

Demo: Mitigating Radio Resource Exhaustion Risks for Critical UEs in Private 5G Networks using Open RAN xApp

Mohit Bidikar
mohit.bidikar@york.ac.uk
University of York
York, United Kingdom

Anthony Moulds
anthony.moulds@york.ac.uk
University of York
York, United Kingdom

Poonam Yadav
poonam.yadav@york.ac.uk
University of York
York, United Kingdom

Abstract

Private 5G networks are increasingly used to support Operational Technology systems such as industrial robots and autonomous inspection platforms. These systems depend on stable, low latency wireless connectivity to maintain predictable behaviour. However, malicious or misconfigured devices can still degrade system performance by generating excessive traffic demand. While such devices cannot directly control radio resources, they can influence scheduler behaviour and consume a disproportionate share of Physical Resource Blocks (PRBs), reducing the reliability of other critical devices. This work presents an Open RAN xApp that enforces per-device radio resource prioritisation to preserve the stability of critical UE traffic in contested wireless environments.

CCS Concepts

• **Networks** → **Network architectures**; • **Hardware** → *Communication hardware, interfaces and storage*; • **Security and privacy** → **Systems security**; • **Applied computing** → *Telecommunications*.

Keywords

Open RAN, 5G, networks, xApps, RIC, srsRAN, SDR

ACM Reference Format:

Mohit Bidikar, Anthony Moulds, and Poonam Yadav. 2026. Demo: Mitigating Radio Resource Exhaustion Risks for Critical UEs in Private 5G Networks using Open RAN xApp. In *The 24th Annual International Conference on Mobile Systems, Applications and Services (MobiSys Companion '26)*, June 21–25, 2026, Cambridge, United Kingdom. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3812835.3814973>

1 Introduction

In 5G networks, Physical Resource Blocks (PRBs) are allocated by the base station scheduler rather than explicitly requested by User Equipment (UEs). However, compromised or misbehaving UEs can indirectly manipulate this allocation by continuously generating traffic. Such a UE does not need to break encryption or bypass authentication to degrade system performance. Instead, it can:

- (1) generate sustained uplink traffic,
- (2) maintain artificially high buffer occupancy,
- (3) trigger repeated scheduling requests, and
- (4) exploit fairness-based scheduling mechanisms.

For critical UEs, the impact extends beyond reduced throughput. It can disrupt stability, compromise control-loop predictability, and ultimately affect operational safety. To address this, we demonstrate how an Open RAN¹ Near-RT RIC xApp can enforce prioritised radio resource allocation at the gNB scheduler, protecting critical Operational Technology (OT) devices from performance degradation caused by malicious or noisy UEs.

2 Implementation

We built a fully functional 5G Standalone (SA) testbed using the srsRAN stack configured in Split 8 architecture [1], running entirely on a local server. This server also hosted a UDP client, which transmitted serialized motion control commands (JSON) originating from a hand-held wireless controller. For the radio setup, we used a USRP X310 SDR synchronized to an OctoClock-G CDA-2990 for frequency and phase stability. The core network was powered by Open5GS, providing full control and user plane functionality. For the critical UE, we used a Quectel RMU500-EK 5G modem with a programmable sysmoISIM, achieving successful registration and full internet connectivity over our private 5G network. The modem was then mounted on a Jetson Nano-based ROS2 Wheeltec ROSbot 4-wheel robot. A UDP server running on the robot received JSON motion packets over the established 5G link. The embedded commands were decoded and executed as ROS2 instructions, demonstrating low-latency, real-time control over 5G using only open-source components and off-the-shelf hardware. Figures 1 and 2 present the demo setup in the SYSTRON Lab² showing the robot UE connection to a 5G network.

The PRB-control xApp runs inside the OSC Near-RT RIC and continuously monitors UE-level RAN telemetry received from the gNB through the E2 interface. Based on this telemetry, the xApp first identifies active UEs and classifies them as either critical or non-critical using a predefined list of UE identifiers. Since subscriber-level identifiers such as IMSI or GUTI are not currently exposed to the Near-RT RIC in srsRAN, this classification is performed using session-level UE identifiers observed via E2 messages. In future production deployments, this mechanism can be extended to use slice identifiers (SST/SD) once they become available to the Near-RT RIC, enabling scalable prioritisation across multiple UEs and traffic classes.

Since the number of PRBs available is limited, the xApp assigns a larger share of PRBs to critical UEs during initialisation, while non-critical UEs operate with best-effort access to the remaining PRBs. The xApp then continuously monitors throughput and PRB utilisation using E2SM-KPM telemetry and dynamically adjusts



This work is licensed under a Creative Commons Attribution 4.0 International License. *MobiSys Companion '26, Cambridge, United Kingdom*
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2711-5/2026/08
<https://doi.org/10.1145/3812835.3814973>

