



UNIVERSITÀ POLITECNICA DELLE MARCHE
Repository ISTITUZIONALE

Zero Trust Meets the Real-Time Loop: The Latency Price of Never Trusting Anything in the O-Cloud

This is the peer reviewed version of the following article:

Original

Zero Trust Meets the Real-Time Loop: The Latency Price of Never Trusting Anything in the O-Cloud / Riggio, R.. - In: IEEE NETWORK. - ISSN 0890-8044. - (2026). [Epub ahead of print] [10.1109/mnet.2026.3732160]

Availability:

This version is available at: 11566/363532 since: 2026-09-29T05:18:28Z

Publisher:

Published

DOI:10.1109/mnet.2026.3732160

Terms of use:

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. The use of copyrighted works requires the consent of the rights' holder (author or publisher). Works made available under a Creative Commons license or a Publisher's custom-made license can be used according to the terms and conditions contained therein. See editor's website for further information and terms and conditions.

This item was downloaded from IRIS Università Politecnica delle Marche (<https://iris.univpm.it>). When citing, please refer to the published version.

Publisher copyright:

IEEE - Postprint/Author's Accepted Manuscript

©2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. To access the final edited and published work see 10.1109/mnet.2026.3732160

(Article begins on next page)

Zero Trust Meets the Real-Time Loop: The Latency Price of Never Trusting Anything in the O-Cloud

Roberto Riggio

Università Politecnica delle Marche, Ancona, Italy

Email: r.riggio@univpm.it

Abstract—The disaggregation that makes Open RAN attractive replaces the sealed base station with a multi-vendor software ecosystem, yet its security baseline still protects the boundary far more thoroughly than what happens inside it: once a third-party application is onboarded and a protected channel established, its subsequent actions on the RAN go essentially unchecked. Zero trust is the natural remedy, but it collides with the hard real-time budgets of the RAN control loop, and this collision has not been quantified. We identify four east-west attack paths that perimeter-oriented protection leaves open, and map the zero-trust primitives of workload identity, per-message authorization, and continuous attestation onto the O-RAN interfaces. Where prior work has measured the cost of encrypting O-RAN interfaces, we measure the cost of authorizing each action. On an O-RAN Software Community Near-RT RIC, authorization evaluated at a remote decision point inflates the 99th-percentile E2 loop latency to 14.2 ms and pushes 4% of transactions past the 10 ms deadline. Co-locating policy evaluation with the enforcement point and bounding decision staleness returns the 99th percentile to 7.5 ms while holding post-revocation exposure below 625 ms. The security-latency trade-off is therefore an operator-tunable parameter rather than an architectural absolute, and we close with guidelines for selecting an operating point.

Index Terms—Open RAN, O-Cloud, zero trust, RAN Intelligent Controller, xApps, workload identity, remote attestation, authorization, network security, real-time control loops.

I. INTRODUCTION

The Open Radio Access Network (O-RAN) architecture decomposes the base station into disaggregated functions connected by open interfaces and steered by two RAN Intelligent Controllers (RICs) operating at non-real-time and near-real-time timescales [1]. That decomposition is the source of O-RAN's appeal and equally of its security problem: the sealed appliance has been replaced by a cloud platform, the O-Cloud, on which software of heterogeneous provenance executes side by side and exercises direct control over the radio. The O-RAN Alliance's security specifications protect that boundary well, mandating mutual authentication, transport protection, and access control on the open interfaces and subjecting candidate applications to validation before admission.

What they do not address is what happens after admission. Once an xApp has been onboarded and a protected channel established, its behaviour is constrained by the scope it declared at registration rather than by any continuing verification of what it is and what it may do at the moment it acts. The consequences are not hypothetical. Systematic threat analyses identify the RICs and their hosted applications as the dominant

risk area [2], and experimentation against a deployed O-RAN stack has shown that an adversary positioned inside the platform can degrade service without ever crossing the perimeter [3].

