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 VÀ MÔ PHỎNG GIAO THỨC OPENFLOW TRÊN NS-3**

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

**Sinh viên thực hiện : Phùng Sỹ Hoàng Sơn**

**Mã số sinh viên : 61136475**

KHÁNH HÒA - 2022

LỜI CAM ĐOAN

Tôi xin được cam đoan : Đề tài báo cáo thực tập cơ sở “Tim hiểu và mô phỏng giao thức OpenFlow trên NS-3 ” 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 02 tháng 01 năm 2022*

**Sinh viên thực hiện**

Phùng Sỹ Hoàng Sơn

LỜI CẢM ƠN

Trước hết, em xin gửi đến quý Thầy (Cô) Khoa Công nghệ Thông tin – Trường Đại Học Nha Trang lời cảm ơn chân thành vì đã giúp em có thêm kiến thức để hoàn thành được vấn đề em nghiêng cứ xuyên suốt trong thời gian qua.

Đặc biệt, Em xin gửi lời cảm ơn sâu sắc nhất đến cô Cấn Thị Phượng đã dành thời gian và công sức tận tình hướng dẫn, giúp đỡ em có thể hoàn thành báo cáo này một cách tốt nhất.

Tuy nhiên trong quá trình em nghiên cứu và thực hiện đề tài, cũng như trong quá trình làm bài báo cáo thực tập cơ sở, do kiến thức cũng như kinh nghiệm thực tiễn của bản thân còn hạn chế, khó tránh khỏi thiếu sót, rất mong các thầy (cô) thông cảm. 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

Vì những lí do trên nên em viết định chọn đề tài “ **Tìm hiểu và mô phỏng giao thức OpenFlow trên NS-3**” để báo cáo.

Chương 1: Tổng quan về vấn đề nghiêng cứu

Chương 2: Tìm hiểu NS-3

2.1. Giới thiệu NS-3

NS-3 là một mô phỏng mạng sự kiện rời rạc được nhắm mục tiêu chủ yếu cho nghiên cứu và sử dụng giáo dục. Dự án NS3, bắt đầu vào năm 2006, là một dự án nguồn mở phát triển ns-3.

2.2. Một số khái niệm trong NS3

2.2.1 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.

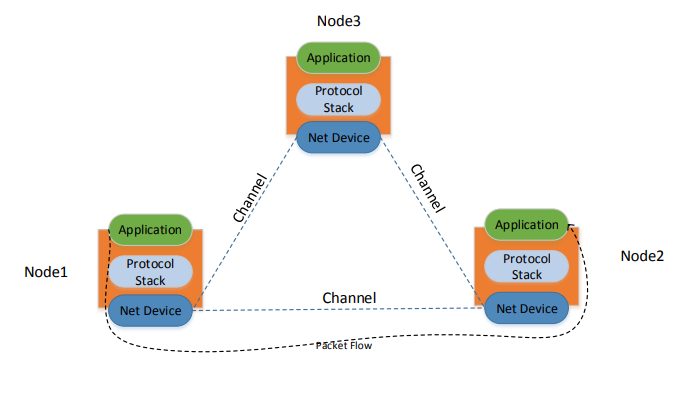
2.2.2 Application

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 hiên, 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 năngmô phỏng ở tầng ứng dụng mong muốn (Vd: FTP, HTTP…). Các nhà phát triển có thểmở rộng lớp Application để tạo ra ứng dụng cụ thể hơn phục vụ cho việc giải lập.

2.2.3 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.



Hình: Mô hình các node được kết nối qua mạng thông qua một kênh truyền – Channel

2.2.4 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 Nodeđể 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, Csma Net Device làm việc với Csma Channel, Point To Point Net Device thì làm việc với Point To Point Channel và Wifi Net Device tương thích với Wifi Channel.

