BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC NHA TRANG

KHOA CÔNG NGHỆ THÔNG TIN



BÀI BÁO CÁO THỰC TẬP CƠ SỞ

CHỦ ĐỀ

TÌM HIỂU HỆ MÔ PHỎNG MẠNG NS-3 VÀ MÔ PHỎNG WLAN

**Giảng viên hướng dẫn: Cấn Thị Phượng**

**Sinh viên thực hiện: Nguyễn Thanh Sơn**

**Mã số sinh viên: 61134297**

KHÁNH HÒA - 2021

LỜI CAM ĐOAN

Tôi xin được cam đoan : Đề tài báo cáo thực tập cơ sở “Tim hiểu hệ mô phỏng NS-3 và mô phỏng WLAN” là kết quả dựa trên sự cô gắng, nỗ lực của bản thân với sự hướng dẫn nhiệt tình của giảng viên Cấn Thị Phượng. Các kết quả và nghiên cứu trong đề tài là trung thực và hoàn toàn không sao chép hoặc sử dụng kết quả của đề tài nghiên cứu nào tương tự.

Những phần sử dụng tài liệu tham khảo trong đồ án đã trích dẫn đầy đủ.

Nếu phát hiện có sự sao chép kết quả nghiên cứu của đề tài khác, tôi xin chịu hoàn toàn trách nhiệm và kỷ luật của giảng viên đưa ra.

Khánh Hòa, ngày 30 tháng 12 năm 2021

Sinh viên thực hiện

Nguyễn Thanh Sơn

LỜI CẢM ƠN

Để hoàn thành đề tài báo cáo thực tập cơ sở này, trước hết em xin gửi đến quý thầy, cô Khoa Công nghệ Thông tin – Trường Đại Học Nha Trang lời cảm ơn chân thành.

Em xin gửi đến cô Cấn Thị Phượng, người đã tận tình hướng dẫn, giúp đỡ em hoàn thành đề tài báo cáo này lời cảm ơn sâu sắc nhất.

Trong quá trình nghiên cứu thực hiện đề tài, cũng như trong quá trình làm bài báo cáo thực tập cơ sở, khó tránh khỏi sai sót, rất mong các thầy, cô bỏ qua. Đồng thời do kiến thức cũng như kinh nghiệm thực tiễn của bản thân còn hạn chế, bài báo cáo này khó tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp từ quý thầy, cô để em học hỏi thêm được nhiều kinh nghiệm, cũng như kỹ năng cần thiết.

Em xin chân thành cảm ơn!

LỜI MỞ ĐẦU

Chương I: Cơ sở lý thuyết

Chương 2: Tìm hiểu giao thức định tuyến NS-3, Wlan

A. Giới thiệu NS-3

Ns-3 simulator là một chương trình mô phỏng mã nguồn mở, hỗ trợ mô phỏng các sự kiện truyền thông rời rạc (discrete event simulation) của nhiều giao thức mạng khác nhau như WLAN, AODV, OSLR, …. Ns-3 được viết bằng ngôn ngữ C++, tuy nhiên người sử dụng chỉ cần nắm ngôn ngữ C++ ở mức cơ bản để có thể hiểu và xây dựng các kịch bản mô phỏng mong muốn.

Một số khái niệm cần quan tâm trong Ns-3:

1. Node

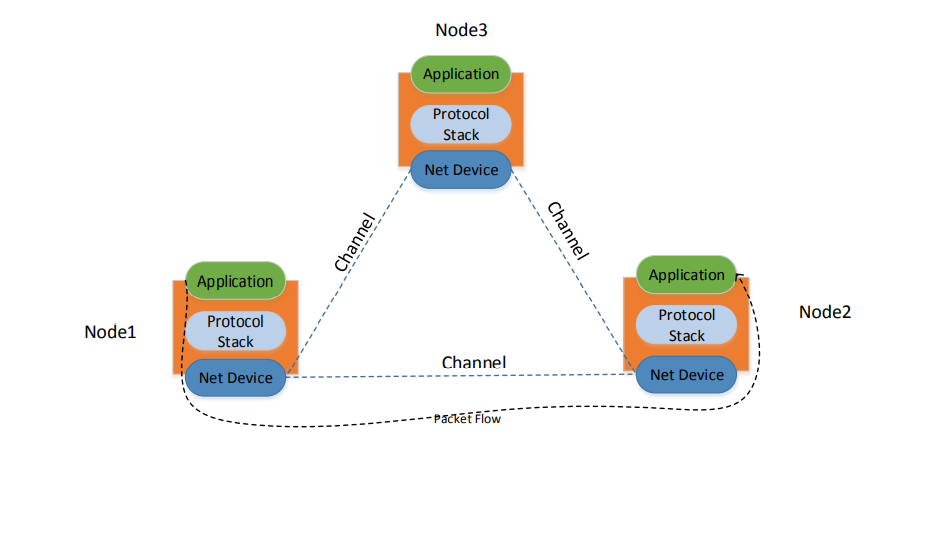
Trong Ns-3, chúng ta sử dụng một khái niệm chung thường được sử dụng bởi các trình mô phỏng mạng khác là node để biểu diễn cho một thiết bị tính toán cơ bản. Khái niệm được biểu diễn bằng lớp Node của ngôn ngữ C++ trong Ns-3. C thể xem môt Node như một máy tính mà chúng ta có thể thêm các chức năng cho nó bằng việc cài thêm ứng dụng dụng mạng, thêm các tầng giao thức, gắn thêm các card wifi, Bluetooth với các driver tương thích.

1. Application

Trong Ns-3, chúng ta không cần quan tâm đến các node sử dụng hệ điều hành gì hoặc cấu trúc lệnh hệ thống để lập trình mô phỏng giao thức mạng mà mình mong muốn. Tuy nhiên, trong Ns-3, vẫn tồn tại khái niệm Application được biểu diễu với lớp cùng tên trong C++. Lớp này cung cấp các phương thức để hỗ trợ xây dựng được những tính năng mô phỏng ở tầng ứng dụng mong muốn (Vd: FTP, HTTP…). Các nhà phát triển có thể mở rộng lớp Application để tạo ra ứng dụng cụ thể hơn phục vụ cho việc giải lập.

1. Channel

Trong thực tế, để kết nối một máy tính vào mạng, thông thường dữ liệu truyền dẫn qua mạng thông qua một kênh truyền – Channel. Ví dụ khi chúng ta kết nối cáp Ethernet vào ổ cắm trên tường, chúng ta đang kết nối vào một kênh giao tiếp Ethernet. Trong trình mô phỏng ns-3, chúng ta sẽ kết nối Node tới một đối tượng biểu diễn cho kênh truyền – Channel. Lớp Channel cung cấp phương thức để quản lý, lập trình những cách thức truyền thông thông giữa các Nodes. Những lớp Channel cụ thể có thể đặc tả được những mô hình phức tạp như Ethernet Switch, hoặc môi trường truyền dẫn mạng không dây trong thực thế (không gian 3 chiều). Một số lớp thông dụng mô tả các kênh truyền thông trong thực tế như CsmaChannel, PointToPointChannel và WifiChannel. Ví dụ CsmaChannel, mô hình một phiên bản của môi trường mạng không dây mà chúng ta có thể can thiệp vào điều chỉnh các thông số liên quan đến tính năng Carrier Sense Multiple Access /Collision Avoidance phục vụ cho mô phỏng.



1. Net Device

Trong ns-3, net device là một mô hình dùng mô phỏng cho các card mạng (NIC) mô phỏng cả thông số card và software drivver. Một net device được cài đặt lên một Node để nó có thể truyền thông được với node khác khác trong môi trường mô phỏng thông qua các kênh (Channel). Ns-3 cung cấp lớp NetDevice với các phương thức để quản lý các kết nối đến các đối tượng Node và Channel; và được thể được đặc tả củ thể hơn bởi nhà phát triển trong khi lập trình. Trong nội dung thực hành, chúng ta sẽ sử dụng nhiều phiên bản khác nhau của NetDevice như CsmaNetDevice, PointToPointNetDevice và WifiNetDevice. Lưu ý: NetDevice nào thì sử dụng Channel tương thích, Ethernet NIC được thiết kế để sử dụng với Ethernet Network, CsmaNetDevice làm việc với CsmaChannel, PointToPointNetDevice thì làm việc với PointToPointChannel và WifiNetDevice tương thích với WifiChannel.

