

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG



ĐỒ ÁN

TỐT NGHIỆP ĐẠI HỌC

Đề tài:

**XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG
ẢO HÓA (SFC) TRÊN NỀN TẢNG OPENSTACK**

Sinh viên thực hiện: NGUYỄN THỊ THANH TÂM
LỚP ĐTTT 06 – K58

Giảng viên hướng dẫn: TS. NGUYỄN XUÂN DŨNG

Hà Nội, 6-2018

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN ĐIỆN TỬ - VIỄN THÔNG



ĐỒ ÁN

TỐT NGHIỆP ĐẠI HỌC

Đề tài:

XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG ẢO HÓA (SFC) TRÊN NỀN TẢNG OPENSTACK

Sinh viên thực hiện: NGUYỄN THỊ THANH TÂM
LỚP ĐTTT 06 – K58

Giảng viên hướng dẫn: TS. NGUYỄN XUÂN DŨNG

Cán bộ phản biện:

Hà Nội, 6-2018

Đánh giá quyền đồ án tốt nghiệp

(Dùng cho giảng viên hướng dẫn)

Giảng viên đánh giá: TS. Nguyễn Xuân Dũng

Họ và tên Sinh viên: Nguyễn Thị Thanh Tâm MSSV: 20133430

Tên đồ án: Xây Dựng Chuỗi Chức Năng Mạng Ảo Hóa (SFC) Trên Nền Tảng OpenStack

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:
Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)					
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án	1	2	3	4 5
2	Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế)	1	2	3	4 5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề	1	2	3	4 5
4	Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được	1	2	3	4 5
Có khả năng phân tích và đánh giá kết quả (15)					
5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống	1	2	3	4 5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng.	1	2	3	4 5
7	Trong phần kết luận, em chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai.	1	2	3	4 5
Kỹ năng viết (10)					
8	Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định	1	2	3	4 5
9	Kỹ năng viết xuất sắc (cấu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.)	1	2	3	4 5
Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)					
10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế	5			
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest.	2			
10c	Không có thành tích về nghiên cứu khoa học	0			
Điểm tổng					/50
Điểm tổng quy đổi về thang 10					

***Nhận xét thêm của Thầy/Cô** (giảng viên hướng dẫn nhận xét về thái độ và tinh thần làm việc của sinh viên)*

.....

.....

.....

.....

.....

.....

Ngày: / /201

Người nhận xét

(Ký và ghi rõ họ tên)

Đánh giá quyền đồ án tốt nghiệp

(Dùng cho cán bộ phản biện)

Giảng viên đánh giá:

Họ và tên Sinh viên: Nguyễn Thị Thanh Tâm MSSV: 20133430

Tên đồ án: Xây Dựng Chuỗi Chức Năng Mạng Ảo Hóa (SFC) Trên Nền Tảng OpenStack.

Chọn các mức điểm phù hợp cho sinh viên trình bày theo các tiêu chí dưới đây:

Rất kém (1); Kém (2); Đạt (3); Giỏi (4); Xuất sắc (5)

Có sự kết hợp giữa lý thuyết và thực hành (20)					
1	Nêu rõ tính cấp thiết và quan trọng của đề tài, các vấn đề và các giả thuyết (bao gồm mục đích và tính phù hợp) cũng như phạm vi ứng dụng của đồ án	1	2	3	4 5
2	Cập nhật kết quả nghiên cứu gần đây nhất (trong nước/quốc tế)	1	2	3	4 5
3	Nêu rõ và chi tiết phương pháp nghiên cứu/giải quyết vấn đề	1	2	3	4 5
4	Có kết quả mô phỏng/thực nghiệm và trình bày rõ ràng kết quả đạt được	1	2	3	4 5
Có khả năng phân tích và đánh giá kết quả (15)					
5	Kế hoạch làm việc rõ ràng bao gồm mục tiêu và phương pháp thực hiện dựa trên kết quả nghiên cứu lý thuyết một cách có hệ thống	1	2	3	4 5
6	Kết quả được trình bày một cách logic và dễ hiểu, tất cả kết quả đều được phân tích và đánh giá thỏa đáng.	1	2	3	4 5
7	Trong phần kết luận, em chỉ rõ sự khác biệt (nếu có) giữa kết quả đạt được và mục tiêu ban đầu đề ra đồng thời cung cấp lập luận để đề xuất hướng giải quyết có thể thực hiện trong tương lai.	1	2	3	4 5
Kỹ năng viết (10)					
8	Đồ án trình bày đúng mẫu quy định với cấu trúc các chương logic và đẹp mắt (bảng biểu, hình ảnh rõ ràng, có tiêu đề, được đánh số thứ tự và được giải thích hay đề cập đến trong đồ án, có căn lề, dấu cách sau dấu chấm, dấu phẩy v.v), có mở đầu chương và kết luận chương, có liệt kê tài liệu tham khảo và có trích dẫn đúng quy định	1	2	3	4 5
9	Kỹ năng viết xuất sắc (câu trúc câu chuẩn, văn phong khoa học, lập luận logic và có cơ sở, từ vựng sử dụng phù hợp v.v.)	1	2	3	4 5
Thành tựu nghiên cứu khoa học (5) (chọn 1 trong 3 trường hợp)					
10a	Có bài báo khoa học được đăng hoặc chấp nhận đăng/đạt giải SVNC khoa học giải 3 cấp Viện trở lên/các giải thưởng khoa học (quốc tế/trong nước) từ giải 3 trở lên/ Có đăng ký bằng phát minh sáng chế	5			
10b	Được báo cáo tại hội đồng cấp Viện trong hội nghị sinh viên nghiên cứu khoa học nhưng không đạt giải từ giải 3 trở lên/Đạt giải khuyến khích trong các kỳ thi quốc gia và quốc tế khác về chuyên ngành như TI contest.	2			
10c	Không có thành tích về nghiên cứu khoa học	0			
Điểm tổng					/50
Điểm tổng quy đổi về thang 10					

Nhận xét thêm của Thầy/Cô

.....

.....

.....

.....

.....

.....

Ngày: / /201

Người nhận xét
(Ký và ghi rõ họ tên)

LỜI NÓI ĐẦU

Với sự phát triển nhanh chóng của Internet và số lượng người dùng, việc phát triển thêm các dịch vụ giá trị gia tăng (VAS - Value-Added Services) cho các khách hàng của mình dần trở thành ưu tiên hàng đầu của các nhà cung cấp dịch vụ. Để cung cấp một dịch vụ đầu cuối hoàn chỉnh cho khách hàng, các nhà cung cấp dịch vụ mạng viễn thông và công nghệ thông tin phải thiết lập và cấu hình hệ thống các dịch vụ mạng, chức năng mạng phù hợp để đảm bảo dịch vụ tới người dùng hoạt động ổn định, tin cậy. Cùng sự ra đời của ảo hóa chức năng mạng (NFV), các thành phần dịch vụ mạng được cung cấp trên nhiều đám mây cho mục đích nâng cao hiệu suất và cân bằng tải. Việc kết nối các chức năng mạng này để tạo thành một dịch vụ mạng từ đầu đến cuối hoàn chỉnh là nhiệm vụ phức tạp, tốn thời gian và tốn kém. Chuỗi chức năng dịch vụ (SFC) là một cơ chế cho phép các chức năng dịch vụ khác nhau được kết nối với nhau để tạo thành một dịch vụ cho phép các nhà khai thác hưởng lợi từ cơ sở hạ tầng đã được ảo hóa. SFC là một công cụ hỗ trợ cho NFV, cung cấp sự thay thế linh hoạt và kinh tế cho môi trường tĩnh ngày nay cho các nhà cung cấp dịch vụ đám mây (CSP), Nhà cung cấp dịch vụ ứng dụng (ASP) và Nhà cung cấp dịch vụ Internet (ISP).

Vì vậy, sau quá trình học tập tại trường, đồng thời nhận được sự chỉ dẫn tận tình của thầy Nguyễn Xuân Dũng cũng như các thầy cô trong Viện Điện tử Viễn thông, em xin chọn đề tài: “Xây dựng chuỗi chức năng mạng ảo hóa trên môi trường OpenStack” để làm đồ án tốt nghiệp.

Em xin chân thành cảm ơn TS. Nguyễn Xuân Dũng đã tận tình giúp đỡ, hướng dẫn, tạo điều kiện để em thực hiện đề tài đồ án tốt nghiệp. Ngoài ra em cũng xin chân thành cảm ơn thầy Nguyễn Hữu Thanh cùng các thành viên Future Internet Lab đặc biệt là các bạn trong nhóm Network Function Virtualization đã chỉ bảo, giúp đỡ, chia sẻ trong suốt thời gian qua.

TÓM TẮT ĐỒ ÁN

Chuỗi chức năng dịch vụ (Service Function Chaining - SFC) là công nghệ quy định các chính sách chuyển tiếp lưu lượng tách biệt trong vùng chuỗi dịch vụ. Chuỗi các chức năng dịch vụ mạng SFC định nghĩa và tạo ra một chuỗi các dịch vụ mạng theo thứ tự mong muốn và điều khiển lưu lượng truy cập thông qua chúng. Hiện nay, kiến trúc mạng của các nhà khai thác mạng được phân bổ nhiều thiết bị phần cứng độc quyền. Ứng với mỗi loại dịch vụ, mỗi chức năng mạng lại có các thiết bị phần cứng chuyên trách đảm nhận. Các thiết bị này còn đang có một số hạn chế như: chi phí thiết bị cao, khó quản lý tập trung, độ tương thích thấp với hệ thống của nhiều hãng khác dẫn tới doanh nghiệp phải phụ thuộc vào một số hãng phần cứng nhất định. Do đó, để cung cấp các dịch vụ mạng khác nhau một cách linh hoạt cho khách hàng vẫn còn là một vấn đề gây đau đầu cho các nhà cung cấp mạng.

Trong đồ án này, em trình bày về công nghệ ảo hóa chức năng mạng (NFV) và các lợi ích của chuỗi dịch vụ mạng (SFC), các trường hợp sử dụng hai công nghệ đó đem lại lợi ích như thế nào cho các nhà cung cấp dịch vụ mạng và khách hàng. Đồng thời, em cũng xây dựng hệ thống chuỗi chức năng mạng sử dụng các chức năng mạng ảo hóa trên nền tảng OpenStack với quy mô nhỏ để kiểm thử các tính năng và đánh giá hiệu năng hoạt động của chuỗi chức năng. Mô hình hệ thống triển khai với chức năng mạng ảo Tường lửa và Hệ thống phát hiện xâm nhập để bảo vệ máy chủ web trong trung tâm dữ liệu. Giám sát số liệu thống kê từ các chức năng mạng ảo để tính toán được lượng tài nguyên phù hợp sử dụng trong triển khai thực tế, cũng như đo đạc, đánh giá chất lượng dịch vụ của chuỗi. Từ đó, tạo bước đệm để nghiên cứu sâu hơn về lĩnh vực này.

ABSTRACT

Service Function Chaining (SFC) is a mechanism that allows various service functions to be connected to each to form a service enabling carriers to benefit from virtualized software defined infrastructure. At present, the network infrastructure of the network operators are distributed many proprietary hardware devices. Corresponding to each type of service, each network function has a specialized hardware undertake. These devices have a number of limitations: high cost of equipment, difficulty in centralized management, low compatibility with many other systems leading to the enterprise depends on specific hardware vendors.

In this project, I present the definition of NFV and the benefits of the network service chain (SFC), the use cases of two technologies and the benefits that they bring to network service providers and customers. At the same time, I also built a testbed about network functional sequence system on OpenStack, using small-scale network virtualization functions to test the features and assess the performance of SFC. The deployment model has a virtual network function Firewall and Intrusion Detection System to protect the web server in the data center. Monitoring statistics from these virtual network functions to calculate the amount of resources used in actual deployment, as well as measuring and evaluating the QoS of the chain. From there, create a stepping stone for further research in this field.

MỤC LỤC

LỜI NÓI ĐẦU	1
TÓM TẮT ĐỒ ÁN	2
ABSTRACT	3
MỤC LỤC.....	4
DANH MỤC HÌNH VẼ	6
DANH MỤC BẢNG BIỂU	8
DANH SÁCH CÁC TỪ VIẾT TẮT	9
CHƯƠNG 1: MỞ ĐẦU.....	10
1.1. Đặt vấn đề.....	10
1.2. Đề xuất hướng giải quyết	11
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT CHUNG	13
2.1. Tổng quan về công nghệ điện toán đám mây.....	13
2.1.1. Giới thiệu về công nghệ điện toán đám mây	13
2.1.2. Các đặc trưng của công nghệ điện toán đám mây	14
2.2. Giới thiệu OpenStack	18
2.3. Tổng quan về công nghệ ảo hóa chức năng mạng – Network Function Virtualization (NFV)	20
2.3.1. Giới thiệu về công nghệ ảo hóa chức năng mạng.....	20
2.3.2. Lợi ích của công nghệ ảo hóa chức năng mạng.....	21
2.3.3. Các trường hợp sử dụng của NFV	22
2.4. Tổng quan về chuỗi chức năng mạng – Service Function Chaining (SFC)	30
2.4.1. Giới thiệu về chuỗi chức năng mạng.....	30
2.4.2. Ứng dụng của SFC trong một số trường hợp cụ thể	31
2.5. Kết luận và đưa ra định hướng giải pháp xây dựng đề tài đồ án	37
CHƯƠNG 3: XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG	38
3.1. Xây dựng nền tảng cloud quản lý các chức năng mạng ảo hóa – OpenStack.....	39

3.1.1.	Dựng OpenStack để triển khai NFV	39
3.2.	Xây dựng luồng đi chuỗi chức năng mạng SFC	40
3.2.1.	SFC trong trung tâm dữ liệu	40
3.2.2.	Công nghệ xây dựng chuỗi chức năng mạng trên OpenStack	41
3.3.	Xác định các công nghệ sử dụng làm các chức năng mạng	43
3.3.1.	Chức năng mạng Firewall – Iptables	43
3.3.2.	Chức năng mạng Phát hiện xâm nhập (IDS) – Suricata	45
3.3.3.	Chức năng mạng giám sát – Grafana	47
3.4.	Kịch bản thử nghiệm	50
3.4.1.	Đồ hình vật lý	50
3.4.2.	Kịch bản kiểm thử	51
3.4.3.	Kết quả dựng mô hình trên giao diện của OpenStack	52
3.4.4.	Kết quả dựng networking-SFC trên OpenStack	53
3.5.	Kết luận	54
CHƯƠNG 4: KẾT QUẢ ĐO ĐẠC VÀ ĐÁNH GIÁ		56
4.1.	Kiểm chứng luồng lưu lượng đi theo đúng mô hình SFC đã dựng	56
4.2.	Kết quả đo lượng tài nguyên sử dụng trên Suricata	59
4.3.	Kết quả đo độ trễ và tỷ lệ mất gói của lưu lượng khi đi qua chuỗi các chức năng mạng	60
4.4.	Đánh giá kết quả	62
4.5.	Hạn chế	62
KẾT LUẬN		63
TÀI LIỆU THAM KHẢO		64

DANH MỤC HÌNH VẼ

Hình 2.1: Mô hình cơ sở hạ tầng của điện toán đám mây.....	14
Hình 2.2: 3 mô hình dịch vụ của điện toán đám mây	16
Hình 2.3: Mô hình triển khai OpenStack với các project chính.	19
Hình 2.4: Mô hình Ảo hóa chức năng mạng NFVIaaS.....	22
Hình 2.5: Mô hình triển khai CPE truyền thống và vCPE ^[2]	23
Hình 2.6: Các doanh nghiệp sử dụng chung nền tảng để thực hiện phát triển các dịch vụ mạng của chính mình.	24
Hình 2.7: VNF Forwarding Graph	25
Hình 2.8: Ứng dụng của NFV trong môi trường mạng gia đình ^[2]	26
Hình 2.9: Các nút CDN ảo hóa được triển khai trong một môi trường ảo hóa ^[2]	27
Hình 2.10: Một chuỗi SFC đơn giản	31
Hình 2.11: Ứng dụng của SFC trong mạng Fixed Broadband Network.....	32
Hình 2.12: Triển khai SFC trong mạng di động ^[4]	33
Hình 2.13: Các mạng dùng chung chuỗi dịch vụ mạng ^[4]	34
Hình 2.14: SFC trong trung tâm dữ liệu ^[4]	35
Hình 2.15: SFC trong các thiết bị Cloud CPE ^[4]	36
Hình 3.1: Mô hình testbed mức logic.....	38
Hình 3.2: Mô hình triển khai OpenStack mức logic	39
Hình 3.3: Chuỗi các chức năng mạng trong trung tâm dữ liệu	41
Hình 3.4: Cấu trúc networking-SFC trong OpenStack ^[7]	42
Hình 3.5: Quá trình xử lý gói tin của IPtables	44
Hình 3.6: Hai loại hệ thống IDS.....	46
Hình 3.7: Giám sát số liệu hệ thống với Grafana.....	47
Hình 3.8: Đồ hình vật lý triển khai của testbed	50
Hình 3.9: Đồ hình triển khai các máy ảo trong OpenStack	52
Hình 3.10: Đồ hình chuỗi các chức năng mạng trên giao diện OpenStack	53
Hình 3.11: Kết quả tạo port pair trên các cổng nối với VM	53
Hình 3.12: Kết quả tạo port-pair-group	54
Hình 3.13: Kết quả tạo Flow Classifier.....	54

Hình 4.1: Thực hiện ping từ một máy bên ngoài vào máy chủ Web	57
Hình 4.2: Luồng lưu lượng đi qua Iptables	57
Hình 4.3: Luồng lưu lượng gửi qua Suricata	58
Hình 4.4: Lưu lượng được gửi tới máy Webserver.....	58

DANH MỤC BẢNG BIỂU

<i>Bảng 2.1: Bảng so sánh mô hình mạng truyền thống và mô hình mạng ứng dụng NFV</i>	<i>21</i>
<i>Bảng 3.1: Thông số phần cứng và cấu hình hai nút server</i>	<i>51</i>
<i>Bảng 3.2: Thông số máy ảo triển khai trong OpenStack.....</i>	<i>52</i>
<i>Biểu đồ 4.1: Kết quả sử dụng CPU trên Suricata khi tải tăng dần</i>	<i>59</i>
<i>Biểu đồ 4.2: Lượng bộ nhớ sử dụng để xử lý các gói tin trên Suricata khi tải tăng dần.....</i>	<i>60</i>
<i>Biểu đồ 4.3: Kết quả đo độ trễ trong mạng</i>	<i>61</i>
<i>Biểu đồ 4.4: Tỷ lệ mất gói trong mạng.</i>	<i>61</i>

DANH SÁCH CÁC TỪ VIẾT TẮT

Từ viết tắt	Tiếng Anh	Tiếng Việt
NFV	Network Function Virtualization	Ảo hóa chức năng mạng
SFC	Service Function Chain	Chuỗi dịch vụ
CSP	Cloud Service Provider	Nhà cung cấp dịch vụ đám mây
ASP	Application Service Provider	Nhà cung cấp dịch vụ ứng dụng
ISP	Internet Service Provider	Nhà cung cấp dịch vụ Internet
CNTT		Công nghệ thông tin
CPU	Central Processing Unit	Bộ xử lý trung tâm
RAM	Random Access Memory	Bộ nhớ truy cập ngẫu nhiên
OS	Operating System	Hệ điều hành
NAT	Network Address Translation	Chuyển đổi địa chỉ mạng
IDS	Intrusion Detection System	Hệ thống phát hiện xâm nhập
VPN	Virtual Private Network	Mạng riêng ảo
DPI	Deep Packet Inspection	Phân tích gói sâu
WAN	Wide Area Network	Mạng diện rộng
VNF	Virtual Network Function	Chức năng mạng ảo
CPE	Customer Premises Equipment	Thiết bị mạng đặt phía khách hàng
CDN	Content Delivery Network	Mạng phân phối nội dung
FTTH	Fiber to the Home	Mạng viễn thông băng rộng dùng cáp quang
DC	Data Center	Trung tâm dữ liệu
NIDS	Network-based IDS	
HIDS	Host-based IDS	
VM	Virtual Machine	Máy ảo
IP	Internet Protocol	
DMZ	Demilitarized Zone	Vùng mạng riêng máy chủ
QoS	Quality of Service	Chất lượng dịch vụ
DHCP	Dynamic Host Configuration Protocol	Giao thức cấp phát động địa chỉ IP

CHƯƠNG 1: MỞ ĐẦU

1.1. Đặt vấn đề

Ngày nay, triển khai các chức năng mạng bảo vệ các ứng dụng bên trong của trung tâm dữ liệu là mối quan tâm hàng đầu của các nhà cung cấp dịch vụ. Tùy thuộc vào từng ứng dụng và nhu cầu truy cập ứng dụng của khách hàng, mà mỗi ứng dụng dịch vụ bên trong lại cần có các cơ chế bảo mật khác nhau, luồng lưu lượng từ bên ngoài vào từng ứng dụng dịch vụ bên trong lại cần đi qua các chuỗi dịch vụ mạng bảo vệ khác nhau.

