

ĐÁNH GIÁ HIỆU NĂNG HỆ THỐNG CÂN BẰNG TẢI CHO MẠNG TRUYỀN THÔNG VÀ MẠNG ĐỊNH NGHĨA BẰNG PHẦN MỀM (SDN)

Vũ Đức Như*, Dương Văn Dân*, Tạ Minh Thanh*, Trần Nam Khánh*

* Đại học kỹ thuật Lê Quý Đôn

Tóm tắt—Ngày nay, sự phát triển của mạng kết nối vạn vật IoT và sự gia tăng của các thiết bị di động kết nối mạng đòi hỏi các máy chủ cung cấp dịch vụ phải đáp ứng khả năng phản hồi số lượng yêu cầu trong thời gian ngắn và đặt ra vấn đề thách thức không nhỏ cho mạng truyền thông và các hệ thống cân bằng tải. Trong bối cảnh đó, sự ra đời của mạng định nghĩa bằng phần mềm (SDN) với sự tập trung quản lý điều khiển và khả năng cân bằng tải tốt hơn phần nào giải quyết được sự bùng nổ số lượng yêu cầu dịch vụ tới các máy chủ. Để làm sáng rõ hơn vấn đề này, nghiên cứu đã triển khai thực nghiệm để đánh giá khả năng cân bằng tải trong mạng truyền thông và mạng SDN sử dụng cùng một kiến trúc mạng, so sánh trên cùng hệ thống cân bằng tải và thuật toán cân bằng tải. Nghiên cứu dựa trên việc tạo ra nhiều luồng đồng thời bằng công cụ Distributed Internet Traffic Generator (D-ITG). Kết quả thu được từ các thí nghiệm cho thấy cân bằng tải trên mạng SDN vượt trội hơn mạng truyền thông trong việc xử lý yêu cầu, thời gian đáp ứng, tính ổn định và tốc độ truyền dữ liệu trong ứng dụng thực tế.

Từ khoá—Mạng định nghĩa bằng phần mềm, SDN, cân bằng tải, hiệu năng mạng

I. GIỚI THIỆU

A. Tổng quan

Hệ thống cân bằng tải là một phần không thể thiếu trong hệ thống mạng hiện nay. Nó đóng vai trò quan trọng trong việc điều phối các luồng yêu cầu của người dùng đến máy chủ. Hệ thống cân bằng tải sẽ chuyển các yêu cầu của người dùng đến các máy chủ sao cho phù hợp với khả năng đáp ứng của từng máy chủ. Do

đó, nó giúp hệ thống mạng hoạt động ổn định hơn, tăng khả năng sẵn sàng phục vụ và tin cậy hơn cũng như tối ưu hóa hiệu suất của hệ thống mạng hơn. Vì vậy, hệ thống cân bằng tải giúp giải quyết bài toán đáp ứng nhu cầu của người dùng về băng thông và khả năng đáp ứng các yêu cầu.

Tuy nhiên, khi số lượng thiết bị tham gia kết nối Internet càng nhiều thì ngày càng gây nên sự phức tạp cho người quản trị mạng trong quá trình vận hành và điều khiển. Cùng với sự thay đổi về mô hình mạng, sự gia tăng của các dịch vụ đám mây và nhu cầu phát triển của các nhà khai thác băng thông dịch vụ đưa ra, yêu cầu cần phải phát triển một giải pháp mới để quản trị mạng hiệu quả là một vấn đề cấp thiết. Mạng định nghĩa bằng phần mềm (SDN) ra đời cung cấp giải pháp giúp nâng cao hiệu quả và giảm chi phí vận hành và quản trị hệ thống mạng. SDN có thể kiểm soát toàn bộ mạng từ một nút trung tâm để tự động hóa quản trị và bảo mật bằng cách thiết lập các chính sách.

Đi sâu vào nghiên cứu khả năng cân bằng tải của mạng SDN, đã có nhiều nghiên cứu của các nhà khoa học có thể kể đến như: Arkar Soe Linn và cộng sự nghiên cứu khả năng cân bằng tải máy chủ trong mạng SDN [6], nghiên cứu về kỹ thuật cân bằng tải trong mạng SDN [7] của Anish Ghosh và cộng sự; Omran M và cộng sự đã nghiên cứu đánh giá hiệu suất mạng cân bằng tải mạng SDN theo từng mô hình thiết kế khác nhau [8].

Nhưng chưa có nghiên cứu nào đánh giá so sánh hiệu năng của hệ thống cân bằng tải cho mạng truyền thông và mạng SDN. Vì vậy, chúng tôi đã xây dựng và thử nghiệm đo hiệu suất cân bằng tải mạng truyền thông và mạng SDN trên cùng một mô hình triển khai và cùng một biện pháp cân bằng tải. Nghiên cứu sẽ tập trung vào việc so sánh khả năng cân bằng tải mạng SDN và mạng truyền thông trong một môi trường thực tế. Chúng tôi sẽ xây dựng các mô hình mạng tương ứng cho cả SDN và mạng truyền thông và triển khai

Contact author: Tạ Minh Thanh

Email: thanhtm@lqdtu.edu.vn

Manuscript received: 12/2023, revised: 2/2024, accepted: 3/2024

HAProxy để thực hiện cân bằng tải trong cả hai môi trường. Sau đó, chúng tôi sẽ thu thập dữ liệu liên quan đến hiệu suất, lưu lượng mạng từ các thí nghiệm thực tế và phân tích kết quả.

