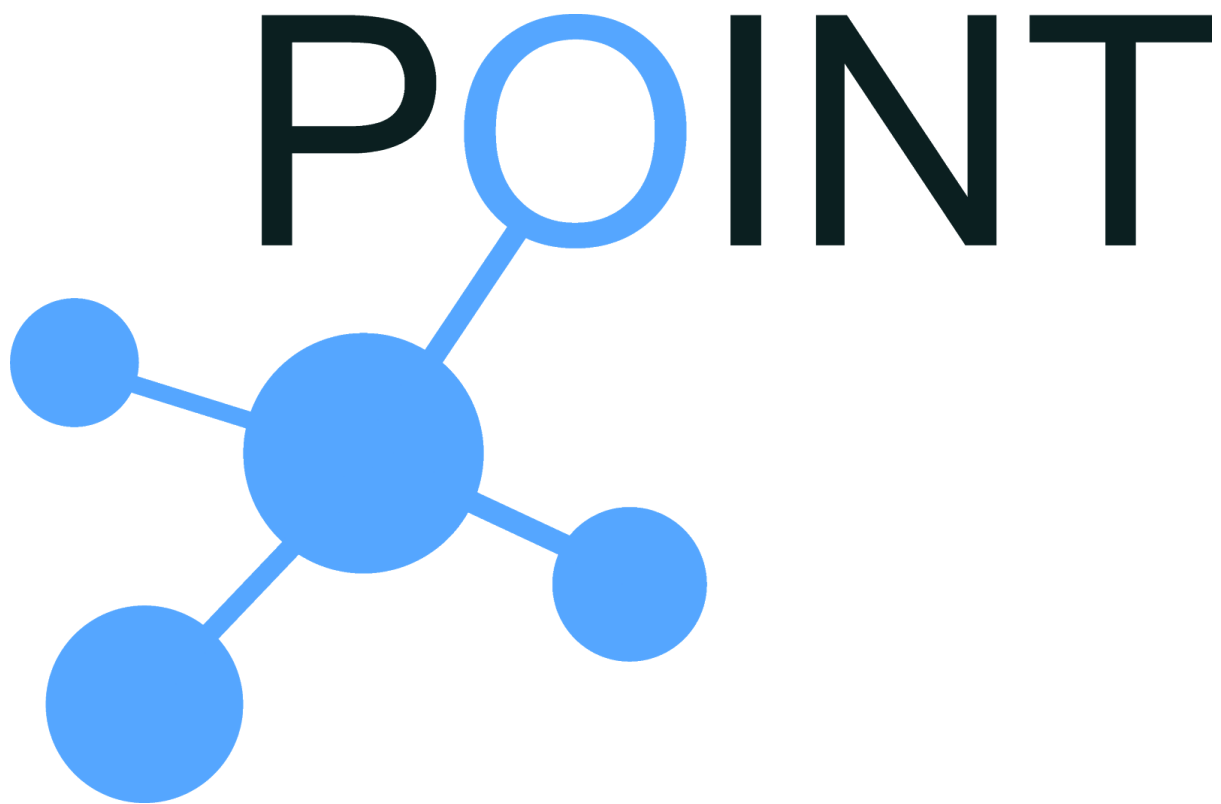


H2020 iP Over ICN- the betTer IP (POINT)

## Examples

Deploy an ICN-over-SDN Network with the SDN Controller OpenDayLight

---



**Author:** George Petropoulos

[1. Overview](#)

[2. Prerequisites](#)

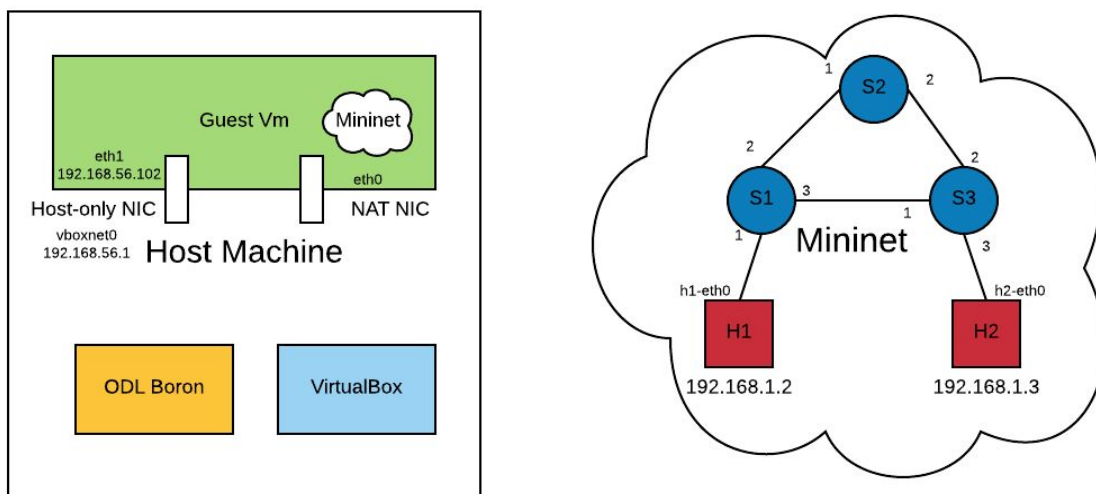
[3. Deploy the Network](#)

# 1. Overview

This example is about the ICN-over-SDN deployment capabilities of POINT, using ODL that has the ABM feature published - in the Borron release. The example is based on a Mininet-emulated SDN topology, but could be replicated in any virtualized or physical SDN topology.