Zero trust is the established answer to this class of problem: no trust conferred by network position, authorisation evaluated per request rather than per session, and continuous rather than one-time verification of the requester. Prior O-RAN work has packaged zero-trust services as xApps, demonstrating authentication, intrusion detection, and secure slicing on a research platform [4], and has proposed scalable authorisation for the xApp population [5], establishing that zero trust is architecturally compatible with O-RAN. Complementing this, Groen *et al.* have measured what protecting O-RAN costs, quantifying the impact of encrypting the E2 interface and the Open Fronthaul and finding the E2 penalty modest [6], [7]. Their subject is the channel; ours is the actor. Verifying who is acting, and whether the action falls within their scope, is work that recurs with every control message rather than once per session, and it is therefore the mechanism most likely to collide with the Near-RT RIC's 10 ms budget.

This article treats that collision as its subject. We characterise the east-west attack paths the perimeter-oriented baseline leaves open once an application is inside the O-Cloud, and map the zero-trust primitives of workload identity, per-message authorisation, and continuous attestation onto the O-RAN interfaces, distinguishing the enforcement topologies whose costs differ by an order of magnitude. We then quantify those costs on an O-RAN Software Community Near-RT RIC, showing the security-latency trade-off to be a design parameter rather than an architectural absolute, and close with guidelines for selecting an operating point against an explicit exposure-window objective.

II. THE O-CLOUD AS AN UNTRUSTED INTERIOR

O-RAN's control plane spans two timescales: the Non-RT RIC hosts rApps and acts over A1, O1, and R1 on loops of a second or longer, while the Near-RT RIC hosts xApps that subscribe to reports and issue control actions over E2 within 10 ms to one second [1]. Both controllers execute as containerised workloads on the O-Cloud. Existing treatments proceed interface by interface [8], and the most thorough also quantify what protecting each interface costs [6]. What they do not address is an adversary whose position is not on an interface but inside the platform terminating them, i.e., an

insider [9], [10]. Four attack paths follow, none requiring the boundary to be crossed. The count is not arbitrary: the shared platform beneath the applications affords one path that no admission decision governs, and admission itself, which binds an identity to a scope and presumes both endure, affords one for each element that can fail.

The first is lateral movement through shared platform services. The Near-RT RIC multiplexes its applications over a shared data layer, a message router, and a single E2 termination through which every subscription and control action passes, so an adversary who compromises the least privileged xApp acquires a foothold adjacent to all of it. Security testing of open-source Near-RT RICs has shown that co-residency alone suffices to establish covert channels between nominally separated applications [11]. Isolation that holds at the network boundary does not hold between tenants of the same controller.

The second is over-scoped authority that is never revisited. An xApp declares the RAN functions and report types it needs at registration, and that declaration is checked once. It is generous in practice, since it must cover every situation the application might encounter, and it persists unchanged as the network and the application evolve. An xApp onboarded to optimise handovers in one cluster retains the ability to read measurements for cells it will never act upon. The gap between authority granted and authority exercised is the attacker's working room.

The third is impersonation. Where east-west protection exists at all, it commonly authenticates workloads by network position, a weak credential in an orchestrated environment where addresses are recycled and workloads rescheduled. Interception and manipulation of control-plane traffic between O-RAN components has been demonstrated on the inter-controller path [12]; the weakness applies with greater force between microservices inside a single controller, where the assumption of a trusted network is strongest.

The fourth is divergence after admission. Onboarding validates an artefact, but that guarantee expires the moment the artefact begins to execute: a workload modified in place, restarted from a substituted image, or updated through a compromised supply chain continues to hold the authorisations granted to the version that was validated.

These paths differ in mechanism but share a structure. Authority is conferred once, at admission, on an identity established by position or by a check that is not repeated, and it is thereafter exercised without verification at the moment of action. The consequence is severe in O-RAN specifically: an xApp acting outside its intent need not send malformed messages, since a handover threshold or resource allocation issued through the correct interface in the correct format is indistinguishable at the wire level from legitimate control, and its effect on the radio is immediate. What is missing is not stronger protection of the channel but verification of the actor and the action at the point of use.

