

## Installing necessary software:-

Please install the following software-

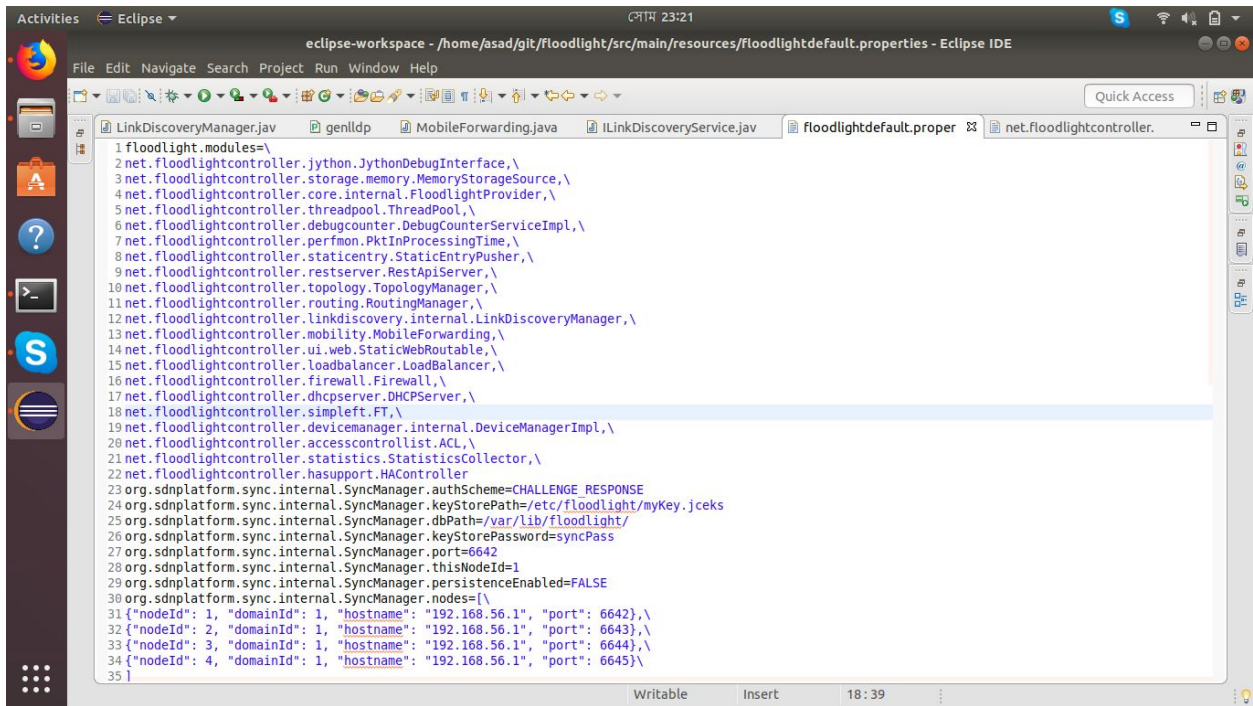
1. Mininet WiFi
2. Floodlight SDN controller (must install from source)
3. Eclipse IDE
4. Java SE 8
5. Python 2.7
6. Scapy

## Installing the project:-

The project contains two modules for the Floodlight SDN controller and a script for enabling LLDP for mobile hosts. To install the floodlight modules, just put the java files in appropriate directory in floodlight source code and then enable the modules. The files that need to be placed in floodlight source directory are following-

1. ILinkDiscoveryService.java [put in  
    'floodlight/src/main/java/net/floodlightcontroller/linkdiscovery']
2. LinkDiscoveryManager.java [put in  
    'floodlight/src/main/java/net/floodlightcontroller/linkdiscovery/internal']
3. MobileForwarding.java [put in 'floodlight/src/main/java/net/floodlightcontroller/mobility']  
    You need to make a folder named 'mobility' at  
    'floodlight/src/main/java/net/floodlightcontroller'

