

# rslidar\_sdk教程——Nanny版

## rslidar\_sdk使用教程

### 一、驱动简介

#### 1. rslidar\_sdk适配环境：

- Ubuntu 16.04 - ROS Kinetic desktop
- Ubuntu 18.04 - ROS Melodic desktop
- Ubuntu 20.04 - ROS Noetic desktop

#### 2. 支持雷达型号

- RS-LiDAR-16
- RS-LiDAR-32
- RS-Bpearl
- RS-Helios
- RS-Helios-16P
- RS-Ruby-Plus-80
- RS-Ruby-128
- RS-Ruby-80
- RS-Ruby-48
- RS-Ruby-Plus-128
- RS-Ruby-Plus-48
- RS-LiDAR-M1
- RS-LiDAR-M2
- RS-LiDAR-E1

#### 3. 支持点的类型

- XYZI - x, y, z, intensity(反射率)
- XYZIRT - x, y, z, intensity(反射率), ring(激光通道), timestamp(时间戳)

### 二、驱动剖析

#### 1. config.yaml参数介绍请查看：[rslidar\\_sdk/doc/intro](#)

02参数介绍	<a href="#">02_parameter_intro.md</a> <a href="#">02_parameter_intro_CN.md</a>
03隐藏参数介绍	<a href="#">03_hiding_parameters_intro.md</a> <a href="#">03_hiding_parameters_intro_CN.md</a>

#### 2. 使用说明请查看：[rslidar\\_sdk/doc/howto](#)

04如何与rslidar_sdk_node v1.3.x共存	<a href="#">04_how_to_work_along_with_rslidar_sdk_node_v1.3.x_CN.md</a>
05如何改变点类型的定义	<a href="#">05_how_to_change_point_type.md</a> <a href="#">05_how_to_change_point_type_CN.md</a>
06如何连接在线雷达	<a href="#">06_how_to_decode_online_lidar.md</a> <a href="#">06_how_to_decode_online_lidar_CN.md</a>
07在线雷达 - 高级主题	<a href="#">07_online_lidar_advanced_topics.md</a> <a href="#">07_online_lidar_advanced_topics_CN.md</a>

08如何解码PCAP文件	<a href="#">08_how_to_decode_pcap_file.md</a> <a href="#">08_how_to_decode_pcap_file_CN.md</a>
09PCAP文件 - 高级主题	<a href="#">09_pcap_file_advanced_topics_CN.md</a> <a href="#">09_pcap_file_advanced_topics.md</a>
10如何使用坐标变换功能	<a href="#">10_how_to_use_coordinate_transformation.md</a> <a href="#">10_how_to_use_coordinate_transformation_CN.md</a>
11如何录制与回放 Packet rosbag	<a href="#">11_how_to_record_replay_packet_rosbag_CN.md</a> <a href="#">11_how_to_record_replay_packet_rosbag.md</a>

3. 驱动源码解析请查看：[rslidar\\_sdk/src\\_intro](#)

rslidar_sdk v1.5.* 源代码解析	<a href="#">rslidar_sdk_intro_CN.md</a>
--------------------------	---