Tuy nhiên, hiện tại, việc triển khai chuỗi các dịch vụ chức năng còn một số hạn chế sau: Phụ thuộc vào cấu trúc vật lý: bởi hầu hết các chức năng dịch vụ mạng được đặt tại các thiết bị phần cứng: như tường lửa, cân bằng tải, ... những dịch vụ này được kết hợp bằng cấu trúc liên kết vật lý cơ bản. Các chức năng mạng được thêm vào chuỗi cần phải theo một thứ tự nghiêm ngặt. Do đó, việc xóa hoặc bổ sung các chức năng này trở nên cực kì phức tạp vì nó yêu cầu thay đổi trong cấu trúc mạng vật lý. Điều này làm ngăn cản các nhà khai thác mạng sử dụng tối ưu các tài nguyên mạng. Sự phụ thuộc vào cấu trúc vật lý cũng dẫn đến hạn chế tính linh hoạt trong việc phân phối dịch vụ vì nó có thể yêu cầu những thay đổi đáng kể trong cấu hình mạng hiện tại. Các nhà khai thác mạng không có cách nhất quán để áp đặt và xác minh vị trí và thứ tự của các chức năng dịch vụ mạng. Từ đó dẫn tới việc cung cấp dịch vụ chậm, không linh hoạt.

Tại các trung tâm dữ liệu ngày nay, các máy chủ web server phải đối mặt với nhiều hiểm họa từ Internet. Đó là các cuộc tấn công mạng diễn ra ngày càng tinh vi, mức độ lớn hơn và khó đối phó hơn.

Trong đồ án này, em xin đề xuất triển khai chuỗi chức năng mạng gồm các chức năng Tường lửa và Hệ thống phát hiện xâm nhập dựa trên công nghệ NFV để bảo vệ máy chủ web server bên trong trung tâm dữ liệu.

1.2. Đề xuất hướng giải quyết

Triển khai SFC bằng cách kết hợp NFV với điều khiển luồng lưu lượng giữa các VNF là một giải pháp tuyệt vời để xử lý những hạn chế còn tồn tại nêu trên. Với mục đích bảo vệ máy chủ web server khỏi các hiểm họa từ bên ngoài, giải pháp đưa ra là triển khai các chuỗi chức năng mạng có tính năng tường lửa và hệ thống phát hiện xâm nhập để lọc luồng lưu lượng trước khi đưa vào máy chủ web.

Trong đồ án này, em thực hiện tìm hiểu các vấn đề liên quan tới NFV, SFC và dựng mô hình testbed chuỗi chức năng mạng ảo hóa ở quy mô nhỏ trên nền tảng OpenStack để kiểm thử các tính năng và đánh giá hiệu năng hoạt động của chuỗi chức năng. Mô hình hệ thống triển khai với chức năng mạng ảo Tường lửa và Hệ thống phát hiện xâm nhập để bảo vệ máy chủ web trong trung tâm dữ liệu. Giám sát số liệu thống kê từ các chức năng mạng ảo để tính toán được lượng tài nguyên phù hợp sử dụng trong triển khai thực tế, đo đạc các thông số hiệu năng mạng của chuỗi.

Đồ án các phần sau:

- Tìm hiểu công nghệ NFV và một số trường hợp sử dụng NFV
- Tìm hiểu SFC và các trường hợp triển khai SFC
- Xây dựng hệ thống thử nghiệm các chức năng mạng ảo NFV trên nền tảng OpenStack, chức năng mạng có tính năng của Tường lửa và Hệ thống phát hiện xâm nhập để bảo vệ máy chủ web bên trong trung tâm dữ liệu.
- Đánh giá hiệu năng của các chức năng mạng ảo trong chuỗi (lượng CPU, RAM sử dụng) và chất lượng dịch vụ của chuỗi chức năng mạng (độ trễ, tỉ lệ mất gói, băng thông) trong trường hợp tải đầu vào tăng dần.

Nội dung đồ án được chia thành các phần như sau:

- **Chương 1:** Nêu vấn đề, phân tích từ đó trình bày lý do chọn đề tài và đề xuất hướng giải quyết, khắc phục.
- **Chương 2:** Trình bày tóm tắt về cơ sở lý thuyết chung, các thành phần liên quan tới việc thực hiện đề tài đồ án. Từ đó đưa ra kết luận và định hướng xây dựng mô hình kiểm thử.

- **Chương 3:** Thực hiện xây dựng mô hình kiểm thử chuỗi chức năng mạng ảo hóa. Trình bày các thành phần trong mô hình kiểm thử.
- **Chương 4:** Trình bày các kết quả đã đạt được, đánh giá chất lượng của chuỗi chức năng mạng đã xây dựng được. Nêu các hạn chế chưa xử lý triệt để.
- Phần cuối cùng đưa ra kết luận và đề xuất các định hướng phát triển đồ án trong tương lai.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT CHUNG

Đầu tiên, Chương 2 sẽ trình bày tổng quan về cơ sở chung của các công nghệ điện toán đám mây, nền tảng quản lý điện toán đám mây OpenStack, công nghệ ảo hóa chức năng mạng và chuỗi các dịch vụ mạng SFC liên quan tới đồ án. Từ đó đưa ra kết luận và định hướng xây dựng mô hình SFC kiểm thử.

2.1. Tổng quan về công nghệ điện toán đám mây

2.1.1. Giới thiệu về công nghệ điện toán đám mây

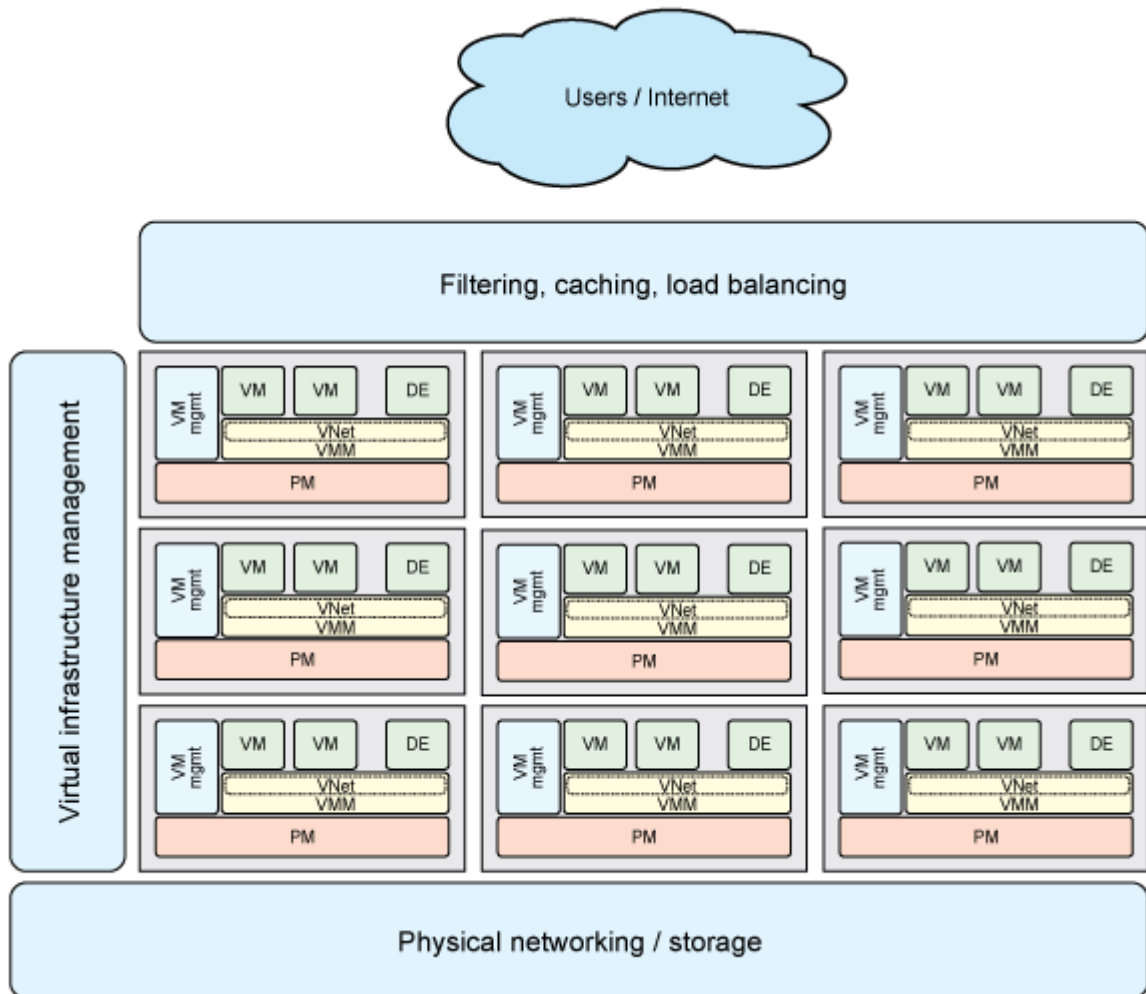
Điện toán đám mây (Cloud Computing), còn gọi là điện toán máy chủ ảo: là mô hình tính toán sử dụng các công nghệ máy tính và phát triển dựa vào mạng Internet.

Thuật ngữ "*Cloud Computing*" ra đời giữa năm 2007, dùng để khái quát lại các hướng phát triển của cơ sở hạ tầng CNTT vốn đã và đang diễn ra từ nhiều năm qua. Quan niệm này có thể được diễn giải một cách đơn giản như sau: các nguồn tài nguyên tính toán khổng lồ như các phần cứng (máy chủ), phần mềm, và các dịch vụ (chương trình ứng dụng), ... sẽ nằm tại các máy chủ ảo (đám mây - cloud) trên Internet thay vì trong máy tính gia đình và văn phòng (trên mặt đất) để mọi người kết nối và sử dụng mỗi khi họ cần dù ở bất cứ đâu miễn có kết nối Internet.

Nói cách khác, đây là một mô hình cho phép truy cập qua mạng để chia sẻ chung nguồn tài nguyên (mạng, dịch vụ, ứng dụng, máy chủ, không gian lưu trữ,...) một cách nhanh chóng, thuận tiện và đồng thời cho phép kết thúc sử dụng dịch vụ, giải phóng tài nguyên dễ dàng, giảm thiểu sự ảnh hưởng quản lý và giao tiếp với nhà cung cấp. Mọi khả năng liên quan đến công nghệ thông tin đều được cung cấp dưới dạng các "dịch vụ", cho phép người sử dụng truy cập các dịch vụ công nghệ thông tin từ một nhà cung cấp nào đó "trong đám mây" mà không cần phải biết về công nghệ đó, cũng như không cần quan tâm đến các cơ sở hạ tầng phục vụ công nghệ đó.

Điện toán đám mây được hình thành bằng cách kết hợp công nghệ ảo hóa trên các nút máy chủ vật lý lại với nhau trên mạng vật lý, kết hợp lưu trữ có chia sẻ và

phối hợp quản lý trên toàn bộ cơ sở hạ tầng, rồi cung cấp cân bằng tải. Mô hình cơ sở hạ tầng của điện toán đám mây được mô tả trong hình sau:



Hình 2.1: Mô hình cơ sở hạ tầng của điện toán đám mây

2.1.2. Các đặc trưng của công nghệ điện toán đám mây

Mô hình điện toán đám mây – Cloud Computing đặc trưng bởi 5 đặc tính, 3 mô hình dịch vụ và 4 mô hình triển khai.

5 đặc tính:

- **On-demand self service** : Khả năng tự phục vụ : người dùng có khả năng tự tính toán được nhu cầu sử dụng máy chủ lưu trữ, khi cần thiết có thể thực hiện tự động mà không cần phải có sự giao tiếp với nhà cung cấp dịch vụ.

- **Broad network access:** khả năng truy cập nhiều kiểu hạ tầng mạng khác nhau, hỗ trợ nhiều nền tảng thiết bị, nhiều hạ tầng vật lý (như điện thoại di động, laptop, tablet, máy trạm..)
- **Resource pooling:** Dùng chung tài nguyên mạng. Tài nguyên tính toán được của nhà cung cấp được dùng chung cho nhiều khách hàng sử dụng dịch vụ đa thuê bao, với những nguồn tài nguyên vật lý và ảo khác nhau được phân công và bố trí theo nhu cầu của người sử dụng. Khách hàng không thể xác định và biết được vị trí chính xác của nguồn tài nguyên nhưng mà có thể dễ dàng xác định được vị trí cung cấp nguồn tài nguyên một cách tương đối (quốc gia, liên bang, trung tâm dữ liệu).

Gộp chung nguồn tài nguyên của nhiều máy chủ với nhau thành một khối thống nhất, rồi sau đó sẽ san sẻ tài nguyên cho người dùng. (mục đích để quản lý, cấp phát tài nguyên dễ dàng, nhanh chóng).

- **Rapid Elasticity** (có tính đàn hồi cao) : Có khả năng cung cấp và giải phóng tài nguyên một cách nhanh chóng, trong một số trường hợp có thể tự động thu hẹp hoặc mở rộng phạm vi tùy nhu cầu người dùng. Khách hàng có khả năng có thể được cung cấp không giới hạn và chiếm tài nguyên bất kì lúc nào.
- **Measured service** (Tính toán dịch vụ) : Hệ thống cloud tự động điều khiển và tối ưu hóa nguồn tài nguyên sử dụng bằng tận dụng khả năng đo lường ở một vài cấp độ dịch vụ. tính toán mức độ sử dụng dịch vụ, kiểm soát thời gian phục vụ, giám sát, điều khiển, báo cáo, ... Từ đó có thể tính toán được chi phí của người sử dụng.

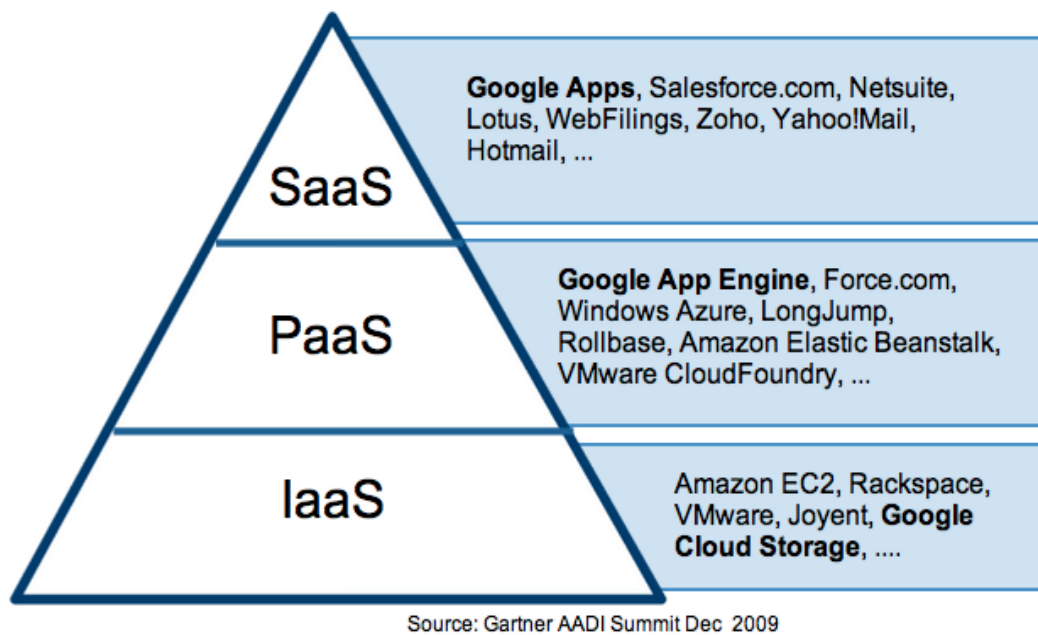
4 mô hình triển khai:

- **Private cloud** : Nền tảng cloud được cung cấp độc quyền cho 1 tổ chức bao gồm nhiều đối tượng (như công ty kinh doanh..). Nó có thể được làm chủ, điều khiển và vận hành bởi một tổ chức - một thành viên thứ 3, hoặc là có sự kết hợp giữa 2 bên, có thể tồn tại hoặc không ...
- **Community cloud:** Nền tảng cloud được cung cấp độc quyền cho một tổ chức người sử dụng đến từ các tổ chức có chung một mối quan tâm về mục đích,

nhiệm vụ, chính sách. Nó có thể được quản lý, vận hành bởi một hoặc nhiều tổ chức trong cộng đồng kết hợp quản lý với nhau.

- **Public Cloud:** Hạ tầng Cloud đc cung cấp cho tất cả mọi người dùng thông thường. Được làm quản lý và vận hành bởi chính người dùng, có tính chất thương mại.. Có sự ràng buộc giữa người dùng và nhà cung cấp.
- **Hybrid Cloud:** Đây là hạ tầng cloud được phối hợp giữa 2 hoặc nhiều hạ tầng cloud riêng biệt. Những hạ tầng này vẫn giữ đặc điểm riêng biệt nhưng có thể phối hợp cùng nhau bởi việc được chuẩn hóa hoặc công nghệ phù hợp mà dữ liệu và ứng dụng có thể lưu động được.

3 mô hình dịch vụ:



Hình 2.2: 3 mô hình dịch vụ của điện toán đám mây

Như mô tả trên Hình 2.2, các mô hình dịch vụ của điện toán đám mây bao gồm:

- **SaaS (Software as a Service):** Khách hàng sử dụng những ứng dụng chạy trên nền tảng cloud của nhà cung cấp. Các ứng dụng có thể truy cập được từ các thiết bị thông qua giao diện với người dùng, Khách hàng không có quyền quản lý và điều khiển kiến trúc hạ tầng của cloud như mạng, server, lưu trữ, hệ thống vận hành và thậm chí cả một vài ứng dụng cá nhân ... do đó hạn chế người dùng về

việc cấu hình đặc biệt cho các ứng dụng. (người dùng không cần quan tâm xem cloud triển khai như thế nào, chỉ cần thuê và sử dụng dịch vụ thông qua các phần mềm client (web browser..)

- ***PaaS (Platform as a Service)*** : Nền tảng như một dịch vụ. cung cấp các dịch vụ về nền tảng, môi trường lập trình, database, etc. để khách hàng phát triển các ứng dụng của mình (thông thường nền tảng này cung cấp cho các developer). Ví dụ: AWS, GAE, Azure, Bluemix, OpenShift, etc.. Nhưng khách hàng không được quản lý các kiến trúc hạ tầng của cloud, được phép điều khiển các ứng dụng triển khai và có thể đồng bộ cài đặt cho môi trường quản lý ứng dụng.
- ***IaaS (Infrastructure as a Service)*** : Cung cấp các dịch vụ về hạ tầng, các máy chủ, tài nguyên tính toán (RAM, CPU), lưu trữ. Trên đó người dùng sẽ tạo các máy ảo với hệ điều hành, triển khai ứng dụng theo nhu cầu của mình. (Người dùng không được điều khiển hạ tầng cloud nhưng có thể điều khiển được hệ thống OS, khả năng lưu trữ, và các ứng dụng triển khai, và có thể có một số quyền kiểm soát chọn mạng).

2.2. Giới thiệu OpenStack

OpenStack là một dự án phần mềm mã nguồn mở dùng để triển khai private và public cloud. Nó bao gồm nhiều thành phần (project) do các công ty, tổ chức và các lập trình viên tự nguyện xây dựng và phát triển.

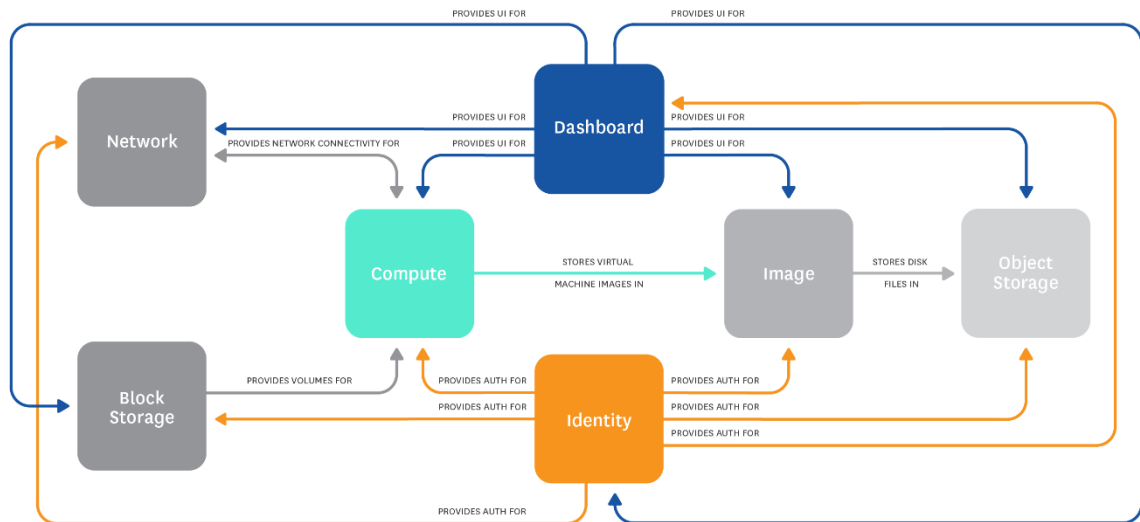
OpenStack hoạt động theo hướng mở: công khai lộ trình phát triển, công khai mã nguồn mở... OpenStack được phát triển và phát hành phiên bản mới trong vòng 6 tháng, hiện tại đã có 13 phiên bản OpenStack. Tên các phiên bản được đặt theo thứ tự chữ cái A, B, C, ... phiên bản hiện tại là Queen. Tới cuối tháng 8/2018, dự định sẽ có phiên bản mới là Rocky.

