Exploiting Controller APIs:

Lesson 3

Please don't write anything on this paper!

Objectives:
Understand the northbound API of the SDN controller by exploiting and accessing this API using a topology viewer application. You won't get any points on this lesson, however there's a homework for points.

Tasks:
Run a larger mininet topology
sudo mn --switch ovsk,protocols=OpenFlow13 --controller remote --topo tree,3

Start the controller with the GUI topology application.
sudo ryu-manager ryu.app.gui_topology.gui_topology --observe-links

- Check whether the GUI works on [controller_ip]:8080
- Analyze what the –observe-links parameter does
- Try ping between the hosts
- Does it work? Yes/no? Why?

Ryu is capable of chaining multiple application. Kill the previous controller and start a new one chaining multiple applications with the switching module:
sudo ryu-manager --observe-links ryu.app.gui_topology.gui_topology ryu.app.simple_switch_13

- Analyse the REST api of the REST_topology application
- Find the REST API endpoints in the sources of the ryu application
- Use CURL / any other REST api connector to execute GET commands towards this API (e.g Postman / curl / paw)

Homework (3 pts):
As you can see, the GUI_Topology application does not show the hosts on the topology only the switches with the connectors. By examining the outputs of the REST API calls, you can see, that the API exposes the information about the hosts too. Write a short program in any programming language, which will connect to the API of the RYU application's API and display the full topology once a pingall command was called in the mininet console ('arp' table is populated). Re-using the gui_topology app is forbidden!
You might get some extra points for really nice solutions.
Deadline next week.