B. Đóng góp của bài báo

Những đóng góp của bài báo có thể được trình bày cụ thể như sau:

(1) Trong bài báo này, chúng tôi tập trung nghiên cứu đánh giá khả năng cân bằng tải của mạng truyền thông và mạng định nghĩa bằng phần mềm nhằm đưa ra cho nhà quản trị mạng có lựa chọn phù hợp nhất cho hệ thống mạng của họ khi phát triển những hệ thống dịch vụ thực tế. Khi đó, dựa trên những kết quả đánh giá, nhà quản lý sẽ quyết định được việc đầu tư một hệ thống với thiết bị mạng thực hay phát triển một hệ thống mạng mềm trên nền tảng của những thiết bị sẵn có.

(2) Chúng tôi nghiên cứu, đánh giá so sánh hệ thống cân bằng tải trên mạng truyền thông và mạng định nghĩa bằng phần mềm nhằm đưa ra được giải pháp thực tế cho các ứng dụng mạng. Các kết quả cho thấy việc sử dụng cân bằng tải trên mạng thực tế hay mạng mềm đều có thể hoạt động được hiệu quả trong ứng dụng mạng. Tuy nhiên, về mặt đầu tư và chi phí thì mạng SDN tối ưu hơn do tính mềm dẻo trong cấu hình mạng và phân tải hệ thống.

C. Cấu trúc bài báo

Bài báo được cấu trúc như sau: phần II trình bày về các nghiên cứu liên quan. Trong phần III về cơ sở lý thuyết, chúng tôi trình bày các kiến thức về SDN, so sánh mạng SDN và mạng truyền thông và cân bằng tải. Phần IV về Thử nghiệm và đánh giá, chúng tôi trình bày về mô hình và các biện pháp triển khai thử nghiệm và đánh giá kết quả. Cuối cùng, phần V đưa ra kết luận và một số giới hạn của bài báo.

II. NGHIÊN CỨU LIÊN QUAN

Trong bài báo [6] tác giả Arkar Soe Linn và các cộng sự đã trình bày nghiên cứu của mình về hệ thống cân bằng tải trong mạng định nghĩa bằng phần mềm. Nhóm tác giả xây dựng thí nghiệm của mình trên nền tảng Mininet và POX controller. Mô hình mạng thí nghiệm được đặt ra gồm 16 client và 3 server. Họ thí nghiệm lần lượt trên các thuật toán cân bằng tải như Random, Round-Robin và Weighted Round-Robin. Kết quả qua các lần thí nghiệm cho thấy thuật toán Weighted Round-Robin (với trọng số lần lượt là 1, 3 và 5 cho server 1, 2 và 3) cho kết quả tốt hơn các thuật toán còn lại. Bài báo đã đưa ra được cái nhìn tổng quan về hệ thống cân bằng tải trong mạng định nghĩa bằng phần mềm (SDN). Tuy nhiên, nhóm tác giả chưa có sự so sánh với các nghiên cứu với mạng truyền thông.

Trong bài báo này, nhóm tác giả sử dụng chính SDN controller để xây dựng hệ thống cân bằng tải. Bài báo đã đưa ra được các thí nghiệm để đánh giá cân bằng tải trên mạng SDN, tuy nhiên vẫn chưa đánh giá được mức độ đó so với mạng truyền thông.

Tác giả Anish Ghosh và cộng sự đã trình bày một nghiên cứu về hệ thống cân bằng tải trong bài báo [7]. Bài báo trình bày về kiến trúc của mạng định nghĩa bằng phần mềm, có sự so sánh với mạng truyền thông. Nhóm tác giả còn nêu ra được các lợi ích của mạng định nghĩa bằng phần mềm. Trình bày về một số thuật toán cân bằng tải đang có hiện nay, cách thức hoạt động và ưu nhược điểm của chúng. Việc so sánh thời gian chạy của các thuật toán vô cùng quan trọng, giúp người triển khai mạng có thể đưa ra các phương án tối ưu cho hệ thống mạng mà họ cần. Tuy nhiên, nhóm tác giả chưa xây dựng một mô hình cụ thể mạng nào để đánh giá khả năng cân bằng tải của các thuật toán. Đây là một bài nghiên cứu tổng quan về các công nghệ cân bằng tải trong mạng SDN. Bài báo đưa ra được so sánh với mạng truyền thông và mặt lợi ích của SDN. Tuy nhiên, phần thí nghiệm cụ thể vào một mô hình mạng để đánh giá nhóm tác giả còn thiếu.