Phần lớn mã nguồn của OpenStack là Python.

Có thể coi OpenStack như một hệ điều hành cloud có nhiệm vụ kiểm soát các tài nguyên tính toán (compute), lưu trữ (storage) và networking trong hệ thống lớn Datacenter, tất cả đều có thể được kiểm soát qua giao diện dòng lệnh hoặc một dashboard (do project horizon cung cấp). Ở thời điểm hiện tại, OpenStack có 6 core project và 35 project tùy chọn cài đặt theo nhu cầu. 6 core project của OpenStack bao gồm:

- KEYSTONE (Identity Service) ^[6]: dịch vụ xác thực, ủy quyền người dùng và các dịch vụ khác của OpenStack.
- GLANCE (Images Service) ^[6]: dịch vụ cung cấp các image cho máy ảo của OpenStack.
- NOVA (Compute Service) ^[6]: dịch vụ tính toán, quản lý vòng đời máy ảo bên trong OpenStack.
- NEUTRON (Network Service) ^[6]: dịch vụ cung cấp kết nối mạng cho các thành phần của OpenStack.
- CINDER (Block Service) ^[6]: dịch vụ lưu trữ volume cho các máy ảo.
- HORIZON^[6] (Dashboard): Cung cấp giao diện quản lý cho người dùng.

6 project này có nhiệm vụ quan trọng trong việc hình thành nên môi trường cloud và quản lý một cách hiệu quả.



Hình 2.3: Mô hình triển khai OpenStack với các project chính.

Hình 2.3 mô tả mô hình triển khai cơ bản của OpenStack với 7 project chính (bao gồm cả SWIFT (Object storage)).

2.3. Tổng quan về công nghệ ảo hóa chức năng mạng – Network Function Virtualization (NFV)

2.3.1. Giới thiệu về công nghệ ảo hóa chức năng mạng

Hiện nay, kiến trúc mạng của các nhà khai thác mạng được phân bổ nhiều thiết bị phần cứng độc quyền. Ứng với mỗi loại dịch vụ, mỗi chức năng mạng lại có các thiết bị phần cứng chuyên trách đảm nhận. Các thiết bị này còn đang có một số hạn chế như: chi phí thiết bị cao, khó quản lý tập trung, độ tương thích thấp với hệ thống của nhiều hãng khác dẫn tới doanh nghiệp phải phụ thuộc vào một số hãng phần cứng nhất định. Để triển khai một dịch vụ mạng mới thường, các nhà khai thác cần giải quyết các bài toán về chi phí năng lượng, chi phí đầu tư vốn và các kỹ năng cần thiết để thiết kế, tích hợp, vận hành và quản lý các thiết bị phần cứng. Không những thế, tuổi thọ của các thiết bị phần cứng đang trở nên ngắn hơn khi đổi mới công nghệ hay dịch vụ tăng tốc, làm ngăn cản việc triển khai các dịch vụ mạng mới và hạn chế cải tiến trong thế giới kết nối mạng ngày càng tập trung như hiện nay.

Công nghệ ảo hóa chức năng mạng - Network Function Virtualization (NFV) ra đời nhằm giải quyết các vấn đề này bằng cách tận dụng công nghệ ảo hóa tiêu chuẩn để hợp nhất nhiều loại thiết bị mạng vào các máy chủ xử lý tốc độ cao, các thiết bị chuyển mạch và thiết bị lưu trữ chuẩn công nghiệp - được đặt tại các trung tâm dữ liệu (Data Center), các điểm chuyển mạch lớn (Network node) trên đường truyền hoặc tại vị trí của người dùng cuối. Nhờ có NFV, các chức năng mạng truyền thống như: NAT, Firewall, Router, IDS, ... tách biệt khỏi các thiết bị vật lý chuyên biệt và được triển khai dưới dạng phần mềm có thể hoạt động trong môi trường ảo hóa trên các thiết bị máy chủ phần cứng phổ thông. Từ đây, việc sử dụng các thiết bị chức năng mạng trở nên linh hoạt hơn: dễ dàng khởi tạo và điều phối các chức năng mạng, giảm thiểu khả năng phụ thuộc vào các nhà cung ứng độc quyền, đồng thời tận dụng nguồn tài nguyên phần cứng hiện có, chi phí vận hành, duy trì và quản lý cũng được giảm thiểu đáng kể.

2.3.2. Lợi ích của công nghệ ảo hóa chức năng mạng

Để thấy rõ được lợi ích của công nghệ ảo hóa chức năng mạng, ta cùng so sánh mô hình mạng truyền thống với mô hình mạng khi áp dụng triển khai NFV thông qua Bảng 2.1 sau:

Bảng 2.1: Bảng so sánh mô hình mạng truyền thống và mô hình mạng ứng dụng NFV

Tiêu chí	Mô hình mạng truyền thống	Mô hình mạng áp dụng NFV
Chi phí thiết bị và chi phí năng lượng	Chi phí cao do phải đầu tư vào nhiều thiết bị phần cứng chuyên biệt, năng lượng tiêu thụ lớn	Chi phí thấp do sử dụng phần mềm, tận dụng được hiệu suất phần cứng phổ thông sẵn có.
Khả năng tùy biến, quản trị, mở rộng và nâng cấp	Khó khăn do phụ thuộc hoàn toàn vào nhà cung cấp thiết bị. Quản lý phân tán gây mất thời gian và công sức của người quản trị. Việc nâng cấp thiết bị gần như là thay thế toàn bộ.	Cao do sử dụng phần mềm hầu như là các giải pháp mã nguồn mở hoặc từ các hãng có cung cấp phần mềm điều khiển. Khả năng mở rộng linh hoạt đáp ứng nhu cầu với các mạng thay đổi thường xuyên. Việc nâng cấp cũng dễ dàng hơn.
Hiệu năng, độ ổn định	Cao do sử dụng các thiết bị phần cứng chuyên biệt cho từng chức năng mạng.	Thấp hơn do ảo hóa trên các thiết bị phần cứng phổ thông. Tuy nhiên, về lâu dài, hiệu năng sẽ dần được cải thiện
Yêu cầu nhân sự	Yêu cầu có kỹ năng sử dụng và vận hành của các hãng thiết bị riêng biệt có trong hạ tầng mạng nhờ tham gia các khóa đào tạo.	Dễ dàng tiếp cận được tài liệu và thông tin để vận hành và quản lý hệ thống mạng. Yêu cầu khả năng cao hơn so với môi trường mạng truyền thống.

Từ Bảng 2.1 trên, ta có thể tóm gọn lại được một số lợi ích tuyệt vời mà công nghệ NFV mang lại như sau: giảm chi phí đầu tư và vận hành, tận dụng nguồn tài

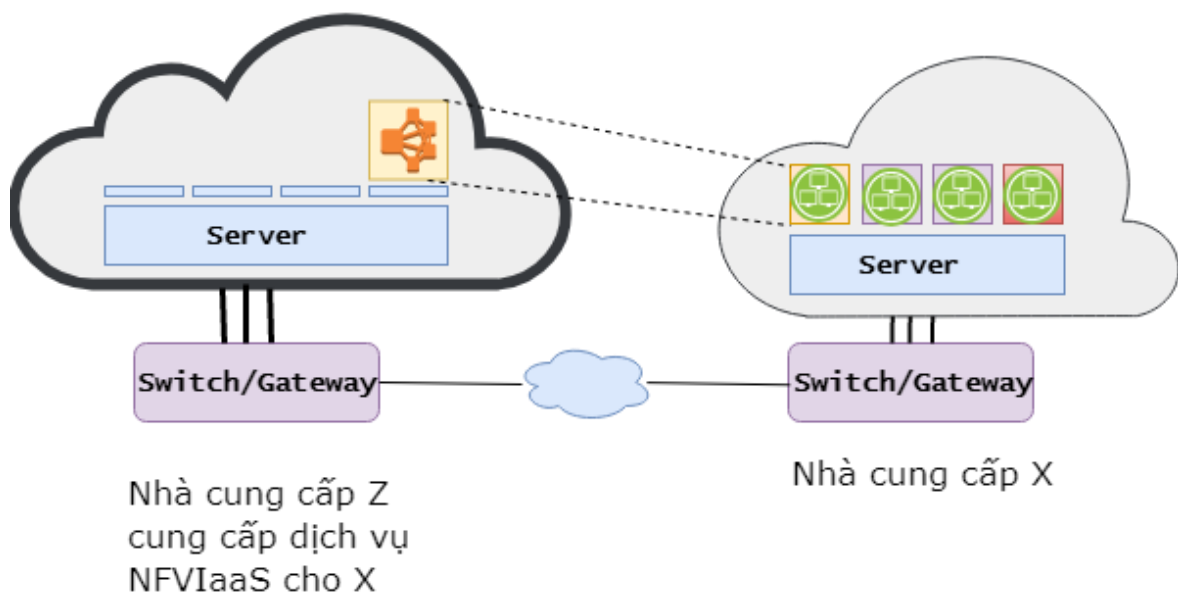
nguyên phần cứng sẵn có, cung cấp nhanh chóng và linh hoạt và nhờ giảm bớt thời gian triển khai các dịch vụ mạng mới, các nhà khai thác mạng nắm bắt những cơ hội thị trường mới để tăng lợi nhuận cho doanh nghiệp khi đầu tư vào các dịch vụ mới đó.

2.3.3. Các trường hợp sử dụng của NFV

9 trường hợp sử dụng NFV như sau:

1) Ảo hóa chức năng mạng: Hạ tầng như một dịch vụ

Network Functions Virtualization Infrastructure as a Service: Các nhà cung cấp dịch vụ mạng cung cấp cho các nhà cung cấp khác hoặc khách hàng của họ hạ tầng ảo hóa chức năng mạng của mình để sử dụng lại như một dịch vụ điện toán đám mây theo mô hình IaaS (Infrastructure as a Service).



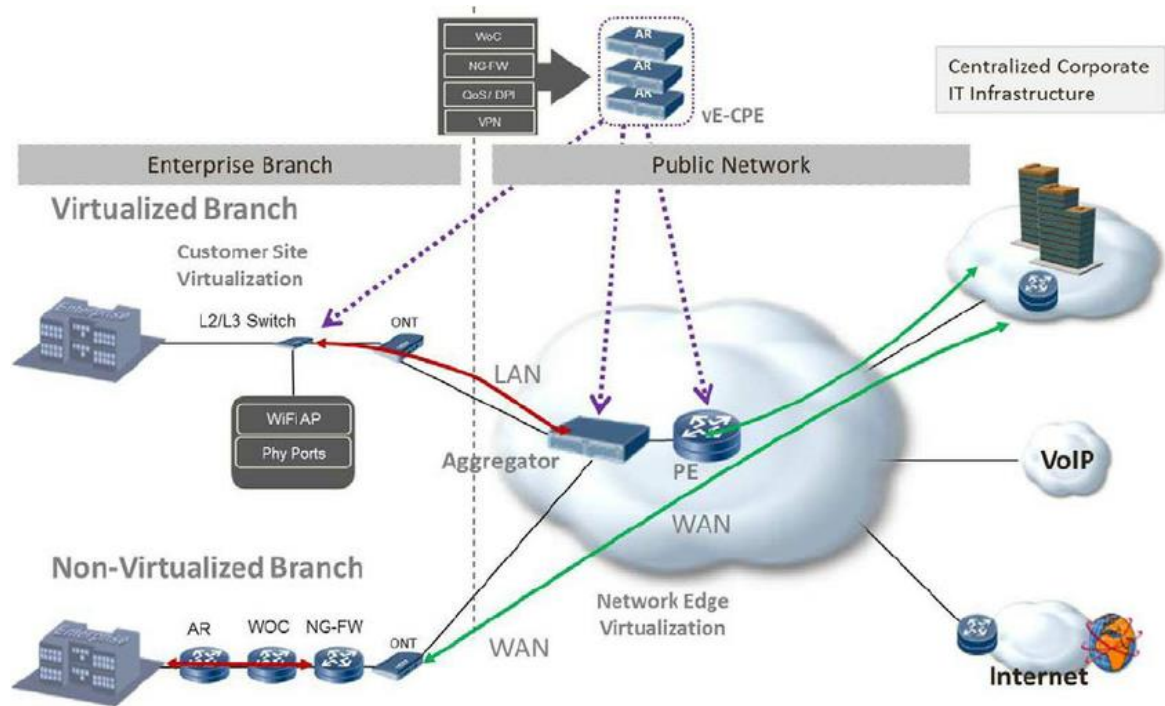
Hình 2.4: Mô hình Ảo hóa chức năng mạng NFVIaaS

Ở Hình 2.4, chức năng mạng ảo hóa bên nhà cung cấp Z được cấp cho bên X sử dụng như cơ sở hạ tầng điện toán đám mây.

2) Ảo hóa chức năng mạng như một dịch vụ (NFV as a Service - NFVaaS)

Các doanh nghiệp hiện nay triển khai nhiều dịch vụ mạng ở các chi nhánh văn phòng khá nhau. Việc đầu tư, vận hành và bảo dưỡng các thiết bị chức năng mạng

chuyên dụng truyền thống như: Router, NAT, Firewall, IDS/IPS, ... gây tốn kém khá nhiều chi phí về vật chất và nhân lực. Trong trường hợp này, các nhà cung cấp mạng viễn thông có thể sử dụng NFV để cung cấp các chức năng mạng CPE (Customer Premises Equipment) dưới dạng điện toán đám mây theo mô hình SaaS (Software as a Service).



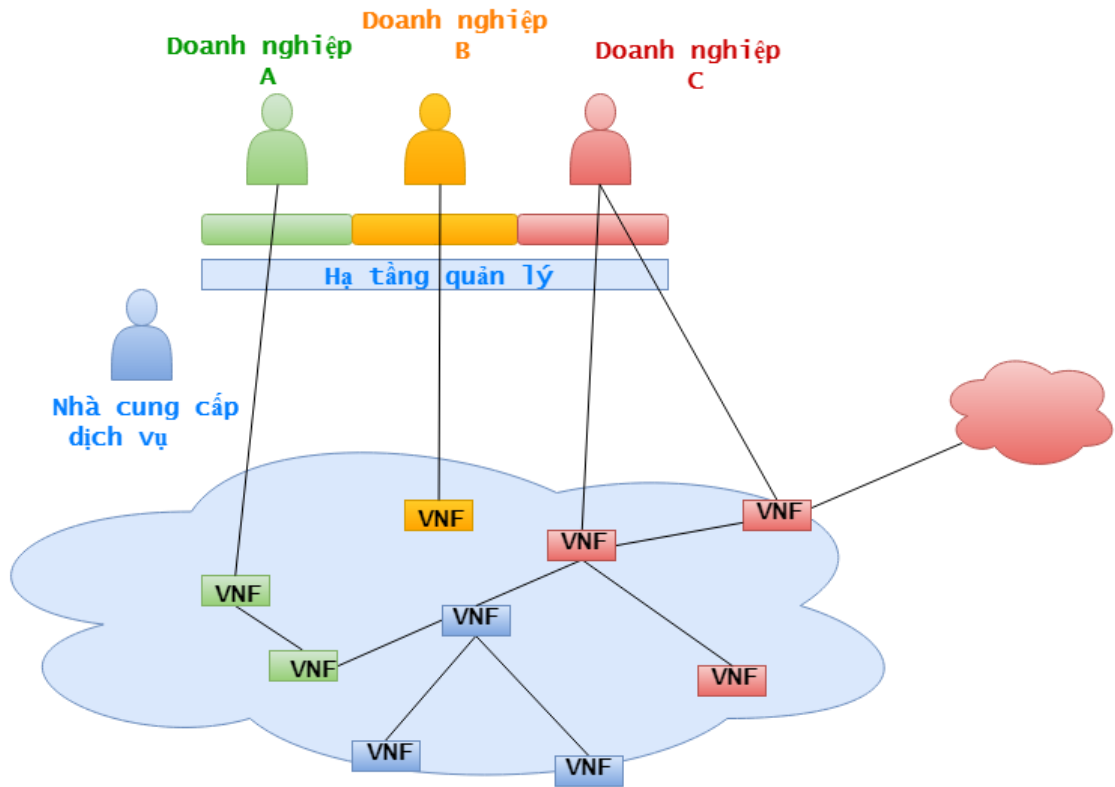
Hình 2.5: Mô hình triển khai CPE truyền thống và vCPE [2]

Như mô tả trên Hình 2.5, lưu lượng truy cập cục bộ của doanh nghiệp được xử lý bởi các thiết bị CPE ảo (vCPE) cung cấp chức năng kết nối lớp 2, lớp 3; và được tích hợp thêm các công nghệ truy cập wifi, hỗ trợ VPN, DPI, WAN Optimazation Controller, QoS. Trong khi triển khai mô hình mạng truyền thống với các thiết bị chuyên dụng, việc vận hành và chi phí thiết bị hẳn sẽ làm đau đầu các chủ doanh nghiệp.

3) Virtual Network Platform as a Service (VNPaaS)

Tài nguyên mạng ngày càng trở nên không bị độc quyền bởi các nhà cung cấp. Các doanh nghiệp hiện nay đã có thể lưu trữ trên cơ sở hạ tầng của nhiều nhà cung cấp. Ảo hóa các chức năng mạng (VNF) làm tăng độ linh hoạt khi chia sẻ các tài nguyên và giảm chi phí thiết lập và quản lý bằng cách cung cấp nền tảng theo mô

hình điện toán đám mây PaaS. Các nhà cung cấp có thể đảm bảo độ phù hợp của cơ sở hạ tầng và các ứng dụng như một nền tảng mà trên đó các doanh nghiệp có thể triển khai các ứng dụng mạng của họ. Với nền tảng này, các doanh nghiệp có thể phát triển tùy ý các dịch vụ mạng theo mục đích kinh doanh của họ.



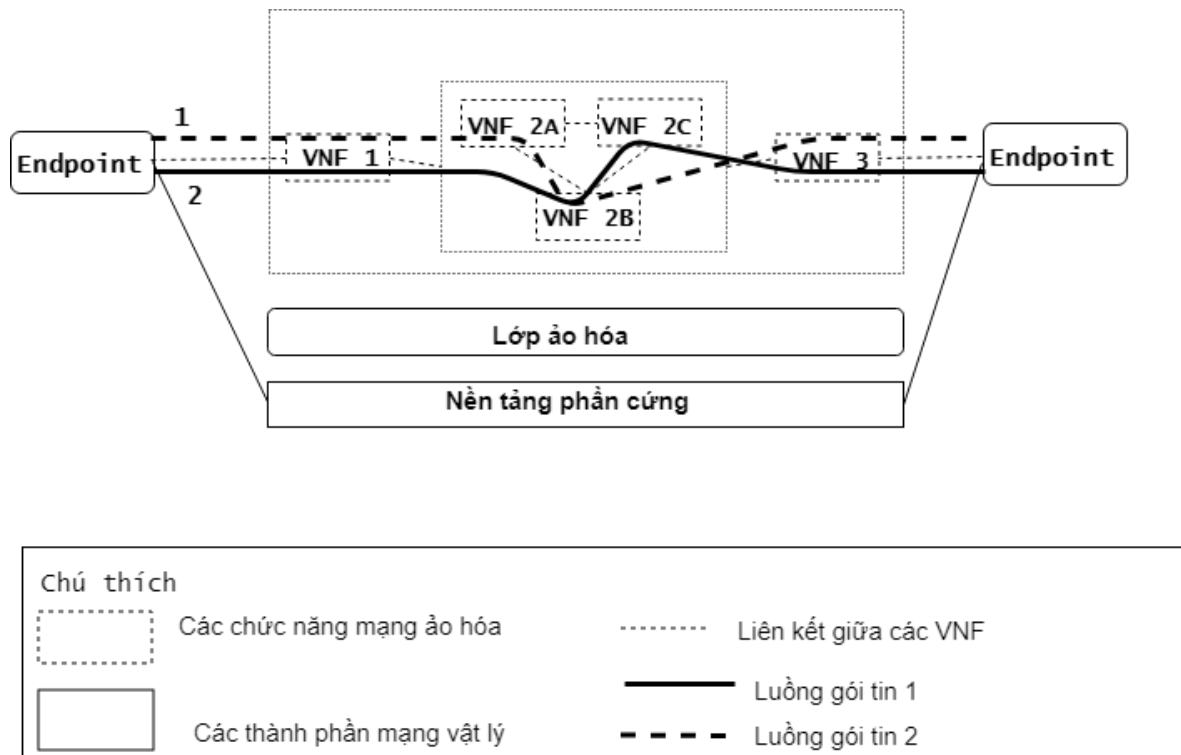
Hình 2.6: Các doanh nghiệp sử dụng chung nền tảng để thực hiện phát triển các dịch vụ mạng của chính mình.

