Niels Bouten and Rashid Mijumbi
Fed4FIRE – GENI Research Experiment Summit (FGRE 2015)

Network Function Virtualization experiments on the Fed4FIRE testbed

A Proof of Concept

1. RSpec

In this tutorial we use the following RSpec:

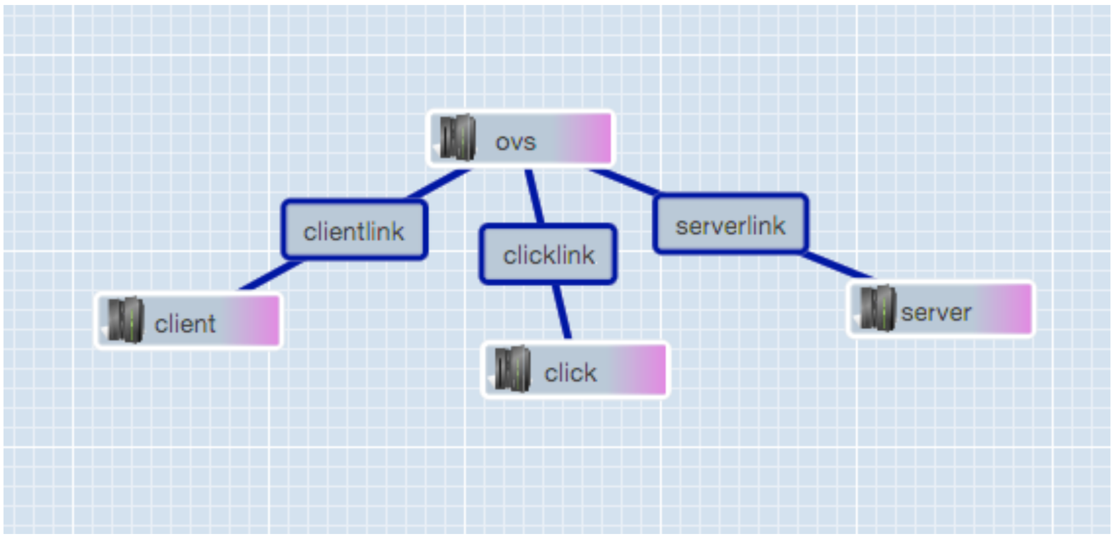
> http://doc.ilabt.iminds.be/fgre/_downloads/nfv_tutorial.rspec

The topology contains 4 nodes. The "ovs" node is an Open vSwitch instance that will be pre-installed by the rspec by using the following commands:

```
<services>
  <execute shell="sh" command="sudo sh /local/nfvfiles/install-ovs-
handler.sh"/>
  <execute shell="sh" command="sudo /local/install-script.sh"/>
  <install install_path="/local"
url="http://www.gpolab.bbn.com/experiment-support/OpenFlowOVS/of-
ovs.tar.gz"/>
  <install install_path="/local"
url="http://users.ugent.be/~nbouten/nfvfiles.tar.gz"/>
</services>
```

The required files are downloaded using the "install"-command both for the OVS installation and the tutorial installation.

The layout of the topology looks like the following:



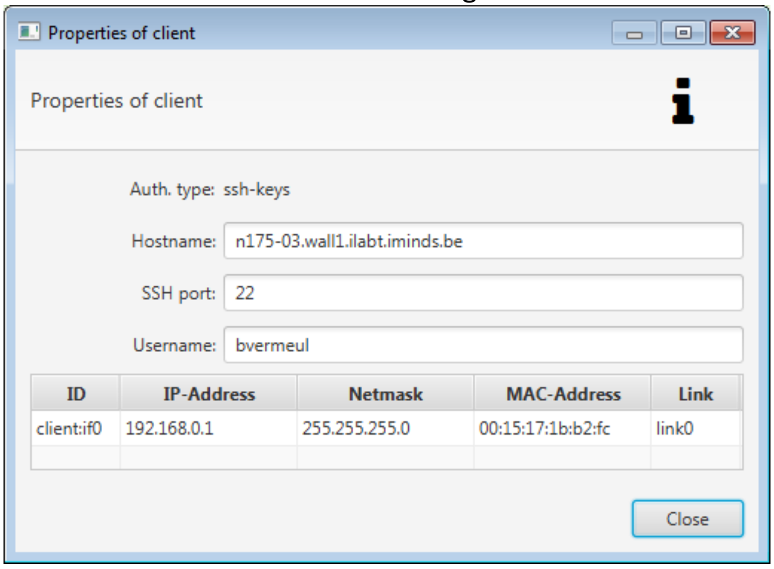
The click node uses an image where Click modular router by using the following command:

```
<sliver_type name="raw-pc">
```

```
  <disk_image
name="urn:publicid:IDN+wall1.ilabt.iminds.be+image+emulab-
ops:DEB60-CLK"/>
</sliver_type>
```

