

# E2 Simulator

- [E2 Simulator](#)
  - [Building docker image and running simulator instance](#)
  - [Building docker image and running simulator instance as helm chart](#)

## E2 Simulator

...

## Description

...

Usage

How to configure E2Sim to initiate the connection towards E2Term(nRT-RIC)

- Get endpoint of Kong service

MyExample: script

```
KONG_SVC=$(kubectl get svc -A | grep "kong" | awk '{print $2}')
NODE_PORT=$(kubectl get --namespace ricplt -o jsonpath="{.spec.ports[0].nodePort}" services $KONG_SVC)
NODE_IP=$(kubectl get nodes -o jsonpath="{.items[0].status.addresses[0].address}")
KONG_ADDRESS=${NODE_IP}:${NODE_PORT}
```

- Get endpoint of E2Term List from E2Mgr

MyExample: script

```
NUM_OF_E2T=$(curl -sX GET "http://${KONG_ADDRESS}/e2mgr/v1/e2t/list" -H
"accept: application/json" | jq '.')
```

```
for id in $(seq 1 ${NUM_OF_E2T}); do
  NODE_IP=$(kubectl get pod -n=ricplt -l app=ricplt-e2term-"${id}" -o
  jsonpath="{.items[0].status.hostIP}")

  NODE_PORT=$(kubectl get svc -n=ricplt service-ricplt-e2term-sctp-"${id}" -
  o jsonpath="{.spec.ports[0].nodePort}")

  E2T_ADDRESSES[$id]="${NODE_IP}:${NODE_PORT}"
done
```

- Set E2Term endpoint as parameter for E2 Simulator executable (Ref.: [https://gerrit.o-ran-sc.org/r/gitweb?p=sim/e2-interface.git;a=blob;f=e2sim/e2sm\\_examples/kpm\\_e2sm/Dockerfile;h=97353530f2f1495177747ea657cc1e0b53538673;hb=HEAD](https://gerrit.o-ran-sc.org/r/gitweb?p=sim/e2-interface.git;a=blob;f=e2sim/e2sm_examples/kpm_e2sm/Dockerfile;h=97353530f2f1495177747ea657cc1e0b53538673;hb=HEAD))

MyExample: helm chart

```
spec:
  containers:
    - name: ....
      image: ....
      imagePullPolicy: ....
      restartPolicy: ....
      command: ["/bin/sh"]
      args: ["-c", "${EXECUTABLE with E2TERM ENDPOINT} "]
  ....
```

## Building docker image and running simulator instance

To start building docker image one should generate the .deb packages using following commands :

```
$ cd build // create folder if not present
```

Generate .deb packages and move it to e2sm\_examples/kpm\_e2sm folder (on plain Ubuntu you might need to install "apt install cmake c++ libsctp-dev")

```
cmake .. && make package && cmake .. -DDEV_PKG=1 && make package
cp *.deb ../e2sm_examples/kpm_e2sm/
```

Now we are ready to build the docker image using below command. In the Dockerfile replace 10001 10002.

```
cd ../e2sm_examples/kpm_e2sm/
docker build -t <simulator-image-name> .
```

if you wish to change the e2t address to connect then modify the Dockerfile in e2sm\_examples /kpm\_e2sm/ path.

```
CMD kpm_sim 10.110.102.29 36422
```

Create instance of simulator :

```
$ docker run <simulator-image-name>
```

It will connect to specified e2t instance.

## Building docker image and running simulator instance as helm chart

o start building docker image one should generate the .deb packages using following commands :

```
$ cd build // create folder if not present
```

Generate .deb packages and move it to e2sm\_examples/kpm\_e2sm folder (on plain Ubuntu you might need to install "apt install cmake c++ libsctp-dev")

```
cmake .. && make package && cmake .. -DDEV_PKG=1 && make package
cp *.deb ../e2sm_examples/kpm_e2sm/
```

Now we are ready to build the docker image using below command. In the Dockerfile replace 10001 10002.

```
cd ../e2sm_examples/kpm_e2sm/
docker build -t e2simul:0.0.2 .
```

if you wish to change the e2t address to connect then modify the Dockerfile in e2sm\_examples /kpm\_e2sm/ path.

CMD kpm\_sim 10.110.102.29 36422

Find the correct IP address to specify there using: `$ kubectl get svc -n ricplt | grep -i e2term-sctp`

Start the helm chart:

```
helm install e2sim ./helm -n ricplt
```