Technical manual

Application Outcomes

1. Enable network compression for point-to-point links in ns-3.
2. Implement the network application that detects the presence of network compression by end-hosts.
3. Verify and validate your simulated compression link and compression detection application.c

Components

Compression Link

Compression link is responsible for compression and decompression of incoming and outgoing packets. Functionality: it takes a PPP packet, and first checks the protocol number in the header. If it determines that the checked packet type matches one to compress, it pre-processes then compresses it. Decompressor, at the other side of the compression link, then reverses all the pre-processing steps performed at the compressor, to retrieve the original incoming packet, before pushing it to the next interface.

Compression Detection Application

The network compression detection is implemented only in the cooperative environment. The network application is a client/server application where the sender sends two sets of 6000 UDP packets back-to-back (called packet train), and the receiver records the arrival time between the first and last packet in the train. Then the application detects is there a compression or not.

Topology

The 4-nodes topology implemented and build using ns-3 environment. In topology there are Node S - sender, Node R- receiver and Nodes 1, R2 - routers. Nodes S and R are the end-hosts running the network application. Nodes R1 and R2 are the intermediate routers where the link between them is compression-enabled.

Simulations

- Transmit low entropy data over a network topology without a compression link.
- Transmit high entropy data over a network topology without a compression link.
- Transmit low entropy data over a network topology with a compression link.
- Transmit high entropy data over a network topology with a compression link

Installation

Prerequisites

Installation is divided into parts

- Part 1: Install

Virtual Box (Oracle), version 6.0.
Install Ubuntu, version 18.04.01.

- Part 2: Prerequisites of ns3 installation

C++ compiler, Python, Git, tar, bunzip2

- Part 3: Installtest NS-3 using the link:

https://www.nsnam.org/docs/tutorial/html/getting-started.html

- Part 4: Install zlib using commands

```
wget http://www.zlib.net/zlib-1.2.11.tar.gz
tar -xvzf zlib-1.2.11.tar.gz
cd zlib-1.2.11
./configure –prefix=/usr/local/zlib
make
sudo make install
```

Program Installation

It is important to clone the repository from GitHub:

```
git clone
https://github.com/icgranger9/cs621.git
cd cs621
./waf configure –build-profile=debug
–enable-examples –enable-tests
./waf build
```

To run: ./waf –run p2p.cc

File Changed/Created	Components
ns-3-dev/src/applications/model/	Topology
p2p-server	
p2p-client	
ns-3-dev/dev/src/applications/helper/	Topology
p2p-helper	
ns-3-dev/src/point-to-point/model/	Compression
point-to-point-net-device	
ns-3-dev/src/applications/model/	Compression Detection
compression-detection-client	
compression-detection-server	
ns-3-dev/src/applications/helper/	Compression Detection
compression-detection-server-client-helper	
ns-3-dev/src/applications/model/	Modified
udp-client.cc	
ns-3-dev/src/applications/	Modified
wscript	

Table 1: Files used in program and relation to components