Như mô tả trên Hình 2.6: nhà cung cấp dịch vụ (màu xanh) cung cấp các dịch vụ VNF cho nhiều doanh nghiệp khác nhau trên cùng một kiến trúc cơ sở hạ tầng.

4) Kết nối các chức năng mạng theo mô hình (VNF Forwarding Graphs)

Sử dụng công nghệ ảo hóa chức năng mạng để định nghĩa thứ tự xử lý gói tin theo ý. Một dịch vụ mạng đơn giản có thể được thực hiện trong môi trường NFV sử dụng các liên kết điểm điểm đã định sẵn hướng đi. Để thực hiện được các cấu trúc mạng với lưu lượng đi theo những luồng phức tạp cần tới VNF Forwarding Graph (hay còn gọi là chuỗi các chức năng mạng).

Các đồ hình chuyển tiếp của VNF cung cấp kết nối logic giữa các chức năng mạng ảo, cho phép kích hoạt điều kiện thứ tự di chuyển của luồng lưu lượng giữa các con NFV đó để tạo ra các dịch vụ mạng hoàn chỉnh có quy tắc xử lý lưu lượng tách biệt cụ thể. Ví dụ: trên nền tảng phần cứng vật lý, triển khai NFV ảo hóa các chức năng mạng và tạo ra hai luồng lưu lượng khác nhau cùng đi qua các con NFV như Hình 2.7:



Hình 2.7: VNF Forwarding Graph

5, 6) Ảo hóa mạng lõi di động, IMS và mạng di động base station:

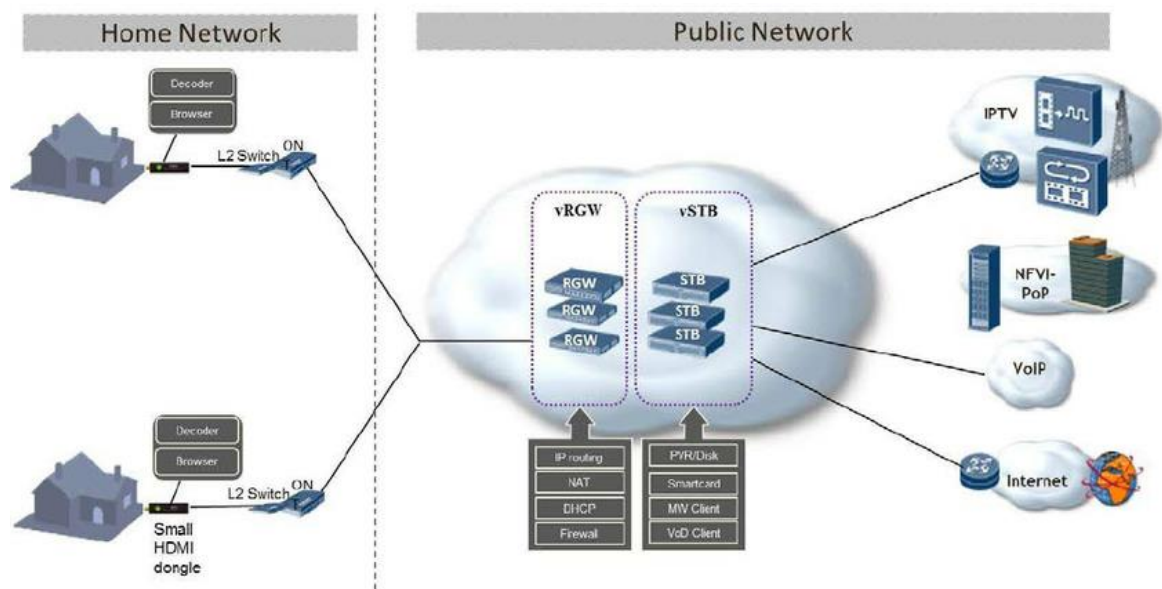
Trong mạng của các nhà khai thác di động lớn, nhiều node mạng truy cập vô tuyến từ nhiều nhà cung cấp khác nhau thường vận hành trong các kiểu mạng di động khác nhau: 3G, 4G LTE và WiMAX, trong cùng một khu vực. Cho phép các nhà mạng viễn thông, di động ảo hóa hạ tầng triển khai bên dưới để giảm chi phí, tăng hiệu năng và tính sẵn sàng, linh hoạt của hệ thống.

7) Ảo hóa trong môi trường mạng gia đình

Công nghệ NFV trở thành ứng cử viên lý tưởng để hỗ trợ khối công việc tính toán từ các chức năng phân tán trước đây với chi phí tối thiểu và đẩy nhanh thời gian

đưa sản phẩm ra thị trường, trong khi các dịch vụ mới có thể được giới thiệu theo yêu cầu ngày càng tăng của khách hàng. Các lợi ích thu được từ việc tránh lắp đặt các thiết bị mới sẽ được nhân lên nếu môi trường mạng gia đình tiếp cận NFV bằng cách thích hợp.

Nhà cung cấp dịch vụ thực hiện ảo hóa một phần hoặc toàn bộ các thiết bị thu nhận/giải mã tín hiệu (Gateway, Modem, Router, ...), các thiết bị CPE chuyên dụng trong mạng gia đình (Setup Box như đầu thu truyền hình Internet). Khách hàng là người dùng cuối trong mạng gia đình không còn cần phải quan tâm tới các bước cấu hình phức tạp cũng như cách thức duy trì và bảo dưỡng các thiết bị đó nữa.



Hình 2.8: Ứng dụng của NFV trong môi trường mạng gia đình ^[2]

Hình 2.8 minh họa ứng dụng của NFV với trường hợp sử dụng trong mạng gia đình. Để sử dụng các dịch vụ như Firewall, DHCP, định tuyến, ... thì khách hàng là các hộ gia đình chỉ cần sử dụng một thiết bị chuyển mạch lớp 2 (CPE từ phía nhà cung cấp) mà không cần phải thực hiện quản lý và cấu hình qua nhiều thiết bị khác nữa.

8) Ảo hóa trong mạng CDN (Content Delivery Networks)

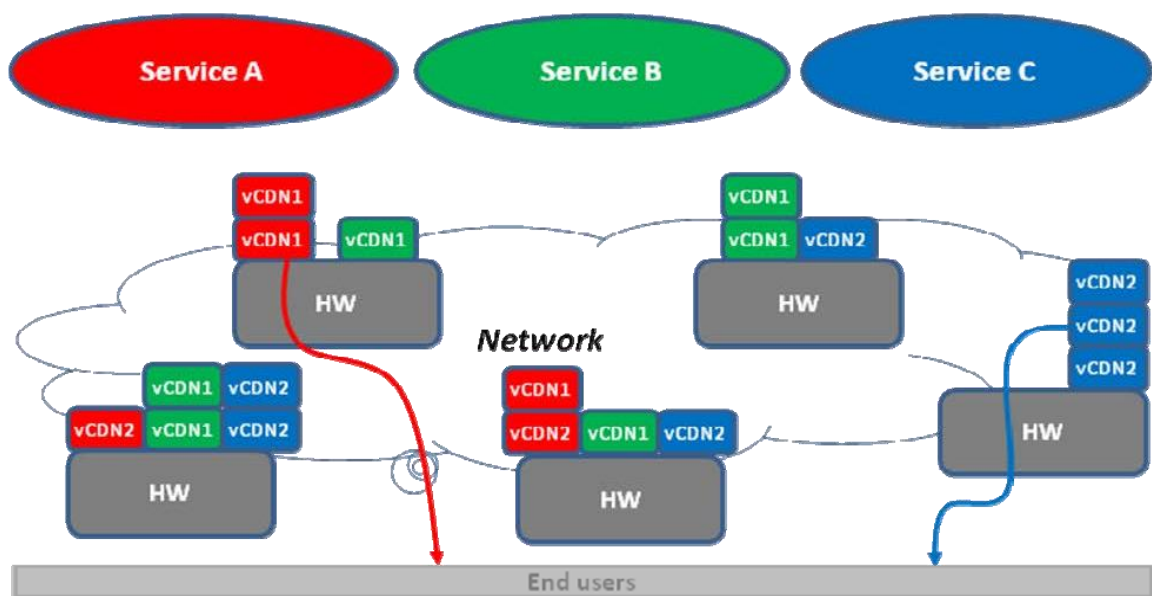
Việc cung cấp nội dung dữ liệu, đặc biệt là video - là một trong những thách thức lớn với tất cả các nhà vận hành mạng do phải phân phối tới khách hàng cuối lưu lượng truy cập ngày càng lớn. Sự gia tăng của lưu lượng video là do sự thay đổi từ

phát broadcast sang unicast, do các thiết bị được sử dụng để trình diễn video và bởi cách tăng chất lượng video ở độ phân giải và tốc độ khung hình. Hơn nữa, các yêu cầu về chất lượng cũng đang phát triển: Dịch vụ phát nội dung trực tiếp và phát theo yêu cầu cho người dùng cuối.

CDN là mạng phân phối nội dung hoạt động bởi một bộ điều khiển CDN (CDN controller) tập trung và các nút cache phân tán trong mạng. CDN controller có nhiệm vụ chọn nút cache phù hợp để trả lời request của người dùng, sau đó chuyển hướng request tới nút cache đã chọn. Nút cache sẽ trả lời request của người dùng cuối và cung cấp nội dung được yêu cầu.

Trong các mô hình triển khai hiện nay, các node CDN cache là các thiết bị phần cứng hoặc phần mềm vật lý chuyên dụng với các yêu cầu cụ thể về tiêu chuẩn phần cứng. Thông thường, các thiết bị và máy chủ vật lý cho các mục đích khác nhau được triển khai song song. Việc này dẫn tới nhiều hạn chế.

Việc tích hợp các node trong mạng CDN vào các mạng của nhà vận hành có thể là một cách hiệu quả và tiết kiệm chi phí để giải quyết những thách thức của việc phân phối lưu lượng truy cập video. Triển khai các nút CDN như một phần ảo trong môi trường ảo hóa tiêu chuẩn sẽ giải quyết được hầu hết các thách thức trên.



Hình 2.9: Các nút CDN ảo hóa được triển khai trong một môi trường ảo hóa ^[2]

Như trên Hình 2.9, các nút CDN phục vụ cho các dịch vụ khác nhau được triển khai ảo hóa trên cùng một thiết bị phần cứng để tiết kiệm chi phí.

9) Ảo hóa các chức năng mạng lớp truy cập cố định - Fixed Access Network Functions Virtualization

Áp dụng NFV sẽ tối ưu tài nguyên kết nối trên các đường truyền cố định lớp Access bằng cách tăng hiệu năng sử dụng, giảm tiêu thụ năng lượng giữa các nút mạng trên đường truyền, tạo ra một nền tảng thống nhất cho các ứng dụng, người dùng và các thuê bao khác nhau. Từ đó, các nhà cung cấp dịch vụ chia sẻ một pool chung quản lý các tài nguyên kết nối và có thể tự động cấp phát và kết hợp các mô hình triển khai tùy theo yêu cầu dịch vụ khách hàng. Tài nguyên băng thông rộng sẽ được triển khai hiệu quả và sẽ là mô hình kinh doanh mới của các nhà cung cấp.

Hiện nay, tại Việt Nam, việc ứng dụng NFV vào thực tiễn có thể giải quyết các bài toán cụ thể sau:

- Đối với các nhà cung cấp dịch vụ Internet và viễn thông như Viettel, VNPT, FPT, CMC,...
- Ảo hóa hạ tầng mạng Mobile Core Network, IMS và Mobile Base Station kết hợp với công nghệ SDN: triển khai, quản lý các dịch vụ viễn thông di động dễ dàng, nhanh chóng, linh hoạt và tối ưu hơn trên nền hạ tầng phần cứng COTS (Commercial off the Shell).
- Fixed Access Network Functions Virtualization: Tối ưu hóa việc truyền dẫn cũng như cắt giảm chi phí nếu có thể chia sẻ tài nguyên về hạ tầng kết nối với nhau.
- Virtualization of Home Environment: quản lý tốt hơn dịch vụ Internet, Thoại, Truyền hình Internet đến người dùng. Cắt giảm chi phí triển khai, bảo trì, sửa chữa, đào tạo con người cho các thiết bị ở tại nhà khách hàng.
- Các công ty kinh doanh nội dung số đa phương tiện như: VNG, FilmPlus, K+,...
 - CDN: tối ưu việc xây dựng hạ tầng CDN cho dịch vụ của mình.

- Sử dụng VNF as a Service như: Load balance, Firewall của các nhà cung cấp dịch vụ khác.
- Đối với các công ty cung cấp dịch vụ máy chủ và dịch vụ public cloud như: Vinahost, Vinadata, CMC, vHost,....
- NFVIaaS hoặc Virtual Network Platform as a Service: xây dựng cloud để cho các nhà cung cấp dịch vụ sử dụng để họ tự xây dựng dịch vụ mạng của riêng mình.
- VNF as a Service cung cấp dịch vụ mạng như: Load balance, Firewall,... cho các doanh nghiệp khác.

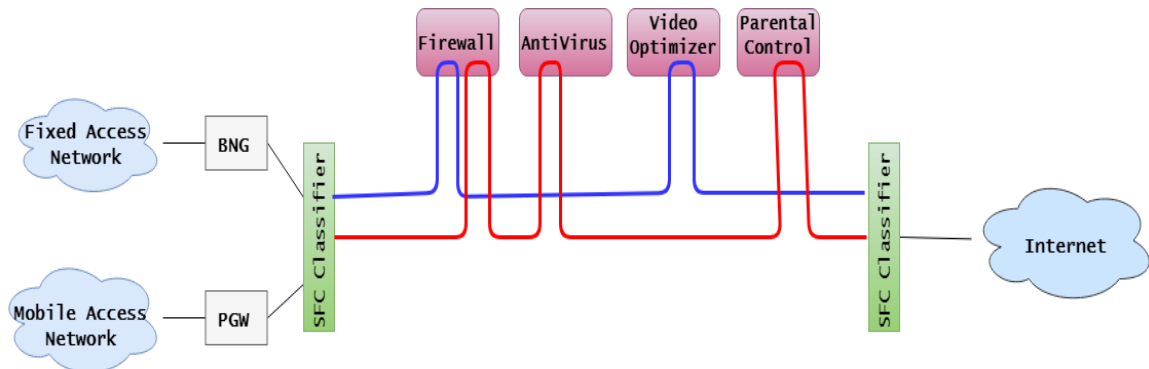
2.4. Tổng quan về chuỗi chức năng mạng – Service Function Chaining (SFC)

2.4.1. Giới thiệu về chuỗi chức năng mạng

Với sự phát triển nhanh chóng của Internet và các công nghệ như thời điểm hiện nay, việc phát triển thêm các dịch vụ giá trị gia tăng (VAS - Value-Added Services) cho các khách hàng của mình dần trở thành ưu tiên hàng đầu của các nhà cung cấp dịch vụ. Thông thường, luồng lưu lượng được chuyển tiếp thông qua các thành phần mạng đã được nhúng trong các dịch vụ:

- Chuyển hướng trực tiếp một phần lưu lượng tới các thành phần mạng có chức năng giám sát và tính phí.
- Trước khi gửi lưu lượng tới các máy chủ trong trung tâm dữ liệu (DC - Data Center), điều chỉnh lưu lượng đi qua một bộ cân bằng tải để phân phối luồng lưu lượng đi qua nhiều liên kết, qua các chức năng mạng để giảm tải.
- Các nhà vận hành mạng di động chia lưu lượng truy cập băng thông rộng và điều khiển chúng để giảm tải đường xuống.
- Sử dụng chức năng tường lửa để lọc các lưu lượng cho hệ thống IDS/IPS (Intrusion Detection System/Intrusion Protection System).
- Sử dụng các cổng bảo mật để mã hóa và giải mã lưu lượng, có thể kích hoạt các tính năng giảm tải SSL.
- Nếu lưu lượng đến từ các mạng hỗ trợ các địa chỉ khác nhau, ví dụ như IPv4 hoặc IPv6 thì chuyển hướng trực tiếp tới CGN (Carrier Grade NAT) hoặc NAT64.
- Một vài nền tảng dịch vụ nội bộ dựa vào nhận dạng dịch vụ ngầm. Chức năng dịch vụ chuyên dụng được kích hoạt để thêm vào các gói tin (ví dụ: thêm vào HTTP header) với định danh thuê bao hoặc thiết bị người dùng UE (User Equipment).
- Nhà cung cấp cho phép các dịch vụ giá trị gia tăng trên cơ sở từng thuê bao. Mong muốn điều khiển luồng lưu lượng truy cập tới từ các thuê bao đã đăng kí VAS được sử dụng các nền tảng dịch vụ có liên quan.

Chuỗi chức năng dịch vụ (Service Function Chaining - SFC) là công nghệ quy định các chính sách chuyển tiếp lưu lượng tách biệt trong vùng chuỗi dịch vụ. Chuỗi các chức năng dịch vụ mạng SFC định nghĩa và tạo ra một chuỗi các dịch vụ mạng theo thứ tự mong muốn và điều khiển lưu lượng truy cập thông qua chúng.



Hình 2.10: Một chuỗi SFC đơn giản

Hình 2.10 ví dụ về một chuỗi chức năng mạng đơn giản gồm các chức năng Firewall, chống virus, tối ưu hóa video và chức năng điều khiển của phụ huynh (Parental Control).

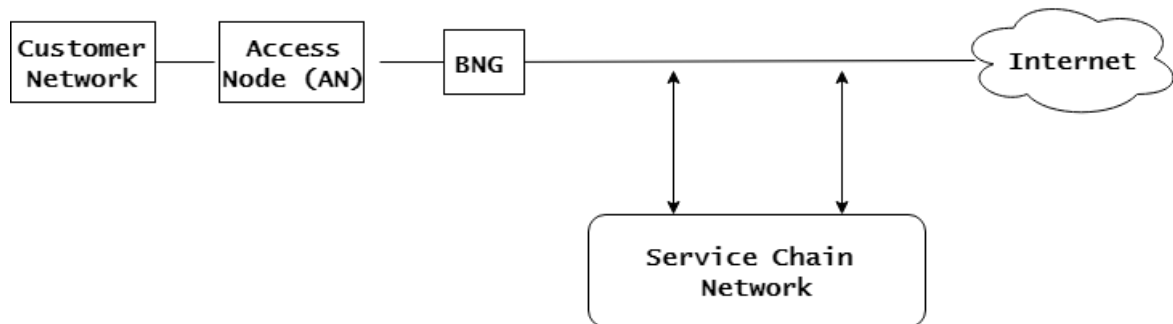
2.4.2. Ứng dụng của SFC trong một số trường hợp cụ thể

Chuỗi chức năng dịch vụ mạng có thể được triển khai trong nhiều mô hình khác nhau như các mạng băng thông rộng (broadband network), các mạng di động (mobile network), và trong các trung tâm dữ liệu (Data Center). Phần này sẽ nêu khái quát một số trường hợp triển khai sử dụng SFC.

1) Ứng dụng SFC vào mạng băng thông rộng cố định (Fixed Broadband Network)

Trong mạng băng thông rộng cố định, người dùng có thể truy cập vào mạng thông qua nhiều công nghệ khác nhau, thường là DSL, Ethernet và PON. Cho dù là bất kỳ cách thức truy cập nào thì kiến trúc cho phép truy cập và các mạng ngầm tương tự đều gồm các nút truy cập (Access Nodes - ANs) và các Broadband Network Gateway(BNG), trong đó AN thường là các thiết bị cho phép người dùng truy cập

mạng và BNG là nút IP đầu tiên cung cấp khả năng xác thực, ủy quyền và tính toán cho người dùng.

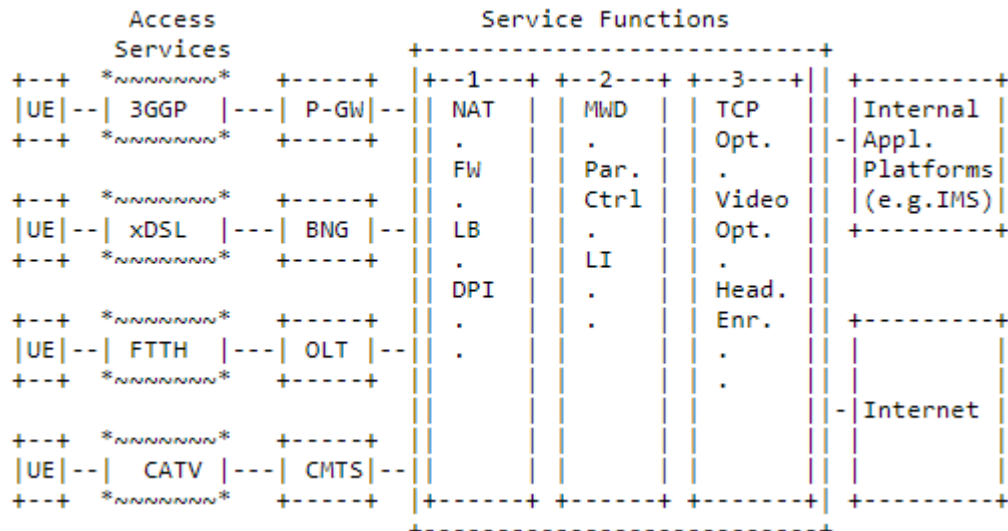


Hình 2. 11: Ứng dụng của SFC trong mạng Fixed Broadband Network

Hình 2.11 mô tả mô hình sử dụng SFC trong mạng băng thông rộng cố định. Chuỗi các dịch vụ mạng được triển khai ngay sau BNG và trước khi ra mạng Internet. Chuỗi dịch vụ có thể bao gồm một số dịch vụ như DPI, DS-Lite, Parental control, Firewall, Cân bằng tải (Load balancer), Cache, ...