### 三、使用步骤方法

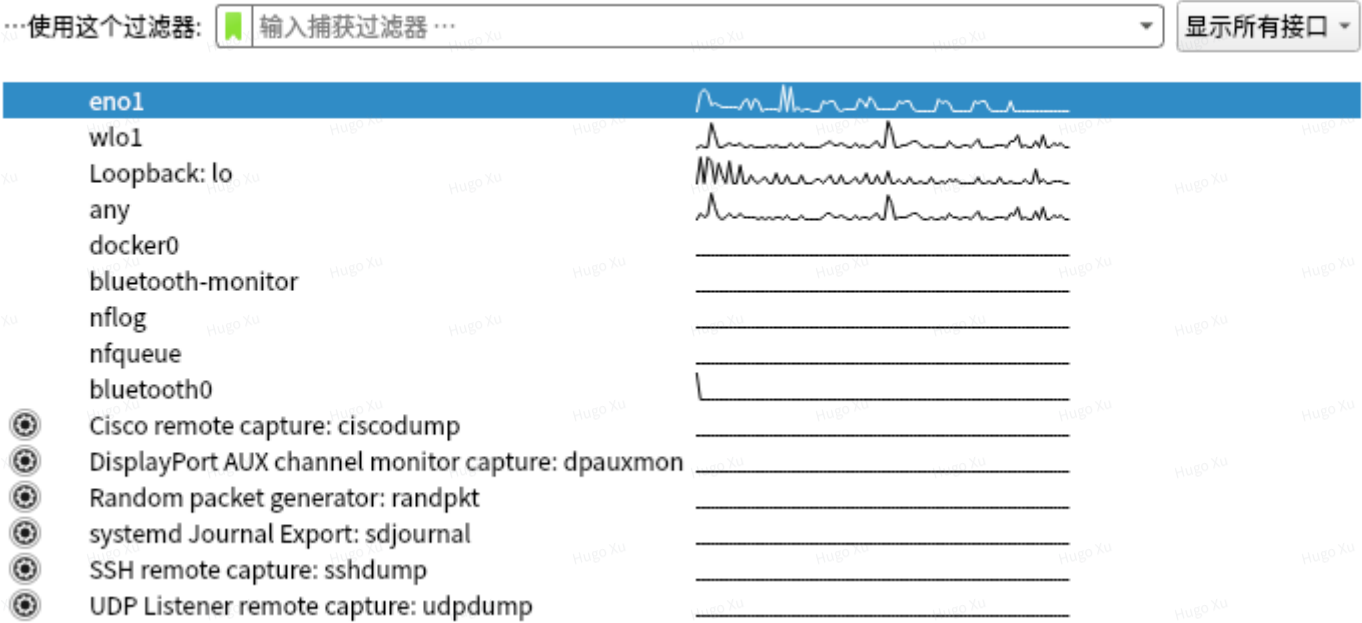
#### 步骤一：主机接收雷达数据

1. 下载安装[wireshark](#)，查看网口报文。

```
1 user@user:~$ sudo apt-get install wireshark
2 user@user:~$ sudo wireshark
```

2. 根据实际安装网口左键双击选择对应网卡。

#### 捕获



3. 进入捕抓界面如无雷达UDP数据（正常接收到雷达UDP数据跳转到“[步骤三/4](#)”），请查看雷达ARP报文，配置主机网卡静态IP为雷达数据目的IP，如下：

\*eno1 1.接收雷达数据网口

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)

应用显示过滤器... <Ctrl-/> + helios-msop | helios-difop | bpearl-msop | bpearl-difop | ruby-msop | ruby-difop |

3.雷达mac地址,一般为40:\*\*\*或suteng开头

2.ARP报文

4.雷达数据目的IP

5.雷达IP

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 192.168.1.200? Tell 192.168.1.200
2	0.401034	SutengIn_0c:6c:04	Broadcast	ARP	60	Who has 192.168.1.102? Tell 192.168.1.200
3	0.768020	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 183.61.243.117? Tell 196.168.20.1
4	0.768031	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 183.61.168.97? Tell 196.168.20.1
5	0.863975	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 183.2.160.211? Tell 196.168.20.1
6	0.868026	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 183.2.160.208? Tell 196.168.20.1
7	0.868098	f8:0d:ac:25:d8:69	Broadcast	ARP	42	Who has 183.2.199.224? Tell 196.168.20.1

4. 修改电脑静态地址为雷达目的地址，并查看修改参数是否生效，修改方式例：

```
1 user@user:~$ sudo ifconfig 192.168.1.102
2 user@user:~$ ifconfig
```

```
eidv@eidv:~$ sudo ifconfig eno1 192.168.1.102 修改命令
eidv@eidv:~$ ifconfig 查看命令
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:a5:e8:21:6a txqueuelen 0 (以太网)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eno1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.102 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::94d2:3ad8:8a19:67b5 prefixlen 64 scopeid 0x20<link>
    ether f8:0d:ac:25:d8:69 txqueuelen 1000 (以太网)
    RX packets 241202 bytes 146264992 (146.2 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4904 bytes 327796 (327.7 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

5. wireshark查看雷达UDP数据MSOP/DIFOP端口号，如下图雷达MSOP端口号为6699，DIFOP端口号为7788.