III. A ZERO-TRUST REFERENCE DESIGN FOR THE O-CLOUD

Zero trust addresses this failure, and NIST SP 800-207 codifies it for enterprise systems. Instantiating it in the O-

Cloud requires three primitives: a cryptographic identity for every workload, an authorisation decision attached to every action, and attestation that keeps the binding between the two honest over time. Fig. 1 places the corresponding components relative to the O-RAN architecture. The principle running through all three, and the one deciding whether the result is deployable, is that the substance of a mechanism and the location of its enforcement point are independent choices: the first is dictated by the threat, the second is free, and it is where the latency is decided.

Identity comes first, because the other two primitives are meaningless without it. Every rApp, xApp, and platform service receives a cryptographic identity naming the workload rather than its location, as SPIFFE established for cloud-native systems [13]. It is short-lived, rotated automatically, and issued to the running workload rather than baked into its image, so that possession of the artefact is not possession of the identity. Issuance attaches to procedures that already exist, i.e., xApp registration with the Near-RT RIC, rApp registration over R1, platform identity over O2, and it is the point at which the declared scope, today recorded and then forgotten, becomes a credential that every subsequent decision can be evaluated against.

Authorisation is where the design departs most sharply from current practice. Each action is evaluated against the requester's identity, its declared scope, and the state of the network at that instant, by a policy decision point logically separate from the code it governs. What remains is where that decision is taken, and three topologies differ by roughly an order of magnitude in cost. A remote decision point is the most flexible and the most expensive, since every decision carries a network round trip. A sidecar co-located with the enforcement point removes the network hop but retains an inter-process boundary. In-process evaluation against a pre-compiled bundle removes both, at the cost of a delay before a policy change takes effect.

Attestation closes the gap admission-time validation leaves open. Measured boot and runtime integrity measurement establish that the O-Cloud node is in a known state, while workload attestation binds the digest of the running image to the identity issued to it, so a workload that has diverged from the validated artefact can no longer present a valid credential; the RATS architecture supplies the vocabulary for this exchange [14]. Attestation is not on the datapath, so its refresh interval trades detection latency against platform resources rather than against control-loop latency.

Table I maps the three primitives onto the O-RAN interfaces. Where loops run at a second or longer, every mechanism is affordable in its most flexible form, and there is no principled reason for the interior of the O-Cloud to be less protected than its boundary. On E2, where the budget is 10 ms and much of it is already committed to the control logic, only the cheapest topologies survive.

IV. WHERE THE COST LANDS

Before measuring the design it is worth being precise about what should be expensive, so the results either confirm a model