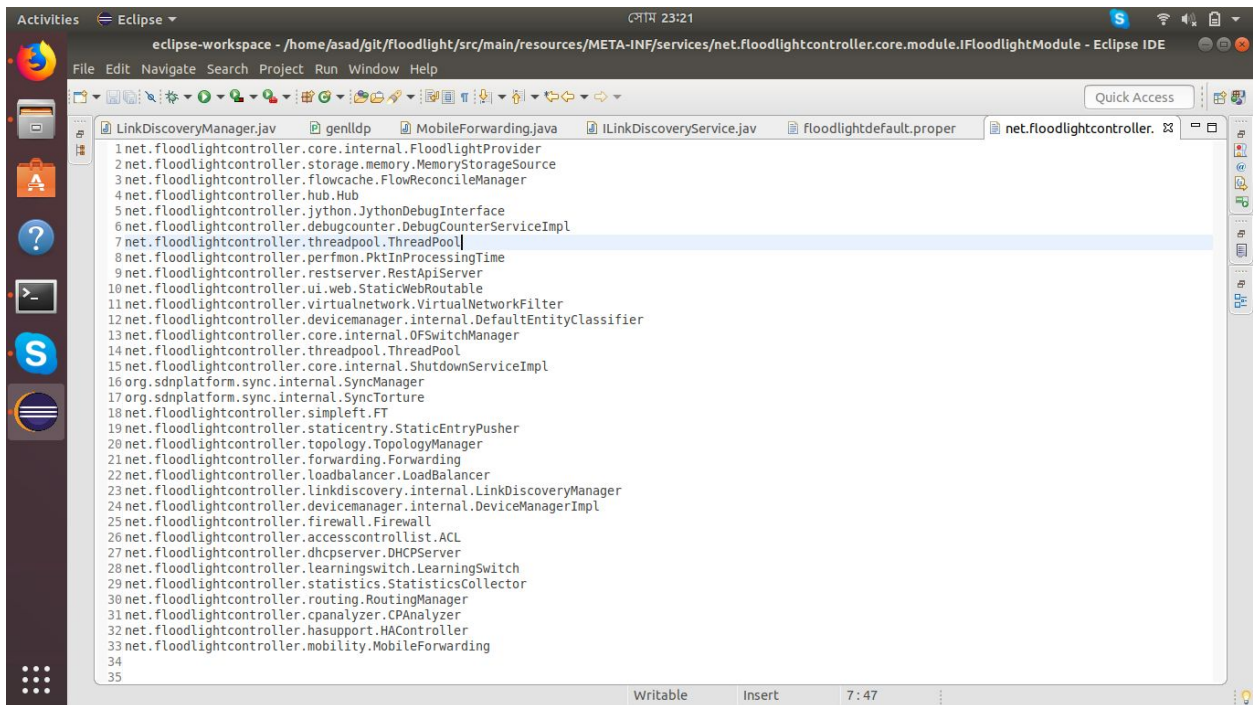
Now, enable these modules by using two configuration files. The appropriate directory for each configuration file is given in following screenshot.



The screenshot shows the Eclipse IDE with the file `floodlightdefault.properties` open. The file contains a list of modules and their configurations. The status bar at the bottom indicates the file is writable and the cursor is at line 18, column 39.

```
1 floodlight.modules=\
2 net.floodlightcontroller.jython.JythonDebugInterface,\
3 net.floodlightcontroller.storage.memory.MemoryStorageSource,\
4 net.floodlightcontroller.core.internal.FloodLightProvider,\
5 net.floodlightcontroller.threadpool.ThreadPool,\
6 net.floodlightcontroller.debugcounter.DebugCounterServiceImpl,\
7 net.floodlightcontroller.perfmon.PktInProcessingTime,\
8 net.floodlightcontroller.staticentry.StaticEntryPusher,\
9 net.floodlightcontroller.restserver.RestApiServer,\
10 net.floodlightcontroller.topology.TopologyManager,\
11 net.floodlightcontroller.routing.RoutingManager,\
12 net.floodlightcontroller.linkdiscovery.internal.LinkDiscoveryManager,\
13 net.floodlightcontroller.mobility.MobileForwarding,\
14 net.floodlightcontroller.ui.web.StaticWebRouteable,\
15 net.floodlightcontroller.loadbalancer.LoadBalancer,\
16 net.floodlightcontroller.firewall.Firewall,\
17 net.floodlightcontroller.dhcpserver.DHCPserver,\
18 net.floodlightcontroller.simpleleft.FT,\
19 net.floodlightcontroller.devicemanager.internal.DeviceManagerImpl,\
20 net.floodlightcontroller.accesscontrol.ACL,\
21 net.floodlightcontroller.statistics.StatisticsCollector,\
22 net.floodlightcontroller.hasupport.HAController
23 org.sdnplatform.sync.internal.SyncManager.authScheme=CHALLENGE_RESPONSE
24 org.sdnplatform.sync.internal.SyncManager.keyStorePath=/etc/floodlight/myKey.jceks
25 org.sdnplatform.sync.internal.SyncManager.dbPath=/var/lib/floodlight/
26 org.sdnplatform.sync.internal.SyncManager.keyStorePassword=syncPass
27 org.sdnplatform.sync.internal.SyncManager.port=6642
28 org.sdnplatform.sync.internal.SyncManager.thisNodeId=1
29 org.sdnplatform.sync.internal.SyncManager.persistenceEnabled=FALSE
30 org.sdnplatform.sync.internal.SyncManager.nodes=\
31 { "nodeId": 1, "domainId": 1, "hostname": "192.168.56.1", "port": 6642},\
32 { "nodeId": 2, "domainId": 1, "hostname": "192.168.56.1", "port": 6643},\
33 { "nodeId": 3, "domainId": 1, "hostname": "192.168.56.1", "port": 6644},\
34 { "nodeId": 4, "domainId": 1, "hostname": "192.168.56.1", "port": 6645}\
35
```