```
1 data.data[0:1]==55 #查看雷达MSOP数据端口号
2 data.data[0:1]==a5 #查看雷达DIFOP数据端口号
```

data.data[0:1]==a5

No.	Time	Source	Destination	Protocol	Length	Info
26127	160.357020	192.168.1.200	192.168.1.102		1290	6699 → 6699 Len=1248
26128	160.358105	192.168.1.200	192.168.1.102		1290	6699 → 6699 Len=1248
26129	160.359496	192.168.1.200	192.168.1.102		1290	7788 → 7788 Len=1248

步骤二：下载rslidar\_sdk最新驱动工程包

- 1. 工程包链接：[https://github.com/RoboSense-LiDAR/rslidar\\_sdk/tree/release](https://github.com/RoboSense-LiDAR/rslidar_sdk/tree/release)
- 2. 建议使用命令下载，如下：

```
1 user@user:~$ git clone https://github.com/RoboSense-LiDAR/rslidar_sdk.git
2 user@user:~$ cd rslidar_sdk
3 user@user:~/rslidar_sdk$ git submodule init
4 user@user:~/rslidar_sdk$ git submodule update
```

步骤三：编译与启动

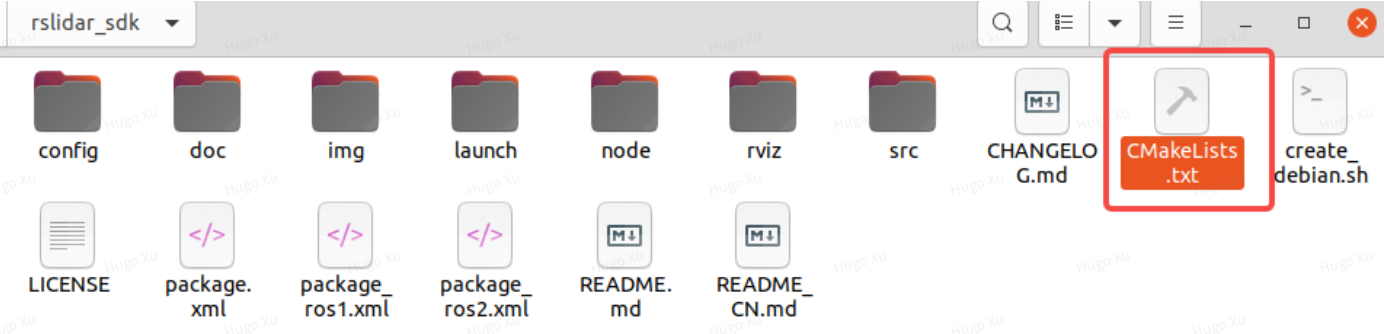
1. 编译前安装环境依赖

```
1 user@user:~$ sudo apt-get update
2 user@user:~$ sudo apt-get install -y libyaml-cpp-dev
3 user@user:~$ sudo apt-get install -y libpcap-dev
```

2. rslidar\_sdk支持三种编译方式，可参考 [rslidar\\_sdk/README\\_CN.md](#) 文件，详情如下：

3.1.1 直接编译

1. 打开工程内的\* [CMakeLists.txt](#) \*文件



2. 将文件顶部的变量 [COMPILE\\_METHOD](#) 设置为 [ORIGINAL](#)。

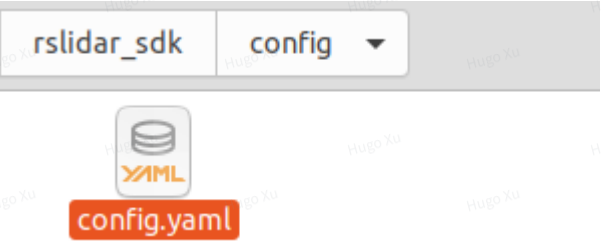
```
1 #=====
2 # Compile setup (ORIGINAL,CATKIN,COLCON)
3 #=====
4 set(COMPILE_METHOD ORIGINAL)
```

3. 编译命令如下：