Năm 2022, bài báo của nhóm tác giả Omran M. A. Alssaheli có tựa đề [8] được công bố. Bài báo trình bày các vấn đề tồn tại trong mạng hiện nay, và chỉ ra xu hướng mạng trong tương lai là mạng định nghĩa bằng phần mềm (SDN). Đánh giá cụ thể các ưu nhược điểm của mạng truyền thông và chỉ ra ưu điểm của mạng định nghĩa bằng phần mềm khắc phục các nhược điểm đó. Họ xây dựng mô hình mạng dựa trên Mininet và thí nghiệm một số kịch bản mạng. Tác giả sử dụng 3 thông số là thông lượng, độ trễ và jitter. Đánh giá trên 3 kịch bản A, B và C với số lượng client và server tăng dần. Họ thực hiện thí nghiệm và tiến hành đo với số lượng yêu cầu tăng dần từ 10 đến 100 với bước nhảy là 10. Kết quả được nhóm tác giả trình bày cho thấy kịch bản A cho thông lượng cao hơn kịch bản B và C. Kịch bản B và C cho jitter thấp và kịch bản C tạo ra độ trễ thấp nhất. Nhưng, kiến trúc mạng duy nhất mà nhóm nghiên cứu thí nghiệm là mô hình mạng star. Điều này chưa thực sự khách quan trong thực tế, vì sẽ có nhiều mô hình mạng khác nhau và nhóm tác giả cũng chưa có sự so sánh với mạng truyền thông trong cùng mô hình.

Một nghiên cứu tổng quan về cân bằng tải trong mạng SDN được nhóm tác giả Prabhjot Kaur trình bày trong bài báo [14]. Một cách tổng quan nhất, nhóm tác giả trình bày về cách thức hoạt động của hệ thống cân bằng tải và ứng dụng của nó trong mạng định nghĩa bằng phần mềm (SDN). Họ xây dựng bảng và so sánh các phương pháp khác nhau, chỉ ra các giới hạn trong các nghiên cứu trước đây. Bài báo đưa ra được cái nhìn tổng quan về các phương pháp, cũng như các nghiên cứu của các tác giả trước.

Một nghiên cứu khảo sát về cân bằng tải trong mạng

định nghĩa bằng phần mềm được thực hiện bởi nhóm nghiên cứu Thabo có tên [15]. Nhóm nghiên cứu đã trình bày từng thành phần trong mạng định nghĩa bằng phần mềm và công nghệ OpenFlow cũng như kiến trúc của OpenFlow. Trình bày tổng quát về các phương pháp, thuật toán cân bằng tải trước đó, đưa ra hiệu năng của từng phương pháp và kết luận về phương pháp đó. Hơn nữa, nhóm tác giả đưa ra được các công nghệ mô phỏng hệ thống mạng định nghĩa bằng phần mềm như Mininet, OMNET++...

Các nghiên cứu trước đó đã có các công trình nghiên cứu quan trọng về vấn đề cân bằng tải trong mạng định nghĩa bằng phần mềm. Tuy nhiên, vẫn còn chưa được đầy đủ các mô hình mạng, các cách thức khác nhau để so sánh giữa các thuật toán. Hơn nữa, chưa có công trình nghiên cứu nào đầy đủ đưa ra so sánh hệ thống cân bằng tải của mạng định nghĩa bằng phần mềm(SDN) và mạng truyền thống. Trong nghiên cứu này, chúng tôi nghiên cứu và thử nghiệm mô hình mạng SDN và mạng truyền thống trên các mô hình mạng khác nhau, các thông số khác nhau để đưa ra các đánh giá cho hệ thống. Chúng tôi sử dụng Ryu Controller để điều khiển lưu lượng trong mạng.

III. CƠ SỞ LÝ THUYẾT

Trong phần này, chúng tôi trình bày về hệ thống cân bằng tải, các nét chính về mạng SDN và so sánh mạng truyền thống với mạng SDN về cấu trúc và tính năng.

A. Cân bằng tải

Cân bằng tải là một kỹ thuật phân chia khối lượng công việc thành nhiều luồng khác nhau để tránh quá tải cho bất kỳ luồng nào[9]. Tối đa hóa thông lượng, giảm thiểu thời gian phản hồi và tối ưu hóa lưu lượng là một số mục tiêu cân bằng tải[10].

Hiện nay, một số thuật toán cân bằng tải đang được ưa thích sử dụng có thể kể đến như[7]: Round Robin, Weighted Round Robin, Least Connection, Weighted Response Time, Source IP Hash. Trong đó, thuật toán Round Robin thường được ưu tiên sử dụng trong các mạng hiện nay.

Một số các công cụ cân bằng tải hay được sử dụng trong hệ thống mạng có thể kể đến là: HAProxy, Nginx đang tỏ ra ưu điểm vượt trội trong các hệ thống mạng truyền thống.