1. Topology Helpers

Trong Ns-3, chúng ta sẽ thấy Nodes sẽ gắn liền với NetDevices. Trong những mô phỏng mạng lớn, chúng ta cần sắp đặt rất nhiều connections giữa các Nodes, NetDevices và Channels. Khi kết nối NetDevice với Nodes, NetDevices với Channels, gán địa chỉ IP, v.v.., là một trong những tác vụ phổ biến trong ns-3, topology helpers được cung câp1 để đơn giản hóa các công việc trên. Ví dụ, chúng ta có thể mất nhiều thao tác ns-3 cơ bản như vừa nêu để tạo một NetDevice, thêm một MAC address, cài đặt net device lên node, cấu hình các tầng giao thức rồi kết nối NetDevice tới một Channel. Và còn cần nhiều thời gian hơn để kết nối những thiết bị này tới nhiều điểm hoặc kết nối các mạng này lại với nhau, hoặc tạo thành một internetwork. Ns-3 cung cấp các lớp topology helper như NodeContainer, NetDeviceContainer, PointToPointHelper hay InternetStackHelper giúp thực hiện các tác vụ trên một cách nhanh chóng và tiện lợi hơn.

II. Cài đặt Ns-3

1. Yêu cầu:

Cài đặt máy ảo Ubuntu trong Vmware

1. Cài đặt gói thư viện:

$ sudo apt-get update

$ sudo apt-get install python-pip python-setuptools python-pip libxml2

$ sudo apt-get install gtk+2.0 qt4-dev-tools

$ sudo apt-get install graphviz libgraphviz-dev pkg-config

$ sudo apt-get install python-dev python-pygraphviz graphviz-dev python-kiwi

python-pygoocanvas python-gnome2 python-gnome2-dev python-rsvg

1. Install PygraphViz:

Download Links: <https://pypi.python.org/pypi/pygraphviz>

$ tar zxvf pygraphviz-1.3.1.tar.gz

$ cd pygraphviz-1.3.1

$ sudo python setup.py install --library-path=/usr/lib/graphviz

**Hoặc sử dụng Pip**

$ sudo pip uninstall pygraphviz

$ pkg-config --libs-only-L libcgraph

$ pkg-config --cflags-only-I libcgraph

$ pip install pygraphviz --install-option="--include-path=/usr/include/graphviz" -- install-option="--library-path=/usr/lib/graphviz/"

**Kiểm tra pygraphviz đã dc install**

$ python

$ >> import pygraphviz as pgv

1. Build Ns-3

Download link: <https://www.nsnam.org/release/ns-allinone-3.25.tar.bz2>

Kiểm tra các gói thư viện đã được cài đặt đầy đủ và configure

$ tar xjvf ns-allinone-3.25.tar.bz2

$ cd ns-allinone-3.25/ns-3.25

$ ./waf clean

$ ./waf –enable-sudo --build-profile=debug --enable-examples --enable-tests configure

$ cd ../

Cài đặt

$ ./build.py

1. Hướng dẫn compile và chạy một kịch bản mẫu

Để compile script các bạn cần copy đoạn script vào thư mục ns-3.25/scratch. Ví dụ muốn compile đoạn script first.cc trong thư mục ns-3.25/examples/tutorial/first.cc

$ cd ns-3.25

$ cp examples/tutorial/first.cc scratch/myfirst.cc

$ ./waf

Output

[614/708] cxx: scratch/myfirst.cc -> build/debug/scratch/myfirst\_3.o

[706/708] cxx\_link: build/debug/scratch/myfirst\_3.o -> build/debug/scratch/myfirst