```
1 user@user:~$ cd rslidar_sdk
2 user@user:~/rslidar_sdk$ mkdir build && cd build
3 user@user:~/rslidar_sdk/build$ cmake .. && make -j4
```

3.1.2 运行节点

1. 打开 [rslidar\\_sdk/config.yaml](#) 文件配置解析参数，yaml参数详解请查看 [rslidar\\_sdk/doc/intro](#) 文件夹。



2. 启动命令如下：

Terminal 1

```
1 user@user:~$ roscore
```

Terminal 2

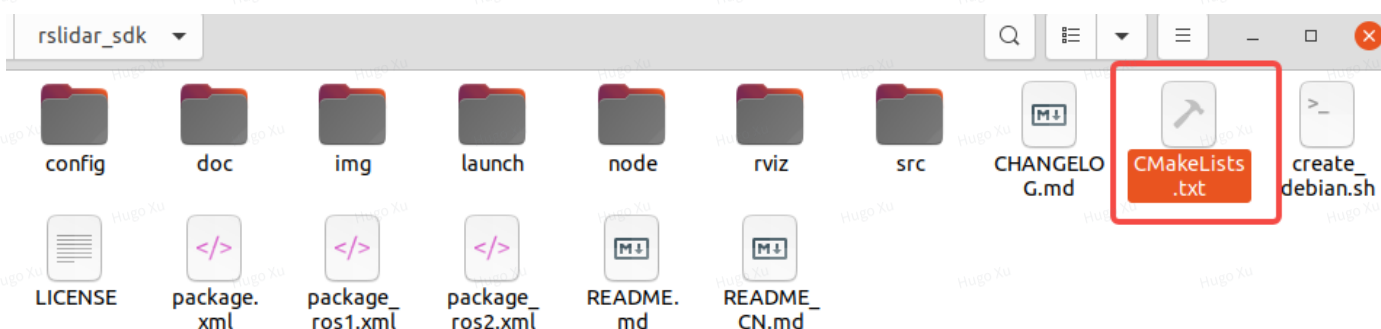
```
1 user@user:~$ cd rslidar_sdk/build
2 user@user:~/rslidar_sdk/build$ ./rslidar_sdk_node
```

### Terminal 3

```
1 user@user:~$ rviz -d rslidar_sdk/rviz/rviz.rviz
```

#### 3.2.1 依赖于ROS-catkin编译

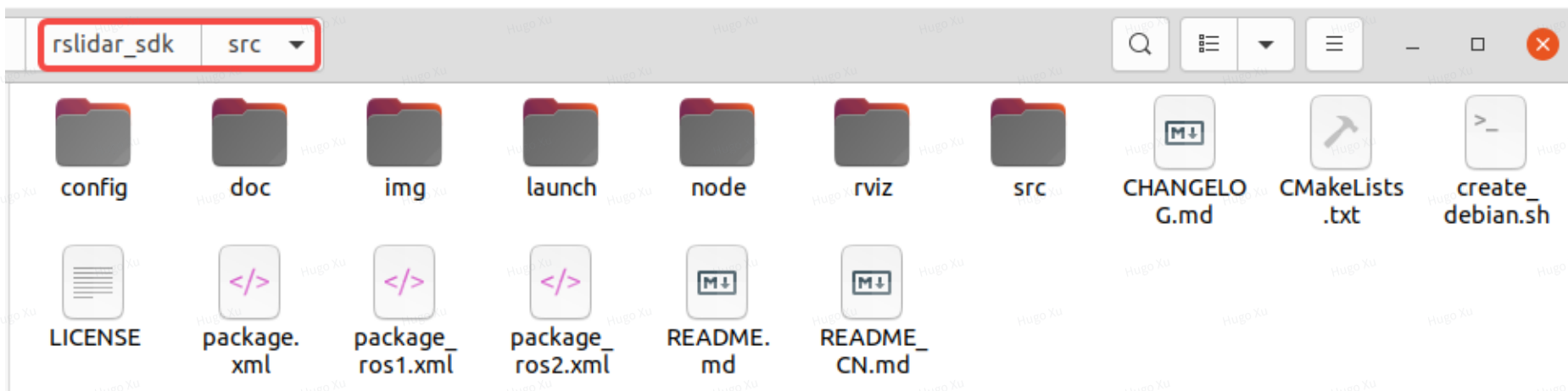
1. 打开工程内的\* **CMakeLists.txt** \*文件。



2. 将文件顶部的变量 **COMPILE\_METHOD** 设置为 **CATKIN**。