Fig. 1: First configuration file



The screenshot shows the Eclipse IDE with the file `net.floodlightcontroller.core.module.IFloodlightModule` open. The file contains a list of modules and their configurations. The status bar at the bottom indicates the file is writable and the cursor is at line 7, column 47.

```
1 net.floodlightcontroller.core.internal.FloodLightProvider
2 net.floodlightcontroller.storage.memory.MemoryStorageSource
3 net.floodlightcontroller.flowcache.FlowReconcileManager
4 net.floodlightcontroller.hub.Hub
5 net.floodlightcontroller.jython.JythonDebugInterface
6 net.floodlightcontroller.debugcounter.DebugCounterServiceImpl
7 net.floodlightcontroller.threadpool.ThreadPool
8 net.floodlightcontroller.perfmon.PktInProcessingTime
9 net.floodlightcontroller.restserver.RestApiServer
10 net.floodlightcontroller.ui.web.StaticWebRouteable
11 net.floodlightcontroller.virtualnetwork.VirtualNetworkFilter
12 net.floodlightcontroller.devicemanager.internal.DefaultEntityClassifier
13 net.floodlightcontroller.core.internal.OFSwitchManager
14 net.floodlightcontroller.threadpool.ThreadPool
15 net.floodlightcontroller.core.internal.ShutdownServiceImpl
16 org.sdnplatform.sync.internal.SyncManager
17 org.sdnplatform.sync.internal.SyncTorture
18 net.floodlightcontroller.simpleleft.FT
19 net.floodlightcontroller.staticentry.StaticEntryPusher
20 net.floodlightcontroller.topology.TopologyManager
21 net.floodlightcontroller.forwarding.Forwarding
22 net.floodlightcontroller.loadbalancer.LoadBalancer
23 net.floodlightcontroller.linkdiscovery.internal.LinkDiscoveryManager
24 net.floodlightcontroller.devicemanager.internal.DeviceManagerImpl
25 net.floodlightcontroller.firewall.Firewall
26 net.floodlightcontroller.accesscontrol.ACL
27 net.floodlightcontroller.dhcpserver.DHCPserver
28 net.floodlightcontroller.learningswitch.Learningswitch
29 net.floodlightcontroller.statistics.StatisticsCollector
30 net.floodlightcontroller.routing.RoutingManager
31 net.floodlightcontroller.cpanalyzer.CPANalyzer
32 net.floodlightcontroller.hasupport.HAController
33 net.floodlightcontroller.mobility.MobileForwarding
34
35
```

Fig. 2: Second configuration file

Running the experiment:-

1. First, start the floodlight controller by using the following command: `java -jar target/floodlight.jar` [Note: you should first `cd` to floodlight root directory before running this]