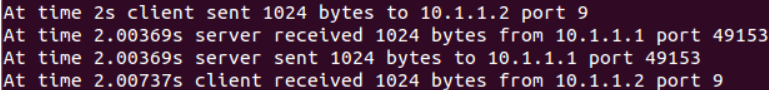
Waf: Leaving directory `/home/lxc/Downloads/ns-allinone-3.25/ns3.25/build'

'build' finished successfully (2.357s)

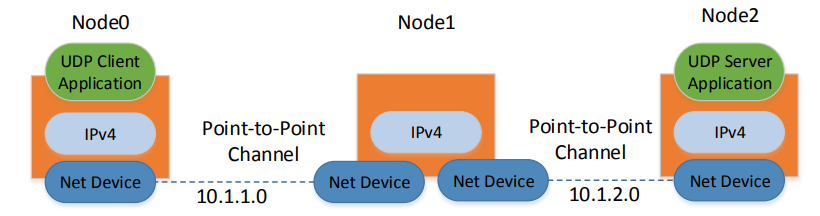
Chạy đoạn kịch bản vừa compile

$ ./waf --run scratch/myfirst

Output



III. Cài đặt và phân tích một mô phỏng đơn giản UDP Client-Server



Mục tiêu: nắm được các lớp và phương thức cơ bản để viết một kịch bản mô phỏng bằng ns3, đọc được các file pcap.

Các bước thực hiện:

1. Tạo một topology gồm 3 node (Node0, Node1, Node2) được kết nối với nhau bằng một đường truyền point-to-point.

NodeContainer n;

// Tạo 3 node

n.Create (3);

// Tạo một kênh kết nối point-to-point

PointToPointHelper p2p;

p2p.SetChannelAttribute("Delay",TimeValue(MilliSeconds (lat)));

p2p.SetDeviceAttribute("DataRate",DataRateValue(DataRate(rate));

p2p.SetDeviceAttribute ("Mtu", UintegerValue (1400));

// Cài đặt các netDevice trên node0, node1

NetDeviceContainer dev = p2p.Install (n.Get(0), n.Get(1));

// Cài đặt các netDevice trên node1, node2

NetDeviceContainer dev2 = p2p.Install (n.Get(1), n.Get(2));

Ghi chú: NodeContainer cho phép chúng ta tạo một tập các Node để chúng ta có thể cài đặt Channel, Protocol Stack và Application lên các node một cách nhanh chóng.

1. Cài Ipv4 Stack lên các node để chạy ứng dụng UDP, Cài đặt một UdpClient trên Node1 và UdpServer (server1) trên Node2. Cài đặt cho đường truyền có một dataRate cố định Rate1.

// Install Internet Stack

InternetStackHelper internet;

internet.Install (n);

Ipv4AddressHelper ipv4;

//…

// Tạo UdpServer trên 2 port 8000

UdpServerHelper server1 (8000);

// Dùng ApplicationContainer để cài đặt server vừa tạo lên Node2 trong mô hình topology phía trên

ApplicationContainer apps;

apps = server2.Install (n.Get (2));

1. Start ứng dụng client, đo lưu lượng end-to-end trong khi thay đổi độ trễ của đường truyền.

// Compile source code

Ns-3.25$./waf cp lab1.cc scratch/lab1.cc

Ns-3.25$./waf

// Chạy mô phỏng

Ns-3.25$./waf –run scratch/lab1 –-vis

--Vis để hiển thị visualization.

4. Giờ cài thêm một ứng dụng client lên Node1 và một server2 lên Node2.

B. Giới thiệu mô phỏng WLAN

1. WLAN là gì ?

WLAN là một loại mạng máy tính nhưng việc kết nối giữa các thành phần trong mạng không sử dụng các loại cáp như một mạng thông thường, môi trường truyền thông của các thành phần trong mạng là không khí. Các thành phần trong mạng sử dụng sóng điện từ để truyền thông với nhau.