```
1 #=====
2 # Compile setup (ORIGINAL,CATKIN,COLCON)
3 #=====
4 set(COMPILMETHOD CATKIN)
```

3. 在rslidar\_sdk目录下新建文件夹，将rslidar\_sdk目录下的文件全部剪切进新建文件夹里面，剪切粘贴完后将新建文件夹重新命名为src。



4. 编译命令如下：

```
1 user@user:~$ cd rslidar_sdk
2 user@user:~/rslidar_sdk$ catkin_make
```

#### 3.2.2 运行节点

1. 打开 **rslidar\_sdk/src/config.yaml** 文件配置解析参数，yaml参数详解请查看 **rslidar\_sdk/src/doc/intro** 文件夹。



2. 启动命令如下：

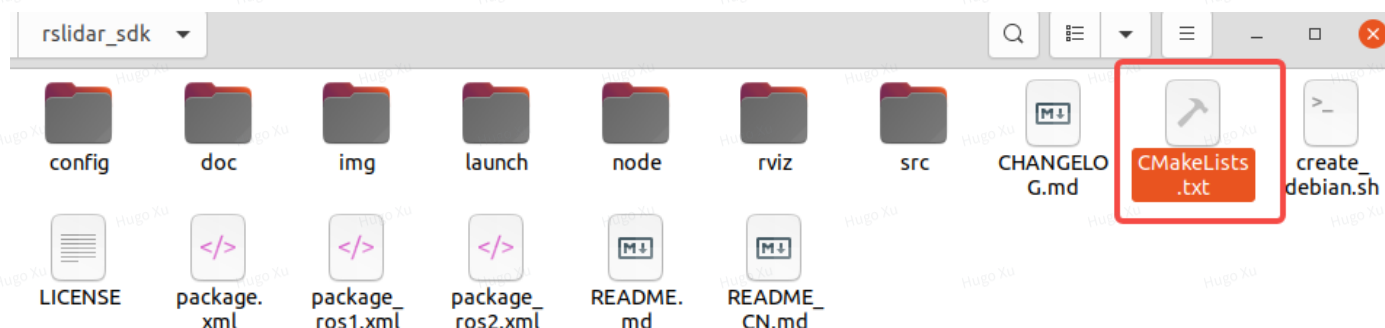
```
1 user@user:~/rslidar_sdk$ source devel/setup.bash
2 user@user:~/rslidar_sdk$ roslaunch rslidar_sdk start.launch
```

### 3.3.1 依赖于ROS2-colcon编译

1. 打开工程删除原有的package.xml，将package\_ros2.xml复制粘贴命名为package.xml。



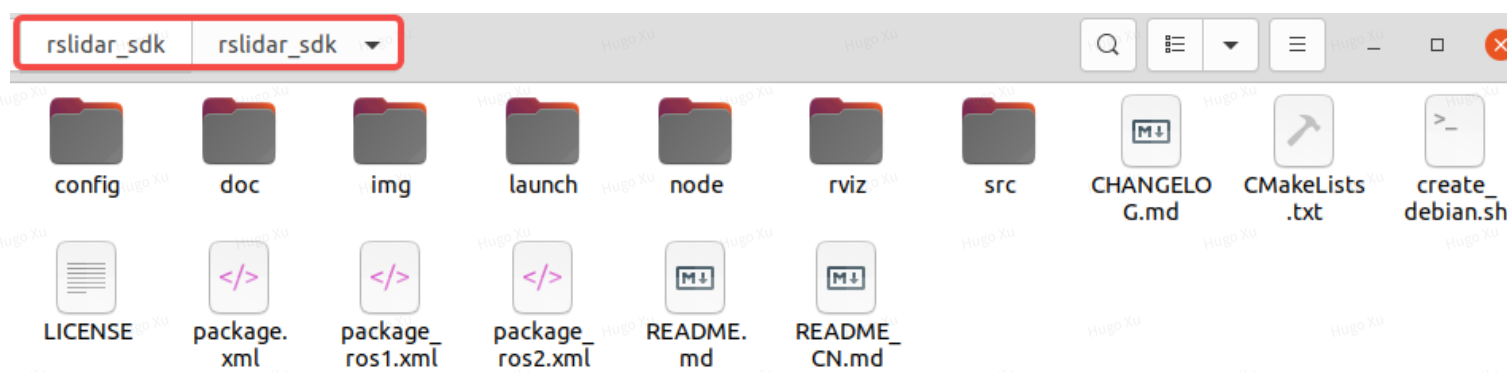
2. 打开工程内的\* **CMakeLists.txt** \*文件



3. 将文件顶部的变量 **COMPILE\_METHOD** 设置为 **COLCON**。

