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

Mô phỏng 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.

Mục đích của hướng dẫn này là giới thiệu người dùng ns-3 mới vào hệ thống một cách có cấu trúc. Đôi khi rất khó để người dùng mới thu thập thông tin cần thiết từ hướng dẫn chi tiết và chuyển đổi thông tin này thành mô phỏng làm việc. Trong hướng dẫn này, chúng tôi sẽ xây dựng một số mô phỏng ví dụ, giới thiệu và giải thích các khái niệm và tính năng chính khi chúng tôi đi.

Khi hướng dẫn mở ra, chúng tôi sẽ giới thiệu tài liệu ns-3 đầy đủ và cung cấp các con trỏ đến mã nguồn cho những người quan tâm đến việc đào sâu hơn vào hoạt động của hệ thống.

Chúng tôi cũng cung cấp một hướng dẫn bắt đầu nhanh chóng cho những người thoải mái lặn ngay trong mà không có quá nhiều tài liệu.

Một vài điểm chính đáng chú ý khi bắt đầu:

* ns-3 là mã nguồn mở, và dự án cố gắng duy trì một môi trường mở để các nhà nghiên cứu đóng góp và chia sẻ phần mềm của họ.
* ns-3 không phải là một phần mở rộng tương thích ngược của Ns2; Đó là một mô phỏng mới. Cả hai mô phỏng đều được viết bằng C ++ nhưng ns-3 là một trình mô phỏng mới không hỗ trợ API ns-2.

Ns-3 đã được phát triển để cung cấp một nền tảng mô phỏng mạng mở, mở rộng, cho nghiên cứu và giáo dục mạng. Tóm lại, Ns-3 cung cấp các mô hình về cách mạng dữ liệu gói hoạt động và hoạt động, và cung cấp một công cụ mô phỏng để người dùng tiến hành các thí nghiệm mô phỏng. Một số lý do để sử dụng ns-3 bao gồm thực hiện các nghiên cứu khó khăn hơn hoặc không thể thực hiện với các hệ thống thực, nghiên cứu hành vi hệ thống trong môi trường được kiểm soát cao, có thể tái tạo và tìm hiểu về cách mạng hoạt động. Người dùng sẽ lưu ý rằng mô hình có sẵn được đặt trong ns-3 tập trung vào việc mô hình hóa cách các giao thức và mạng Internet hoạt động, nhưng ns-3 không giới hạn ở các hệ thống Internet; Một số người dùng đang sử dụng ns-3 để mô hình hóa các hệ thống không dựa trên Internet.

Nhiều công cụ mô phỏng tồn tại cho các nghiên cứu mô phỏng mạng. Dưới đây là một vài tính năng phân biệt của ns-3 trái ngược với các công cụ khác.

* ns-3 được thiết kế như một tập hợp các thư viện có thể được kết hợp với nhau và cũng với các thư viện phần mềm bên ngoài khác. Trong khi một số nền tảng mô phỏng cung cấp cho người dùng một môi trường giao diện người dùng đồ họa tích hợp duy nhất, trong đó tất cả các tác vụ được thực hiện, ns-3 là mô-đun hơn về vấn đề này. Một số họa sĩ hoạt hình bên ngoài và các công cụ phân tích dữ liệu và trực quan hóa có thể được sử dụng với ns-3. Tuy nhiên, người dùng nên mong đợi làm việc tại dòng lệnh và với các công cụ phát triển phần mềm C++ và / hoặc Python.
* ns-3 chủ yếu được sử dụng trên các hệ thống Linux hoặc macOS, mặc dù hỗ trợ tồn tại cho các hệ thống BSD và cũng cho các khung Windows có thể xây dựng mã Linux, chẳng hạn như Windows Subsystem cho Linux hoặc Cygwin. Native Windows Visual Studio hiện không được hỗ trợ mặc dù một nhà phát triển đang làm việc trên hỗ trợ trong tương lai. Người dùng Windows cũng có thể sử dụng máy ảo Linux.
* ns-3 không phải là một sản phẩm phần mềm được hỗ trợ chính thức của bất kỳ công ty nào. Hỗ trợ ns-3 được thực hiện trên cơ sở nỗ lực tốt nhất trên diễn đàn ns-3-users[(ns-3-users@googlegroups. com](mailto:ns-3-users%40googlegroups.com)).

ns-3 là một mô phỏng sự kiện rời rạc thường chạy từ dòng lệnh. Nó được viết trực tiếp bằng C++, không phải bằng ngôn ngữ mô hình cấp cao; các sự kiện mô phỏng chỉ đơn giản là các cuộc gọi chức năng C ++, được tổ chức bởi một người lập lịch trình.