B. Mạng định nghĩa bằng phần mềm - SDN

Mạng định nghĩa bằng phần mềm (SDN - Software Defined Networking) hay mạng điều khiển bằng phần mềm là một kiến trúc mạng mới, linh hoạt, dễ quản lý, tiết kiệm chi phí và dễ thích nghi và phù hợp với yêu cầu ngày càng tăng số lượng người dùng và nhu cầu băng thông cao của các ứng dụng ngày nay. Kiến

Bảng 1
SO SÁNH MẠNG TRUYỀN THỐNG VÀ MẠNG SDN [7]

Mạng truyền thống	Mạng SDN
Mạng tĩnh và không linh hoạt	Có thể lập trình động và linh hoạt
Có mặt phẳng điều khiển phân tán	Có mặt phẳng điều khiển tập trung
Hoạt động bằng cách sử dụng các giao thức	Sử dụng API để định cấu hình theo nhu cầu
Được tạo thành từ các thiết bị phần cứng	Được tạo bằng phần mềm mở

trúc này phân tách các chức năng chuyển tiếp và điều khiển mạng. Cho phép việc điều khiển mạng có thể lập trình trực tiếp và cơ sở hạ tầng độc lập với các ứng dụng và dịch vụ mạng [1].

Mặt phẳng điều khiển là bộ não của mạng SDN chịu trách nhiệm quản lý mạng một cách tập trung. Mặt phẳng dữ liệu là cơ sở hạ tầng bên dưới như Switch và Router[11] [12]. Như được thể hiện trong Hình 2, kiến trúc SDN mô tả với ba tầng chính: Application layer, Control layer, Data layer. Thông qua các giao diện ứng dụng được cung cấp, người dùng đưa ra các yêu cầu đối với người quản trị ví dụ như: điều chỉnh các tham số băng thông, độ trễ, định tuyến.... Các yêu cầu này được gửi đến tầng điều khiển qua các Northbound API. Tầng điều khiển, sau khi nhận các yêu cầu thì ra quyết định thực thi các yêu cầu đó và gửi chúng tới tầng dữ liệu thông qua các Southbound API. Các yêu cầu này được lớp dữ liệu thực thi. Quá trình hoạt động tầng điều khiển luôn giám sát và thu thập thông tin trong quá trình hoạt động, phục vụ quá trình quản lý và ra các quyết định.

C. Mạng SDN và mạng truyền thống

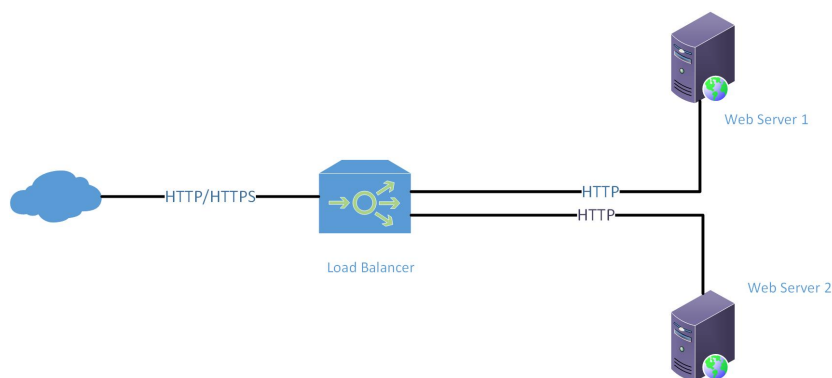
Mạng SDN đang mang lại sự ưu việt hơn mạng truyền thống bởi một số sự khác nhau cơ bản được thể hiện ở Bảng 1.

- Mạng truyền thống thường được cấu hình trước khi triển khai và không dễ dàng thay đổi khi đang hoạt động. Trong khi mạng SDN lại có thể lập trình trước khi triển khai và trong quá trình mạng đang hoạt động.

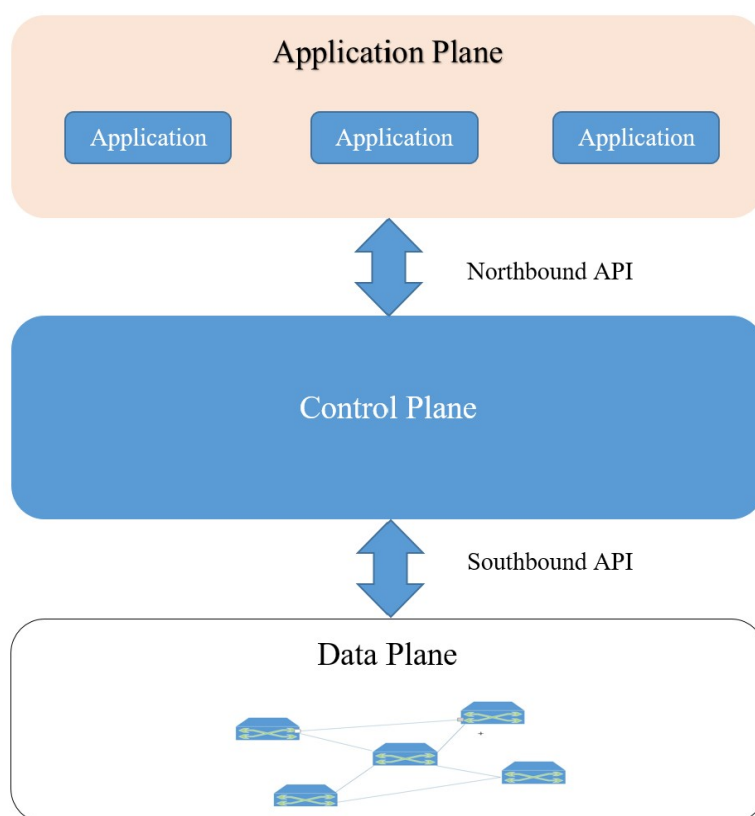
- Quá trình điều khiển hoạt động của mạng truyền thống như chuyển tiếp, định tuyến được thực hiện tại các thiết bị riêng lẻ. Đối với mạng SDN quá trình đó được thực hiện tập trung tại bộ điều khiển của mạng.