```
1 #=====
2 # Compile setup (ORIGINAL,CATKIN,COLCON)
3 #=====
4 set(COMPILY_METHOD COLCON)
```

4. 在rslidar\_sdk目录下新建文件夹命名为"rslidar\_sdk"，将原始rslidar\_sdk目录下的文件全部剪切进新建的rslidar\_sdk文件夹下，如下图：



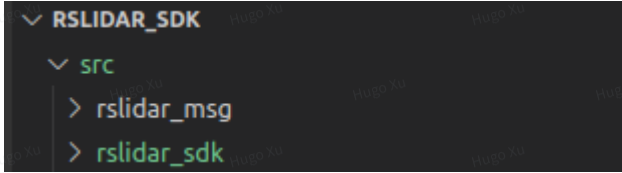
5. 返回上一级目录，目录下新建终端下载工程"rslidar\_msg"。

```
1 user@user:~/rslidar_sdk$ git clone https://github.com/RoboSense-LiDAR/rslidar_msg.git
```





6. 在此目录下新建文件夹src，将rslidar\_sdk与rslidar\_msg剪切粘贴进新建的src文件夹下，结构如图：



7. 编译命令如下：

```
1 user@user:~$ cd rslidar_sdk
2 user@user:~/rslidar_sdk$ ls #终端报文: src
3 user@user:~/rslidar_sdk$ colcon build
```

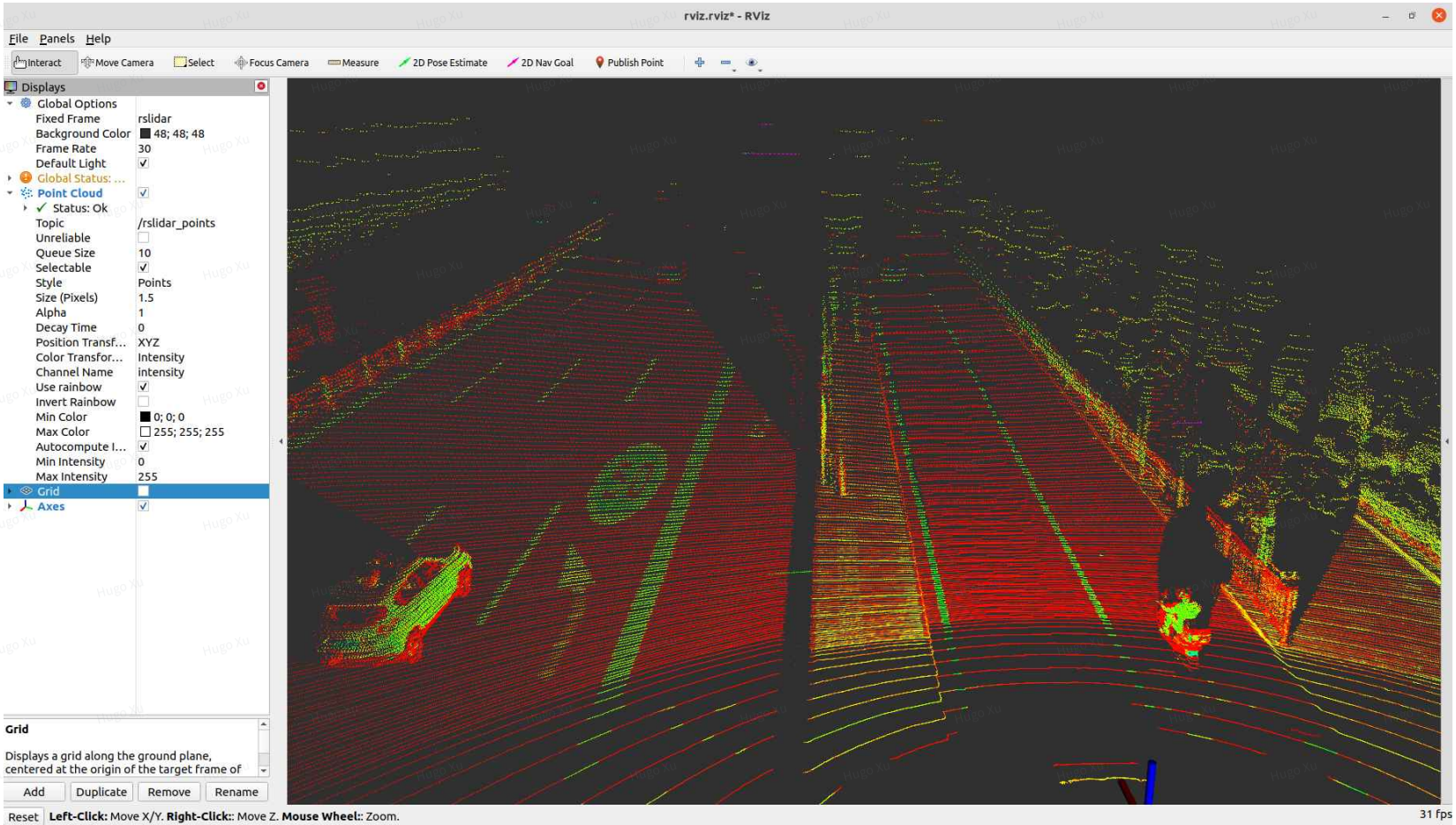
3.2.2 运行节点

1. 打开 `rslidar_sdk/src/rslidar_sdk/config.yaml` 文件配置解析参数，yaml参数详解请查看 `rslidar_sdk/src/rslidar_sdk/doc/intro` 文件夹。



2. 启动命令如下：