¹<https://www.o-ran.org/>

²<https://systronlab.github.io/projects/xapp-driven-ran-optimisation>

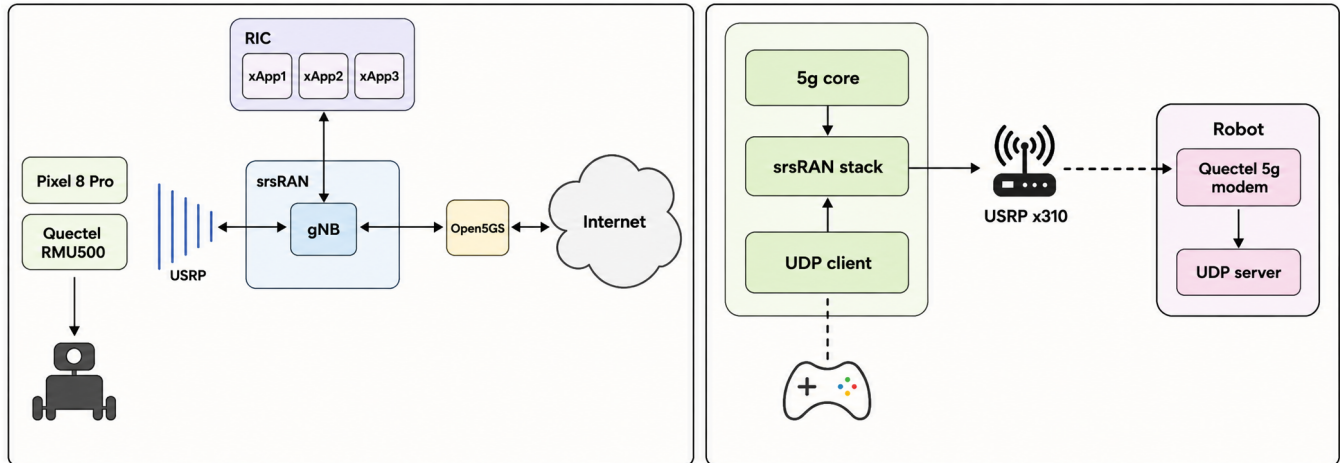


Figure 1: The end-to-end system architecture showing different UEs connection to Open RAN 5G network.

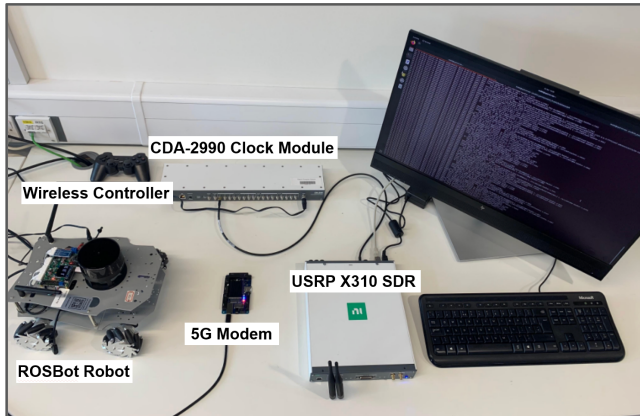


Figure 2: The demonstration setup in the lab showing a robot UE connection to an Open RAN 5G network.

PRB allocations through E2SM-RC control messages if contention from non-critical traffic affects the performance of the critical UEs. Overall, the xApp follows the standard O-RAN closed-loop control workflow consisting of telemetry collection, PRB utilisation evaluation, control decision, and RAN actuation.

System Components

- **Near-RT RIC:** Hosts the PRB-control xApp and manages the E2 connection with the gNB.
- **E2 Node/gNB:** Exposes RAN telemetry and accepts control commands through the E2 interface.
- **E2SM-KPM:** Provides periodic measurements such as PRB utilisation, UE throughput, buffer status, and radio quality indicators.
- **E2SM-RC:** Enables the xApp to send RAN control commands to influence scheduler behaviour.
- **PRB-control xApp:** Prioritises critical UEs, such as robotic or OT devices over non critical UEs.

Control Workflow

- (1) The xApp subscribes to UE-level and cell-level measurements using E2SM-KPM.
- (2) The gNB periodically reports PRB utilisation, throughput, buffer occupancy, and radio condition metrics.
- (3) The xApp classifies UEs as critical or non-critical based on RAN-level UE identifier (UE ID) currently exposed through the E2 interface in srsRAN.
- (4) The xApp assigns a higher baseline PRB allocation to critical UEs during initialisation, while non-critical UEs operate with best-effort access to the remaining resources.
- (5) The xApp sends an E2SM-RC control message to the gNB and the gNB scheduler applies the updated PRB policy.
- (6) The xApp continuously monitors the effect and adjusts the PRB allocation if required.

3 Results

Our PBR xApp successfully delivered PRB allocation commands to the gNB via the E2SM-RC interface, with the scheduler applying them immediately. This enabled real-time prioritisation of the critical robot over the non-critical UE.

UE-1 (Critical - ROSBot):

- Achieved higher and more stable throughput
- Exhibited noticeably smoother motion
- Experienced a significant reduction in jitter and micro-freezes

UE-2 (Non-critical - Google Pixel 8 Pro):

- Underwent controlled throughput reduction
- Maintained a fully stable connection

Acknowledgments

This work is supported, in part, by EPSRC and DSIT under grants EP/X040518/1, EP/Y037421/1, and EP/Y019229/1.

References

- [1] Michele Polese, Leonardo Bonati, Salvatore D'Oro, Stefano Basagni, and Tommaso Melodia. 2023. Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges. *IEEE Communications Surveys & Tutorials* 25, 2 (2023), 1376–1411. doi:10.1109/COMST.2023.3239220