- Mạng truyền thống hoạt động bằng các giao thức như TCP/IP, OSPF, ICMP... Còn mạng SDN sử dụng các API (Southbound API, Northbound API) để cấu hình và kiểm soát mạng.

- Mạng truyền thống dựa vào các thiết bị phần cứng như: Switch, Router, tường lửa..., để thực hiện các chức năng. Thay vào đó, mạng SDN xây dựng, thực thi và quản lý các chức năng trên bộ điều khiển tập trung được xây dựng bằng phần mềm.



Hình 1. Mô hình cân bằng tải



Hình 2. Cấu trúc mạng SDN

IV. THỬ NGHIỆM VÀ ĐÁNH GIÁ

Trong phần này chúng tôi tiến hành xây dựng môi trường thí nghiệm hệ thống cân bằng tải mạng SDN trên môi trường ảo hóa sử dụng Containernet với bộ điều khiển Ryu Controller. Với mạng truyền thống, chúng tôi thiết lập một mạng theo sơ đồ kết nối giống với mạng SDN sử dụng máy PC và Switch.

A. Thiết lập môi trường thí nghiệm

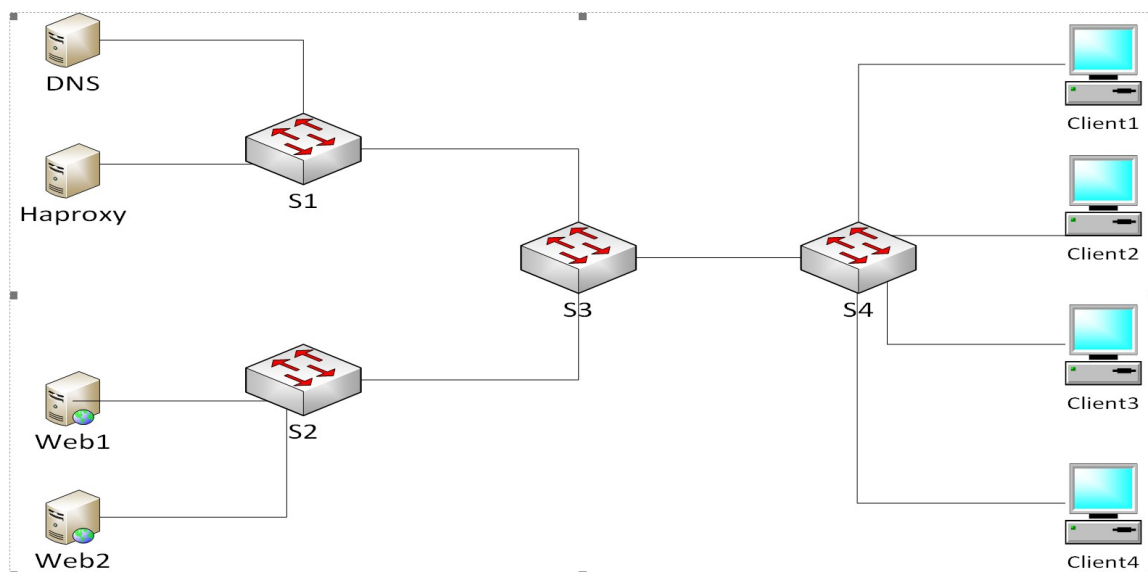
Để đánh giá được khả năng cân bằng tải giữa mạng SDN và mạng truyền thống chúng tôi xây dựng môi trường thực nghiệm như hình 2 để so sánh các thông số một cách chính xác nhất. Một mạng gồm 1 máy chủ

DNS làm nhiệm vụ quản lý và phân giải tên miền, một máy chủ HAProxy làm nhiệm vụ cân bằng tải cho 2 webserver (Web1, Web2) và sử dụng 4 máy Client để kiểm tra hệ thống.

Môi trường thí nghiệm mạng SDN chúng tôi sử dụng bao gồm những công cụ sau:

Trình quản lý ảo hoá: OracleVM Virtualbox, là nền tảng mã nguồn mở cung cấp khả năng tạo, quản lý, thực thi các máy ảo. Công cụ này có hiệu năng cao và nhiều tích năng hỗ trợ tốt cho việc quản lý và sử dụng các máy ảo trên nhiều hệ điều hành khác nhau.

