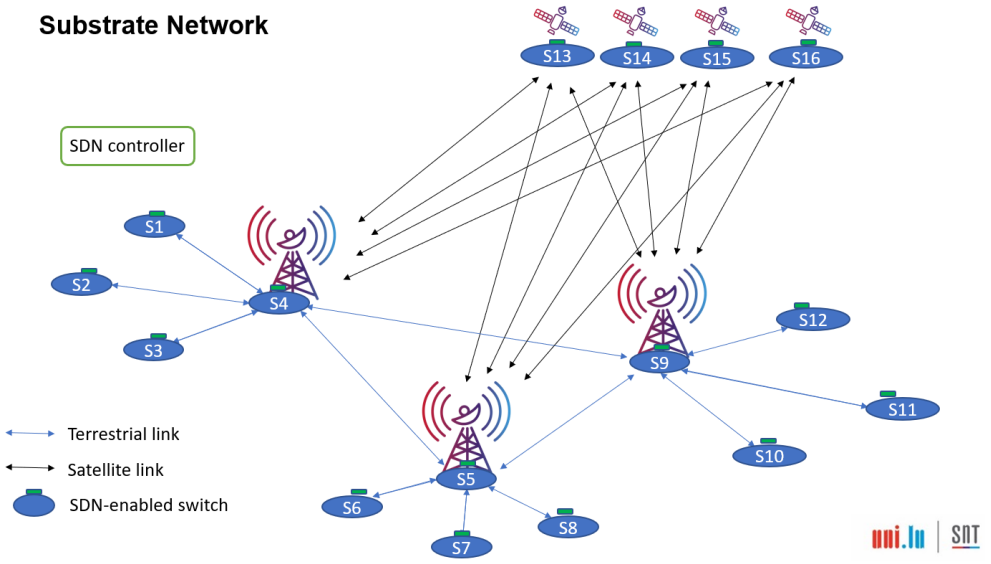
Manual for REST Interface of RYU SDN controller

Author: Mario Minardi
Affiliation: Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg

This document is provided to help the connection between SDN controller and external entity via the REST interface. The steps are listed below.

1. The Substrate network should be initially emulated. Run a Mininet script with OpenFlow switches of the type Open vSwitch. An example is provided in the python file "example_for_REST.py", to be run with the command "sudo python example_for_REST.py".

   The substrate network emulated is the following:



2. Then the Ryu controller may be initiated even with some standard pre-built applications. An example is provided in the following:

```
mario@mario-VirtualBox:~/ryu/ryu/app$ ryu-manager ofctl_rest.py rest_topology.py --observe-links
loading app ofctl_rest.py
loading app rest_topology.py
loading app ryu.controller.ofp_handler
loading app ryu.topology.switches
loading app ryu.controller.ofp_handler
instantiating app None of DPSet
creating context dpset
creating context wsgi
instantiating app ryu.topology.switches of Switches
instantiating app rest_topology.py of TopologyAPI
instantiating app ryu.controller.ofp_handler of OFPHandler
instantiating app ofctl_rest.py of RestStatsApi
(8828) wsgi starting up on http://0.0.0.0:8080
(8828) accepted ('127.0.0.1', 48826)
127.0.0.1 - - [06/Jul/2022 10:28:26] "GET /v1.0/topology/links HTTP/1.1" 200 10893 0.053268
127.0.0.1 - - [06/Jul/2022 10:28:28] "GET /v1.0/topology/switches HTTP/1.1" 200 8590 0.002278
127.0.0.1 - - [06/Jul/2022 10:29:20] "GET /v1.0/topology/links HTTP/1.1" 200 10893 0.001112
127.0.0.1 - - [06/Jul/2022 10:29:21] "GET /v1.0/topology/switches HTTP/1.1" 200 8590 0.001496
```

ATTENTION: The option --observe-links is necessary, otherwise the SDN controller will not be able to externally provide the current substrate topology.

3. Matlab, or any external tool, may be used to get the info on the topology, run an external mapping application, and then use the SDN controller to POST the computed mapping for any traffic request.
   Some internal MATLAB applications have been developed for the GET and POST requests.

   At the following link, https://gitlab.uni.lu/mminardi/rest_interface/-/blob/main/GET_request_sender.m, the source code for the GET requests can be found.

   The **GET_request_sender**() function takes no input and gives 2 output:
   - the adjacency matrix
   - the topology with the port number for each link, which will be used for the POST request. This is in the format of “struct”.

   The **POST_request_sender** function, which gives no output, but takes several inputs, is discussed few lines below. It is important to underline that here a simple case is shown. The mapping is provided for a fixed amount of time and each traffic is differentiated with the Vlan ID. Traffic from service requests may be differentiated by several factors, e.g. MAC/IP source or destination addresses, TCP/UDP port, etc..

   Input:
   - *topology_from_RYU* = topology directly received from RYU in the type of “struct”, see “GET_request_sender.m” above
   - *link_mapping* = this is the sequence of SDN-enabled switch which the traffic has to go through from source to destination (this is typically computed by an external mapping algorithm, e.g. Dijsktra).

- *Lifetime* = this is the hard timeout the OpenFlow flow rules in every switch of the path will be installed for. Not necessary, but typically this is done to avoid any overloading of every switch.
- *Switches* = this is the number of switches in the network.
- *Number_of_VN* = this is the number of the traffic request. Our approach differentiates every traffic request with the VLAN identifier in order to not be dependent on addresses, but different options may be chosen as explained above.

The code for the POST request function can be found at https://gitlab.uni.lu/mminardi/rest_interface/-/blob/main/POST_request_sender.m.

As a result, we can check that the topology is correctly received in Matlab (figure below).
IMPORTANT: only links between SDN-enabled switches are reported.