2.2.5 Topology Helpers

Trong ns-3, chúng ta sẽ thấy Nodes sẽ gắn liền với NetDevices. Trong những mô phỏngmạng lớn, chúng ta cần sắp đặt rất nhiều connections giữa cácNodes,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 hayInternetStackHelpergiúp thực hiện các tác vụ trên một cách nhanh chóng và tiện lợi hơn.

2.3 Cài đặt NS-3 trên Ubuntu giả lập

Mở Teminal và nhập các lệnh

[$] sudo apt-get install gcc g++ python python-dev mercurial bzr gdb valgrind gsl-bin libgsl0-dev libgsl0ldbl flex bison tcpdump sqlite sqlite3 libsqlite3-dev libxml2 libxml2-dev libgtk2.0-0 libgtk2.0-dev uncrustify doxygen graphviz imagemagick texlive texlive-latex-extra texlive-generic-extra texlive-generic-recommended texinfo dia texlive texlive-

[$] mkdir ns3

[$] cd ns3

[$] wget [http://www.nsnam.org/release/ns-allinone-3.19.tar.bz2](http://www.nsnam.org/release/ns-allinone-3.13.tar.bz2)

  [$] tar xjf ns-allinone-3.19.tar.bz2

[$] cd ns-allinone-3.19/

[$] ./build.py --enable-examples --enable-tests

[$] ./waf -d debug --enable-examples --enable-tests configure

[$] ./waf

[$] ./test.py

Chương 3 : Tìm hiểu OpenFlow

3.1 Giới thiệu OpenFlow

OpenFlow là một giao thức truyền thông cho phép truy cập vào mặt phẳng chuyển tiếp của bộ chuyển đổi mạng hoặc bộ định tuyến qua mạng.OpenFlow cho phép truy cập trực tiếp và điều khiển mặt phẳng chuyển tiếp của các thiết bị mạng như switch và router, cả thiết bị vật lý và thiết bị ảo, do đó giúp di chuyển phần điều khiển mạng ra khỏi các switch thực tế tới phần mềm điều khiển trung tâm.

Sự xuất hiện của OpenFlow thực sự là một cuộc cách mạng, đưa sự phát triển của SDN lên một tầm cao mới. OpenFlow là giao thức hoạt động giữa tầng điều khiển (Control Layer) và tầng vật lý (Infrastructure Layer). Trong kiến trúc của SDN, tất cả các thiết bị được liên kết với tầng điều khiển và thông qua OpenFlow. OpenFlow có 2 nhiệm vụ chính:

• Giám sát hoạt động của các thiết bị mạng: Lưu lương mạng, trạng thái hoạt động của các nút mạng, các thông tin cơ bản về các thiết bị …

• Điều khiển hoạt động của thiết bị mạng: Điều khiển luồng dữ liệu (routing), Bảo mật, Quality of Service...

3.2 Các đặc trưng của openflow

+ OpenFlow có thể được sử dụng bởi ứng dụng phần mềm ngoài để điều khiển mặt phẳng chuyển tiếp của các thiết bị mạng, giống như tập lệnh của CPU điều khiển một hệ thống máy tính.

+ Giao thức OpenFlow được triển khai trên cả hai giao diện kết nối giữa các thiết bị cơ sở hạ tầng mạng và phần mềm điều khiển SDN.

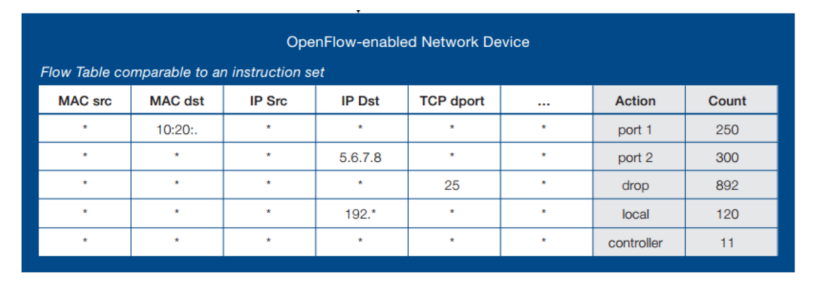
+OpenFlow sử dụng khái niệm “flow” (luồng) để nhận dạng lưu lượng mạng trên cơ sở định nghĩa trước các quy tắc phù hợp (được lập trình sẵn bởi phần mềm điều khiển SDN). Giao thức này còn cho phép định nghĩa các tham số, chẳng hạn như mô hình lưu lượng sử dụng, ứng dụng và tài nguyên cần thiết để xác định cách thức điều hướng lưu lượng truyền qua các thiết bị mạng.