Công cụ mô phỏng mạng Mininet là một công cụ mô phỏng được viết bằng Python và C cho phép chạy một số máy chủ ảo, bộ điều khiển, công tắc và liên



Hình 3. Mô hình triển khai cân bằng tải bằng HAProxy

kết. Nó được sử dụng rộng rãi trong môi trường mô phỏng mạng mã nguồn mở. Mininet sử dụng ảo hóa dựa trên vùng chứa để làm cho một hệ thống duy nhất hoạt động như một mạng hoàn chỉnh. Hơn nữa, nó là một công cụ mạng đơn giản, mạnh mẽ và phù hợp để phát triển và thử nghiệm các ứng dụng dựa trên Open Flow. Nó là thành phần tích hợp của Open vSwitch và công tắc có khả năng OpenFlow. Mininet có thể tạo cấu trúc liên kết mạng phức tạp cho mục đích thử nghiệm mà không cần định cấu hình mạng vật lý. Công cụ này có thể hỗ trợ cấu trúc liên kết tùy chỉnh. Nó hỗ trợ API Python đơn giản và có thể mở rộng để tạo và thử nghiệm mạng [4].

Containernet là một nhánh của trình giả lập mạng Mininet nổi tiếng và cho phép sử dụng các bộ chứa Docker làm máy chủ trong cấu trúc liên kết mạng mô phỏng [5].

Bộ điều khiển Ryu: Đây là một khung mạng định nghĩa bằng phần mềm dựa trên thành phần. Ryu cung cấp các thành phần phần mềm với API được xác định rõ giúp các nhà phát triển dễ dàng tạo các ứng dụng điều khiển và quản lý mạng mới. Ryu hỗ trợ nhiều giao thức khác nhau để quản lý thiết bị mạng, chẳng hạn như OpenFlow, Netconf, OF-config, v.v. Về OpenFlow, Ryu hỗ trợ đầy đủ các Tiện ích mở rộng 1.0, 1.2, 1.3, 1.4, 1.5 và Nicira. Tất cả các mã đều có sẵn miễn phí theo giấy phép Apache 2.0. Ryu được viết hoàn toàn bằng Python 002

Đối với mạng truyền thông chúng tôi sử dụng các máy tính có cùng cấu hình cài đặt Hệ điều hành Ubuntu, các máy tính được kết nối với nhau bởi các switch Cisco (2960L). Các Switch này để ở chế độ mặc định và chỉ có nhiệm vụ chuyển tiếp các gói tin.

Cân bằng tải: trong thí nghiệm chúng tôi sử dụng

công cụ cân bằng tải HAProxy [3] áp dụng thuật toán Round Robin tuần tự, 2 máy chủ Web Server sẽ luân phiên nhau xử lý các yêu cầu.

B. Thu thập dữ liệu

Chúng tôi sử dụng công cụ đo lường là D-ITG để đánh giá hiệu suất và đáp ứng của máy chủ trong hai mạng khác nhau. D-ITG được sử dụng để tạo các luồng yêu cầu mô phỏng thực tế từ nhiều nguồn đến máy chủ, đánh giá hiệu suất của máy chủ trong môi trường tải cao [13] .

Bên máy thu(máy Haproxy): ITGRecv - 1 /tmp/hap.log các dữ liệu thu thập được lưu trữ trong file log có tên hap.log. Bên máy gửi (máy client): ITGSend - T TCP -a www.hvktqs.com -c 5000 -C 10 -t 25000 Trong đó:

-T TCP: Xác định loại giao thức truyền thông là TCP.

-a address: Xác định địa chỉ đích của máy chủ hoặc thiết bị mà gói tin sẽ được gửi tới.

-c 5000: Xác định số lượng gói tin sẽ được gửi đi

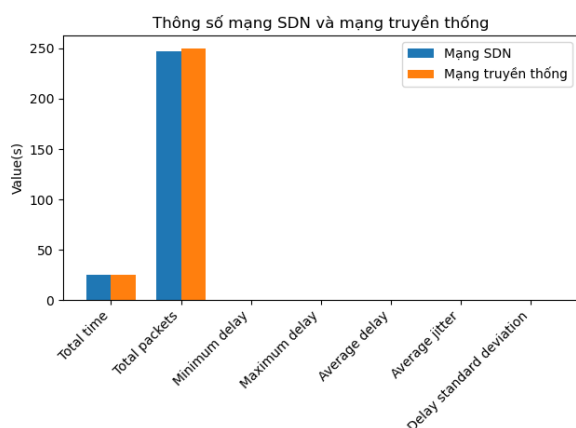
-C 10: Xác định số lượng kết nối sẽ được thiết lập đồng thời

-t 25000: Xác định thời gian (đơn vị là mili giây) mà gói tin sẽ được gửi đi trong mỗi kết nối.

Chúng tôi gửi 5000 gói tin sử dụng giao thức TCP tới địa chỉ website www.hvktqs.com với 10 kết nối đồng thời trong thời gian 25000 ms.

C. Kết quả

Thông số trung bình của 10 lần đo lưu lượng và hiệu suất của mạng sử dụng công cụ D-ITG được thể hiện trong Bảng 2.