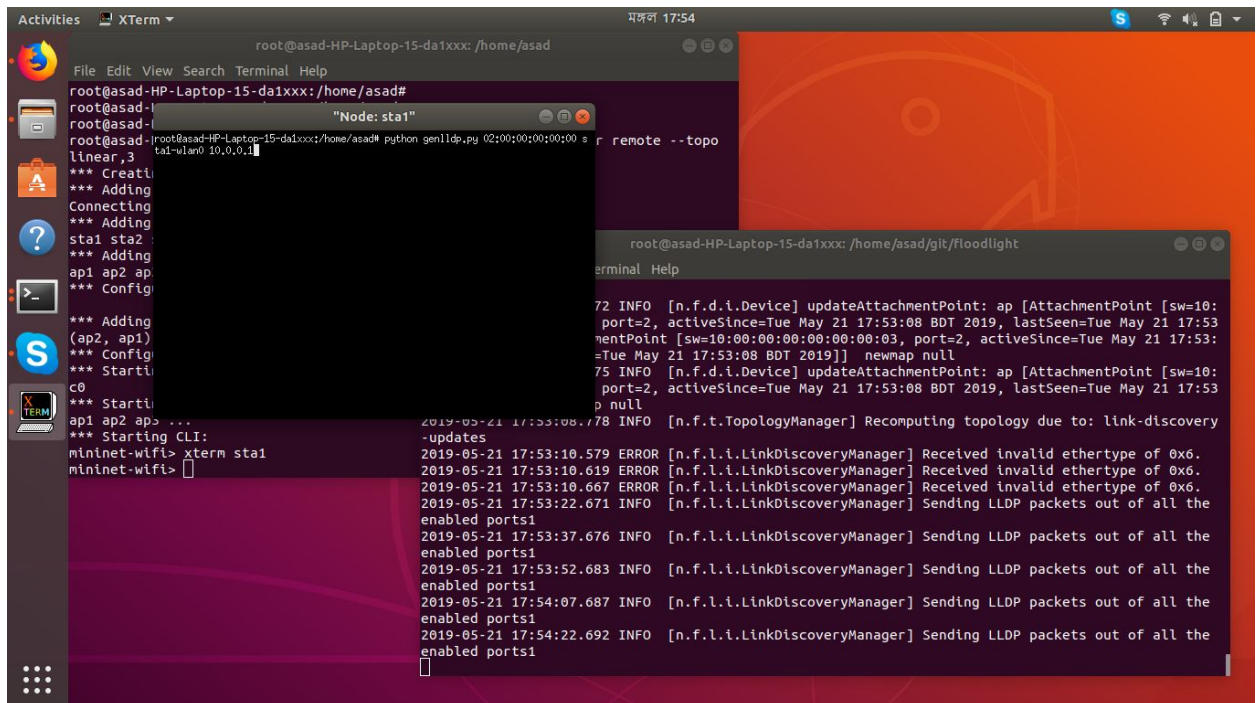
[illegible]

2. Now start mininet: `mn --wifi --controller remote --topo linear,3`

```
root@asad-HP-Laptop-15-da1xxx: /home/asad
File Edit View Search Terminal Help
root@asad-HP-Laptop-15-da1xxx: /home/asad#
root@asad-HP-Laptop-15-da1xxx: /home/asad#
root@asad-HP-Laptop-15-da1xxx: /home/asad#
root@asad-HP-Laptop-15-da1xxx: /home/asad#
root@asad-HP-Laptop-15-da1xxx: /home/asad# mn --wifi --controller remote --topo
linear,3
*** Creating network
*** Adding controller
Connecting to remote controller at 127.0.0.1:6653
*** Adding stations:
sta1 sta2 sta3
*** Adding access points:
ap1 ap2 ap3
*** Configuring wifi nodes...

*** Adding link(s):
(ap2, ap1) (ap3, ap2) (sta1, ap1) (sta2, ap2) (sta3, ap3)
*** Configuring nodes
*** Starting controller(s)
c0
*** Starting switches and/or access points
ap1 ap2 ap3 ...
*** Starting CLI:
mininet-wifi>
```