+Giao thức OpenFlow là một chìa khóa để cho phép các mạng định nghĩa bằng phần mềm và cũng là giao thức tiêu chuẩn SDN duy nhất cho phép điều khiển lớp chuyển tiếp của các thiết bị mạng.

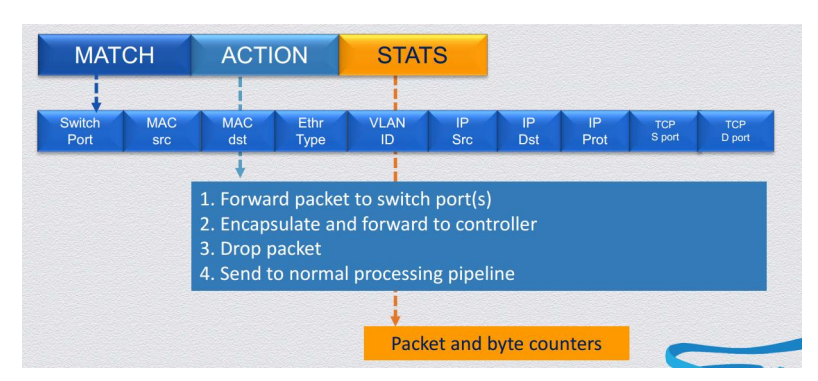
3.3 Cấu tạo và hoạt động của openflow

3.3.1 Cấu tạo

Một thiết bị OpenFlow bao gồm ít nhất 3 thành phần :



Hình 6. Một ví dụ về Flow Table

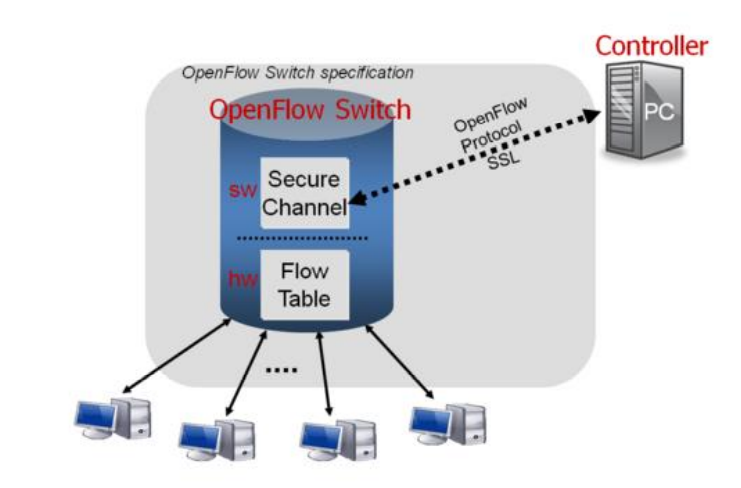
**

Hình 7. Các trường trong bảng flow và cách thức hoạt động

• Flow Table: một liên kết hành động với mỗi luồng, giúp thiết bị xử lý các luồng thế nào.

• Secure Channel: kênh kết nối thiết bị tới controller (controller), cho phép các lệnh và các gói tin được gửi giữa controller và thiết bị

• OpenFlow Protocol: giao thức cung cấp phương thức tiêu chuẩn và mở cho một controller truyền thông với thiết bị.