1. Lịch sử của WLAN

Công nghệ Wlan xuất hiện lần đầu tiên vào cuối năm 1990, khi những nhà sản xuất giới thiệu những sản phẩm hoạt động trong băng tần 900Mhz. Những giải pháp này (không được thống nhất giữa các nhà sản xuất) cung cấp tốc độ truyền dữ liệu 1Mbps, thấp hơn nhiều so với tốc độ 10Mbps của hầu hết các mạng sử dụng cáp hiện thời

Năm 1992, những nhà sản xuất bắt đầu bán những sản phẩm WLAN sử dụng băn tần 2.4 Ghz. Mặc dù những sản phẩm này đã có tốc độ truyền dữ liệu cao hơn nhưng chúng vẫn là giải pháp riêng của mỗi nhà sản xuất không được công bố rộng rãi. Sự cần thiết cho việc hoạt động thống nhất giữa các thiết bị ở những dãy tần số khác nhau dẫn đến một số tổ chức bắt đầu phát triển ra những chuẩn mạng không dây chung.

Năm 1997, Institute of Electrical and Electronics Engineers (IEEE) đã phê chuẩn sự ra đời của chuẩn 802.11, và cũng được biết với tên gọi WIFI (Wireless Fidelity) cho các mạng WLAN. Chuẩn 802.11 hỗ trợ 3 phương pháp truyền tín hiệu vô tuyến ở tần số 2.4Ghz.

Năm 1999, IEEE thông qua 2 sự bổ sung cho chuẩn 802.11 là các chuẩn 802.11a và 802.11b (định nghĩa ra những phương pháp truyền tín hiệu). Và những thiết bị WLAN dựa trên chuẩn 802.11b đã nhanh chóng trở thành công nghệ không dây vượt trội. Các thiết bị WLAN dựa trên chuẩn 802.11b truyền phát ở tần số 2.4Ghz, cung cấp tốc độ truyền dữ liệu có thể lên tới 11Mbps. IEEE 802.11b được tạo ra nhằm cung cấp những đặc điểm về tính hiệu dụng, thông lượng (throughput) và bảo mật để so sánh với mạng có dây.

Năm 2003, IEEE công bố thêm một sự cải tiến là chuẩn 802.11g mà có thể truyền nhận thông tin ở cả 2 dãy tầng 2.4Ghz và 5Ghz và có thể nâng tốc độ truyền dữ liệu lên đến 54Mbps. Thêm vào đó, những sản phẩm áp dụng 802.11h cũng có thể tương thích ngược với các thiết bị chuẩn 802.11b, Hiện nay chuẩn 802.11g đã đạt đến tốc độ 108Mbps – 300Mbps.

1. Các mô hình WLAN

Mô hình mạng độc lập (IBSSs) hay còn gọi là mạng Adhoc

Mô hình mạng cơ sở (BSSs)

Mô hình mạng mở rộng (ESSs)

1. Ưu điểm của WLAN

Sự tiện lợi: Mạng không dây cũng như hệ thống mạng thông thường. Nó cho phép người dùng truy xuất tài nguyên mạng ở bất kỳ nơi đâu trong khu vực được triển khai (nhà hay văn phòng). Với sự gia tăng số người sử dụng máy tính xách tay (laptop), đó là một điều rất thuận lợi.

Khả năng di động: Với sự phát triển của các mạng không dây công cộng, người dùng có thể truy cập internet ở bất cứ đâu. Chẳng hạn ở các quán cafe, người dùng có thể truy cập Internet không dây miễn phí.

Hiệu quả: Người dùng có thể duy trì kết nối mạng khi họ đi từ nơi này sang nơi khác.

Triển khai: Việc thiết lập hệ thống mạng không dây ban đầu chỉ cần ít nhất 1 access point. Với mạng dùng cáp, phải tốn thêm chi phí có thể gặp khó khăn trong việc triển khai hệ thống cáp ở nhiều nơi trong tòa nhà.

Khả năng mở rộng: Mạng không dây có khả năng đáp ứng tức thì khi gia tăng số lượng người dùng. Với hệ thống mạng dùng cáp cần phải gắn thêm cáp.