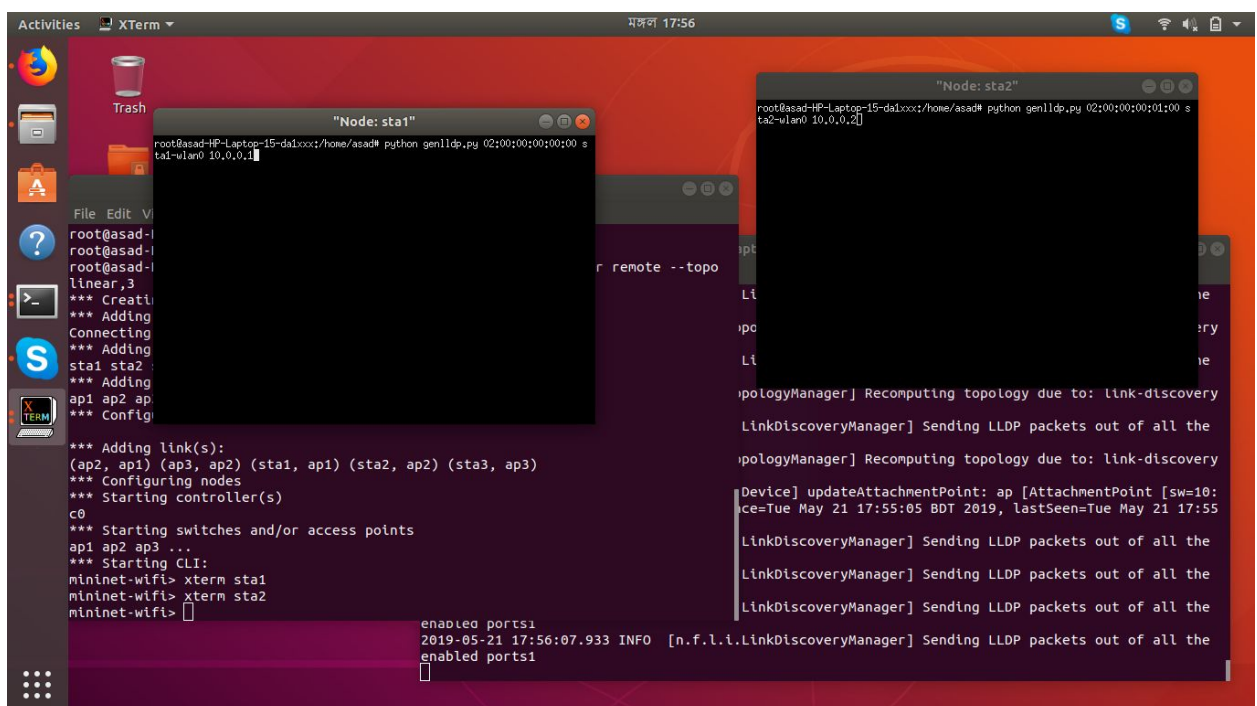
3. Now to enable LLDP in the wifi stations run a command in each of them. But before that you can run a ping just for cross checking.



```
root@asad-HP-Laptop-15-da1xxx: /home/asad#  
root@asad-HP-Laptop-15-da1xxx: /home/asad#  
root@asad-HP-Laptop-15-da1xxx: /home/asad# python genlldp.py 02:00:00:00:00:00 sta1-wlan0 10.0.0.1  
*** Creating node: sta1  
*** Adding link(s):  
*** Adding link(s): (ap2, ap1) (ap3, ap2) (sta1, ap1) (sta2, ap2) (sta3, ap3)  
*** Configuring nodes  
*** Starting controller(s)  
c0  
*** Starting switches and/or access points  
ap1 ap2 ap3 ...  
*** Starting CLI:  
mininet-wifi> xterm sta1  
mininet-wifi> xterm sta2
```

```
2019-05-21 17:53:08.778 INFO [n.f.t.TopologyManager] Recomputing topology due to: link-discovery  
-updates  
2019-05-21 17:53:10.579 ERROR [n.f.l.i.LinkDiscoveryManager] Received invalid ethertype of 0x6.  
2019-05-21 17:53:10.619 ERROR [n.f.l.i.LinkDiscoveryManager] Received invalid ethertype of 0x6.  
2019-05-21 17:53:10.667 ERROR [n.f.l.i.LinkDiscoveryManager] Received invalid ethertype of 0x6.  
2019-05-21 17:53:22.671 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1  
2019-05-21 17:53:37.676 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1  
2019-05-21 17:53:52.683 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1  
2019-05-21 17:54:07.687 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1  
2019-05-21 17:54:22.692 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1
```

You will see that ping is not successful. This is because the algorithm depends on LLDP and it is not enabled on the wifi mobile nodes yet. To enable LLDP in the hosts, run a python script in each of them (using appropriate command line argument for each). Example command for sta1: `python genlldp.py 02:00:00:00:00:00 sta1-wlan0 10.0.0.1` See screenshot for details.



```
root@asad-HP-Laptop-15-da1xxx: /home/asad# python genlldp.py 02:00:00:00:00:00 sta2-wlan0 10.0.0.2
```

```
2019-05-21 17:56:07.933 INFO [n.f.l.i.LinkDiscoveryManager] Sending LLDP packets out of all the  
enabled ports1
```



4. Now that lldp is enabled on sta1 and sta2, you should run the ping again.
5. Now, you will see that ping is successful this time. If you look at the floodlight terminal now, you will see messaging showing detected destination switch, the installed path in the network and more useful messages.

6. To see if node mobility is handled successfully you can now connect one of the mobile hosts to a different access point. To do so, you need three steps-
  - a. Check which access point a host is currently connected to: `sta1 iw dev sta1-wlan0 link`
  - b. Disconnect the host from current access point: `sta1 iw dev sta1-wlan0 disconnect`
  - c. Connect the host to a different access point: `sta1 iw dev sta1-wlan0 connect ssid_ap2`



7. Now run ping again and check floodlight messages to see the difference.