Hình 8. Các thành phần trong thiết bị OpenFlow

3.3.3 Hoạt động

Trong router hay switch cổ điển, các gói tin chuyển tiếp nhanh chóng (data path) và các quyết định định tuyến mức độ cao (control path) xảy ra trên cùng một thiết bị. OpenFlow Switch tách hai chức năng đó ra. Phần data patth vẫn nằm trên switch, trong khi control path được chuyển đến một controller riêng biệt, thường là một máy chủ tiêu chuẩn. OpenFlow Switch và Controller giao tiếp thông qua giao thức OpenFlow, trong đó xác định các tin nhắn, chẳng hạn như gói nhận được, gửi gói tin ra, sửa đổi bảng chuyển tiếp và nhận số liệu thống kê.

Các data path của một OpenFlow Switch được trình bày rõ ràng trong flow table; mỗi mục flow table chứa một tập các trường gói tin phù hợp và một hành động (như gửi ra cổng, sửa đổi trường hoặc hủy). Khi một OpenFlow Switch nhận được một gói tin nó chưa bao giờ thấy trước đây mà không có trong flow table, nó sẽ gửi gói tin này đến bộ điều khiển. Controller sau đó đưa ra quyết định về cách xử lý gói tin này. Nó có thể hủy (drop) các gói tin, hoặc nó có thể thêm một flow entry chỉ đạo việc chuyển đổi trên làm thế nào để chuyển tiếp các gói tin tương tự trong tương lai

3.3.4 Ưu điểm của OpenFlow

Lợi ích mà các doanh nghiệp và nhà khai thác mạng có thể đạt được thông qua kiến trúc SDN trên cơ sở OpenFlow bao gồm:

• Tập trung hóa điều khiển trong môi trường nhiều nhà cung cấp thiết bị: phần mềm điều khiển SDN có thể điều khiển bất kỳ thiết bị mạng nào cho phép OpenFlow từ bất kỳ nhà cung cấp thiết bị nào, bao gồm switch, router và các switch ảo.

• Giảm sự phức tạp thông qua việc tự động hóa: kiến trúc SDN trên cơ sở OpenFlow cung cấp một framework quản lý mạng tự động và linh hoạt. Từ framework này có thể phát triển các công cụ tự động hóa các nhiệm vụ hiện đang được thực hiện bằng tay.

• Tốc độ đổi mới cao hơn: việc áp dụng OpenFlow cho phép các nhà khai thác mạng lập trình lại mạng trong thời gian thực để đạt được các nhu cầu kinh doanh và yêu cầu từ người dùng cụ thể khi có sự thay đổi.

• Gia tăng độ tin cậy và khả năng an ninh của mạng: các nhân viên IT có thể định nghĩa các trạng thái cấu hình và chính sách ở mức cao và áp dụng tới cơ sở hạ tầng thông qua OpenFlow. Kiến trúc SDN trên cơ sở OpenFlow cung cấp điều khiển và tầm nhìn hoàn chỉnh trên mạng, nên có thể đảm bảo điều khiển truy nhập, định hình lưu lượng, QoS, an ninh và các chính sách khác được thực thi nhất quán trên toàn bộ cơ sở hạ tầng mạng không dây và có dây, bao gồm cả các văn phòng chi nhánh, các cơ sở chính và DC.

• Điều khiển mạng chi tiết hơn: mô hình điều khiển trên cơ sở flow của OpenFlow cho phép nhân viên IT áp dụng các chính sách tại mức chi tiết, bao gồm phiên, người dùng, thiết bị và các mức ứng dụng trong một sự trừu tượng hóa cao, tự động điều chỉnh thích hợp. - Tốt hơn với trải nghiệm người dùng: bằng việc tập trung hóa điều khiển mạng và tạo ra trạng thái thông tin có sẵn cho các ứng dụng mức cao hơn, kiến trúc SDN trên cơ sở OpenFlow có thể đáp ứng tốt hơn cho các nhu cầu thay đổi của người dùng.