2) Ứng dụng SFC vào mạng di động

Lưu lượng được chuyển trực tiếp tới/từ Internet đến một hoặc nhiều chức năng dịch vụ mạng. Đó có thể là các chức năng dịch vụ giá trị gia tăng như dịch vụ kiểm soát (Parental Control) hoặc Tường lửa. Các thuê bao có thể chủ động đăng kí hoặc hủy bỏ các dịch vụ này bất kì lúc nào mà không cần tới sự can thiệp với nhà cung cấp dịch vụ. Một số chức năng dịch vụ cơ bản có thể kể đến như: DPI (Deep Packet Intrusion), thanh toán và tính phí, tối ưu hóa TCP tối ưu hóa Web, tối ưu hóa video ...



Hình 2.12: Triển khai SFC trong mạng di động ^[4]

Hình 2.12 là một ví dụ về triển khai SFC trong các mạng: thiết bị người dùng (User Equipment - UE) truy cập vào mạng di động, luồng lưu lượng được dẫn tới cơ sở hạ tầng có chuỗi các dịch vụ và cuối cùng là kết nối với các nền tảng ứng dụng trên Internet hoặc trong trung tâm dữ liệu riêng của nhà cung cấp. Từ trên xuống dưới có một mạng di động 3GPP kết thúc tại P-GW, một mạng xDSL với đường truyền PPP kết thúc tại BNG, một mạng FTTH và cuối cùng là một mạng truyền hình cáp (CATV).

Các chuỗi dịch vụ khác nhau được thực hiện tùy vào nhu cầu của từng loại mạng khác nhau. Chuỗi 1 và 2 có thể áp cho tất cả các mạng trên, nhưng chuỗi 3 với các chức năng tối ưu hóa dữ liệu có thể sẽ chỉ áp dụng với mạng di động.

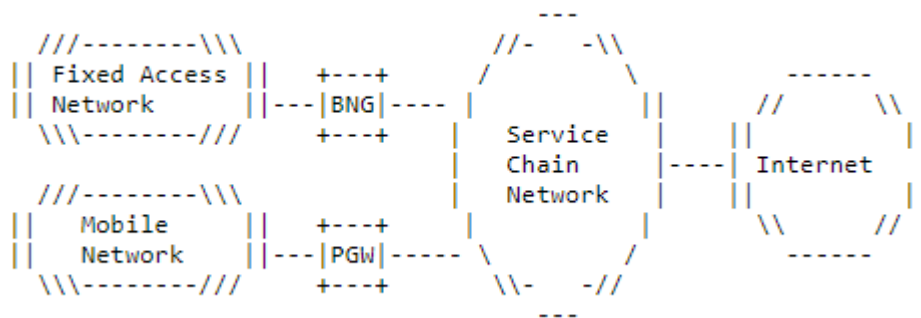
Các chức năng dịch vụ này có thể đặt trên cùng một thiết bị hoặc đặt trong các hộp độc lập. Để nâng cao chất lượng dịch vụ và trải nghiệm người dùng qua các dịch vụ giá trị gia tăng sáng tạo, đáp ứng nhu cầu kinh doanh của các nhà cung cấp, các chức năng dịch vụ ngày càng phát triển.

3) Mạng chuỗi dịch vụ - Công cụ hội tụ mạng

Từ hai trường hợp sử dụng SFC trên, ta có thể thấy sự tương đồng trong chuỗi dịch vụ mạng. Mặc dù mạng băng thông rộng cố định và mạng di động được triển khai tách biệt, với các nhà khai thác tích hợp chạy cả hai mạng này rõ ràng là có lợi khi cung cấp chuỗi dịch vụ dùng chung cho cả hai mạng này.

Ngoài việc tối ưu hóa tài nguyên, một mạng chuỗi các dịch vụ phổ biến có thể cho phép chuyển đổi dịch vụ liên mạch từ mạng này tới mạng khác. Ví dụ: Một khách hàng đang xem trò chơi bóng đá trên điện thoại của mình thông qua mạng di động 4G. Sau khi về nhà, anh ta có thể chuyển qua sử dụng mạng wifi trong nhà của mình - mạng bây giờ chuyển sang mạng cáp quang FTTH (Fiber To The Home) băng thông 100Mbps. Trong trường hợp này, thật dễ dàng khi cung cấp các dịch vụ từ cùng một mạng chuỗi các dịch vụ.

SFC có thể được sử dụng như một công cụ để giải quyết tốt hơn nhu cầu hội tụ.



Hình 2.13: Các mạng dùng chung chuỗi dịch vụ mạng ^[4]

Hình 2.13 minh họa một mạng chuỗi dịch vụ dùng chung được chia sẻ bởi cả mạng băng rộng cố định và mạng di động. Chuỗi dịch vụ mạng dùng chung có thể được triển khai chỉ bao gồm các nút mạng có chức năng cụ thể hoặc trong trung tâm dữ liệu. Trong cả hai trường hợp, các nút dịch vụ, dù là vật lý hay ảo hóa, đều được chia sẻ bởi cả mạng có dây và mạng không dây. Các nhà khai thác quản lý chuỗi dịch vụ dùng chung cho cả hai mạng và lưu lượng truy cập từ cả hai mạng có thể đi qua cùng một chuỗi dịch vụ.

4) Chuỗi chức năng dịch vụ phân tán

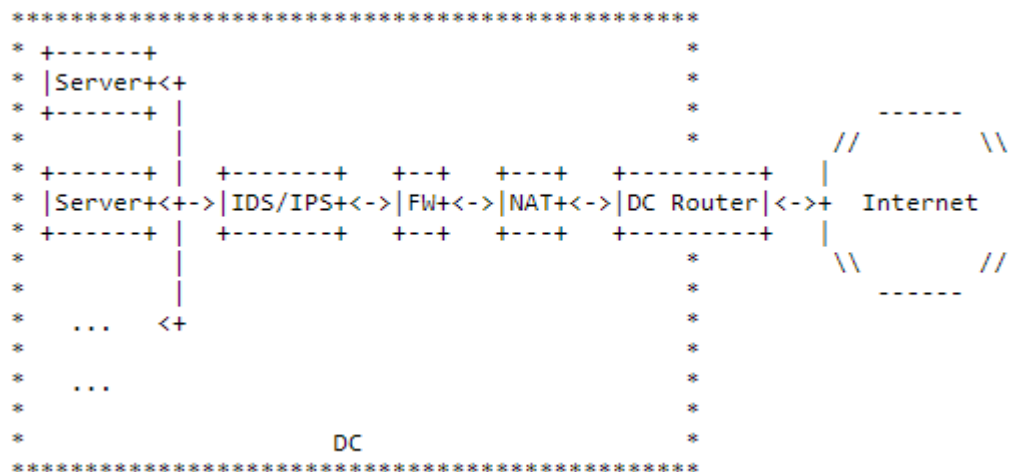
Bên cạnh các trường hợp triển khai kể trên, một chuỗi dịch vụ mạng có thể không cần thiết phải triển khai trong cùng một vị trí mà có thể phân phối qua một số phần của mạng (Data center) hoặc thậm chí sử dụng chức năng dịch vụ đặt gần khách hàng.

Có thể kích hoạt nhiều miền SFC trong cùng một miền quản trị.

5) Triển khai SFC trong trung tâm dữ liệu

Trong trung tâm dữ liệu (DC), như mạng băng rộng và mạng di động, các chuỗi dịch vụ mạng cũng có thể triển khai để cung cấp các dịch vụ giá trị gia tăng.

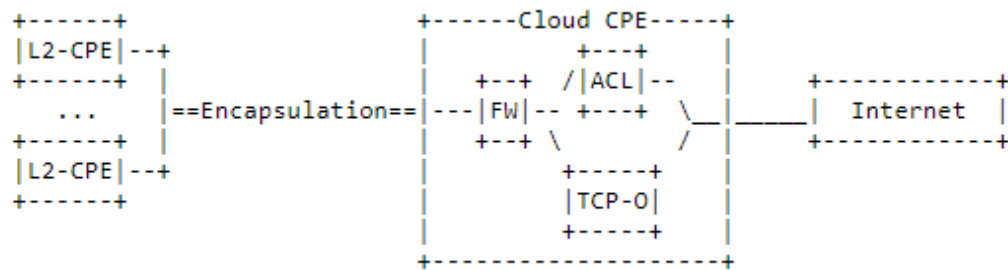
Hình 2.14 mô tả tình huống triển khai SFC trong trung tâm dữ liệu: Các chức năng mạng được đặt giữa router của DC và các máy chủ server. Từ server truy cập ra Internet, có nhiều chức năng dịch vụ như IDS/IPS, Firewall, NAT được sắp xếp tạo thành một SFC nguyên khối xử lý tất cả lưu lượng truy cập tới.



Hình 2.14: SFC trong trung tâm dữ liệu ^[4]

6) Chuỗi chức năng mạng trong các thiết bị Cloud CPE

Cloud CPE là một kịch bản triển khai mà tập trung vào các dịch vụ giá trị gia tăng (được lưu trữ trong mạng hoặc phía cloud), tách khỏi các hộp dịch vụ phía thuê bao với thiết bị chức năng lớp 2/lớp 3 cơ bản. Trong trường hợp này, tất cả các dịch vụ giá trị gia tăng đều được cấu hình bởi người dùng và được kích hoạt ở phía mạng. Người dùng có thể tự định nghĩa các dịch vụ giá trị gia tăng của riêng họ. Thiết bị Cloud CPE sẽ chuyển các yêu cầu đó thành chuỗi các dịch vụ chức năng. Kiến trúc như vậy phải được hỗ trợ để phân biệt các người dùng khác nhau và lưu lượng của họ.



Hình 2.15: SFC trong các thiết bị Cloud CPE ^[4]

2.5. Kết luận và đưa ra định hướng giải pháp xây dựng đề tài đồ án

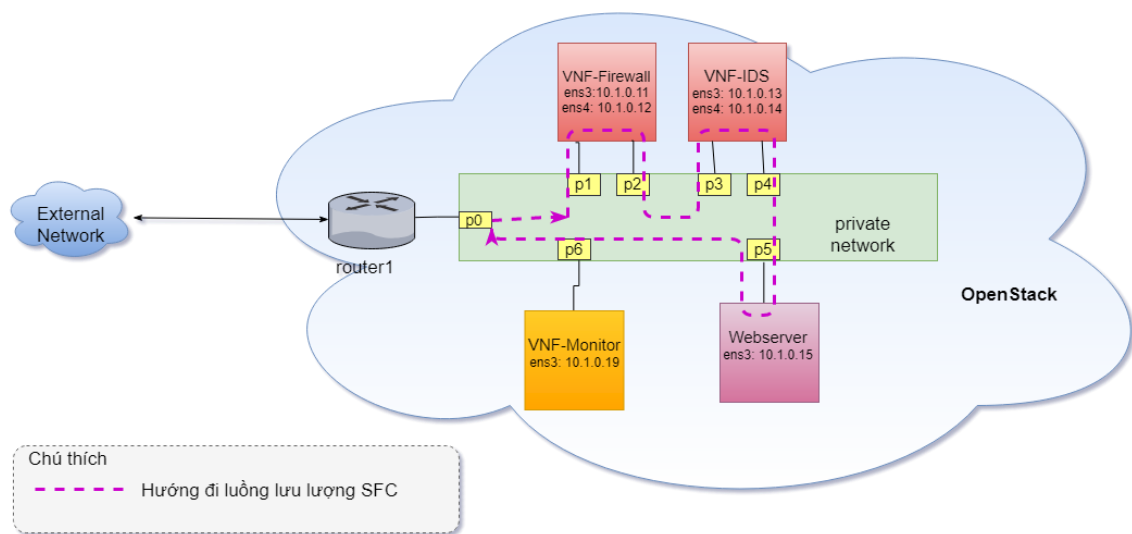
Chuỗi chức năng mạng hiện nay phần lớn vẫn sử dụng các thiết bị chuyên dụng, dẫn đến tốn kém về chi phí và phụ thuộc vào nhà sản xuất. NFV ra đời đã giải quyết được vấn đề này bằng việc thay vì phải sử dụng các thiết bị chuyên dụng để làm các chức năng mạng, chúng ta có thể sử dụng các server vật lý được cài các phần mềm giả lập các chức năng mạng như firewall, loadbalance,...

Khi NFV kết hợp với điều khiển luồng sẽ tạo thành chuỗi chức năng mạng SFC linh hoạt, có khả năng quản lý tập trung mạnh mẽ. Qua đó, các nhà mạng có thể tiết kiệm chi phí vận hành, quản lý và cũng bớt lệ thuộc vào các nhà sản xuất thiết bị mạng.

Phần Chương 2 đã trình bày tổng quan về công nghệ ảo hóa chức năng mạng và Chuỗi chức năng mạng, lợi ích cũng như một số trường hợp ứng dụng của hai công nghệ trên. Qua các trường hợp sử dụng NFV và SFC vào hạ tầng mạng đã trình bày ở trên, em lựa chọn xây dựng chuỗi các chức năng mạng ảo hóa bảo vệ các ứng dụng và máy chủ phía nhà cung cấp (hay trung tâm dữ liệu), và cụ thể trong đề án này là bảo vệ máy chủ web server. Chuỗi chức năng mạng sẽ bao gồm 2 thành phần chính là chức năng mạng Firewall có tác dụng chặn, lọc các gói ở lớp 3 và lớp 4; chức năng hệ thống phát hiện xâm nhập NIDS có tác dụng phát hiện các hành vi khả nghi tấn công vào một mạng. Chuỗi chức năng sẽ bảo vệ máy chủ web server bên trong.

CHƯƠNG 3: XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG

Chương này mô tả phân xây dựng testbed chuỗi chức năng mạng đã đề xuất ở chương 2. Xây dựng mô hình chuỗi chức năng mạng trên nền tảng OpenStack. Chuỗi gồm hai chức năng mạng bảo vệ cho máy chủ Webserver bên trong. Các chức năng mạng đều là máy ảo được tạo bởi OpenStack.



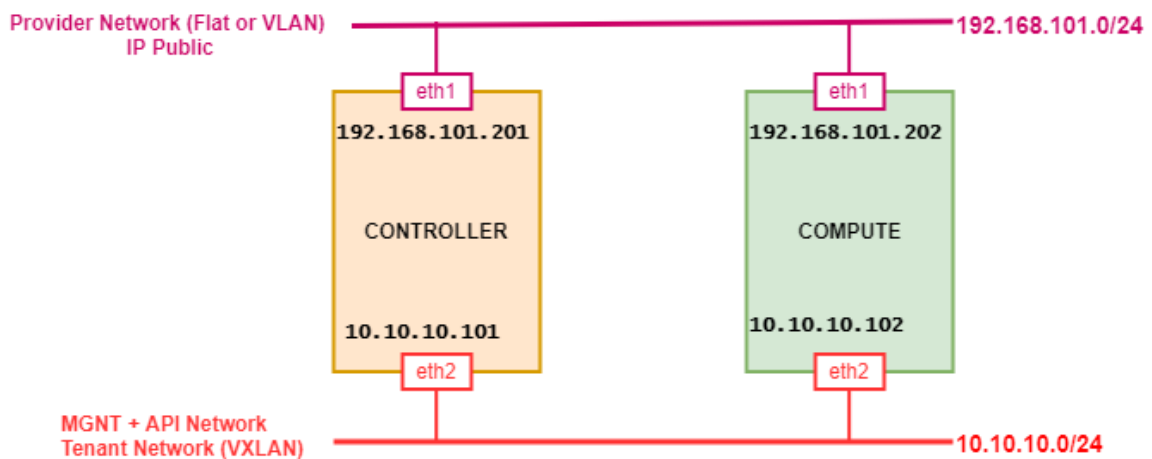
Hình 3.1: Mô hình testbed mức logic

Hình 3.1 mô tả đồ hình testbed mức logic. Chuỗi chức năng mạng triển khai gồm hai chức năng mạng ảo Tường lửa (Firewall) và Hệ thống phát hiện xâm nhập (IDS). Phần tiếp theo trình bày các bước dựng testbed.

3.1. Xây dựng nền tảng cloud quản lý các chức năng mạng ảo hóa – OpenStack

3.1.1. Dựng OpenStack để triển khai NFV

Mô hình triển khai OpenStack gồm nút chính là Controller và Compute. Đồ hình mức logic như sau:



Hình 3.2: Mô hình triển khai OpenStack mức logic

Hệ thống kiểm thử trên mô hình thực tế:

- **Controller:** Là máy chủ server, cài đặt các project: Keystone, Glance, Nova, Neutron, Horizon. Có nhiệm vụ kiểm soát các chức năng chính của OpenStack. Khởi tạo và quản lý các tài nguyên trên máy chủ, cho phép tạo các máy ảo đóng vai trò các khối NFV. Yêu cầu tối thiểu 16G RAM, 500G ổ cứng, 8 core CPU.
- **Compute:** là một máy chủ server khác, cài đặt các project Neutron, Nova. Có chức năng cung cấp các tài nguyên tính toán cho các khối chức năng mạng. Yêu cầu tối thiểu 10G RAM, 100G ổ cứng, 10 core CPU.
- Về hạ tầng mạng gồm 2 dải như sau:
 - Dải Provider Network: cung cấp kết nối ra mạng bên ngoài cho toàn bộ hệ thống và các chức năng mạng.
 - Dải MGNT + API network (Management): cung cấp kết nối giữa các khối trong OpenStack, quản lý và cấp phát dịch vụ mạng cho các VM bên trong.

2 dải này được cấu hình trên 2 VLAN khác nhau của switch vật lý để tiết kiệm chi phí.

3.2. Xây dựng luồng đi chuỗi chức năng mạng SFC

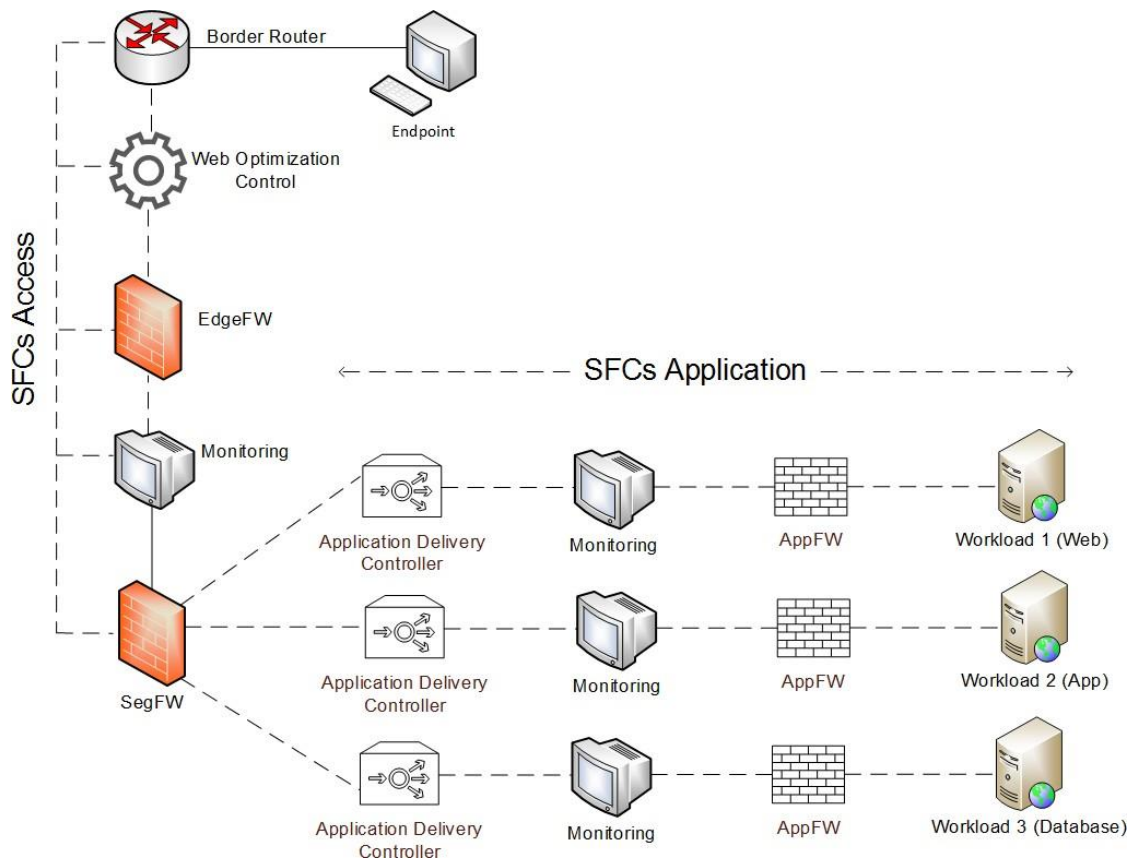
3.2.1. SFC trong trung tâm dữ liệu

Trong trung tâm dữ liệu, nhiều chức năng dịch vụ từ lớp 4 tới lớp 7 được triển khai trên cả thiết bị vật lý và ảo hóa.