2. SFC Management Interface

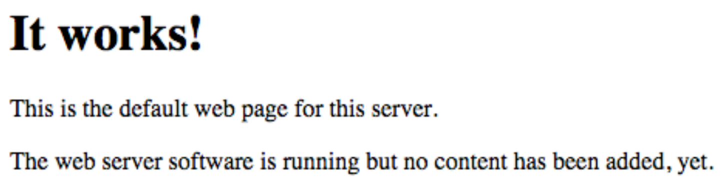
The SFC management interface will be accessible through the browser and will run on the OVS node. To access this interface we need to find out the management address of the OVS node. In jFed you can right click on the OVS node, and click 'show node info'. This will show something similar to:



Auth. type: ssh-keys
Hostname: n175-03.wall1.ilabt.iminds.be
SSH port: 22
Username: bvermeul

ID	IP-Address	Netmask	MAC-Address	Link
client:if0	192.168.0.1	255.255.255.0	00:15:17:1b:b2:fc	link0

You can copy and the hostname and paste it in a browser, which should give this:



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

To start the management interface we need to start the Python based controller for the OVS node, which also deploys the browser-based GUI for the SFC management interface. Login on the OVS node by double clicking it in jFed and execute these commands:

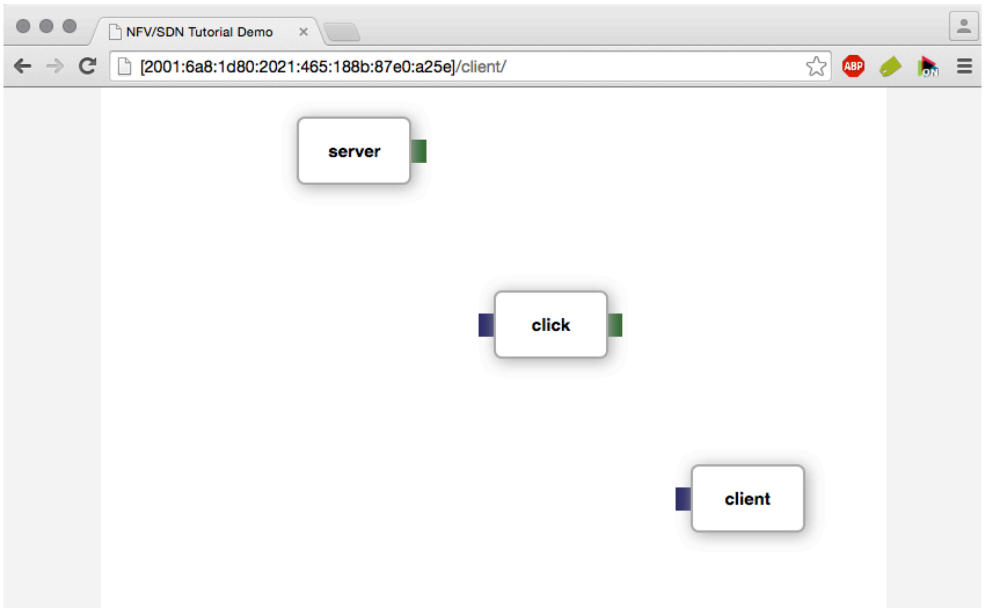
```
ovs> cd /local/nfvfiles
ovs> sudo python generate-ovs.py
```

This configures the OVS switch and runs the python controller which listens to incoming commands:

```
Control network interface:
      MAC: 00:30:48:78:f4:e4
      dev: eth4

Experiment network interfaces:
client
      MAC:  00:15:17:1b:b2:57
      IP: 10.10.1.1
      IF:  eth2
click
      MAC:  00:15:17:1b:b2:1c
      IP: 10.10.1.2
      IF:  eth5
server
      MAC:  00:15:17:1b:b4:98
      IP: 10.10.1.3
      IF:  eth1
Added client to controller
Added click to controller
Added server to controller
```

Pointing your browser to the aforementioned hostname (add /client to the end), e.g. http://n175-03.wall1.ilabt.iminds.be/client yields the following GUI:



By interconnecting the different network functions and clicking "Install Routing" at the bottom, you are able to generate the SFCs and create the routing which is automatically configured by using ovs commands:

```
server started and listening on localhost:8000
Installing client,click;click,server
```

To remove the routing, you can click on "Clear Routing" at the bottom:

```
Clearing route clear:client
```

