

Note: This tutorial assumes that you have completed the previous tutorials: Using rxconsole and roslaunch (/ROS/Tutorials/UsingRxconsoleRoslaunch).

💡 Please ask about problems and questions regarding this tutorial on answers.ros.org (<http://answers.ros.org>). Don't forget to include in your question the link to this page, the versions of your OS & ROS, and also add appropriate tags.

How to Roslaunch Nodes in Valgrind or GDB

Description: When debugging roscpp (/roscpp) nodes (/Nodes) that you are launching with roslaunch (/roslaunch), you may wish to launch the node in a debugging program like gdb or valgrind instead. Doing this is very simple.

Keywords: roslaunch, valgrind, gdb, pdb

Tutorial Level: INTERMEDIATE

Next Tutorial: Profiling roslaunch nodes (/roslaunch/Tutorials/Profiling%20roslaunch%20nodes)

The `launch-prefix` attribute of the `<node>` tag that, among other things, makes it easy to debug a ROS node process. Here are some example `launch-prefix`es you might find useful:

- `launch-prefix="xterm -e gdb --args"` : run your node in a gdb in a separate xterm window, manually type `run` to start it
- `launch-prefix="gdb -ex run --args"` : run your node in gdb in the same xterm as your launch without having to type `run` to start it
- `launch-prefix="stterm -g 200x60 -e gdb -ex run --args"` : run your node in gdb in a new stterm window without having to type `run` to start it
- `launch-prefix="valgrind"` : run your node in valgrind
- `launch-prefix="xterm -e"` : run your node in a separate xterm window
- `launch-prefix="nice"` : nice your process to lower its CPU usage

- `launch-prefix="screen -d -m gdb --args"` : useful if the node is being run on another machine; you can then ssh to that machine and do `screen -D -R` to see the gdb session
- `launch-prefix="xterm -e python -m pdb"` : run your python node a separate xterm window in pdb for debugging; manually type `run` to start it
- `launch-prefix="yappi -b -f pstat -o <filename>"`: run your rospy node in a multi-thread profiler such as yappi.
- `launch-prefix="/path/to/run_tmux"`: run your node in a new tmux window; you'll need to create `/path/to/run_tmux` with the contents:

```
#!/bin/sh

tmux new-window "gdb --args $@"
```

1. Obtaining core dumps

To obtain core dumps when processes crash, first set the core file size limit. To check the limits, run:

```
$ ulimit -a
core file size          (blocks, -c) 0          # <-- Prevents core dumps
data seg size           (kbytes, -d) unlimited
scheduling priority     (-e) 20
file size               (blocks, -f) unlimited
pending signals         (-i) 16382
max locked memory       (kbytes, -l) 64
max memory size         (kbytes, -m) unlimited
open files              (-n) 1024
pipe size               (512 bytes, -p) 8
POSIX message queues    (bytes, -q) 819200
real-time priority      (-r) 0
stack size              (kbytes, -s) 8192
cpu time                (seconds, -t) unlimited
max user processes      (-u) unlimited
virtual memory          (kbytes, -v) unlimited
file locks              (-x) unlimited
```

Set the core size to unlimited:

```
ulimit -c unlimited
```

Now crashing processes will attempt to create core files. Currently (2010-07-02) they will fail to create the file because the default roslaunch working directory, `$ROS_HOME`, contains a directory named "core". This directory prevents the core dump from being created.

To allow core dumps to be created, set the core filename to use the process pid by default. Run the following as root:

```
echo 1 > /proc/sys/kernel/core_uses_pid
```

Now core dumps will show up as `$ROS_HOME/core.PID`

2. Error shooting

- If you get error like below one, you should try to launch node in another windows with gdb. [🌐 link \(https://github.com/ros/ros_comm/issues/223\)](https://github.com/ros/ros_comm/issues/223)
- `launch-prefix="xterm -e gdb --args"`

```
[tcsetpgrp failed in terminal_inferior: Inappropriate ioctl for device]
```

Except where otherwise noted, the ROS wiki is licensed under the

Creative Commons Attribution 3.0 (<http://creativecommons.org/licenses/by/3.0/>)

Wiki: [roslaunch/Tutorials/Roslaunch Nodes in Valgrind or GDB](#) (2018-08-01 06:54:13由Kei Okada (/Kei%20Okada)编辑)

Brought to you by:  Open Robotics

(<https://www.openrobotics.org/>)