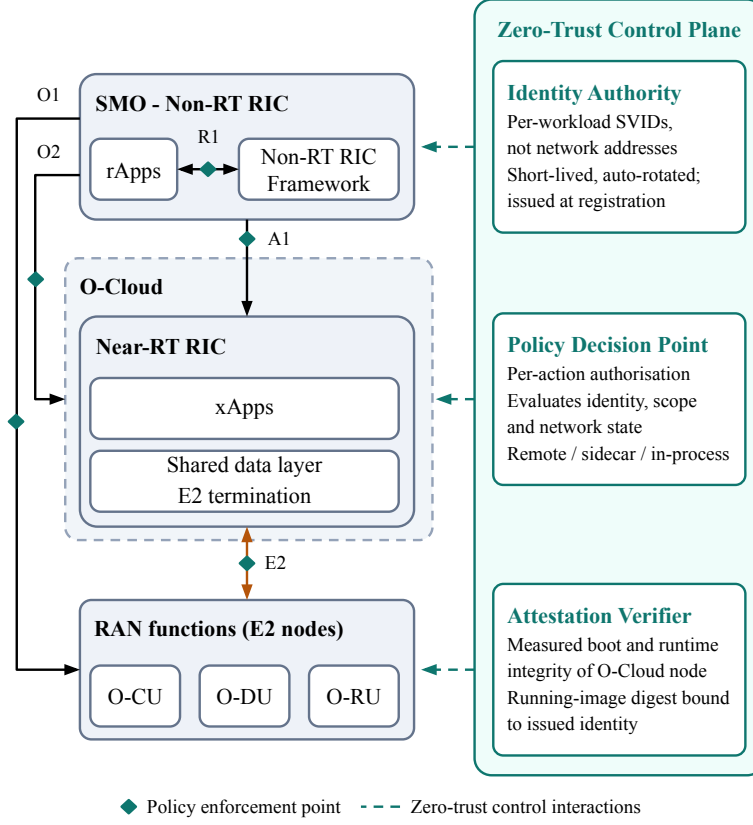


Fig. 1. Zero-trust reference design for the O-Cloud. The identity authority, policy decision point, and attestation verifier are shown relative to the Non-RT RIC, the Near-RT RIC, and the O-Cloud platform, with policy enforcement points marked on each O-RAN interface. The E2 control path, on which the 10 ms budget applies, is highlighted.

TABLE I
ZERO-TRUST ENFORCEMENT PER O-RAN INTERFACE. COST CLASS IS RELATIVE TO THE INTERFACE'S OWN LOOP BUDGET: LOW IS BELOW ONE PERCENT, MODERATE IS A FEW PERCENT, HIGH IS A SIGNIFICANT FRACTION.

Interface	Loop budget	Identity	Authorisation topology	Cost class
A1	> 1 s	Per-workload, rotated	Remote decision point	Low
O1	> 1 s	Per-workload, rotated	Remote decision point	Low
O2	Minutes	Platform-issued	Remote decision point	Low
R1	> 1 s	Per-workload, rotated	Remote or sidecar	Low
E2	10 ms–1 s	Per-workload, rotated	Sidecar or in-process	Moderate to high

or contradict it. The work zero trust adds to an O-RAN control loop falls into four terms that behave very differently.

The first is session establishment. Mutual authentication, certificate validation, and key agreement are expensive relative to a single message but are incurred once per connection, and on a stable platform they amortise to insignificance. Two circumstances qualify that. Short-lived credentials make establishment recurring rather than one-off, and its cost is concentrated in time: a rolling upgrade or a mass reconnection after controller failover peaks exactly when the control loops are themselves re-converging and can least absorb it.

The second is per-message cryptographic protection. Once a session exists, protecting each message is symmetric-key work on a few hundred bytes, measured in microseconds and stable enough to treat as a constant. Its significance is aggregate rather than per-transaction: E2 indication traffic can run to thousands of messages per second per node, so

what is invisible in a latency budget remains visible in CPU accounting.

The third is the policy decision. Its substance is cheap, since evaluating whether a given identity, holding a given scope, may issue a given action in the current network state is a small computation over a small amount of state. What varies by orders of magnitude is where that computation happens. A remote policy decision point costs a network round trip and a scheduling delay at the far end; a sidecar retains process-boundary crossings and serialisation; in-process evaluation against a pre-compiled bundle costs a function call. The same policy, expressing the same security property, lands anywhere from a fraction of a millisecond to several milliseconds depending on a deployment decision that has nothing to do with security strength.

The fourth is attestation, which never appears on the datapath. Evidence is appraised asynchronously, so it adds no

latency to any transaction, but it competes for the CPU and memory bandwidth the RIC and the RAN functions depend on, and that contention surfaces as jitter rather than as a visible security cost. It is the one term of the four the evaluation below does not quantify.

Taken together these terms yield the observation on which the rest of the article rests. The dominant cost scales with the rate at which authorisation decisions are taken, not with the rate at which messages are sent, and the two are independent whenever a decision can be reused. An xApp issuing a thousand control actions of the same type against the same cell under an unchanged policy requires one evaluation, not a thousand, provided the platform may remember the answer. Caching therefore converts the dominant term from a per-message cost into a per-policy-change cost, which is what makes per-action authorisation tractable within 10 ms at all. It does so by spending something else: a cached authorisation stays valid for its full lifetime, so an xApp whose credential has been revoked may continue to act for the remainder of that interval. The cache lifetime is therefore not an implementation detail but the parameter that sets the exposure window.

V. CASE STUDY: ZERO TRUST ON AN O-RAN SOFTWARE COMMUNITY NEAR-RT RIC

We evaluate the design on a containerised O-RAN Software Community Near-RT RIC hosting three xApps, handover optimisation, resource allocation, and a monitor that subscribes without acting, with the radio side supplied by an ns-3/5G-LENA emulation of a multi-cell scenario, so that indication traffic and control actions reflect realistic load. Each workload receives a short-lived identity from a SPIFFE-compatible authority, authorisation decisions are evaluated by Open Policy Agent¹ in the topologies described above, and attestation follows the RATS model.

