# Robotics-as-a-Service Benchmarks over ROS2

## GOALS

Robotics-as-a-service (RaaS) platforms can significantly enhance the value of Autonomous Mobile Robots (AMRs).

We aimed to benchmark our RaaS architecture's performance and security using Fed4Fire testbeds with our AMR prototype.

The Fed4Fire Industrial IoT lab, VirtualWall and wi-lab.t testbeds were used for the experiment.

## CHALLENGES

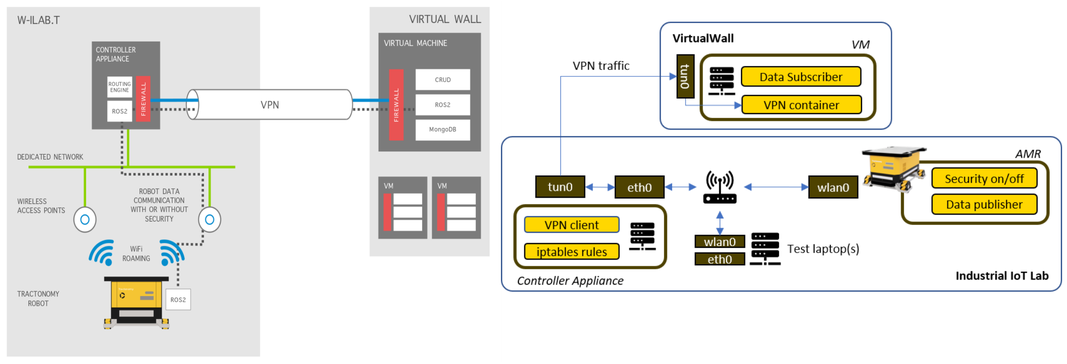
ROS2-security is in active development, but information is limited for deployment in business or industrial networks.

Limited information available on network performance of ROS2 with and without security inside state-of-art VPNs.

Testing in public cloud environments is expensive and many unknowns, e.g. throttling, sniffing, load-balancing

Benchmarks like bandwidth and losses needed before deploying to a production environment, i.e. public clouds.

## DEMO SETUP




A VM was setup in the IMEC/UGent VirtualWall datacenter.

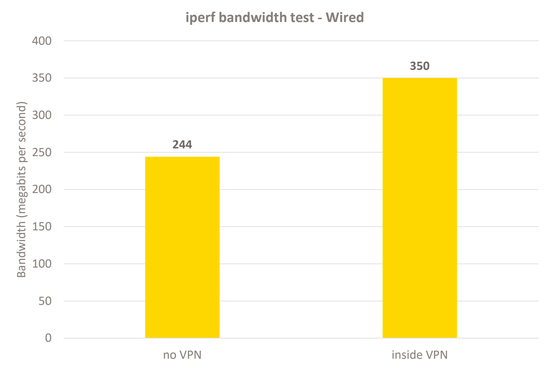
Controller Appliance manages VPN tunnel to VirtualWall VM.

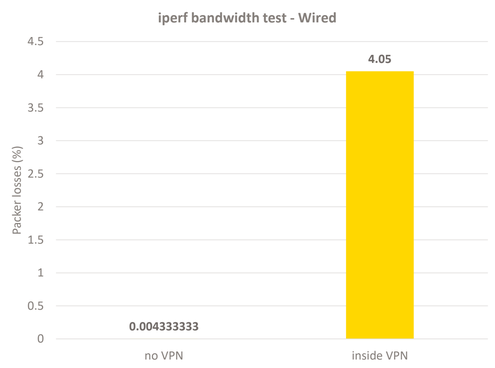
Use case was AMR sending data to cloud; driving in IIoT lab.

All AMR traffic forwarded through VPN tunnel.

wi-lab.t wireless nodes were used for wireless networking.

## RESULTS

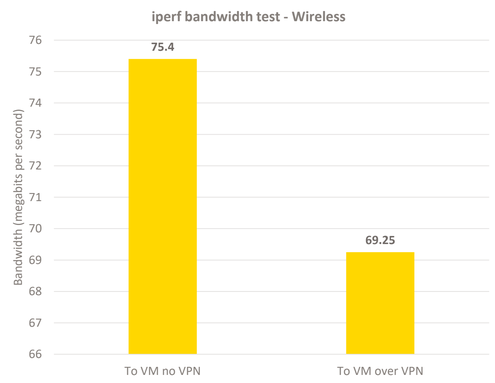



Achieved goal of ROS2 in a representative state-of-art VPN network with security.

No observable overheard with ROS2 security enabled.

No critical issues running ROS2 inside VPN.




