Trav Feller – Lab 5

1. How did your rule that creates the sensor pico install rules in the new child pico?
    a. I chained multiple events together so that each event would send the request to the child node and then raise another event to add another ruleset, 4 times in total.
2. How did you ensure that your sensor picos were created before sending them the event telling them their profile was updated?
    a. On the last of my 4 rulesets installed rules, I raised an event to update the profile. This way, all 4 of my rulesets would for sure be installed in the child pico.
3. How did you create a test harness for your pico system?
    a. I created my test harness by writing up a simple python program that called raised the correct events in the sensor pico. I had to give it the eci of the parent pico, and then the parent pico called the coorelating funcitons of the children picos. I included my python in the github repository.
4. In this set up, the picos representing sensors don't need to talk to each other and the sensor management pico is the parent, so it has channels to each child. How could you provide channels between sensor picos if sensor-to-sensor interaction were necessary?
    a. The parent pico could send a request for each child pico to create an open channel and then pass that channel eci back up to the parent. The parent could keep track of all channels and each child pico could request the channel eci of the other child that it wishes to communicate with. This way, the parent would store all the info and each child could just query the parent for the channel. This also means that the parent would also have to send down a channel eci to each child to communicate with it!