Hình 4. Kết quả đo mạng SDN và mạng truyền thống

Bảng II
KẾT QUẢ ĐO BẰNG CÔNG CỤ D-ITG

Thông số	Mạng SDN	Mạng truyền thống
Total time (s)	24.9497376	24.982123
Total packets (s)	247	250
Minimum delay (s)	0.000085	0.342692
Maximum delay (s)	0.0116792	0.362653
Average delay (s)	0.00029902	0.343829
Average jitter (s)	0.00002386	0.000571
Delay standard deviation (s)	0.0008627	0.001318
Bytes received (s)	1157500	1150000
Average bitrate Kbit/s	371.152308	40.028624
Average packet rate (pkt/s)	9.2777077	10.007156

Từ kết quả kiểm tra hiệu suất mạng và khả năng đáp ứng yêu cầu của máy chủ bằng công cụ D-ITG bảng 2 ta nhận thấy rằng:

Thời gian hoàn thành tất cả các kết nối và yêu cầu (Total time): của mạng SDN là tốt hơn mạng truyền thống. Điều này có thể cho thấy mạng SDN có hiệu suất tốt hơn trong việc xử lý các yêu cầu và kết nối.

Độ trễ tối thiểu giữa việc gửi yêu cầu và nhận phản hồi (Minimum delay): Mạng SDN có độ trễ tối thiểu thấp hơn nhiều so với mạng truyền thống (0.000101 giây so với 0.342692 giây). Điều này cho thấy mạng SDN có khả năng xử lý nhanh hơn trong việc đáp ứng các yêu cầu.

Độ trễ tối đa giữa việc gửi yêu cầu và nhận phản hồi (Maximum delay): Mạng SDN có độ trễ tối đa thấp hơn nhiều so với mạng truyền thống, nhanh hơn khoảng 72 lần (0.005258 giây so với 0.362653 giây). Điều này cho thấy mạng SDN có khả năng xử lý ổn định hơn với các yêu cầu có độ trễ lớn.

Độ trễ trung bình giữa việc gửi yêu cầu và nhận phản hồi (Average delay): Mạng SDN có độ trễ trung bình thấp hơn mạng truyền thống (0.000161 giây so với 0.343829 giây). Điều này cho thấy mạng SDN có thời gian đáp ứng nhanh hơn đối với các yêu cầu.

Độ biến động trung bình của độ trễ giữa các lần gửi

yêu cầu và nhận phản hồi (Average jitter: Mạng SDN có độ jitter thấp hơn mạng truyền thống. Điều này cho thấy mạng SDN có sự ổn định cao hơn trong việc đáp ứng các yêu cầu.

Độ lệch chuẩn của độ trễ giữa các lần gửi yêu cầu và nhận phản hồi (Delay standard deviation): Mạng SDN có độ lệch chuẩn của độ trễ thấp hơn so với mạng truyền thống (0.000333 giây so với 0.001318 giây). Điều này cho thấy mạng SDN có sự ổn định cao hơn trong việc đáp ứng các yêu cầu.

Tổng số byte dữ liệu nhận được trong suốt thời gian đo lường (Bytes received): Mạng truyền thống có lượng dữ liệu nhận được lớn hơn gấp 10 lần so với mạng SDN, cho thấy mạng truyền thống có khả năng truyền dữ liệu lớn hơn.

Tốc độ trung bình của dữ liệu nhận được trong đơn vị Kilobit/giây (Average bitrate). Mạng truyền thống có tốc độ trung bình thấp hơn nhiều so với mạng SDN, cho thấy mạng SDN có khả năng truyền dữ liệu nhanh hơn.

Tốc độ trung bình của gói tin được gửi và nhận trong đơn vị gói tin/giây (Average packet rate): Cả hai mạng đều có tốc độ gửi và nhận gói tin tương đối gần nhau.

V. KẾT LUẬN

Kết quả nghiên cứu cho thấy mạng SDN sử dụng cân bằng tải Haproxy với thuật toán Round Robin vượt trội hơn mạng truyền thống trong việc xử lý yêu cầu, có thời gian đáp ứng nhanh hơn, ổn định cao hơn và tốc độ truyền dữ liệu nhanh hơn. Điều này cho thấy mạng SDN đang mang lại cho chúng ta những tính năng ưu việt hơn nhờ kiến trúc mạng linh hoạt, khả năng quản lý kiểm soát trung tâm tốt hơn mạng truyền thống.

Tuy nhiên, để có một cách nhìn và nhận xét một cách chính xác nhất, chúng tôi sẽ tiếp tục thử nghiệm so sánh khả năng cân bằng tải giữa mạng SDN và mạng truyền thống với những cấu trúc mạng khác nhau phức tạp hơn và sát với thực tế hoạt động của mạng.

Trong thời gian tới, chúng tôi dự định phát triển và thí nghiệm trên một cơ sở hạ tầng thật, để đánh giá mức độ hiệu quả của hệ thống cân bằng tải SDN với các dự án thực tế. Trước hết có thể là mô hình mạng của một công ty nhỏ hoặc mạng máy tính của một phòng, ban nhỏ. Qua đó, chúng tôi sẽ đánh giá khả năng cũng như mức độ đáp ứng của hệ thống cân bằng tải. Dựa trên đó, chúng tôi cũng có thể đánh giá khả năng bảo mật, và xây dựng phương hướng bảo mật cho mạng SDN.

