Poster: Leveraging Isolation and Adaptive Resource Management for Secure Virtualized ORAN

Lam D. Nguyen, Wei Shao, Hajime Suzuki, Shiping Chen, and Wei Ni
CSIRO Data61, Eveleigh, NSW 2015, Australia.

***Abstract*—With its disaggregated and virtualized design, the Open Radio Access Network (ORAN) architecture offers significant flexibility for 5G and emerging 6G networks, but introduces critical security challenges in multi-tenant environments. When hosted on shared hardware, virtualized components such as the O-DU, O-CU, and RIC are vulnerable to side-channel attacks, where malicious actors can exploit timing and resource-based patterns to infer sensitive information across tenant boundaries. This paper proposes a model that enhances isolation and resource management within ORAN deployments by integrating deep reinforcement learning (DRL) with container orchestration technologies. We present a workflow of the system that utilizes DRL to dynamically manage resources and enforce strong tenant separation within Kubernetes-managed environments. Our approach mitigates cross-tenant interference and minimizes exposure to side-channel attacks, supporting the integrity and resilience of ORAN's multi-tenant infrastructure. The proposed solution provides a scalable framework for secure ORAN deployments, setting a foundation for robust 6G network services while suggesting future enhancements such as advanced adaptive monitoring and timing obfuscation to fortify against timing-based exploits.**

***Index Terms*—ORAN, Virtualization, Orchestration, Side-Channel Attack, Deep Reinforcement Learning**

I. Introduction

Context. The Open Radio Access Network architecture is designed to offer flexibility, scalability, and vendor interoperability in 5G and upcoming 6G networks by disaggregating traditional monolithic Radio Access Network (RAN) functions [1]. In ORAN, critical RAN functions such as the Distributed Unit (O-DU), Centralized Unit (O-CU), and the RAN Intelligent Controller (RIC) are virtualized and often deployed on shared hardware infrastructures. This virtualized, multi-tenant environment allows operators to scale and adapt their networks efficiently, deploying specialized applications (xApps) and services tailored for diverse use cases, from high-bandwidth applications to latency-sensitive services [2]. However, this shared infrastructure introduces new security concerns, particularly side-channel attacks exploiting shared resources.

Problems. As ORAN components share underlying physical resources like CPU, memory, and cache, they become susceptible to side-channel attacks [3]. These attacks leverage subtle patterns in resource usage, such as cache timing or memory access, to infer sensitive information across tenants or virtualized functions. For instance, an attacker may use cache timing techniques to monitor usage patterns in a shared environment, gaining insights into the behavior of critical RAN functions or even extracting confidential data. This threat is especially relevant in a multi-tenant ORAN setup, where strong isolation is essential to prevent cross-tenant interference and protect against information leakage. The risk posed by these side-channel vulnerabilities underscores the need for adaptive and intelligent resource isolation and management mechanisms.

This research paper is conducted under the 6G Security Research and Development Project, as led by the Commonwealth Scientific and Industrial Research Organisation (CSIRO) through funding appropriated by the Australian Government's Department of Home Affairs. This paper does not reflect any Australian Government policy position. For more information regarding this Project, please refer to https://research.csiro.au/6gsecurity/.

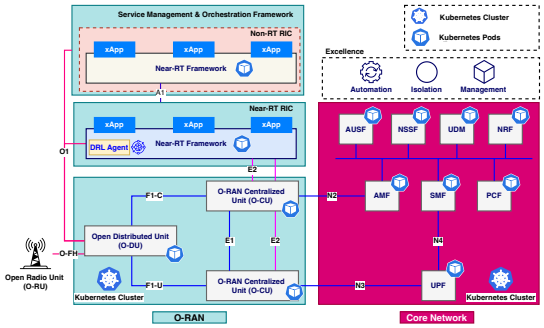

Fig. 1: Leveraging Isolation and Resource Management for securing ORAN components.