Four postures are compared: the platform as it ships, with no mutual authentication between internal components and no authorisation beyond the scope declared at registration; the same with mutual authentication and transport protection on all internal paths; per-action authorisation evaluated at a remote decision point; and that decision point moved to a sidecar co-located with the E2 termination, with results cached for a bounded interval. For each we record 20 000 complete transactions, measured from the arrival of a RIC Indication at the xApp to the departure of the corresponding RIC Control message, and report the distribution rather than the mean alone, since a control loop is characterised by its tail.

Fig. 2 shows those distributions against the 10 ms budget. The baseline completes at a mean of 3.40 ms and a 99th percentile of 6.63 ms, with no transaction exceeding the deadline. Mutual authentication moves the 99th percentile to 6.86 ms, an increase of roughly 0.2 ms accounted for almost entirely by per-message symmetric work: once a session exists, protecting each message is not what makes zero trust expensive. The remote decision point pushes the 99th percentile to 14.15 ms and causes 3.98% of transactions to miss the deadline, while its mean, at 6.04 ms, remains comfortably within budget, which is

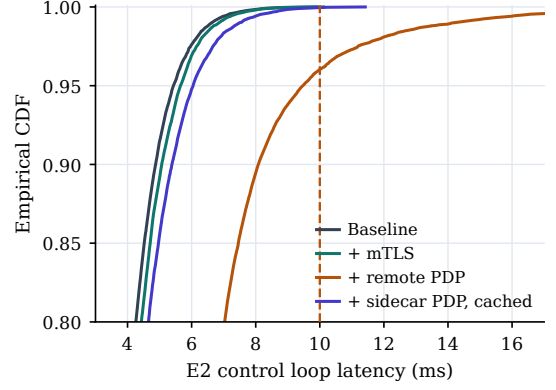


Fig. 2. Distribution of end-to-end E2 control loop latency under the four security postures. The vertical line marks the 10 ms Near-RT budget; the fraction of each distribution to its right is the deadline violation rate.

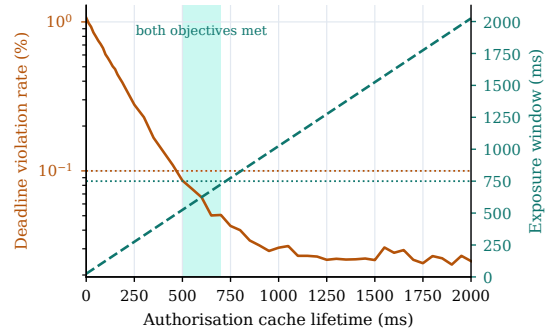


Fig. 3. Deadline violation rate and post-revocation exposure window as a function of authorisation cache lifetime. The shaded region marks lifetimes for which both objectives are met.

why reporting means alone would mislead, the loop appearing healthy on average while failing at a rate no operator would accept. Relocating the decision to a sidecar and permitting caching returns the 99th percentile to 7.50 ms and the violation rate to 0.04%, at a cache hit ratio of 95.0%. The mechanism is unchanged between these two postures; only its placement differs.

Because caching makes that posture viable, its cost deserves to be stated in security terms rather than buried in a configuration file. Fig. 3 sweeps the cache lifetime against the deadline violation rate and the exposure window: the interval during which an xApp whose authorisation has been withdrawn can still act. The two move in opposite directions. With caching disabled the violation rate reaches 1.07%, since almost every action forces a fresh evaluation against current network state; a lifetime long enough to drive violations to their floor leaves a revoked application in control for seconds. Neither extreme is defensible. Reading the figure requires an exposure bound. Under an illustrative 750 ms bound and a violation objective of 0.1%, lifetimes between 500 ms and 700 ms satisfy both, and at 600 ms the violation rate is 0.04% with exposure bounded at 625 ms. We report this as an instance of the method, not a recommendation.

