H2020 i**P** **O**ver Ic**N**- the bet**T**er IP (POINT)

Examples

Simulate an ICN-only POINT Network with the Network Simulator 3 (NS3)

Author: Mohammed Qasim Al-Khalidi

1. Overview

The main objective of this document is to guide the reader through the required steps to set up an ICN network of various sizes and topologies simulated in an NS3 environment and how to run Pub/Sub applications on different nodes setting various traffic loads on the network. For NS3, Click or Blackadder installation and compilation process, please see the "How To.pdf" and "NS3 How To.pdf" documents available in the doc directory of your Blackadder installation "`<BLACKADDER PREFIX>/doc/`". In this setup we assume that the mentioned components are already installed and compiled to support NS3 – Blackadder simulations. Figure 1 shows a simple NS3 – Blackadder simulation example consisting of 5 nodes. In this setup node 3 has the TM/RV role in the network while nodes 1 and 5 run publish, subscribe applications and nodes 2 and 4 are forwarders in the network.

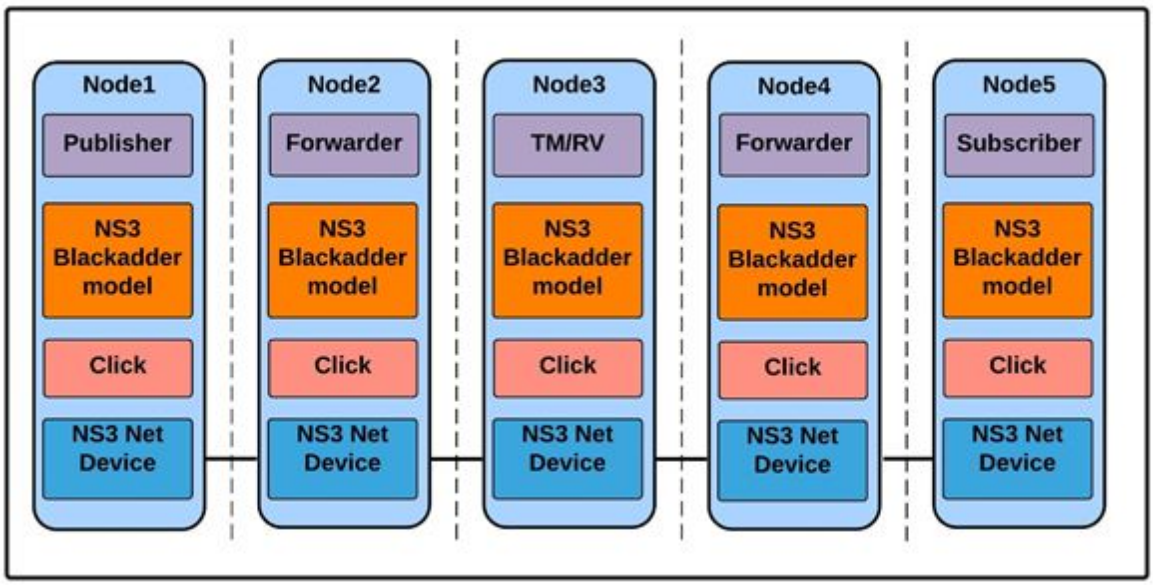

Figure 1: Basic NS3 – Blackadder Simulation Example

2. Configuring an NS3 Network

Creating the Blackadder configuration file can be done manually by defining all the network nodes, their connections and all Pub/Sub applications in the network. This process is described in the “NS3 How To.pdf” document available in the doc directory of Blackadder “`<BLACKADDER PREFIX>/doc/`”. This approach can be suitable for small networks but is found to be a time-consuming and error-prone process for large scale networks with different topologies and node connectivity degrees since multiple parameters must be manually added with a specific format. Therefore a script has been provided that allows for defining different random network topologies with any number of nodes into a Blackadder configuration file. This script is available in:

```
<BLACKADDER PREFIX>/ns3/config-generator/
```

To use this script you will need R programming language installed on your machine, in addition to "graph" and "igraph" packages.