Người dùng ns-3 sẽ có được mã nguồn ns-3 (xem bên dưới), biên dịch nó thành các thư viện được chia sẻ (hoặc tĩnh) và liên kết các thư viện với () chương trình chính mà họ là tác giả. Chương trình chính () là nơi cấu hình kịch bản mô phỏng cụ thể được thực hiện và nơi trình mô phỏng được chạy và dừng lại. Một số chương trình ví dụ được cung cấp, có thể được sửa đổi hoặc sao chép để tạo ra các kịch bản mô phỏng mới. Người dùng cũng thường chỉnh sửa mã thư viện ns-3 (và xây dựng lại các thư viện) để thay đổi hành vi của nó.

ns-3 có các ràng buộc Python tùy chọn để tác giả các chương trình cấu hình kịch bản trong Python (và sử dụng quy trình làm việc dựa trên Python); Sự khởi đầu nhanh chóng này không bao gồm những khía cạnh đó.

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.

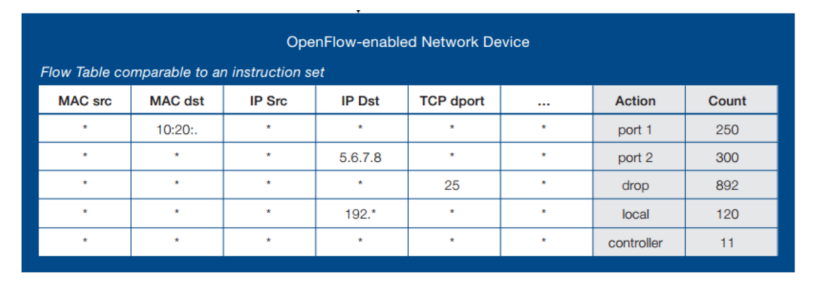
+OpenFlow sử dụng khái niệm “flow” (luồng) để nhận dạng lưu lượng mạng trên cơ sở định nghĩa trước các quy tắc phù hợp (được lập trình sẵn bởi phần mềm điều khiển SDN). Giao thức này còn cho phép định nghĩa các tham số, chẳng hạn như mô hình lưu lượng sử dụng, ứng dụng và tài nguyên cần thiết để xác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

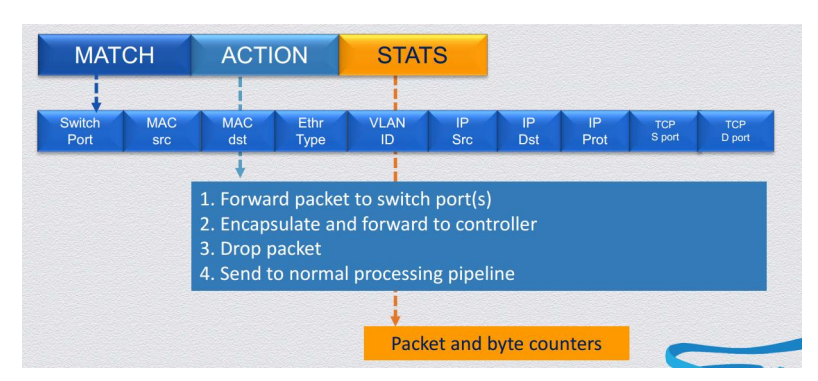
3.3 Cấu tạo và hoạt động của openflow

3.3.1 Cấu tạo

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



Hình 6. Một ví dụ về Flow Table

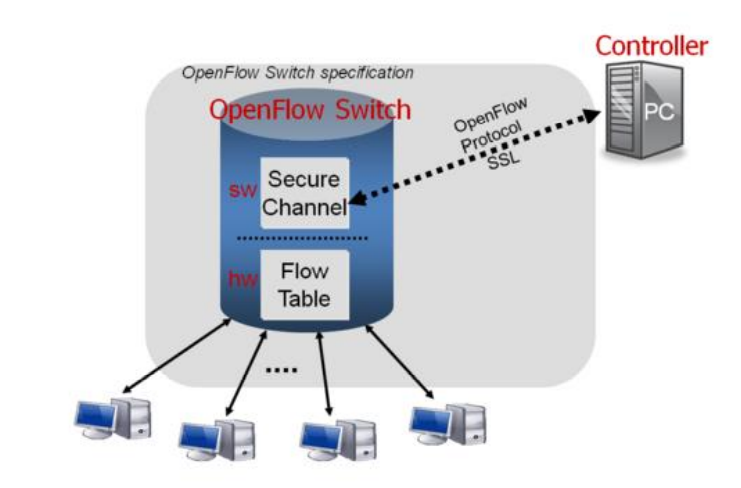
**

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.