Các trung tâm dữ liệu - các doanh nghiệp lớn, cloud hoặc các nhà cung cấp dịch vụ - triển khai các nút dịch vụ tại nhiều điểm khác nhau trong mô hình mạng. Những nút này cung cấp một loạt các chức năng dịch vụ và thiết lập các chức năng dịch vụ được lưu trữ tại một nút nhất định hoặc chồng lấp với các chức năng dịch vụ đã được lưu trữ tại các nút dịch vụ khác.

Ứng dụng SFC trong trung tâm dữ liệu gồm ứng dụng cho 3 kiểu lưu lượng sau:

- Lưu lượng từ bên ngoài truy cập vào trung tâm dữ liệu: là các lưu lượng của người dùng cuối truy cập tới dịch vụ của họ trên trung tâm dữ liệu. Đó có thể là lưu lượng truy cập web, đọc tin tức, mạng xã hội và email. Sự gia tăng xu hướng mang mọi thứ tới thiết bị của bạn (Bring Your Own Device - BYOD) và các ứng dụng mạng xã hội yêu cầu lưu lượng phải được phân tích, người dùng phải được xác thực và ủy quyền, nội dung dữ liệu cần được tối ưu hóa để tăng năng suất hoạt động. Ví dụ: lưu lượng từ ngoài vào cần đi qua các chức năng mạng: Firewall, NAT, các chức năng tối ưu hóa nội dung truyền tải, ...
- Lưu lượng truy cập nội bộ trong trung tâm dữ liệu: là loại lưu lượng chính trong trung tâm dữ liệu, kết nối các nút lại với nhau.
- Môi trường đa người dùng: hỗ trợ môi trường nhiều khách hàng là yêu cầu với mọi trung tâm dữ liệu.



Hình 3.3: Chuỗi các chức năng mạng trong trung tâm dữ liệu

3.2.2. Công nghệ xây dựng chuỗi chức năng mạng trên OpenStack

Trong OpenStack, các máy ảo VM kết nối vào một mạng ảo thông qua các port. Điều này cho phép tạo sử dụng mô hình điều khiển lưu lượng đối với SFC sử dụng các port. Việc kết nối các port này trong một chuỗi các port cho phép điều khiển lưu lượng đi qua một hoặc nhiều VM đóng vai trò là chức năng mạng (service function - SF).

Một chuỗi các port tương đương khái niệm Service Function Path (SFP) bao gồm:

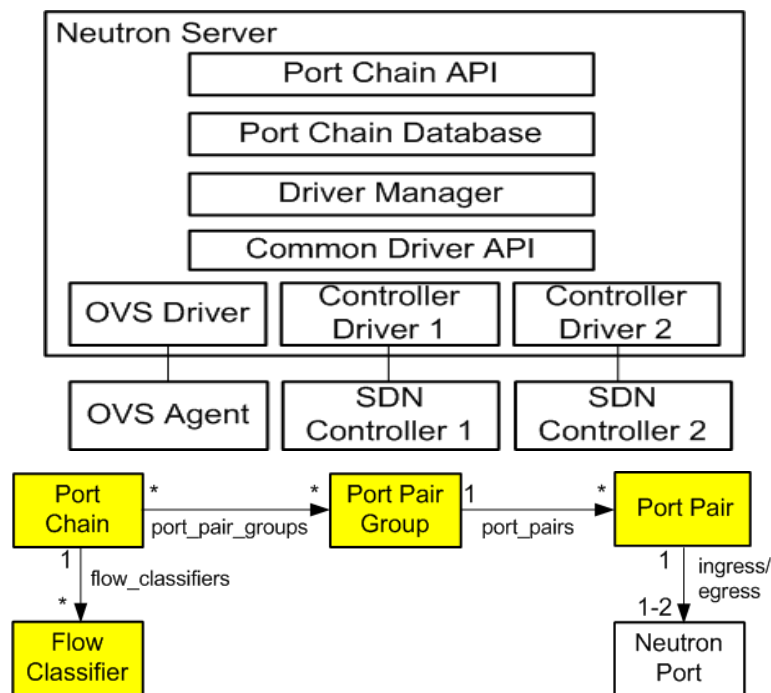
- Một tập các port định nghĩa nên trình tự các chức năng dịch vụ.
- Một tập các flow classifiers (bộ phân loại các luồng): chỉ định các luồng lưu lượng đi vào chuỗi.

Nếu một SF gắn với một cặp port thì phải chỉ rõ ingress port và egress port của SF đó. SF cho phép gắn với một port và port đó đóng cả hai vai trò, coi như là một port cho phép lưu lượng đi theo hai chiều vào và ra.

Một chuỗi các port (port chain) được coi là một service chain vô hướng. Một SFC bao gồm hai port chain vô hướng là SFC hai chiều.

Một flow classifier chỉ thuộc về một port chain để tránh việc bối rối khi hệ thống quyết định xem chain nào sẽ xử lý các gói tin. Một port chain có thể gắn với nhiều classifier vì nhiều loại lưu lượng có thể yêu cầu cùng một SFP.

Project networking-sfc là project con của neutron, triển khai port chain plug-in với Open vSwitch driver và SDN Controller drivers (networking-odl hoặc networking-onos) cho phép tương tác với các SFC provider khác nhau (Open vSwitch agent hoặc SDN Controller như OpenDaylight hoặc ONOS). Project này cũng cung cấp một driver API chung để hỗ trợ các driver khác nhau đó nhằm cung cấp các giải pháp khác nhau triển khai một SFP.



Hình 3.4: Cấu trúc networking-SFC trong OpenStack^[7]

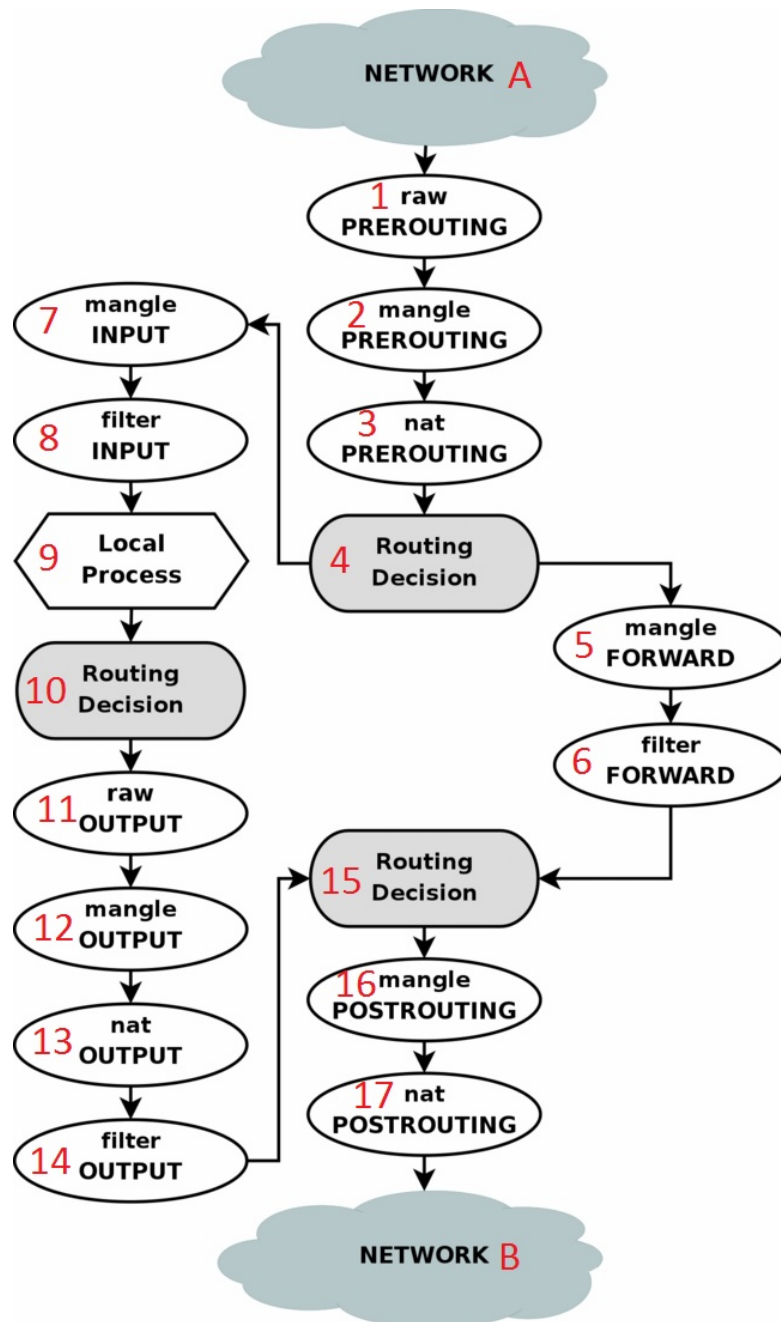
Hình 3.4 mô tả về mối quan hệ giữa các thành phần trong cấu trúc mạng Networking SFC của OpenStack.

3.3. Xác định các công nghệ sử dụng làm các chức năng mạng

3.3.1. Chức năng mạng Firewall – Iptables

Iptables là một tiện ích firewall cực kì linh hoạt được xây dựng trên các hệ điều hành linux. Iptables là chương trình chạy ở không gian người dùng (user space) cho phép người quản trị hệ thống cấu hình các quy tắc để chặn và lọc gói. Iptables miễn phí và được tích hợp sẵn trong nhân linux.

Iptables thuộc loại firewall có khả năng nhận biết được trạng thái các gói tin này. Từ đó có thể đưa ra quyết định chặn, lọc các gói tin một cách thông minh hơn. Hơn thế nữa Iptables còn hỗ trợ khả năng giới hạn tốc độ kết nối đối với các kiểu kết nối khác nhau từ bên ngoài, cực kì hữu hiệu để ngăn chặn các kiểu tấn công từ chối phục vụ (DoS) mà hiện nay vẫn là mối đe dọa hàng đầu đối với các website trên thế giới. Một đặc điểm nổi bật nữa của Iptables là nó hỗ trợ chức năng dò tìm chuỗi tương ứng (string pattern matching), chức năng cho phép phát triển firewall lên một mức cao hơn, có thể đưa ra quyết định loại bỏ hay chấp nhận packet dựa trên việc giám sát nội dung của nó. Chức năng này có thể được xem như là can thiệp được đến mức ứng dụng như HTTP, TELNET, FTP... mặc dù thực sự Netfilter Iptables vẫn chỉ hoạt động ở mức mạng (lớp 3 theo mô hình OSI 7 lớp).



Hình 3.5: Quá trình xử lý gói tin của IPtables

Quá trình xử lý gói tin của IPtables được mô tả trên Hình 3.5 như sau:

Các gói tin đi từ bên ngoài vào (1) ban đầu được xử lý thông qua chain PREROUTING (1, 2, 3). Chain này có chức năng thay đổi nguồn của gói tin (nếu cần) trước khi qua các bước xử lý tiếp theo, để hệ thống nhận biết được gói tin có phải dành cho mình hay chuyển tiếp tới các nút mạng tiếp theo.

Sau khi đi qua chain từ PREROUTING, các gói tin được kernel định tuyến (4). Sẽ có hai trường hợp:

- Nếu gói tin dành cho hệ thống (7), nó sẽ được đưa tới chain INPUT của bảng mangle (8) và filter (9) để thực hiện các quy tắc trên các bảng đó, cuối cùng được đưa tới tiến trình đảm nhận dịch vụ cho gói tin đó để xử lý.
- Nếu gói tin không dành cho các dịch vụ trên hệ thống (5), gói tin sẽ được đưa tới chain FORWARD qua các bảng mangle và filter (5, 6) để thực hiện các quy tắc trước khi chuyển tiếp gói tin sang đích chính xác của nó.

Nếu gói tin được sinh ra từ hệ thống, ban đầu chúng sẽ được định tuyến (bước 10) để xác định được địa chỉ mà nó cần chuyển đến để xác định interface mà gói tin sẽ chuyển ra. Gói tin sau đó được đi qua chain OUTPUT trên các bảng mangle, nat và filter (11, 12, 13, 14) để thực hiện các quy tắc trong các bảng đó.

Sau cùng, các gói tin được đưa tới chain POSTROUTING (đổi nguồn) (15) và gửi gói tin trở lại mạng.

3.3.2. Chức năng mạng Phát hiện xâm nhập (IDS) – Suricata

Ngày nay, khi mạng Internet đã phủ sóng khắp mọi nơi thì thách thức của các vấn đề xâm phạm và tấn công đã khiến các tổ chức phải bổ sung thêm chức năng mạng có tính năng phát hiện và kiểm tra các lỗ hổng bảo mật. Hệ thống phát hiện xâm nhập (IDS) là hệ thống phòng chống có khả năng phát hiện các hành vi khả nghi tấn công vào một mạng. Tính năng chính của hệ thống này là nhận biết những hàng động không bình thường và đưa ra cảnh báo cho người dùng.

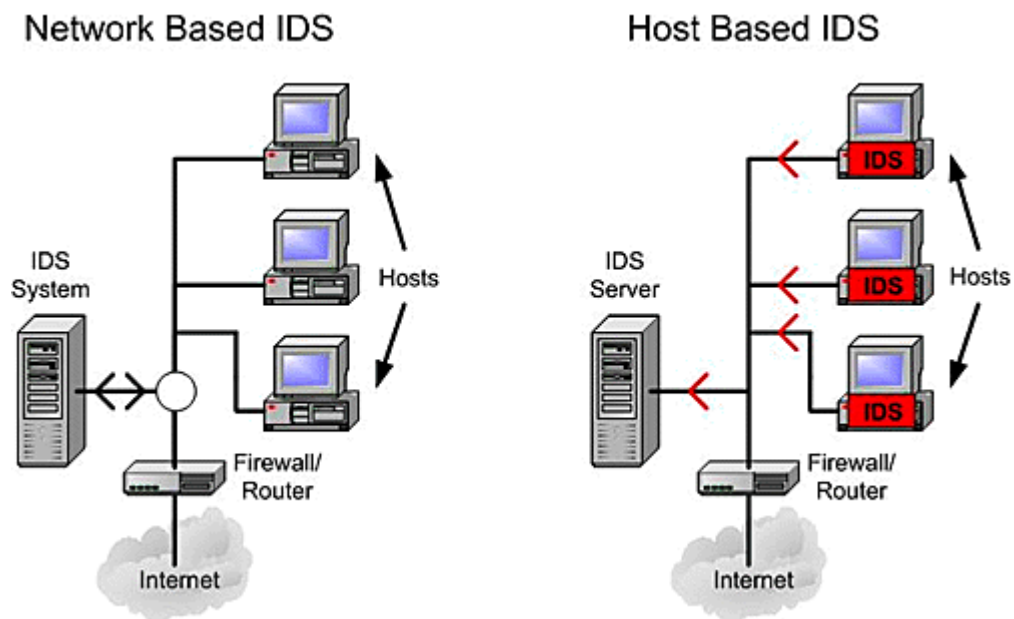
Hệ thống IDS phát hiện xâm nhập dựa trên các dấu hiệu đặc biệt về các mối nguy cơ đã biết trước đó trên các gói tin, hoặc so sánh lưu lượng mạng hiện tại với thông số hoạt động trong trạng thái bình thường để tìm ra các dấu hiệu bất thường.

Hệ thống IDS được chia thành hai loại cơ bản:

- Network-based IDS (NIDS): sử dụng dữ liệu trên toàn bộ lưu lượng trong mạng để phát hiện bất thường. Kiểu NIDS trong suốt với người dùng cuối, cài

đặt và bảo trì đơn giản, không ảnh hưởng tới mạng, có khả năng xác định lỗi ở tầng network và độc lập với hệ điều hành.

- Host based IDS – HIDS: Bằng cách cài đặt phần mềm trên máy chủ, IDS dựa trên máy chủ quan sát tất cả những hoạt động về hệ thống và các file log, lưu lượng mạng thu thập. Hệ thống HIDS theo dõi hệ điều hành, các lời gọi hệ thống và các thông điệp báo lỗi trên máy chủ. HIDS thường được đặt trên một máy tính nhất định thay vì giám sát hoạt động của một mạng, HIDS thường được đặt trên các máy chủ quan trọng và các server trong vùng DMZ.



Hình 3.6: Hai loại hệ thống IDS

Hình 3.6 mô tả về hai loại hệ thống IDS được triển khai trong thực tế: NIDS và HIDS.

Suricata là một công cụ phát hiện mối đe dọa mạng miễn phí, nguồn mở, mạnh mẽ và nhanh chóng.

Suricata có khả năng phát hiện xâm nhập thời gian thực (IDS), phòng chống xâm nhập (IPS), giám sát an ninh mạng (Network Security Monitoring - NSM) và xử lý pcap ngoại tuyến.

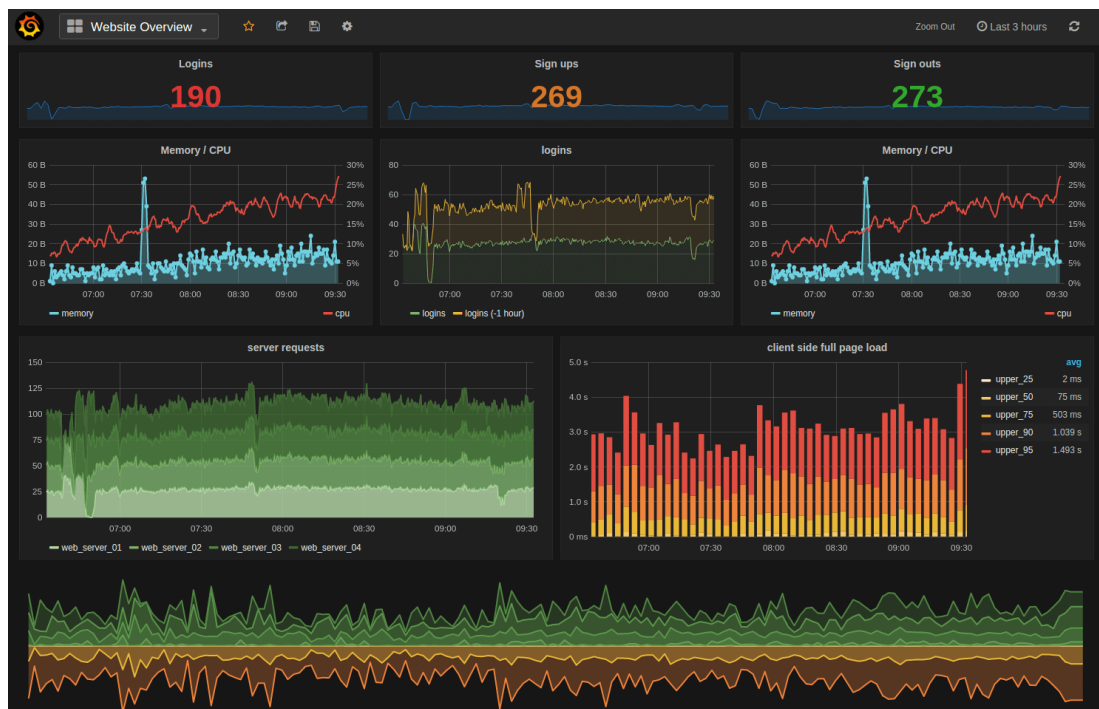
Suricata kiểm tra lưu lượng mạng bằng cách sử dụng bộ các quy tắc (rule) xử lý mạnh mẽ và có hỗ trợ kịch bản LUA để phát hiện các mối đe dọa phức tạp. Các rule của Suricata được cập nhật thường xuyên khi phát hiện các

Lý do em lựa chọn Suricata thực hiện chức năng IDS trong mô hình bởi sự phát triển nhanh chóng của Suricata dựa trên cộng đồng và tập trung vào tính bảo mật, khả năng sử dụng và hiệu quả, hỗ trợ đa tiến trình, do đó có thể sử dụng nhiều hơn 1 CPU trong cùng một thời điểm, hỗ trợ trên nhiều nền tảng hệ điều hành như Unix/Linux, FreeBSD và Windows.

3.3.3. Chức năng mạng giám sát – Grafana

Grafana là công cụ được tin tưởng và yêu thích bởi cộng đồng, là nền tảng phân tích tất cả các loại metric.

Grafana cho phép truy vấn, visualize (hiển thị), cảnh báo và giúp người quản trị hiểu metric dù chúng được lưu ở bất kì đâu. Tạo, khám phá và chia sẻ dashboard với nhóm và thúc đẩy văn hóa luồng dữ liệu.



Hình 3.7: Giám sát số liệu hệ thống với Grafana

Hình 3.7 là giao diện giám sát số liệu hệ thống hiển thị trên Grafana.

