This is the documents which identify the vulnerabilities of my current implementation for part 3 and part 4. As always, I identify them separately.

**Part 3:**

**Layer-3 “Shortest-Path Switching” Routing Application Implementation**

I use Bellman-Ford algorithm to calculate the shortest path from a switch to another. I know that the link connected two switches is undirected. Bellman-Ford algorithm can work for undirected graph; however, it cannot work with undirected graph with negative weight. Currently I assume that the weight of each link is equal to one. When in the future, if you want to set the weight of some links to negative. It will cause problems.

**Part 4:**

**Distributed Load Balance Routing Application Implementation**

The Part 4 implementation has one problem. When you are not using a fresh mininet, it may cause that some hosts are not reachable. Most of the time the problem can be resolved by restarting the virtual box and restarts the Floodlight and SDN applications and restarts the mininet for couple of times. If it doesn’t, try restart the virtual machine. It happens every now and then in my virtual box, and I don’t find out what causes the problems. Running the ShortestPathSwitching alone won’t causes the problem.

There is another vulnerability. You can ping a virtual IP from a host and get the ARP response packet back. However, for some reason, the host cannot “recognize” the ARP response packet. I check the format and content of the ARP response packets carefully. It’s correct as far as I’m concerned, but the host cannot recognize it, therefore cause the ping packet to be considered lost.