3. Installing click functions

To install click functions, log in on the click node, and go the the folder containing the click-scripts:

```
click> cd /local/nfvfiles/clickfiles
```

The following can be used to instantiate a click bandwidth shaper at 50Mbps:

```
click> sudo bash install-click-bandwidthshaper.sh 50Mbps
```

The following can be used to instantiate a click delay shaper at 50ms one-way-delay:

```
click> sudo bash install-click-delayshaper.sh 50ms
```

The following can be used to block port 8080 using a click firewall:

```
click> sudo bash install-click-firewall.sh 8080
```

The following can be used to instantiate a click shaper at bandwidth 50Mbps and one-way-delay of 50ms:

```
click> sudo bash install-click-shaper.sh 50Mbps 50ms
```

4. Checking the SFC Functionality

To check the functionality of the bandwidthshaper, we will use iperf. It was pre-installed by the installation scripts at both the client and the server. Run iperf at the server:

```
server> iperf -s
```

Run iperf at the client:

```
client> iperf -c server
```

For a click bandwidthshaper of 50Mbps this is the result of iperf:

```
Client connecting to server, TCP port 5001
TCP window size: 23.5 KByte (default)
------------------------------------------------------------
[  3] local 10.10.1.1 port 43346 connected with 10.10.1.3 port
5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  60.4 MBytes  50.5 Mbits/sec
```

To check the functionality of the delayshaper, we will use ping. Run ping at the client or the server:

```
client> ping server
```

```
server> ping client
```

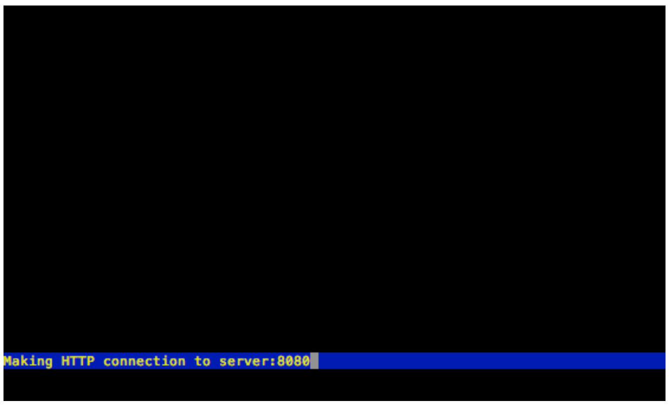
For a click delayshaper of 50ms one-way-delay this is the result of a ping:

```
client> ping server
PING server-serverlink (10.10.1.3) 56(84) bytes of data.
64 bytes from server-serverlink (10.10.1.3): icmp_req=1 ttl=64
time=101 ms
64 bytes from server-serverlink (10.10.1.3): icmp_req=2 ttl=64
time=101 ms
64 bytes from server-serverlink (10.10.1.3): icmp_req=3 ttl=64
time=101 ms
```

To check the functionality of the firewall, we will use the lynx browser. Both port 8080 and 8081 are valid websites at the server. They were preinstalled by the installscript and point to "facebook" and "nfv tutorial" website respectively. If port 8080 was blocked, accessing it using lynx:

```
client> lynx server:8080
```

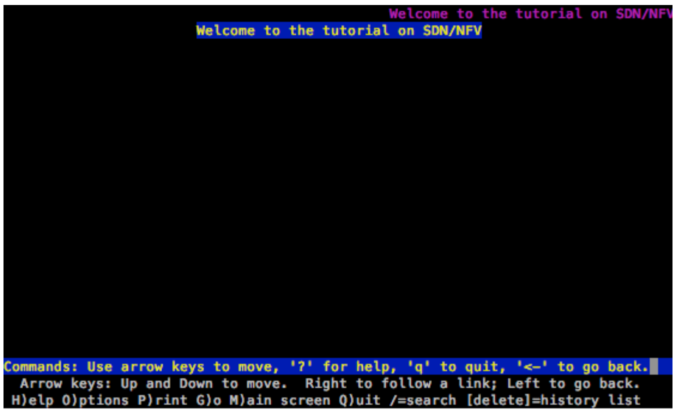
will yield an inaccessible page:



```
Making HTTP connection to server:8080
```

While the other site is still available:

```
client> lynx server:8081
```



```
                                                  Welcome to the tutorial on SDN/NFV
                         Welcome to the tutorial on SDN/NFV

Commands: Use arrow keys to move, '?' for help, 'q' to quit, '<-' to go back.
  Arrow keys: Up and Down to move.  Right to follow a link; Left to go back.
 H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```