Các tính năng:

- Visualize (trực quan hóa) : Vẽ biểu đồ từ metric được cung cấp. Grafana có rất nhiều tùy chọn visualize giúp người dùng vẽ biểu đồ một cách nhanh chóng và linh hoạt. Các panel plugin với nhiều cách khác nhau để trực quan hóa các metric và log hệ thống.
- Alerting - Cảnh báo : Giúp người dùng xác định các ngưỡng metric, hiển thị ngưỡng metric cảnh báo và định nghĩa các quy tắc cảnh báo. Grafana liên tục đánh giá metric và gửi cảnh báo khi metric vượt quá ngưỡng cho phép. Cảnh báo có thể được gửi qua Slack, Mail, PagerDuty, Telegram, ...
- Unify – Hợp nhất : Kết hợp dữ liệu để có cái nhìn toàn cảnh tốt hơn. Grafana hỗ trợ hàng chục loại database một cách tự nhiên, kết hợp chúng với nhau trong cùng một giao diện dashboard.
- Open - Mở: Grafana đưa bạn nhiều tùy chọn. Nó hoàn toàn là nguồn mở, được hỗ trợ bởi cộng đồng sôi động. Có thể dễ dàng cài đặt Grafana hoặc sử dụng Hosted Grafana trên bất kì nền tảng nào.
- Extend: Khám phá hàng trăm dashboard và plugin trong thư viện chính thức. Nhờ đam mê và động lực của cộng đồng, một dashboard hoặc plugin mới được thêm vào mỗi tuần.
- Collaborate - Cộng tác: mang mọi người lại với nhau, chia sẻ dữ liệu và các dashboard với các nhóm. Grafana trao quyền cho người dùng và giúp nuôi dưỡng một nền văn hóa hướng dữ liệu.
- Dynamic Dashboards: Tạo và sử dụng lại các dashboards với các biến template xuất hiện ở phần đầu của dashboard
- Annotations - Chú thích : Biểu đồ chú thích có sự kiện phong phú từ các nguồn dữ liệu khác nhau. Di chuột qua các sự kiện cho bạn thấy siêu dữ liệu sự kiện đầy đủ và các thẻ tag.

Tóm lại: Grafana là công cụ được sử dụng với nhiệm vụ chính là trực quan hóa và phân tích dữ liệu thời gian thực – tức là nó không phải đi thu thập metric từ hệ thống cần giám sát mà chỉ công cụ để hiển thị và phân tích dữ liệu. Với giao diện đẹp mắt và nhiều tính năng tuyệt vời, Grafana được cộng đồng tin tưởng và yêu thích sử dụng.

dụng. Có hơn 150000 hệ thống grafana đang hoạt động trên toàn thế giới với nhiều usecase sử dụng phong phú.

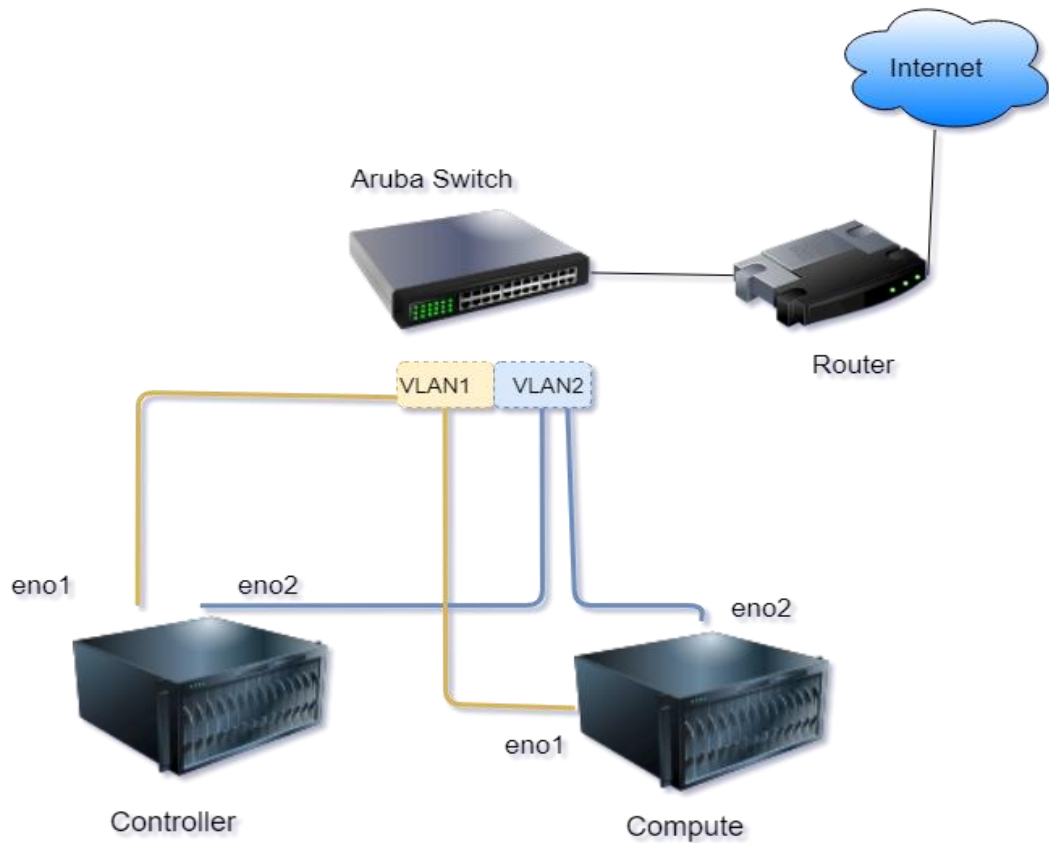
Với những lợi ích như trên, em chọn Grafana làm chức năng giám sát các thông số trong mạng. Grafana sẽ biểu thị thông tin về metric được thu thập bởi Collectd được cài trên các chức năng mạng Iptables và Suricata. Để thu thập các số liệu thống kê (metric) của hệ thống, các máy ảo trong mạng đều cài đặt công cụ collectd, cấu hình gửi metric tới máy Grafana. Máy Grafana tích hợp cài Graphite (là công cụ kết hợp với Collectd để thu thập metric và giám sát metric, Grafana hiển thị các thông số mà Graphite ghi lại).

3.4. Kịch bản thử nghiệm

Sau khi đã xác định được các công nghệ để xây dựng chuỗi chức năng mạng đề ra. Tiếp theo, em thực hành triển khai trên hệ thống thật.

3.4.1. Đồ hình vật lý

Hệ thống triển khai trên 2 máy vật lý thật có đồ hình như sau:



Hình 3.8: Đồ hình vật lý triển khai của testbed

Hình 3.8 mô tả đồ hình vật lý và kết nối mạng của các nút server triển khai xây dựng nền tảng OpenStack.

Các thông số vật lý: Thông số về tài nguyên phần cứng và thông tin cấu hình IP được mô tả trong Bảng 3.1:

Bảng 3.1: Thông số phần cứng và cấu hình hai nút server

Nút	Thông số tài nguyên		Thông số cấu hình IP (Cấu hình tĩnh)	
	Core CPU	RAM (GB)	eno1	eno2
Controller	24	64	192.168.101.201	10.10.10.201
Compute	32	64	192.168.101.201	10.10.10.202

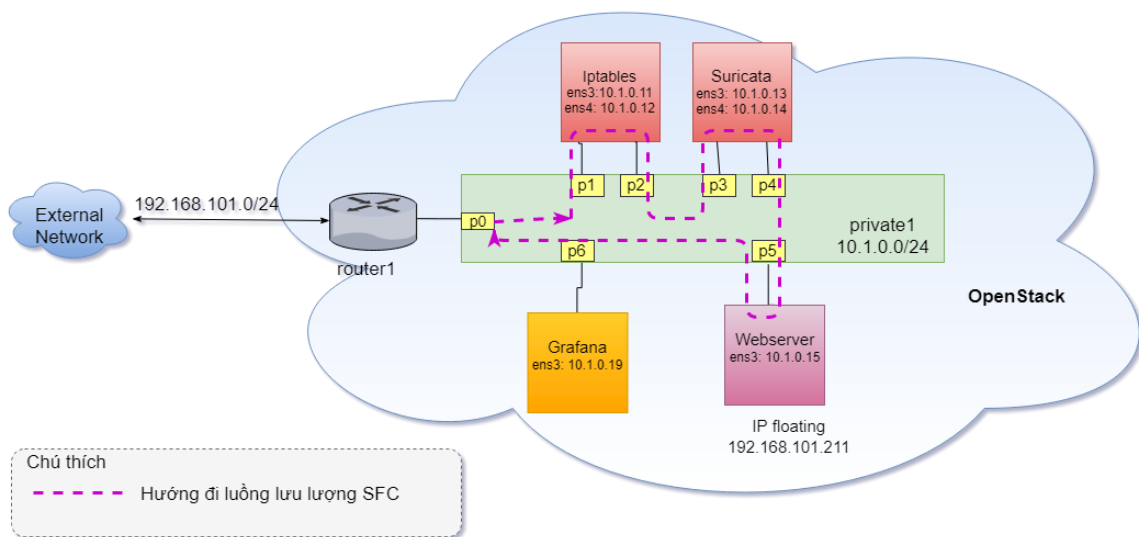
Hạ tầng mạng được kết nối sử dụng một switch được chia thành 2 VLAN.

- **Dải Vlan 1:** là Vlan mặc định trong switch, kết nối với router để có thể ra ngoài Internet. IP của dải Vlan 1 được cấp phát từ router – địa chỉ 192.168.101.0/24.
- **Dải Vlan 2:** được cấu hình để làm mạng Management phục vụ các giao tiếp dịch vụ kết nối giữa Controller và Compute. Cấu hình IP của dải Vlan2: 10.10.10.0/24

Cấu hình 2 máy chủ Controller và Compute nối với các dải mạng tương ứng. Thực hiện cài đặt OpenStack theo hướng dẫn từ trang chủ của OpenStack^[11].

3.4.2. Kịch bản kiểm thử

Sau khi cài đặt, triển khai các máy ảo VM với các thông số sau:



Hình 3.9: Đồ hình triển khai các máy ảo trong OpenStack

Các máy ảo được cài hệ điều hành Ubuntu server 16.04. Các máy ảo được kết nối chung với nhau thông qua mạng ảo private1 (cấp phát dải IP: 10.1.0.0/24). Thông số cấu hình các máy ảo được mô tả trong Bảng 3.2:

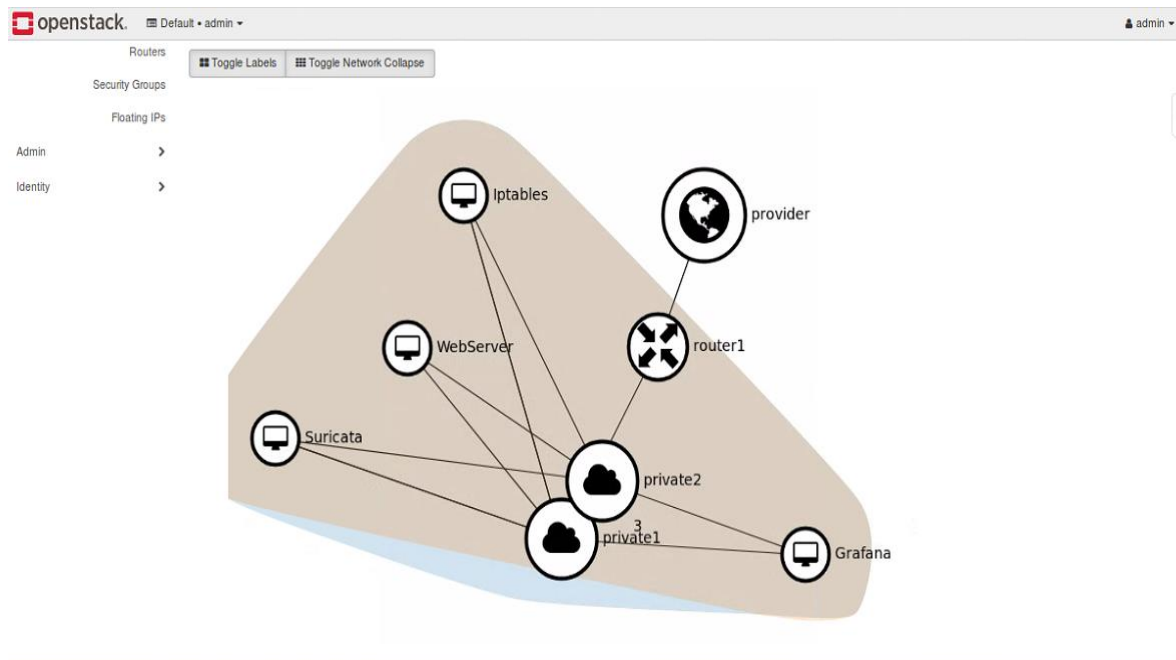
Bảng 3.2: Thông số máy ảo triển khai trong OpenStack

Máy ảo	Core CPU	RAM (GB)	IP ens3	IP ens4	Floating IP
Iptables	2	2	10.1.0.11	10.1.0.12	
Suricata	2	2	10.1.0.13	10.1.0.14	
Grafana	1	1	10.1.0.19		
Webserver	3	3	10.1.0.15		192.168.101.211

Máy ảo Webserver được gán thêm Floating IP (IP lấy từ mạng vật lý có thể ra ngoài Internet) để làm IP public của ứng dụng Web.

3.4.3. Kết quả dựng mô hình trên giao diện của OpenStack

Sau khi dựng xong mô hình các máy ảo, cài đặt và cấu hình các chức năng mạng Iptables, Suricata, Grafana ta được kết quả đồ hình mạng trên giao diện dashboard của OpenStack như Hình 3.10 sau:



Hình 3.10: Đồ hình chuỗi các chức năng mạng trên giao diện OpenStack

Các máy ảo trong cùng một mạng riêng nội bộ là mạng private1, các kết nối ra ngoài mạng được đưa ra thông qua router ảo router1 đến mạng provider của nhà cung cấp để ra ngoài Internet.

3.4.4. Kết quả dựng networking-SFC trên OpenStack

Theo mô hình đã thiết kế, em đã tạo ra trên port chain và port group trên các cổng p1 (ingress port), p2 (egress port) (nối vào VM Iptables) và p3, p4 (nối vào Suricata). Kết quả như sau:

```
root@controller:~#
root@controller:~# neutron port-pair-list
neutron CLI is deprecated and will be removed in the future. Use openstack CLI instead.
+-----+-----+-----+-----+
| id | name | tenant_id | ingress | egress |
+-----+-----+-----+-----+
| 93005bb9-3b4c-4381-bf52-76f112686828 | PP2 | 2c85a9d757b54b118cc4f905f8af42d9 | 6022be08-3040-4167-af6d-9c077c1818f6 | c6275895-759b-486c-8119-7082cac61184 |
| a93536bf-12f5-4131-b79b-1cdd8b911cde | PP1 | 2c85a9d757b54b118cc4f905f8af42d9 | 4d7a2c9b-e68c-4dea-b701-b5c66e0df1d4 | c7c7effe-9717-470f-9ee8-9f60fdbdc7de |
+-----+-----+-----+-----+
```

Hình 3.11: Kết quả tạo port pair trên các cổng nối với VM

Trong đó, PP1 là port pair nối giữa p1 và p2 của VM Iptables, PP2 và port pair nối giữa cổng p3 và p4 của VM Suricata.

Tạo port pair group chứa các port pair (đại diện cho một hoặc nhiều service function). Nhiều port pair cho phép cân bằng tải hoặc phân bổ lưu lượng xử lý trên

một tập các service function instance cùng chức năng. Ở đây là đại diện cho chức năng Firewall (Iptables) và chức năng NIDS (Suricata)

```
root@controller:~# neutron port-pair-group-list
neutron CLI is deprecated and will be removed in the future. Use openstack CLI instead.
+-----+-----+-----+-----+
| id | name | tenant_id | port_pairs |
+-----+-----+-----+-----+
| 81881e85-1afb-4f3a-a34a-7d779f58835b | PG2 | 2c85a9d757b54b118cc4f905f8af42d9 | [u'93005bb9-3b4c-4381-bf52-76f112686828'] |
| 8f1f3add-b8d9-4f49-bd37-e6c6bae11e5c | PG1 | 2c85a9d757b54b118cc4f905f8af42d9 | [u'a93536bf-12f5-4131-b79b-1cdd0b911cde'] |
+-----+-----+-----+-----+
root@controller:~#
```

Hình 3.12: Kết quả tạo port-pair-group

Tạo Flow Classifier: Tập hợp các thuộc tính định nghĩa nên flow, gồm cả các thuộc tính về nguồn và đích của flow. Phân biệt giữa các luồng lưu lượng khác nhau sẽ đi qua các chức năng mạng khác nhau.

```
root@controller:~# neutron flow-classifier-list
neutron CLI is deprecated and will be removed in the future. Use openstack CLI instead.
+-----+-----+-----+-----+
| id | name | tenant_id | summary |
+-----+-----+-----+-----+
| 19d6fbc1-fa93-42d2-bc36-445580a8ce41 | FC1 | 2c85a9d757b54b118cc4f905f8af42d9 | protocol: ICMP,
| | | | source[port]: 0.0.0.0[any:any],
| | | | destination[port]: 10.1.0.16/32[any:any],
| | | | neutron_source_port: 0efd07e8-8607-45c9-b218-53470ffbef67,
| | | | neutron_destination_port: None,
| | | | l7_parameters: {}
| | | |
| 49d9ff89-6d01-4b6b-b2a1-c403d4e9db06 | FC7 | 2c85a9d757b54b118cc4f905f8af42d9 | protocol: ICMP,
| | | | source[port]: 0.0.0.0[any:any],
| | | | destination[port]: 10.1.0.15/32[any:any],
| | | | neutron_source_port: 0efd07e8-8607-45c9-b218-53470ffbef67,
| | | | neutron_destination_port: None,
| | | | l7_parameters: {}
| | | |
| 5dcab42a-5878-41db-862c-2faae35d3c3d | FC2 | 2c85a9d757b54b118cc4f905f8af42d9 | protocol: TCP,
| | | | source[port]: 0.0.0.0[any:any],
| | | | destination[port]: 10.1.0.16/32[80:80],
| | | | neutron_source_port: 0efd07e8-8607-45c9-b218-53470ffbef67,
| | | | neutron_destination_port: None,
| | | | l7_parameters: {}
| | | |
| 7cc7abb8-db56-493a-86f9-3b8d6051eafa | FC6 | 2c85a9d757b54b118cc4f905f8af42d9 | protocol: UDP,
| | | | source[port]: 0.0.0.0[any:any],
| | | | destination[port]: 10.1.0.17/32[80:80],
| | | | neutron_source_port: 0efd07e8-8607-45c9-b218-53470ffbef67,
| | | | neutron_destination_port: None,
| | | | l7_parameters: {}
| | | |
| 7ec9c41c-5bf5-4255-b6b4-53b443d7c53c | FC9 | 2c85a9d757b54b118cc4f905f8af42d9 | protocol: UDP,
| | | | source[port]: 0.0.0.0[any:any],
| | | |
+-----+-----+-----+-----+
```

Hình 3.13: Kết quả tạo Flow Classifier

Ở mô hình này, em tạo chuỗi phân loại các luồng ICMP, UDP và TCP.

Cuối cùng là tạo port-chain gồm các chuỗi Flow Classifier vừa tạo kết hợp cùng 2 port group PG1 và PG2. Trên switch ảo duy trì mạng private giữa các máy ảo, thêm luồng điều khiển để lưu lượng đi theo đúng trình tự mong muốn.

3.5. Kết luận

Chương 3 đã mô tả quá trình thực hiện tìm hiểu và xác định các yếu tố cần thiết, dựng mô hình testbed SFC có hai chức năng mạng là Iptables và Suricata có tác dụng bảo vệ máy chủ Web server bên trong. Testbed dựng trên nền tảng OpenStack với

các chức năng mạng là các máy ảo bên trong OpenStack, luồng lưu lượng được điều khiển bởi project cung cấp mạng của OpenStack là Neutron.

CHƯƠNG 4: KẾT QUẢ ĐO ĐẠC VÀ ĐÁNH GIÁ

Sau khi hoàn thiện mô hình testbed, phần tiếp theo sẽ thực hiện một số kiểm chứng và đo đạc hiệu năng hoạt động của hệ thống.

Kịch bản kiểm chứng bao gồm:

- 1) Kiểm chứng luồng lưu lượng đi theo đúng mô hình SFC đã dựng.
- 2) Thực hiện đo đạc các thông số hoạt động của chuỗi SFC khi đẩy tải vào chuỗi.
 - Thực hiện đo đạc tài nguyên sử dụng trên Suricata khi hoạt động ở các tải và đánh giá để biết được mức tài nguyên cần thiết cấp phát cho chức năng mạng này.
 - Thực hiện đo đạc thông và đánh giá các thông số chất lượng dịch vụ của chuỗi chức năng: độ trễ và tỉ lệ mất gói khi lưu lượng đi qua chuỗi chức năng mạng.