## 2. Prerequisites

The example assumes a basic setup including a host machine with Ubuntu 14.04 or 16.04, with Virtualbox installed, and a Guest Virtual Machine (VM) with Ubuntu 15.04 installed. The Guest VM is configured to include at least one network interface, ideally two network interfaces. One host-only NIC to connect to the host machine, and one NAT NIC to connect to the Internet. For the host-only NIC, it is assumed that prefix 192.168.56.1/24 is used, and Guest VM is given (statically) an IP address 192.168.56.102, while the Host Machine has 192.168.56.1.



In the host machine, it is assumed that the reader has successfully installed the prerequisites to run the Opendaylight Boron SDN controller, as indicated in [HowTo-SDN](#) document.

In the Guest VM, it is assumed that the reader has successfully installed Blackadder, following the instructions of the [HowTo](#) document.

In addition, in the Guest VM, the example have been tested with Mininet version 2.2.1 and Openvswitch version 2.4.0.

## 3. Deploy the Network

Assuming that the reader has already cloned in the Mininet VM the POINT repository to a folder *point*, and has already built and installed it, follow the next steps.

In the Mininet VM, transfer the `demo_topo_deployment_tool.py` file from the *mininet-example* folder to the home directory.

```
$ cp ~/point/sdn/mininet-example/demo_topo_deployment_tool.py
~/demo_topo_deployment_tool.py
```

**In the host machine, run the OpenDaylight Boron release executable:**

```
$ cd distribution-karaf-0.5.2-Boron-SR2
$ ./bin/karaf
```

**At the OpenDaylight console, install the appropriate features to install flow rules:**

```
> feature:install odl-openflowplugin-flow-services
odl-openflowplugin-southbound
odl-openflowplugin-flow-services-rest odl-l2switch-all
```

**Check that OpenDaylight runs properly by checking the OpenDaylight console:**

```
> log:tail
```

**In the Mininet VM, run the custom python Mininet topology:**

```
$ cd ~
$ sudo python demo_topo_deployment_tool.py
```

**In Mininet console, ping all hosts:**

```
> pingall
```

To enable the communication between the SDN controller in the Host Machine, and the 2 Mininet hosts (h1 and h2) in the Guest VM, add a route via the appropriate interface and IP address of your setup:

```
$ sudo ip route add 192.168.1.0/24 via 192.168.56.102 dev vboxnet0
```

Replace 192.168.56.102 and vboxnet0 with the IP address of your VM and the host-only interface of Virtualbox. To test it from your host machine, ping 192.168.1.2 and ping 192.168.1.3 (Mininet hosts).

**In your host machine, use the deployment file of**

[https://github.com/point-h2020/point-2.0.0/tree/master/deployment/examples/odl\\_sample.cfg](https://github.com/point-h2020/point-2.0.0/tree/master/deployment/examples/odl_sample.cfg) to deploy an ICN topology over SDN:

```
$ cd deployment/
$ make all
$ ./deploy -c examples/odl_sample.cfg -o --nokill --nostart
```

**In your Mininet VM, check that ABM Openflow rules are configured:**

```
> sudo ovs-ofctl dump-flows s1 -O OpenFlow13
> sudo ovs-ofctl dump-flows s2 -O OpenFlow13
> sudo ovs-ofctl dump-flows s3 -O OpenFlow13
```

You should see entries matching packets of IPv6 type and source or destination addresses.

Then execute the following commands in the Mininet console:

```
> h1 /usr/local/bin/click /tmp/00000001.conf > /tmp/h1_log 2>&1 &  
> h2 /usr/local/bin/click /tmp/00000005.conf > /tmp/h2_log 2>&1 &  
> h1 ./point-cycle1/TopologyManager/tm /tmp/topology.graphml >  
/tmp/tm_log 2>&1 &  
> h1 ./point-cycle1/examples/samples/publisher > /tmp/pub_log &  
> h2 ./point-cycle1/examples/samples/subscriber
```

You should see correct published data in the Mininet console. In addition, if you dump-flows in switches s1 and s3 with the aforementioned commands, you should check increased packet number on rules with output ports 3 and 1 respectively, while dump-flows in switch s2 indicates no traffic.