexity and multi-container overhead (E2 Term, Submgr, Rtmgr), we use FlexRIC for simpler E2 based testing.

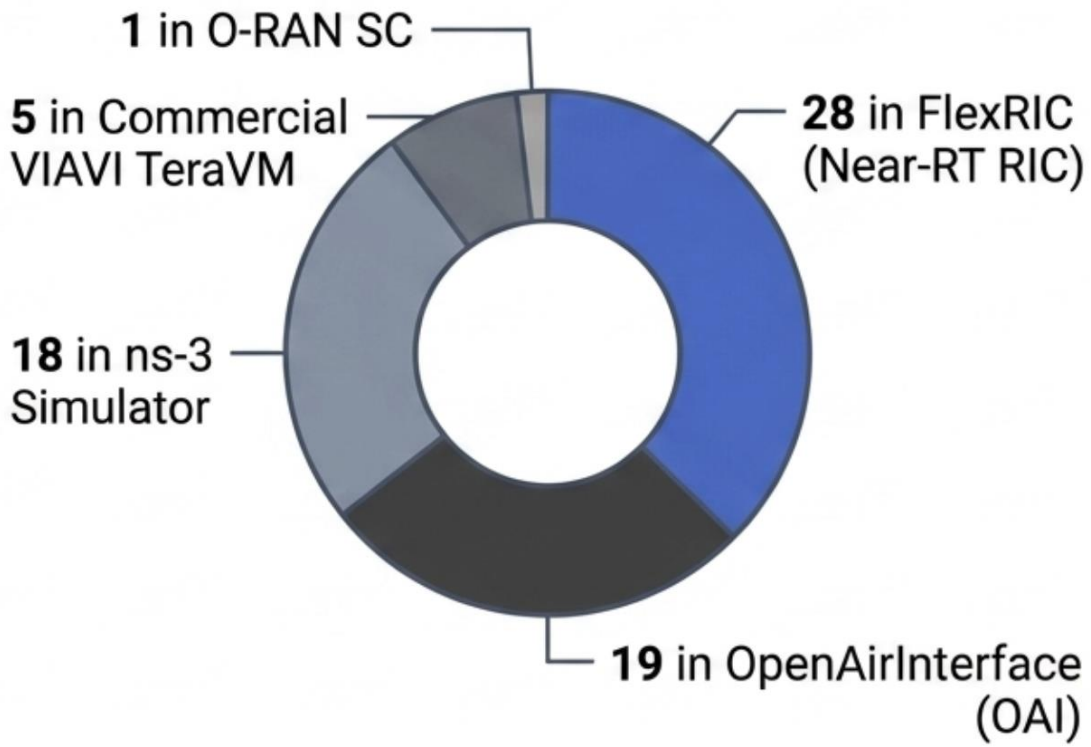
- However, we also evaluate O-RANClaw on full O-RAN-SC RIC for generalizability

- We evaluate OpenAirInterface and ns-3 gNB implementations.

- To show ORANClaw's real-world applicability, we evaluated VIAVI TeraVM RAN Simulator (v2.4-40470-43580c20), a commercial O-RAN testing tool, and found vulnerabilities in proprietary gNB and RIC.

ORANClaw: Results

- ORANClaw uncovered 71 new issues. 8 CVEs assigned so far.
- 68 out of 71 resulted in persistent DoS. A single malformed packet is capable of triggering a fault that requires manual restart of the system.



Code & Experiments Artifact Availability

Github: https://github.com/asset-group/ORANClaw-E2-MitM-Fuzzing

Website: https://asset-group.github.io/ORANClaw-E2-MitM-Fuzzing/

Questions?