The Node Definition Language

NoDL: an API for ROS 2 node description

CANONICAL ubuntu

1

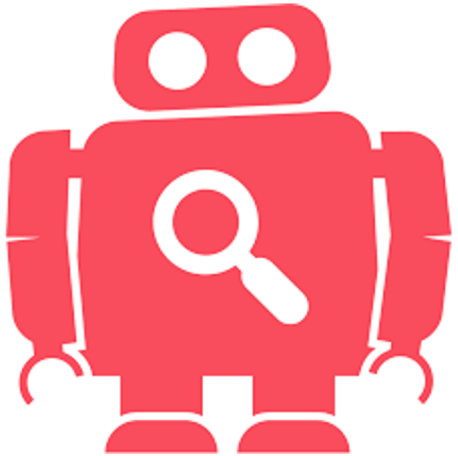
What is NoDL?

What is NoDL?

An API for ROS 2 node description

What problem does it solve?

- Provides a standardized way of declaring the node interfaces of ROS 2 packages
- Allows developers to programmatically export ROS 2 interfaces (action/message/parameter/service)

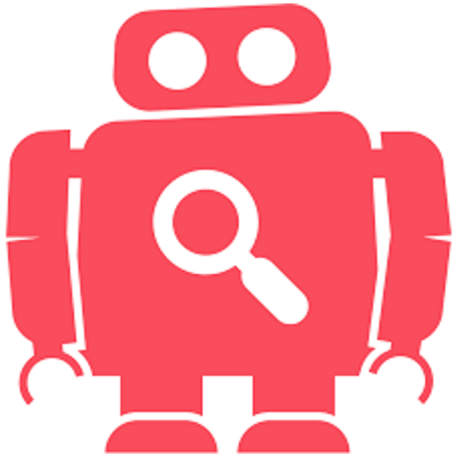


Why NoDL?

Security automation, and more

Why do we want NoDL?

- Enables automatic security policy generation with sros2
- And countless other potential tools, ie: a graphical interface to organize ROS 2 nodes, automating documentation generation etc.



NoDL in code

- ROS 2 design document draft: “ROS 2 Node Definition Language”

NoDL repo:

- A Python library - `nodl_python`
- A ROS 2 CLI tool - `ros2nodl`

Related projects:

- A ROS 2 launch integration - `ros2launch_security`
- A security policy generator - `nodl_to_policy`

NoDL in code

ros2launch_security

- ROS 2 launch integration - secure launch
  - Adds two new optional arguments to ros2 launch (extensions)
  - `--secure [keystore_path]`: Enable security (ie, launch with encryption)
  - `--no-create-keystore`: Point to an existing keystore.

nodl_to_policy

- Security policy generator
  - Tooling to generate a ROS 2 Access Control Policy based on NoDL node descriptions and sros2

2

Usage & CLI

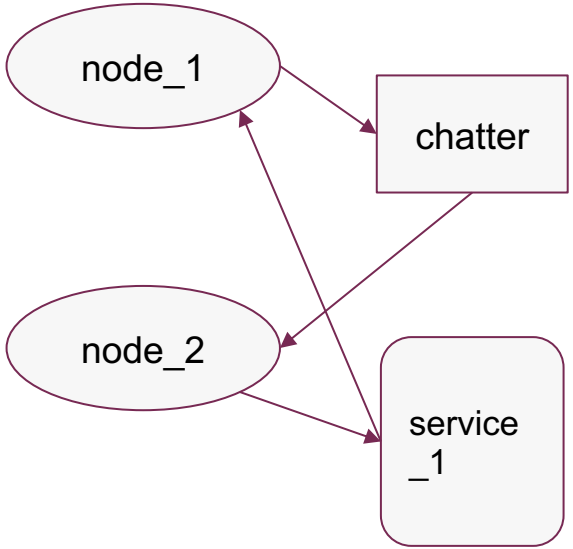
Node declaration

NoDL is embodied by an input XML file with a package's node declarations.

`package.nodl.xml`:

```
<interface version="1">
 <node name="node_1" executable="talker">
   <parameter name="verbose" type="bool" />
   <topic name="chatter" type="std_msgs/msg/String" role="publisher" />
   <service name="service_1" type="std_srvs/srv/Empty" role="client" />
 </node>

 <node name="node_2" executable="service">
   <topic name="chatter" type="std_msgs/msg/String" role="subscription" />
   <service name="service_1" type="std_srvs/srv/Empty" role="server" />
 </node>
</interface>
```



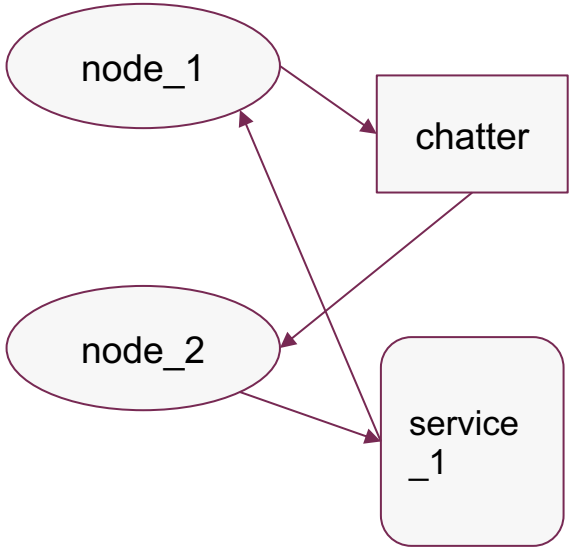
Node declaration

NoDL is embodied by an input XML file with a package's node declarations.