1. Nhược điểm của WLAN

Bảo mật: Môi trường kết nối không dây là không khí nên khả năng bị tấn công của người dùng là rất cao.

Phạm vi: Một mạng chuẩn 802.11g với các thiết bị chuẩn chỉ có thể hoạt động tốt trong phạm vi vài chục mét. Nó phù hợp trong một căng nhà, nhưng với một tòa nhà lớn thì không đáp ứng được nhu cầu. Để đáp ứng cần phải mua thêm Repeater hay acsess point, dẫn đến chi phí gia tăng.

Độ tin cậy: Vì sử dụng sóng vô tuyến để truyền thông nên việc bị nhiễu, tín hiệu bị giảm do tác động của các thiết bị khác (lò vi sóng...) là không tránh khỏi.

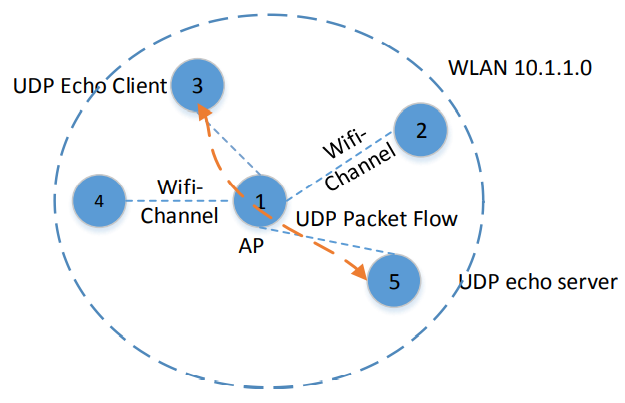
Làm giảm đáng kể hiệu quả hoạt động của mạng.

Tốc độ: Tốc độ của mạng không dây (1 – 125Mbps) rất chậm so với mạng sử dụng cáp (100Mbps đến hàng Gbps).

-Hướng dẫn các bước mô phỏng:

* Khởi động máy tính
* Khởi động chương trình mô phỏng OPNET v 14.5
* Sau khi OPNET đã được khởi động, chúng ta sẽ thực hiện quá trình xây dựng chương trình mô phỏng WLAN
* Chọn menu File/New
* Chọn mục Project- Ấn OK
* Đánh tên của Project vào và kịch bản của Project
* Project Name: WLAN\_LAB
* Scenario Name: \_WLAN\_
* Chọn Create empty scenario - Ấn next
* Chọn Office - Ấn Next
* Thay đổi các đơn vị tùy theo mục đích của bạn, nếu không thay đổi ta chọn Next
* Kéo xuống dưới, chọn mục Wireless\_lan\_adv, phần Include chọn Yes
* Chọn Finish để hoàn tất quá trình khởi tạo Project
* Cửa sổ mô phỏng hiển thị. Bao gồm các công cụ mô phỏng cần thiết đã được liệt kê sẵn
* Lập mô hình mạng theo sơ đồ sau đây:
* Trong mục Node models/chọn các đối tượng sau:
* Application
* Config
* Fixed Node Application Configuration
* Profile Config Fixed Node Profile Configuration
* Wlan\_wkstn\_adv Mobile Node Wriless Lan WorkSation
* Wlan\_wkstn\_adv Fixed Node Wriless Lan Worksation
* Cách kéo các đối tượng ra vùng mô phỏng để sử dụng:
* Giữ chặt chuột trái để vào công cụ được chọn, kéo rê ra ngoài vùng mô phỏng thả chuột công cụ sẽ được hiển thị trong vùng mô phỏng.
* Ấn delete để xóa đối tượng không cầm thiết
* Ấn Ctrl + S, để lưu dự án
* Chỉnh cấu hình cho hai đối tượng Application Config & Profile Config
* Đối tượng Application Config:
* Chuột phải vào đối tượng Profile Config: Set name & đặt tên cho đối tượng là proConfig
* Chuột phải vào đối tượng chọn: Edit Attributes & tùy chỉnh các tham số
* Chỉnh cấu hình cơ bản cho hai đối tượng wlan\_wkstn\_adv (Fixed Node) & wlan\_wkstn\_adv (Mobile Node):
* Wlan\_wkstn\_adv (Fixed Node)

1. Mô hình các node trong mô phỏng:



Trong mô hình sẽ có tổng cộng 5 node, chúng ta sẽ dùng một node làm Access Point (AP) – cụ thể là node đâu tiên. Sau đó, chúng ta sẽ cài đặt một kênh WifiChannel (cung cấp bởi ns-3) cho toàn bộ các node trong mô hình.