To install R language from terminal :

apt-get install r-base

Then to install graph and igraph packages:

~$ R

> source("http://bioconductor.org/biocLite.R")

> biocLite("graph")

> install.packages("igraph")

After installation, open the script file and edit the number of nodes you would like to be simulated (5 in this example) at line 11 of the code as follows:

```
g <- barabasi.game(5, directed = FALSE)
```

Make sure that the node links always stay undirected (directed=FALSE) in order to allow the auto generator code to create bidirectional links for the blackadder topology. Worthy to note that the random network graph function above is just an

example and you are free to use any other available functions you may find suitable for your network topology. Also the node connectivity links are generated randomly and do not necessarily reflect the links of this example.

Now run the script using R :

```
~$ R -f <BLACKADDER \\
PREFIX>/ns3/config-generator/blackadder-topology-autogenerator.R
```

and the blackadder configuration file (ns3_topology.cfg) will be automatically generated in your workspace. After generating the Blackadder configuration file, you will need to define the TM/RV node and Pub/Sub applications to run on the nodes. In this example Node 3 will be the TM/RV of the network, Node 1 will run a publisher application and Node 5 will run a subscriber application, Thus the configuration file should be edited as follows (needed modifications are highlighted in red):

1) Add TM/RV to the role definition of node 3 as follows:

```
{
      label = "00000003";
      role = ["RV", "TM"];
      connections = (
      {
            to = "00000002";
            Mtu = 1500;
            DataRate = "100Mbps";
            Delay = "10ms";
      },
      {
            to = "00000004";
            Mtu = 1500;
            DataRate = "100Mbps";
            Delay = "10ms";
      }
      );
},
```

2) Add the publisher application to Node 1 as follows:

```
{
      label = "00000001";
      role = [];
      connections = (
      {
            to = "00000002";
            Mtu = 1500;
            DataRate = "100Mbps";
            Delay = "10ms";
      }
      );
      applications = (
      {
            name = "Publisher";
            start = "2.34";
            stop = "14.87";
      }
      );
},
```

3) Add the subscriber application to Node 5 as follows:

```
{
      label = "00000005";
      role = [];
      connections = (
      {
            to = "00000004";
            Mtu = 1500;
            DataRate = "100Mbps";
            Delay = "10ms";
      }
      );
      applications = (
```

```
        {
                name = "Subscriber";
                start = "2.34";
                stop = "14.87";
        }
        );
}
```

3. Deploying the NS3 Network

After defining the pub/sub applications of the example, the configuration file is ready for deployment. To deploy the NS3 example use the configuration file created earlier and deploy by running:

```
./deploy -c /<PATH TO CONFIG FILE>/ns3_topology.cfg –s
```

By using -s (--simulate) flag, the deployment tool will create all necessary click configuration files, the topology file and NS3 simulation (C++) code. The generated file "topology.cpp" will contain the NS3 code necessary to run the network configuration.

4. Running the NS3 Example

First of all, you will need to copy topology.cpp that was created during the deployment process into the appropriate location in the NS3 directory `<NS3 PREFIX>/ns-3.X/examples/blackadder/ .`

It will be called examples/blackadder/example3 in NS3.

```
cp /tmp/topology.cpp <NS3
PREFIX>/ns-3.X/examples/blackadder/example3.cc
```

Now edit `<NS3 PREFIX>/ns-3.X/examples/blackadder/wscript` to add the example by appending the following lines:

```
obj = bld.create_ns3_program('example3', ['core',
'point-to-point', 'blackadder', 'applications'])

obj.source = ['example3.cc', 'publisher.cc', 'subscriber.cc',]
```

Now build NS3:

```
cd <NS3_PREFIX>

./waf build
```

And finally you can run the example simulation with

```
./waf --run examples/blackadder/example3
```