## MORE RESULTS

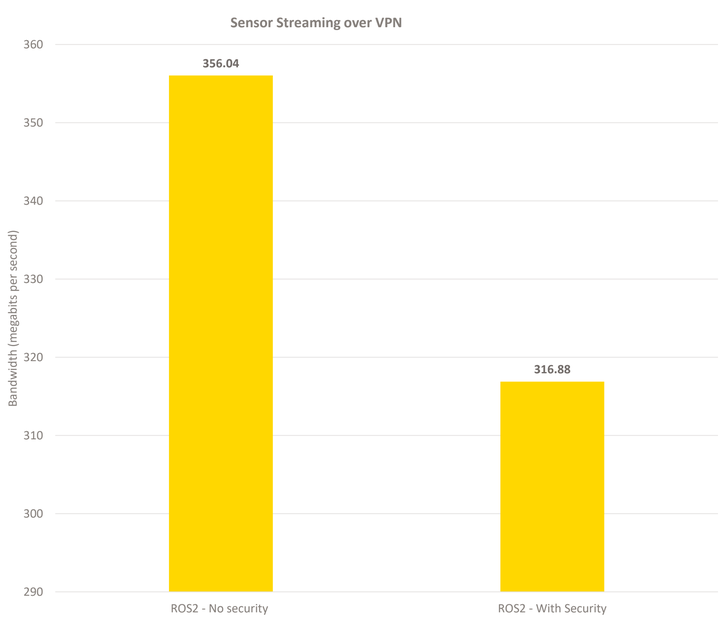
Contradiction 1 - VPN bandwidth use contradicted in wired/wireless tests

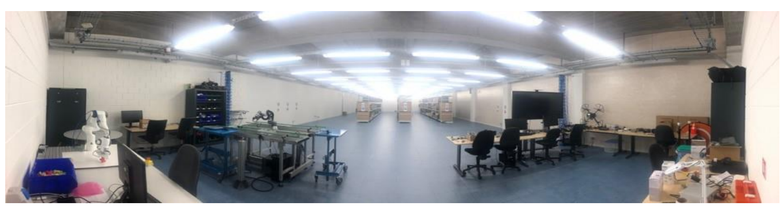
Contradiction 2 – Traffic inside encrypted tunnel used less traffic than

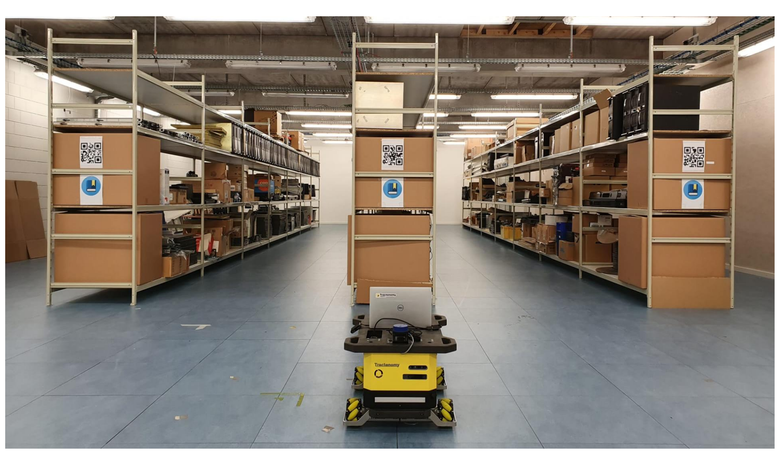
Losses increased in VPN with bandwidth higher than 20mbps.

Roaming test failed due to issue with wireless radios on robot laptop.

Penetration and fuzzing tests showed ROS2 security resilient to intrusion.





## CONCLUSIONS

Despite some testing issues, we achieved a robust and scalable ROS2/VPN framework for our RaaS.

Resolved several undocumented issues with ROS2 topic multi-cast inside a VPN.

Captured details on ROS2 security that need to be resolved before commercialization.

Discovered limitations with our wireless hardware and defined actions to resolve them.

Using default ROS2 Quality-of-Service (QoS) settings may have influenced the identified issues.

## POST MORTEM

ROS2/DDS and security features offer a powerful framework for RaaS implementations.

Fed4Fire testbeds offer easy-to-setup and unrestricted test environments for limit testing.

In retrospection, could have benefited from an end-to-end dry-run to avoid contradictory results.

Next experiments must account for ROS QoS settings, but QoS management requires needs tooling and automation.

Raas-o-ROS2 (phase 2 awarded) will develop a tooling to automate tuning of ROS2 QoS and security at scale.