1. Hướng dẫn xây dựng mô hình mô phỏng WLAN trong ns-3

Một số phương thức trong ns-3 dùng để tạo các node, tạo kênh kết nối Wifi có thể tham khảo thêm ở đây.

NodeContainer: <https://www.nsnam.org/doxygen/classns3_1_1_node_container.html>

YansWifiChannelHelper:

[https://www.nsnam.org/doxygen/classns3\_1\_1\_yans\_wifi\_channel\_hel per.html](https://www.nsnam.org/doxygen/classns3_1_1_yans_wifi_channel_hel%20per.html)

WifiHelper:

<https://www.nsnam.org/doxygen/classns3_1_1_wifi_helper.html>

1. Đầu tiên tạo các Node và lấy một node đầu tiên (Node0) làm AP

NodeContainer wifiStaNodes;

wifiStaNodes.Create (nWifi);

// Dùng node0 làm AP

NodeContainer wifiApNode = wifiStaNodes.Get (0);

1. Tạo kênh kết nối WLAN (YansWifiChannel trong ns-3)

YansWifiChannelHelper channel = YansWifiChannelHelper::Default ();

YansWifiPhyHelper phy = YansWifiPhyHelper::Default ();

phy.SetChannel (channel.Create ());

WifiHelper wifi;

wifi.SetRemoteStationManager ("ns3::AarfWifiManager");

// Cài đặt các thông số cho kênh truyền Wifi như SSID

WifiMacHelper mac;

Ssid ssid = Ssid ("ns-3-ssid");

mac.SetType ("ns3::StaWifiMac","Ssid", SsidValue (ssid),

"ActiveProbing", BooleanValue (false));

1. Cài đặt kênh kết nối WLAN lên các NetDevice của các node và AP

// Cài đặt kênh truyền WLAN lên các node

NetDeviceContainer staDevice=wifi.Install(phy,mac,wifiStaNodes);

// Cài đặt kênh truyền WLAN lên node AP

NetDeviceContainer apDevices=wifi.Install(phy, mac, wifiApNode);

1. Cài đặt IPv4\_Stack lên tất các nodes và cấp phát IP Addr cho các NetDevice Interfaces vừa cài đặt ở bước 3

// Cài đặt IPv4 Stack cho các node

InternetStackHelper stack;

stack.Install (wifiStaNodes);

Ipv4AddressHelper address;

// Cấp IP cho các nodes

address.SetBase ("10.1.1.0", "255.255.255.0");

Ipv4InterfaceContainer wifiInterfaces;

wifiInterfaces = address.Assign (staDevices);

address.Assign (apDevices);

1. Cài đặt UDP Echo Client và Server

// Cài đặt UDPEchoServer trên port 9 ở node4

UdpEchoServerHelper echoServer (9);

ApplicationContainer serverApps = echoServer.Install (wifiStaNodes.Get (nWifi-1));

// Cài lên node4

// Cài đặt UDPEchoClient trên một node thuộc WLAN

UdpEchoClientHelper echoClient (wifiInterfaces.GetAddress (nWifi-1), 9);

// Cài lên node1

ApplicationContainer clientApps = echoClient.Install (wifiStaNodes.Get (1));

1. Compile, chạy mô phỏng và quan sát kết quả thu được (Xem các file pcap)

Ns-3.25$ cp lab2.cc scratch/lab2.cc

Ns-3.25$ ./waf

Ns-3.25$ ./waf –-run scratch/lab2 –-vis