Three findings follow. Transport security is affordable on every O-RAN interface including E2, so unprotected east-west paths inside the O-Cloud reflect habit rather than any

¹<https://www.openpolicyagent.org>.

performance constraint. Per-action authorisation is affordable on the slow interfaces in any topology, and on E2 only when evaluated locally with a reusable result. And the residual cost is concentrated in a single quantity, the exposure window, which can be measured and bounded.

VI. DESIGN GUIDELINES AND OPEN CHALLENGES

Four guidelines follow. The placement of the policy decision point should be a function of the loop it governs rather than a platform-wide convention: on paths bounded at 10 ms the decision must be evaluated within the enforcement point's address space or alongside it, a remote point being not a stricter configuration of the same design but an unusable one. Policy should be compiled and distributed ahead of use rather than queried at the moment of use, which recasts distribution as a problem O-RAN already solves for AI policy, with the same tolerance for propagation delay and the same expectation that enforcement survives an unreachable source. The cache lifetime should be exposed as an operator-facing security parameter rather than buried as a performance knob, and derived rather than guessed, on two grounds. Cache staleness is the last term of exposure budget beginning when a compromised xApp is detected and running through revocation and its propagation; where detection takes seconds, as it generally does, driving the cache below that term buys no security and costs deadlines. The lifetime should be small against the dominant term, not small absolutely. And the bound belongs in control actions rather than milliseconds, the exposure divided by the application's control cadence, since the harm is set by how many actions a revoked xApp can still issue and how far each moves the radio. An operator who can say how many rogue adjustments the network absorbs before service degrades has the bound. And the posture should differ across interfaces by design, since uniformity either imposes the 10 ms loop's constraints on interfaces that could afford far more or relaxes the slow-path configuration until it fits E2.

Three challenges remain open. The most immediate is that attestation, alone among the mechanisms considered here, has a cost this article has not measured. Because its resources are shared with the RIC and the RAN functions, its effect appears as contention rather than as latency on any single transaction. Determining how much O-Cloud capacity must be reserved so that a useful refresh interval does not perturb the loops attestation protects is a question the design makes precise but does not answer. It sharpens where workloads execute on accelerators, since attesting a partitioned GPU is not the established procedure attesting a container image has become, and AI workloads sharing an O-Cloud with RAN functions will make this the common case.

The second is standardisation. Nothing in the O-RAN specifications requires a workload identity distinct from network position, and until the security specifications name one, deployments will interoperate at the level of the interface while diverging at the level of the platform [15]. A per-workload identity is exactly the kind of requirement that is cheap to mandate early and expensive to retrofit.

The third is the one this design does not address. Per-action authorisation verifies each action against the identity

and scope of the application issuing it, yet every action a colluding pair of xApps might use to degrade a cell can be individually well-formed, correctly scoped, and legitimately authorised. Oscillating a handover threshold and a transmit power setting in concert is an attack composed entirely of permitted operations. Attack graphs, the established tool for multi-step compromise, do not reach it: their edges are exploits chained by precondition, and here there is neither. Detecting such an attack requires reasoning over sequences of authorised actions and their aggregate effect on the radio, for which the RIC's own visibility into the consequences of RAN actions may prove the more promising instrument.

VII. CONCLUSION

O-RAN's security baseline defends a boundary that disaggregation has made progressively less meaningful. Inside it, applications of independent provenance exercise authority granted once at admission and act on the radio through interfaces that verify the channel but not the actor. Zero trust answers this directly, and what has stood in the way of adopting it is not whether its mechanisms fit the architecture but whether they fit the deadlines.

They do, but not uniformly and not by default. What decides feasibility is where enforcement happens rather than how strong it is: the same authorisation policy is unusable on E2 when evaluated remotely and comfortable when evaluated locally against a bounded-staleness cache. The cost that remains is neither diffuse nor hidden. It is the interval during which a withdrawn authorisation is still honoured, and because that interval is configured it can be bounded, declared, and held to account like any other operational objective.