`package.nodl.xml`:

```
<interface version="1">
  <node name="node_1" executable="talker">
    <parameter name="verbose" type="bool" />
    <topic name="chatter" type="std_msgs/msg/String" role="publisher" />
    <service name="service_1" type="std_srvs/srv/Empty" role="client" />
  </node>

  <node name="node_2" executable="service">
    <topic name="chatter" type="std_msgs/msg/String" role="subscription" />
    <service name="service_1" type="std_srvs/srv/Empty" role="server" />
  </node>
</interface>
```



CLI: Available verbs for ros2 nodl

```
$ ros2 nodl show
```

```
$ ros2 nodl show cpp_pubsub talker
{'name': 'talker',
'executable': 'talker',
'actions': {},
'parameters': {},
'services': {},
'topics':
    {'topic':
    {'name': 'topic',
    'type': 'std_msgs/msg/String',
    'role': <PubSubRole.PUBLISHER: 'publisher'>}}}
```

CLI: Available verbs for ros2 nodl

`$ ros2 nodl **validate**`

```
$ ros2 nodl validate cpp_pubsub.nodl.xml
Validating cpp_pubsub.nodl.xml...
  Success
All files validated
```

ros2launch_security

```
$ ros2 launch cpp_pubsub_launch.xml.launch --secure
```

```
$ ros2 launch cpp_pubsub cpp_pubsub_launch.py --secure
[INFO] [launch]: All log files can be found below /home/flor/.ros/log/2022-03-07-04-02-02-039923-Flor-Ubuntu-714203
[INFO] [launch]: Default logging verbosity is set to INFO
[INFO] [talker-1]: process started with pid [714205]
[INFO] [listener-2]: process started with pid [714207]
[listener-2] [INFO] [1646636522.507395154] [rcl]: Found security directory: /tmp/tmplyvbnnbb/enclaves/listener
[talker-1] [INFO] [1646636522.511481663] [rcl]: Found security directory: /tmp/tmplyvbnnbb/enclaves/talker
[talker-1] [INFO] [1646636523.031227271] [talker]: Publishing: 'Hello, world! 0'
[listener-2] [INFO] [1646636523.031896352] [listener]: I heard: 'Hello, world! 0'
[talker-1] [INFO] [1646636523.531239256] [talker]: Publishing: 'Hello, world! 1'
[listener-2] [INFO] [1646636523.531705801] [listener]: I heard: 'Hello, world! 1'
...
```

Example


ROS 2 Secure Launch
Security in ROS has never been so easy
Ted Kern
CANONICAL
ubuntu

nodl_to_policy

```
$ ros2 nodl_to_policy convert cpp_pubsub.nodl.xml
```

package.nodl.xml

```
<interface version="1">

  <node name="talker" executable="talker">
    <topic name="topic" type="std_msgs/msg/String"
role="publisher"/>
  </node>

  <node name="listener" executable="listener">
    <topic name="topic" type="std_msgs/msg/String"
role="subscription"/>
  </node>

</interface>
```

```
<policy version="0.2.0">
  <enclaves>
    <enclave path="/listener">
      <profiles>
        <profile node="listener" ns="/">
          <topics subscribe="ALLOW">
            <topic>clock</topic>
            <topic>parameter_events</topic>
            <topic>topic</topic>
          </topics>
          <topics publish="ALLOW">
            <topic>parameter_events</topic>
            <topic>rosout</topic>
          </topics>
        </profile>
      </profiles>
    </enclave>
    <enclave path="/talker">
      <profiles>
        <profile node="talker" ns="/">
          <topics subscribe="ALLOW">
            <topic>clock</topic>
            <topic>parameter_events</topic>
          </topics>
          <topics publish="ALLOW">
            <topic>parameter_events</topic>
            <topic>rosout</topic>
            <topic>topic</topic>
          </topics>
          ...
```

#3

What’s next for NoDL?

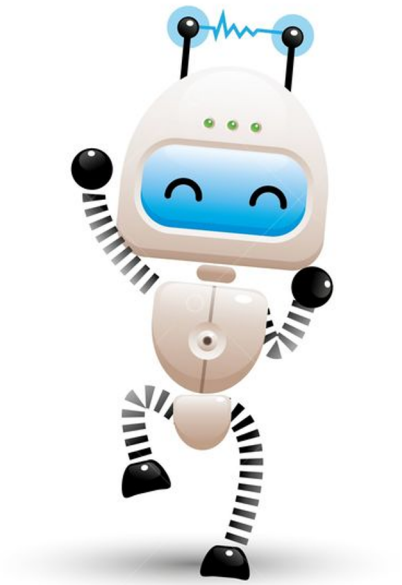
Our goals

- Promote security to the ROS community
- Leverage NoDL to ease the use of ROS 2 security;

  ie, to automate SROS2 policy generation.

What’s next?

- Improve documentation
- Merge design document
- Blog/workshop to promote its use
- `ament_nodl` linter
- Implement NoDL for ROS 2 examples and REP 2005 packages.
- Enable `ros2launch_security` for the reference implementation



#4

References

Relevant links

- **Design document**

https://github.com/ros2/design/pull/266

- **Repo and libraries**

https://github.com/ubuntu-robotics/nodl

https://github.com/ubuntu-robotics/nodl/pull/43

https://index.ros.org/p/ros2nodl/

https://index.ros.org/p/nodl_python/

- **ROS2 Launch Security**

https://github.com/osrf/ros2launch_security

- **NoDL to Policy**

https://github.com/aprotyas/nodl_to_policy

Thank you. Questions?