```
1 user@user:~$ cd rslidar_sdk
2 user@user:~/rslidar_sdk$ ls #终端报文: build install log src
3 user@user:~/rslidar_sdk$ source install/setup.bash
4 user@user:~/rslidar_sdk$ ros2 launch rslidar_sdk start.py
```



## 四、ubuntu防火墙

如经过以上步骤在线解析雷达数据仍不能显示点云，节点终端持续报**MSOP TIMEOUT**和**DIFOP TIMEOUT**，请将系统防火墙关掉，ubuntu常见防火墙如下：

### 1. UFW

```
1 user@user:~$ sudo ufw status #查看防火墙状态
2 user@user:~$ sudo ufw disable #关闭防火墙
3 user@user:~$ sudo ufw enable #打开防火墙
```

### 2. Firewall

```
1 user@user:~$ sudo firewall-cmd --state #查看防火墙状态
2 user@user:~$ sudo systemctl stop firewalld.service #关闭防火墙
3 user@user:~$ sudo systemctl disable firewalld.service #关闭自启动
4 user@user:~$ sudo systemctl restart firewalld.service #打开防火墙
5 user@user:~$ sudo systemctl enable firewalld.service #打开自启动
```

### 3. Iptables

```
1 user@user:~$ sudo service iptables status #查看防火墙状态
2 user@user:~$ sudo iptables -F #清空所有规则
3 user@user:~$ sudo iptables -X #删除所有用户自定义链
4 user@user:~$ sudo iptables -Z #将所有计数器清零
5 user@user:~$ sudo iptables -P INPUT ACCEPT #开放所有端口
6 user@user:~$ sudo iptables -P INPUT OUTPUT ACCEPT #开放所有端口
7 user@user:~$ sudo iptables -P OUTPUT ACCEPT #开放所有端口
8 user@user:~$ sudo iptables -P FORWARD ACCEPT
9 user@user:~$ sudo modprobe -r iptables
10 user@user:~$ sudo service iptables stop #暂时关闭防火墙
11 user@user:~$ sudo chkconfig iptables off #永久关闭防火墙
```