Proposed Solution. This paper proposes a novel model solution that enhances isolation and resource management within ORAN deployments by integrating deep reinforcement learning (DRL) with container orchestration technologies. By leveraging Kubernetes' advanced features, such as namespaces for logical separation, network policies to control inter-component communication, and resource quotas to limit CPU and memory usage, we enhance the isolation between ORAN components as shown in Fig. 1. The integration of DRL enables dynamic and intelligent resource allocation, allowing the system to detect and mitigate side-channel threats proactively. We present a system workflow that utilizes DRL to manage resources adaptively and enforce strong tenant separation within Kubernetes-managed environments.

II. Solusion Design

In this section, we present the new solution that enhances the security of multi-tenant ORAN environments by integrating Kubernetes-based isolation and resource management

with DRL. This design aims to mitigate side-channel attacks by enforcing strong tenant isolation, dynamically managing resources, and proactively detecting potential threats.

A. Kubernetes-Based Isolation and Resource Management

Kubernetes serves as the foundational platform for orchestrating containerized ORAN components. Using its features, we achieve robust isolation and efficient resource management, including:i) **Feature 1**: Kubernetes *namespaces* provide a scope for names within the cluster, allowing for logical separation of ORAN components and tenants. Each tenant and ORAN function (e.g., O-DU, O-CU, RIC) operates within its own namespace, ensuring that resources and workloads are isolated at the cluster level; ii) **Feature 2**: Network policies in Kubernetes define how pods communicate with each other and with external services. By configuring strict network policies, we limit the communication paths between components, preventing unauthorized access and potential lateral movement by malicious actors; iii) **Feature 3**: We implement resource quotas and limits to prevent resource exhaustion and reduce the risk of timing-based side-channel attacks. *Quotas at the namespace* level and limits at the pod level control the maximum CPU, memory, and storage resources that can be consumed, ensuring fair resource distribution among tenants.

B. Integration of Deep Reinforcement Learning

We integrate DRL into the Kubernetes-managed ORAN environment to enhance security and optimize resource utilization dynamically: i) DRL agents are trained to monitor resource usage patterns and adjust real-time allocations. By learning from the environment, these agents can allocate resources efficiently while introducing variability that obscures usage patterns exploitable by side-channel attacks; ii) The DRL system continuously evaluates the state of the cluster to enforce network policies and resource limits adaptively. This proactive approach allows the system to respond to anomalous behaviors or changing workload demands promptly.

The workflow of the integrated system is described in Fig. 2 below:

- **Step 1:** Collect real-time metrics (CPU, memory, traffic patterns) from ORAN components (O-DU, O-CU, Near-RT RIC, Non-RT RIC) via O1, A1, E2 interfaces (1).
- **Step 2:** Preprocess the data (cleaning, normalization, feature extraction) to make it ready for training and inference (2).
- **Step 3:** Train DRL agents using algorithms like Proximal Policy Optimization (PPO) to develop policies for resource management, tenant isolation, and security (3).
- **Step 4:** Publish the trained policies through the **DRL Agent Management** module (4). Deploy the agents to the DRL Inference module for real-time decision-making. The agents are integrated with Kubernetes APIs to manage system resources.
- **Step 5:** The DRL inference system processes real-time inputs to evaluate the network state and produce actionable insights. These insights are translated into: i) configuration Updates (O1) for resource management, ii) control Actions (E2) to optimize scheduling or traffic flow, iii) policy Updates (A1/E2) for tenant-specific and operational policies (5).
- **Step 6:** Execute these actions dynamically on ORAN components to optimize performance, isolation, and security (6).
- **Step 7:** Monitor performance and collect feedback metrics (resource utilization, latency, interference) (7).
- **Step 8:** Refine DRL policies based on feedback and updated observations from O1, A1, and E2 (8).
- **Step 9:** Iterate continuously, retraining policies to adapt to network conditions and emerging threats (9), (10).

