

1. General Description

This document contains the CPG for the work "On the (In)Security of Secure ROS2", submitted to ACMCCS 2022. It contains our analysis, visualization dot files as well as the original output from Joern [1].

2. CPG Generation and Outputs

We follow the Joern default setting and conduct our analysis in a virtual machine with Ubuntu 20.04 system. Specifically, we analyze 4 ROS2 libraries:

- (1) RCL general library: <https://github.com/ros2/rcl>
- (2) RCL API for C++ (rclcpp): <https://github.com/ros2/rclcpp>
- (3) RMW general library: <https://github.com/ros2/rmw>
- (4) RMW support for FastRTPS: https://github.com/ros2/rmw_fastrtps

Without any abstraction, we include the original CPG generated in the **source_CPG** folder. Each .dot file stores the graph information for one specific method, and the file name is the method index generated by Joern by default.

3. Graph Reduction and CPG Analysis

It can be noticed that the CPG result is huge: there are more than 50k source CPG files, most of which are not related to message communication. We follow the methodology as described in Section 4.1 to identify key functions.

Specifically, we study the ROS2 documentation [2-4] and source code to summarize the error handling classes and the error handling functions coding standard. We summarize this in *error_handling_labels.txt*.

After analysis, we obtain 23 key functions for the further manual communication flow analysis. We also manually examine the call relations between these functions to ensure the correctness of the generated CPG. Below we present the functions. To keep consistency, we rename the .dot files of these key functions in the following format:

library_name::original_function_name

For instance, the **rcl_node_init** function from the **rcl** library is named as **rcl::rcl_node_init**, while the same name function from the **rclcpp** library is named as **rclcpp::rcl_node_init**. Below we attach the name of the 23 identified key functions, as well as their original index number generated by Joern. Due to the naming convention of Windows system, ":" cannot appear in file directory. By default, they are replaced by "_" if you view the files on Windows systems.

rclcpp library:

rclcpp::rcl_node_init (41351-cpg.dot)

rcl library:

rcl::rcl_node_init (490-cpg.dot)
rcl::rcl_publisher_init (203-cpg.dot)
rcl::rcl_subscription_init (596-cpg.dot)
rcl::rcl_publish (208-cpg.dot)
rcl::rcl_take_sequence (609-cpg.dot)
rcl::rmw_create_publisher (720-cpg.dot)
rcl::rmw_create_subscription (927-cpg.dot)
rcl::rmw_take_sequence (940-cpg.dot)
rcl::rmw_publish (726-cpg.dot)

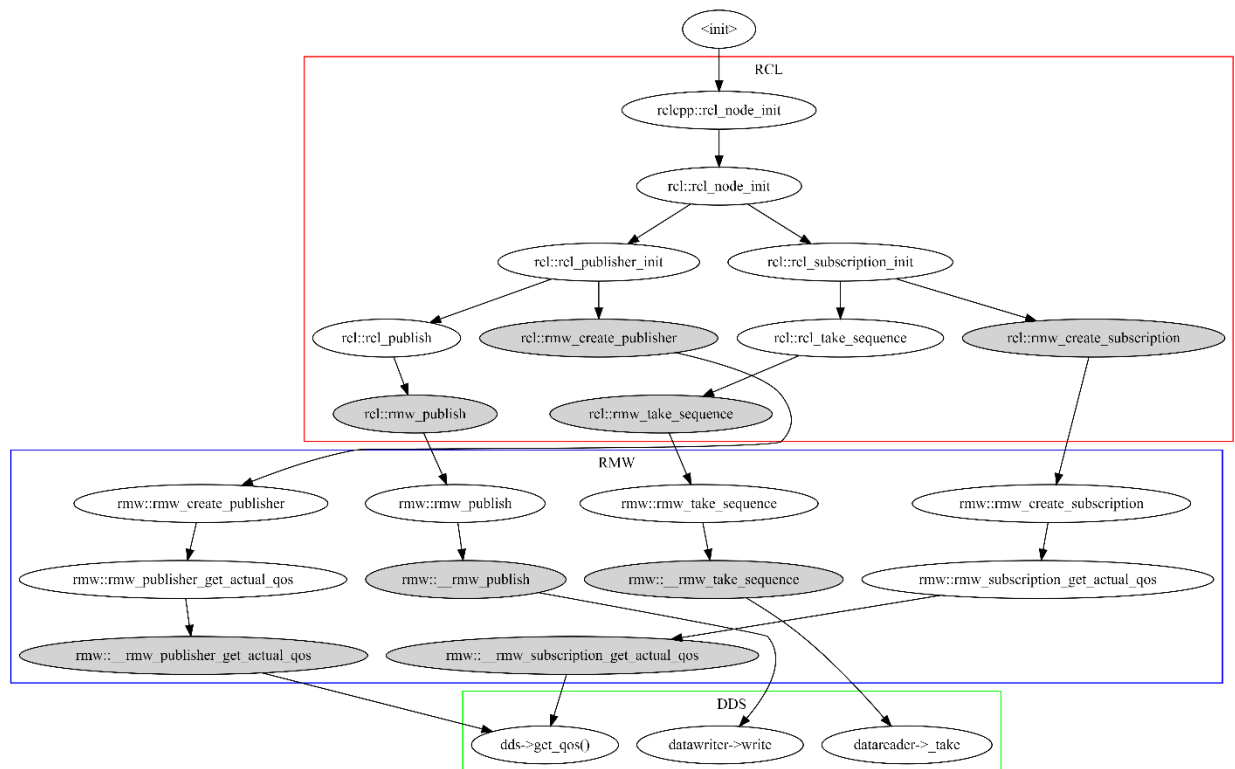
rmw library

rmw::create_publisher (9675-cpg.dot)
rmw::rmw_publish (4-cpg.dot)
rmw::rmw_take_sequence (1835-cpg.dot)
rmw::rmw_create_publisher (9675.dot)
rmw::rmw_create_subscription (9866-cpg.dot)
rmw::rmw_publisher_get_actual_qos (9679-cpg.dot)
rmw::__rmw_publish (11371-cpg.dot)
rmw::__rmw_take_sequence (12030.dot)
rmw::__rmw_publisher_get_actual_qos (13273-cpg.dot)
rmw::__rmw_subscription_get_actual_qos (rmw_fastrtps_shared_cpp/9528-cpg.dot)
dds-get_qos (rmw_fastrtps_shared_cpp/21422-cpg.dot)
data_writer-write (rmw_fastrtps_shared_cpp/21827-cpg.dot)
data_reader-take (rmw_fastrtps_shared_cpp/21377-cpg.dot)

4. Result Interpretation

We use the following dot graph to visualize the ROS2 code structure, which is an abstraction of code dependency graph (CDG) from CPG. It reflects the function call relations between the key functions. We divide the functions into RCL level, RMW level and DDS level. The white color

nodes are the functions, and the grey color nodes are the API entry functions that interacts with the functions on another layer.



- [1] Joern: <https://joern.io/>
- [2] RCL Documentation (auto-generated by Doxygen): <https://docs.ros2.org/beta3/api/rclcpp>
- [3] RMW Documentation (auto-generated by Doxygen): <https://docs.ros2.org/foxy/api/rmw/>
- [4] RMW_FastRTPS Documentation (auto-generated by Doxygen): https://docs.ros2.org/galactic/api/rmw_fastrtps_dynamic_cpp