TÀI LIỆU

- [1] <https://opennetworking.org/sdn-definition/>. [04/4/2024]
- [2] <https://ryu.readthedocs.io/en/latest/index.html> [04/4/2024]
- [3] <https://www.haproxy.org/> [04/4/2024]
- [4] <https://github.com/mininet/mininet/wiki/Introduction-to-Mininet/> [04/4/2024]
- [5] <https://github.com/containernet/containernet/> [04/4/2024]
- [6] Arkar Soe Linn, Su Hlaing Win, Su Thawda Win, Server Load Balancing in Software Defined Networking, National Journal of Parallel and Soft Computing, Volume 01, Issue 01, March-2019.

- [7] Anish Ghosh, Mrs. T. Manoranjitham, A study on load balancing techniques in SDN, International Journal of Engineering Technology, 7 (2.4) (2018) 174-177
- [8] Omran M. A. Alssaheli, Z. Zainal Abidin, N. A. Zakaria, Z. Abal Abas, Software Defined Network based Load Balancing for Network Performance Evaluation, (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 13, No. 4, 2022.
- [9] Bilal Ashfaq Ahmed, Norris Syed Abdullah, Mohd Foad Rohani, Saima Waseem, "Fractional Political Optimizer-Based Switch Migration in Software-Defined WAN for Load Balancing with Deep Q Network", Cybernetics and Systems, pp.1, 2022.
- [10] Xiaodan Wang, "Rural financial service promotion mechanism based on data center network", Physical Communication, pp.101970, 2022.
- [11] Sahar Abdollahi, Arash Deldari, Hamid Asadi, AhmadReza Montazerolghaem, Sayyed Majid Mazinani, "Flow-Aware Forwarding in SDN Datacenters Using a Knapsack-PSO-Based Solution", IEEE Transactions on Network and Service Management, vol.18, no.3, pp.2902-2914, 2021.
- [12] B Rukmini Bhat, N S Sneha, Keerthana Bhat, Chaithra C Kamath, Chaitha Naik, "Improving the Efficiency of Software Defined Network through Load Balancing Algorithms", 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), pp.124-131, 2021.
- [13] S. Avallone S. Guadagno D. Emma A. Pescap' G. Ventre; D-ITG Distributed Internet Traffic Generator; 1st International Conference on Quantitative Evaluation of Systems (QEST 2004), 27-30 September 2004, Enschede, The Netherlands
- [14] Chahal, Jasmeen Bhandari, Abhinav. (2019). Load Balancing in Software Defined Networking: A Review. Asian Journal of Computer Science and Technology. 7. 10.51983/ajcst-2018.7.2.1859.
- [15] Semong, Thabo Maupong, Thabiso Anokye, Stephen. (2020). Intelligent Load Balancing Techniques in Software Defined Networks: A Survey. Electronics. 9. 10.3390/electronics9071091.

Evaluating Load Balancing Performance for Traditional and Software-Defined Networking Systems

Abstract—Today, the development of the Internet of Things (IoT) and the proliferation of mobile devices connecting to networks require service-providing servers to respond to a large volume of requests in a short time, posing significant challenges for traditional networks and load balancing systems. In this context, the emergence of Software-Defined Networking (SDN) with centralized management and improved load balancing capabilities somewhat addresses the surge in service requests to servers. To further elucidate this issue, a study conducted experiments to evaluate load balancing capabilities in both traditional networks and SDN networks using the same network architecture, comparing the same load balancing system and algorithm. The study involved generating multiple concurrent streams using the Distributed Internet Traffic Generator (D-ITG) tool. Results from the experiments indicate that load balancing in SDN networks outperforms traditional networks in request processing, response time, stability, and data transmission speed in real-world applications.

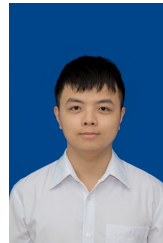
Keywords— Software-Defined Network, SDN, Load Balancing, Network Performance.



Duc Nhu Vu is a Master student of Le Quy Don Technical University in Vietnam. His research interests lie in the area of network architecture, and computer vision.
E-mail: ducnhu.pro@gmail.com



Van Dan Duong is a cadet of Le Quy Don Technical University in Vietnam. His research interests lie in the area of deep learning, steganography, and computer vision.
E-mail: duongvandan2806@gmail.com



Nam Khanh Tran obtained an impressive IT engineering degree from Le Quy Don Technical University in Vietnam in 2020. He has since been contributing to the academic community as a lecturer at the same university, specializing in the research fields of cybersecurity, technology network, and image processing.
E-mail: khanhtn107195@lqdtu.edu.vn



Minh Thanh Ta is currently an associate professor and vice dean of Institute of Information and Communication Technology in Le Quy Don Technical University, Vietnam. He is also a Postdoctoral Fellow of the Department of Mathematical and Computing Sciences at Tokyo Institute of Technology. He received his B.S. and M.S in Computer Science from National Defense Academy, Japan, in 2005 and 2008 and his Ph.D. from Tokyo Institute of Technology, Japan, in 2015, respectively. He is a member of IPSJ Japan and IEEE. His research interests lie in the area of watermarking, network security, and computer vision.
E-mail: thanhmt@lqdtu.edu.vn