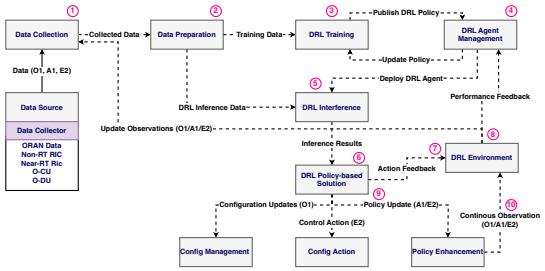

Fig. 2: End-to-End Deep Reinforcement Learning workflow in ORAN

III. Conclusion

This paper presents a robust integration of Deep Reinforcement Learning (DRL) and Kubernetes to address resource management and security challenges in multi-tenant ORAN environments. The proposed framework effectively mitigates side-channel attacks and optimizes network performance by leveraging DRL agents for dynamic resource allocation and adaptive policy enforcement, combined with Kubernetes' isolation features. This scalable solution lays the groundwork for secure, efficient, and adaptive ORAN deployments, paving the way for future 6G networks. As a next step, we plan to implement a prototype of the proposed approach to evaluate its performance in real-world scenarios. This will involve investigating its effectiveness in meeting the intended objectives, such as enhanced security, improved resource utilization, and scalability in dynamic ORAN environments. These efforts will further validate the feasibility of this framework and inform future advancements in 6G virtualization.

References

[1] W. Jiang, B. Han, M. A. Habibi, and H. D. Schotten, "The road towards 6g: A comprehensive survey," *IEEE Open Journal of the Communications Society*, vol. 2, pp. 334–366, 2021.

[2] A. A. Khan, A. A. Laghari, A. M. Baqasah, R. Alroobaea, T. R. Gadekallu, G. A. Sampedro, and Y. Zhu, "Oran-b5g: A next generation open radio access network architecture with machine learning for beyond 5g in industrial 5.0," *IEEE Transactions on Green Communications and Networking*, 2024.

[3] M. Liyanage, A. Braeken, S. Shahabuddin, and P. Ranaweera, "Open ran security: Challenges and opportunities," *Journal of Network and Computer Applications*, vol. 214, p. 103621, 2023.

LEVERAGING ISOLATION AND ADAPTIVE RESOURCE MANAGEMENT FOR SECURE VIRTUALIZED ORAN

Lam D. Nguyen, Wei Shao, Hajime Suzuki, Shiping Chen, and Wei Ni

CSIRO Data61, Eveleigh, NSW 2015, Australia.

Abstract

With its disaggregated and virtualized design, the Open Radio Access Network (ORAN) architecture offers significant flexibility for 5G and emerging 6G networks, but introduces critical security challenges in multi-tenant environments. When hosted on shared hardware, virtualized components such as the O-DU, O-CU, and RIC are vulnerable to side-channel attacks, where malicious actors can exploit timing and resource-based patterns to infer sensitive information across tenant boundaries. This paper proposes a model that enhances isolation and resource management within ORAN deployments by integrating deep reinforcement learning (DRL) with container orchestration technologies.

We present a workflow of the system that utilizes DRL to dynamically manage resources and enforce strong tenant separation within Kubernetes-managed environments. Our approach mitigates cross-tenant interference and minimizes exposure to side-channel attacks, supporting the integrity and resilience of ORAN's multi-tenant infrastructure.

Proposed Solution

1. **Enhanced Isolation and Resource Management**: The proposed model integrates deep reinforcement learning (DRL) with:
   - Kubernetes container orchestration technologies
   - leveraging features like namespaces, network policies,
   - resource quotas
2. **Dynamic and Adaptive Resource Allocation**: DRL enables intelligent resource management, allowing the system to proactively detect and mitigate security threats while enforcing strong tenant separation in Kubernetes-managed environments.