What zero trust cannot supply is judgement about actions that are individually legitimate. Securing the O-Cloud against an adversary acting entirely within its permissions remains the harder problem, and the RIC's own view of what its applications actually cause may be where the answer lies.

REFERENCES

- [1] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia, "Understanding O-RAN: Architecture, interfaces, algorithms, security, and research challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1376–1411, 2023.
- [2] D. Mimran, R. Bitton, Y. Kfir, E. Klevansky, O. Brodt, H. Lehmann, Y. Elovici, and A. Shabtai, "Security of open radio access networks," *Computers & Security*, vol. 122, p. 102890, 2022.
- [3] P. Baguer, G. M. Yilma, E. Municio, G. Garcia-Aviles, A. Garcia-Saavedra, M. Liebsch, and X. Costa-Pérez, "Attacking o-ran interfaces: Threat modeling, analysis and practical experimentation," *IEEE Open Journal of the Communications Society*, vol. 5, pp. 4559–4577, 2024.
- [4] A. S. Abdalla, J. Moore, N. Adhikari, and V. Marojevic, "Ztran: Prototyping zero trust security xapps for open radio access network deployments," *IEEE Wireless Communications*, vol. 31, no. 2, pp. 66–73, 2024.
- [5] T. O. Atalay, S. Maitra, D. Stojadinovic, A. Stavrou, and H. Wang, "Securing 5G OpenRAN with a scalable authorization framework for xApps," in *Proc. IEEE INFOCOM*, 2023.
- [6] J. Groen, S. D'Oro, U. Demir, L. Bonati, M. Polese, T. Melodia, and K. Chowdhury, "Implementing and evaluating security in O-RAN: Interfaces, intelligence, and platforms," *IEEE Network*, vol. 39, no. 1, pp. 227–234, 2025.
- [7] J. Groen, S. D'Oro, U. Demir, L. Bonati, D. Villa, M. Polese, T. Melodia, and K. Chowdhury, "Securing O-RAN open interfaces," *IEEE Transactions on Mobile Computing*, vol. 23, no. 12, pp. 11 265–11 277, 2024.

- [8] A. S. Abdalla and V. Marojevic, "End-to-end O-RAN security architecture, threat surface, coverage, and the case of the open fronthaul," *IEEE Communications Standards Magazine*, vol. 8, no. 1, pp. 36–43, 2024.
- [9] O-RAN Alliance, "Zero Trust Architecture for Secure O-RAN," O-RAN Alliance e.V., White Paper O-RAN.WP.ZTA for secure O-RAN-v1.0, 2024.
- [10] K. Ramezanpour and J. Jagannath, "Intelligent zero trust architecture for 5g/6g networks: Principles, challenges, and the role of machine learning in the context of O-RAN," *Computer Networks*, vol. 217, p. 109358, 2022.
- [11] K. Thimmaraju, A. Shaik, S. Flück, P. J. Fullana Mora, C. Werling, and J.-P. Seifert, "Security testing the O-RAN near-real time RIC and A1 interface," in *Proc. 17th ACM Conf. Security and Privacy in Wireless and Mobile Networks (WiSec)*, 2024.
- [12] W. Tiberti, E. Di Fina, A. Marotta, and D. Cassioli, "Impact of man-in-the-middle attacks to the O-RAN inter-controllers interface," in *Proc. IEEE Future Networks World Forum (FNWF)*, 2022.
- [13] SPIFFE Project, "SPIFFE: Secure production identity framework for everyone," <https://spiffe.io>, 2026, accessed: 2026-08-10.
- [14] H. Birkholz, D. Thaler, M. Richardson, N. Smith, and W. Pan, "Remote ATtestation procedureS (RATS) architecture," Internet Engineering Task Force, RFC 9334, Jan. 2023.
- [15] O-RAN Alliance, "O-RAN security requirements and controls specification," O-RAN Alliance e.V., Technical Specification O-RAN.WG11.TS.SRCS.0-R005-v15.00, 2026.

Roberto Riggio is an Associate Professor at Università Politecnica Delle Marche, Ancona, Italy. He has published more than 130 papers in internationally refereed journals and conferences. His research interests revolve around optimization and algorithmic problems in networked and distributed systems.