



Useful mininet setups

Mininet is a network emulation platform that is very useful to test SDN applications that you build. It can support different types of topologies. Here we showcase three popular configurations that will be helpful for testing.

1. Single switch

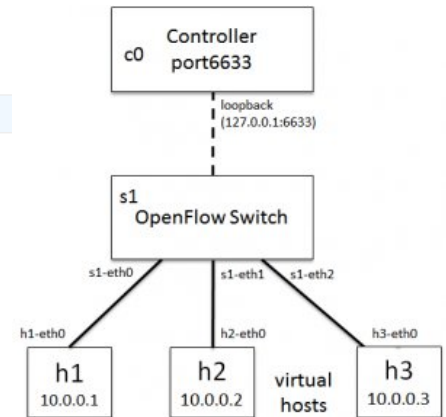
Following command spawns a single switch with 3 hosts attached to it. The hosts will be assigned static IP addresses and MAC addresses.

```
$ sudo mn --arp --topo single,3 --mac --switch ovsk --controller remote
```

In the above command, there are some important keywords worth paying attention to:

- **--mac:** Auto set MAC addresses
- **--arp:** Populate static ARP entries of each host in each other
- **--switch:** ovsk refers to kernel mode OVS
- **--controller:** remote controller can take IP address and port number as options

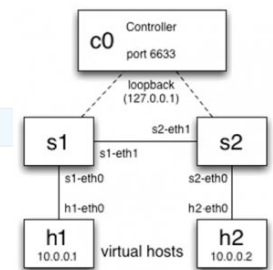
You can now perform *ping* between hosts *h1* and *h2* using command **h1 ping h2**.



2. Two linear switches

Following command spawns two switches connected to each other with a link and has one host on each switch. All other options are similar to the last setup.

```
$ sudo mn --topo linear --switch ovsk --controller remote
```



3. Load-balancer

Following command spawns a switch that has 3 servers and 1 client connected to it. This switch can be controller to act as a load-balancer. However, there are some additional steps to take care of.

```
$ sudo mn --arp --topo single,4 --mac --switch ovsk --controller remote
```

- **Virtual IP/MAC:** Pick a virtual IP (VIP) and MAC for the load-balancer. This is the IP address to which the clients will make a HTTP request. The controller will push rules to rewrite the VIP with the selected HTTP server. To make this work, you need to static set an ARP entry for the VIP in the client. If 'h1' is the client and 10.0.0.5 is the VIP, the following command will add the static ARP entry:

```
mininet> h1 arp -s 10.0.0.5 00:00:00:00:00:05
```

- **Server setup:** The *--arp* keyword is very important to populate MAC addresses in each host. Besides that we need to run the following commands within mininet:

```
mininet> h2 python -m CGIHTTPServer &
mininet> h3 python -m CGIHTTPServer &
mininet> h4 python -m CGIHTTPServer &
```

- **Warm-up controller learning:** After the hosts are up, it is important to make the controller learn the location of each host. You can do this through a *pingall* command in mininet:

```
mininet> pingall
```

- **Client request:** In our custom VM, we have CGI script configured to report back which server is handling a particular client request. Thus, when a client performs the following command, you will receive the IP address of the handling server.

```
mininet> h1 curl http://10.0.0.5:8000/cgi-bin/serverip.cgi
```