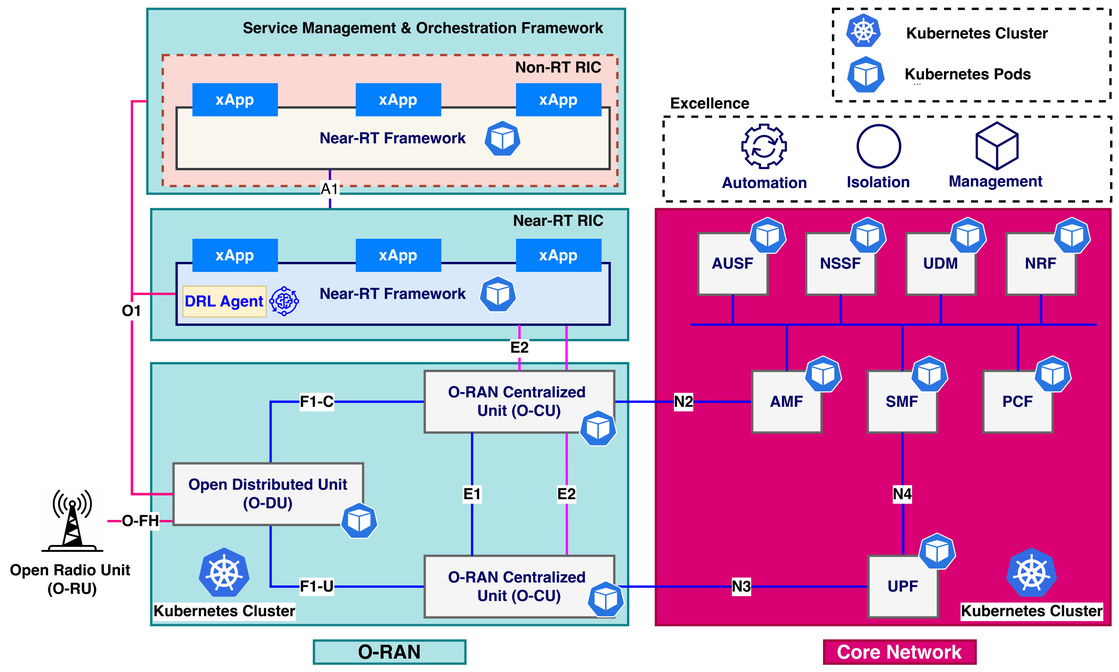

Figure 1: Leveraging Isolation and Resource Management for securing ORAN components.

Integration of Deep Reinforcement Learning

The workflow of the integrated system is described below:

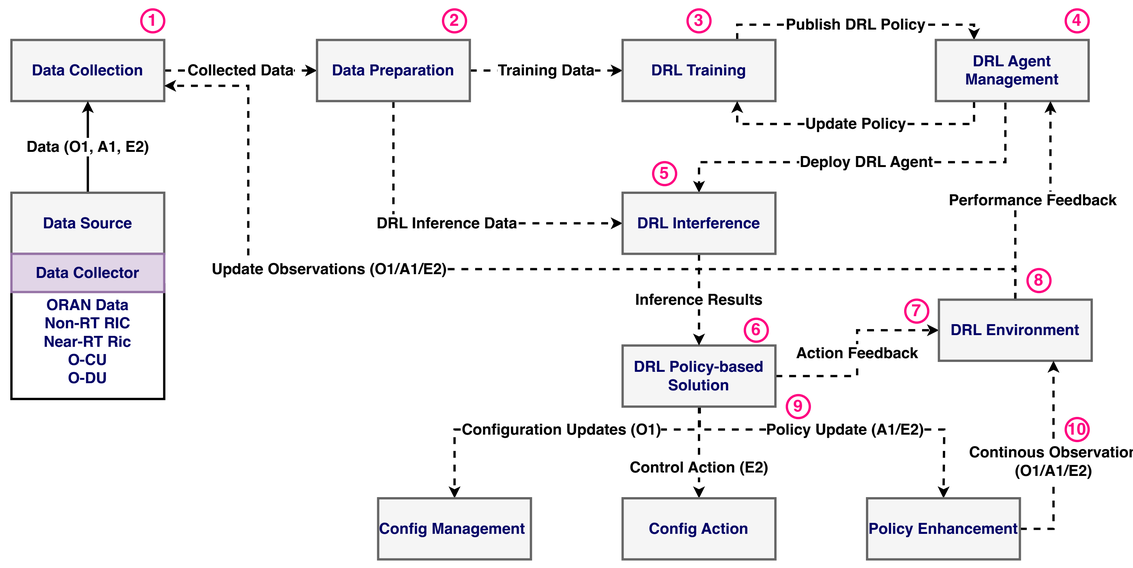

Figure 2: End-to-End Deep Reinforcement Learning workflow in ORAN

1. **Data Collection and Preprocessing**: Gather real-time metrics from ORAN components via O1, A1, E2 interfaces and preprocess for training and inference.
2. **DRL Training and Deployment**: Train DRL agents (e.g., PPO) for resource management and deploy policies via Kubernetes APIs for real-time decision-making.
3. **Action Execution**: Apply insights to manage resources, optimize traffic, and enforce tenant policies dynamically on ORAN components.
4. **Monitoring and Refinement**: Continuously monitor performance, collect feedback, and refine DRL policies to adapt to changing network conditions and threats.

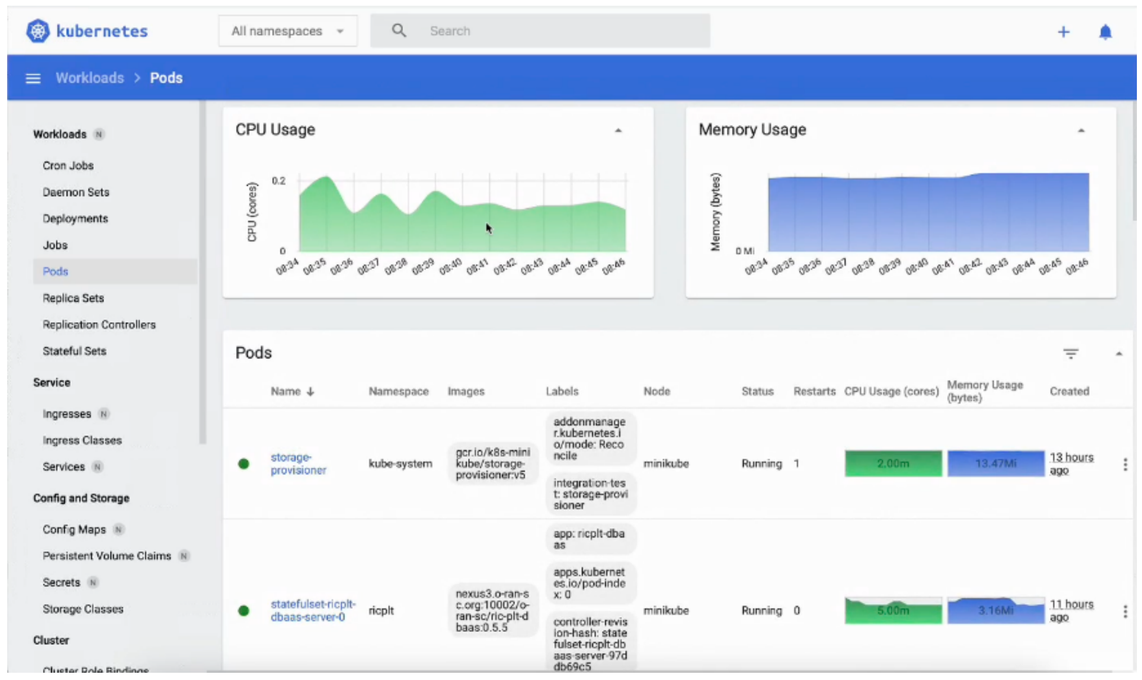

Figure 3: Kubernetes Dashboard

Conclusion and Future Plan

1. **Prototype Implementation**: Develop and test a prototype of the proposed framework in real-world ORAN scenarios.
2. **Performance Evaluation**: Assess the framework's effectiveness in enhancing security, improving resource utilization, and achieving scalability.
3. **Framework Validation**: Validate the feasibility of the approach and refine it based on real-world observations.