4.1. Kiểm chứng luồng lưu lượng đi theo đúng mô hình SFC đã dựng

Để kiểm chứng lưu lượng từ bên ngoài truy cập vào ứng dụng web, lưu lượng đã đi qua các chức năng mạng Iptables, Suricata rồi mới tới Web server bên trong. Từ bên ngoài, thực hiện ping vào địa chỉ IP public của máy chủ Webserver, thấy lưu lượng đi đã đi qua chuỗi chức năng mạng như sau:

```

root@bonesi:~#
root@bonesi:~# ping 192.168.101.211
PING 192.168.101.211 (192.168.101.211) 56(84) bytes of data.
64 bytes from 192.168.101.211: icmp_seq=1 ttl=63 time=2.35 ms
64 bytes from 192.168.101.211: icmp_seq=2 ttl=63 time=1.89 ms
64 bytes from 192.168.101.211: icmp_seq=3 ttl=63 time=1.81 ms
64 bytes from 192.168.101.211: icmp_seq=4 ttl=63 time=1.53 ms
64 bytes from 192.168.101.211: icmp_seq=5 ttl=63 time=1.65 ms
64 bytes from 192.168.101.211: icmp_seq=6 ttl=63 time=1.36 ms
64 bytes from 192.168.101.211: icmp_seq=7 ttl=63 time=1.53 ms
64 bytes from 192.168.101.211: icmp_seq=8 ttl=63 time=1.48 ms
64 bytes from 192.168.101.211: icmp_seq=9 ttl=63 time=1.48 ms
64 bytes from 192.168.101.211: icmp_seq=10 ttl=63 time=1.51 ms
^X^C
--- 192.168.101.211 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9015ms
rtt min/avg/max/mdev = 1.366/1.664/2.359/0.278 ms
root@bonesi:~#

```

Hình 4.1: Thực hiện ping từ một máy bên ngoài vào máy chủ Web

```

root@iptables:~# tcpdump -i ens3 -p icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on ens3, link-type EN10MB (Ethernet), capture size 262144 bytes
18:51:53.133660 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 30, length 64
18:51:54.135379 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 31, length 64
18:51:54.552018 IP 192.168.101.211 > 10.1.0.15: ICMP host 8.8.8.8 unreachable, l
length 83
18:51:54.552055 IP 192.168.101.211 > 10.1.0.15: ICMP host 8.8.8.8 unreachable, l
length 83
18:51:55.137073 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 32, length 64
18:51:56.138783 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 33, length 64
18:51:57.140389 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 34, length 64
18:52:13.166263 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 50, length 64
18:52:14.167647 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 51, length 64
18:52:33.197064 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq
 70, length 64
18:52:34.198684 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq

```

Hình 4.2: Luồng lưu lượng đi qua Iptables

Như trên Hình 4.2, sử dụng công cụ tcpdump để bắt các gói tin vào interface ens3 (tức port p1) thấy có lưu lượng của máy client từ bên ngoài gửi đến Webserver (vào mạng private bên trong đã được NAT ra thành địa chỉ private là 10.1.0.15).

Kiểm tra trên Suricata, ta cũng thấy có lưu lượng chuyển qua như Hình 4.3:

```
root@suricata:~#  
root@suricata:~# tcpdump -i ens3 -p icmp  
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on ens3, link-type EN10MB (Ethernet), capture size 262144 bytes  
18:53:46.274655 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
143, length 64  
18:53:47.276320 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
144, length 64  
18:53:48.278144 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
145, length 64  
18:53:49.276638 IP 192.168.101.211 > 10.1.0.15: ICMP host 8.8.8.8 unreachable, l  
ength 83  
18:53:49.276706 IP 192.168.101.211 > 10.1.0.15: ICMP host 8.8.8.8 unreachable, l  
ength 83  
18:53:49.279820 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
146, length 64  
18:53:50.281438 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
147, length 64  
18:53:51.283209 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
148, length 64  
18:53:52.284857 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
149, length 64  
18:53:53.286567 IP 192.168.101.221 > 10.1.0.15: ICMP echo request, id 25150, seq  
150, length 64
```

Hình 4.3: Luồng lưu lượng gửi qua Suricata

Cuối cùng, lưu lượng được gửi tới đích là Webserver để xử lý:

```
root@web:~#  
root@web:~#  
root@web:~# tcpdump -i ens3 -p icmp  
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on ens3, link-type EN10MB (Ethernet), capture size 262144 bytes  
18:55:09.445025 IP 192.168.101.221 > web: ICMP echo request, id 25150, seq 227, l  
ength 64  
18:55:09.445066 IP web > 192.168.101.221: ICMP echo reply, id 25150, seq 227, l  
ength 64  
18:55:10.446789 IP 192.168.101.221 > web: ICMP echo request, id 25150, seq 228,  
length 64  
18:55:10.446830 IP web > 192.168.101.221: ICMP echo reply, id 25150, seq 228, l  
ength 64  
18:55:11.448348 IP 192.168.101.221 > web: ICMP echo request, id 25150, seq 229,  
length 64  
18:55:11.448399 IP web > 192.168.101.221: ICMP echo reply, id 25150, seq 229, l  
ength 64  
^C
```

Hình 4.4: Lưu lượng được gửi tới máy Webserver

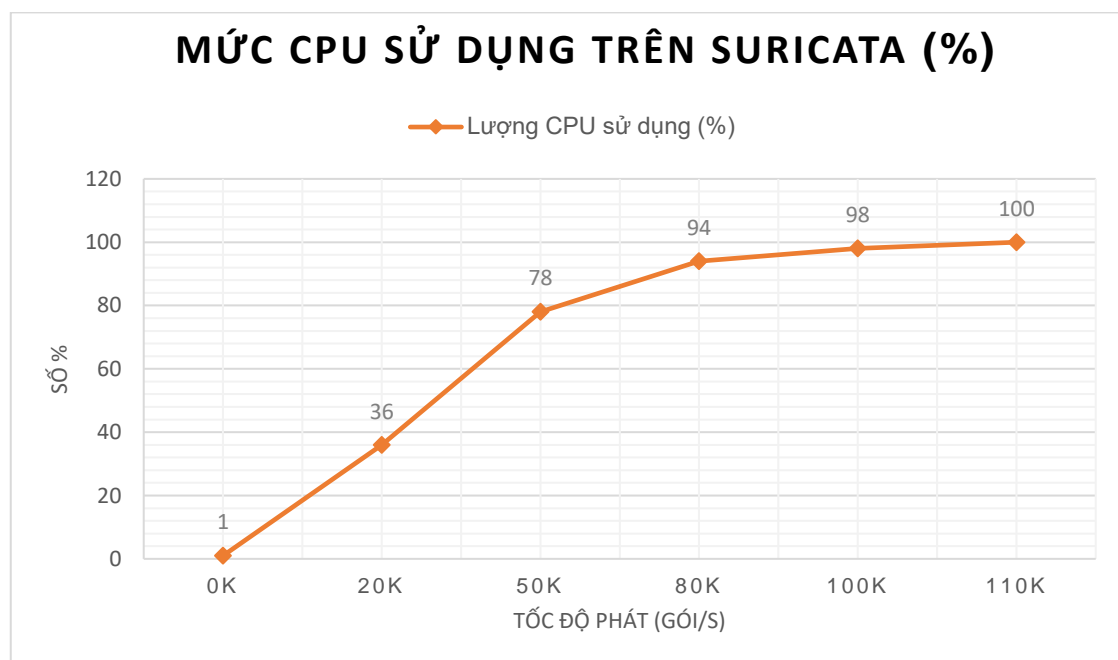
Như vậy, luồng lưu lượng từ ngoài vào đã được đi qua chức năng mạng Iptables và Suricata. Hoạt động đúng theo mô hình SFC đã dựng.

4.2. Kết quả đo lường tài nguyên sử dụng trên Suricata

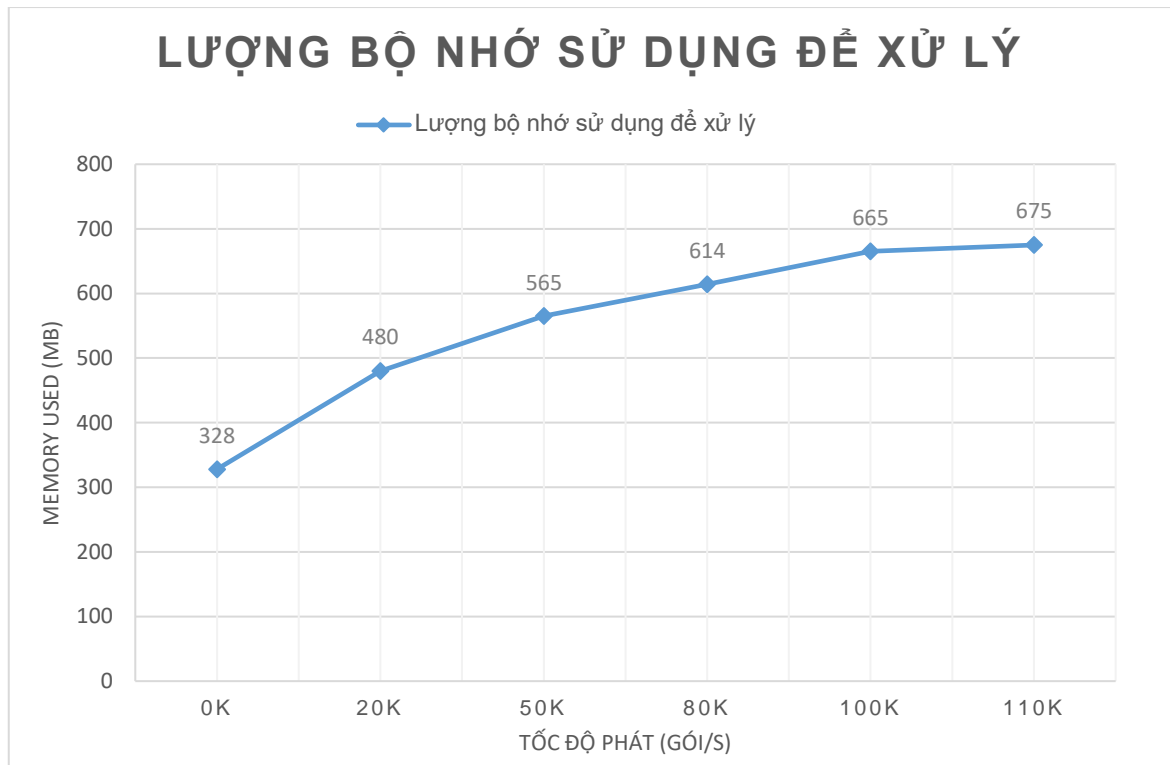
Sau khi xây dựng mô hình, thực hiện đo đạc các thông số sử dụng về CPU và memory trên chức năng mạng Suricata. Suricata kích hoạt sử dụng 15 000 rule phát hiện các hành vi khả nghi tấn công vào mạng

Trên Suricata, Iptables và máy chủ Webserver, cài đặt thêm công cụ collectd để thu thập tham số hệ thống, gửi về máy Grafana (có cài đặt Graphite). Số liệu được hiển thị trên Grafana cho phép giám sát các tài nguyên của các máy ảo.

Để kiểm chứng khả năng hoạt động của các chức năng mạng trong hệ thống, lần lượt phát gói TCP vào chuỗi SFC với tốc độ tăng dần để tăng tải (sử dụng công cụ phát gói: Bonesi^[10]). Kết quả thu được như sau:



Biểu đồ 4.1: Kết quả sử dụng CPU trên Suricata khi tải tăng dần

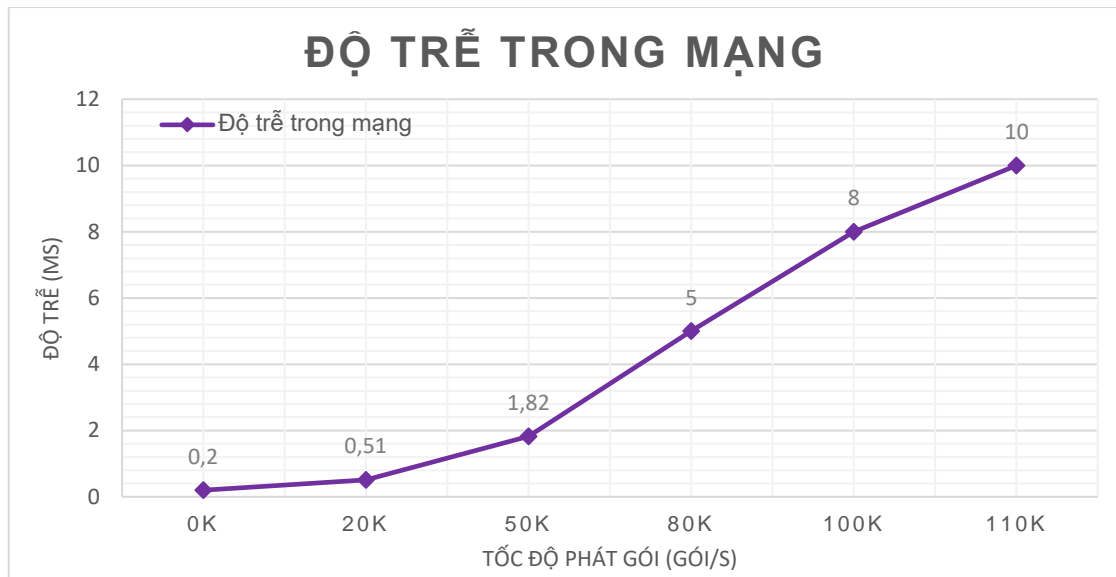


Biểu đồ 4.2: Lượng bộ nhớ sử dụng để xử lý các gói tin trên Suricata khi tải tăng dần

Qua hai biểu đồ trên ta thấy, lượng tài nguyên CPU, memory sử dụng của Suricata tăng dần khi tải đầu vào tăng dần. Như vậy, với thông số cấu hình 2G RAM, 2 core CPU, Suricata chịu được tải tối đa là 110 000 gói TCP/s. Lượng RAM sử dụng là không đáng kể.

4.3. Kết quả đo độ trễ và tỷ lệ mất gói của lưu lượng khi đi qua chuỗi các chức năng mạng

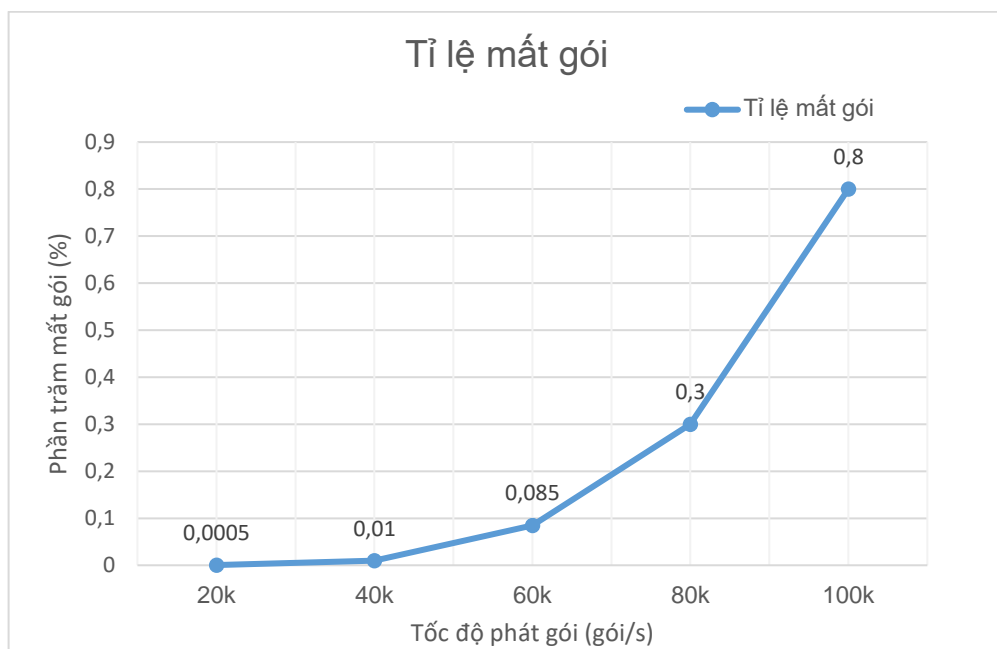
Phát lần lượt số lượng các gói tin TCP tăng dần vào chuỗi SFC. Thực hiện kích hoạt plugin ping (của collectd) trên máy client từ bên ngoài mạng. Cấu hình client cứ mỗi giây ping tới web server một lần để đo thông số thời gian round trip time của gói tin trả về. Độ trễ gói tin trả về tăng dần khi tải hệ thống tăng dần như sau:



Biểu đồ 4.3: Kết quả đo độ trễ trong mạng

Đo tỉ lệ mất gói: Sử dụng công cụ **hping3**, thực hiện gửi liên tục 1000 gói ICMP request trong 1s gửi cùng thời điểm phát gói tin vào chuỗi chức năng. Độ mất gói sẽ tính bằng tỉ lệ giữa hiệu giữa gói tin ICMP request gửi đi và gói tin ICMP reply gửi về với tổng gói tin ICMP request gửi đi.

Kết quả độ mất gói hiển thị như Biểu đồ 4.4 sau:



Biểu đồ 4.4: Tỷ lệ mất gói trong mạng.

Từ Biểu đồ 4.4 ta thấy, tỉ lệ mất gói tin tăng dần khi tải tăng. Khi tải ở mức thấp (chỉ 20 000 gói/s, tỉ lệ mất gói rất nhỏ, chỉ cỡ 0,0005%)

4.4. Đánh giá kết quả

Từ các kết quả kiểm chứng trên ta thấy: mô hình chuỗi các chức năng mạng đã dựng thành công. Luồng lưu lượng trước khi tới máy chủ Web server đã được đưa qua các chức năng mạng Iptables và Suricata để lọc lưu lượng.

Mức tiêu thụ tài nguyên của các chức năng mạng ảo tăng dần khi tải lưu lượng qua chuỗi SFC tăng dần. Với lượng tài nguyên 2 core CPU, Suricata chịu được tải TCP tối đa là 100k gói/s. Lượng RAM sử dụng của Suricata trong khoảng không tới 700MB.

Độ trễ và tỉ lệ mất gói của chuỗi tăng dần theo mức tải tăng. Nguyên nhân là do switch ảo trong OpenStack chưa đủ mạnh để hỗ trợ chuyển gói nhanh hơn. Hạn chế này là do tài nguyên vật lý khi thực hiện đồ án còn hạn chế.

Nhìn chung, chuỗi SFC dựng lên có thể hoạt động ở mức chấp nhận được khi bảo vệ ứng dụng máy chủ Webserver bên trong của các nhà cung cấp. Giúp tiết kiệm chi phí phần cứng, đồng thời linh động hơn trong quá trình tạo thêm và sửa đổi thứ tự dịch vụ mạng trong chuỗi.

4.5. Hạn chế

Đề tài đồ án thực hiện trong thời gian có hạn nên vẫn còn hạn chế:

- Mô hình nhỏ, chỉ có 2 chức năng mạng là Iptables và Suricata.
- Việc điều khiển luồng lưu lượng trong switch để thực hiện SFC còn thực hiện thủ công bằng luồng tĩnh. Không linh hoạt khi thực hiện mở rộng mô hình.
- Chưa làm chủ được switch ảo để tăng hiệu năng chuyển mạch của switch, khiến tỉ lệ mất gói và độ trễ chưa được ở mức tối thiểu.

KẾT LUẬN

Như vậy, qua quá trình thực hiện đồ án, em đã có được hiểu biết tương đối về công nghệ ảo hóa chức năng mạng và chuỗi các dịch vụ mạng. Triển khai thành công mô hình chuỗi chức năng mạng bảo vệ máy chủ Web server bên trong trung tâm dữ liệu.

Tuy đồ án vẫn còn nhiều hạn chế, nhưng trong tương lai, em sẽ tiếp tục phát triển theo các hướng sau để tối ưu hơn nữa:

- Kết hợp công nghệ Mạng định nghĩa bằng phần mềm (Software Defined Networking) để điều khiển luồng lưu lượng trong chuỗi SFC linh hoạt hơn.
- Kích hoạt tính năng chuyển tiếp nhanh của Switch ảo để tăng hiệu năng hoạt động, giảm độ trễ và tỉ lệ mất gói.
- Triển khai nhiều luồng và nhiều chuỗi dịch vụ với các tính năng khác nhau.
- Sau đó tính toán lượng tài nguyên tối thiểu cần thiết để một chức năng mạng có thể hoạt động bình thường ở mức tải đưa ra. Tiến tới sau này đẩy chức năng mạng vào container để cung cấp thành CPE chuyên dụng cho khách hàng.

TÀI LIỆU THAM KHẢO

- [1] Peter Mell, Tim Grance, “*The NIST Definition of Cloud Computing*”, NIST Special Publication 800-145, September – 2011.
- [2] ETSI Industry Specification Group, *Network Functions Virtualisation (NFV): Use cases*. [Online]. Available:
http://www.etsi.org/deliver/etsi_gs/NFV/001_099/001/01.01.01_60/gs_NFV001v010101p.pdf.
- [3] Margaret Chiosi, “*Network Functions Virtualisation – Introductory White Paper*”, presented at the “SDN and OpenFlow World Congress”, Darmstadt-Germany, October 22-24, 2012.
- [4] Service Function Chaining (SFC) General Use Cases. [Online]. Available:
<https://tools.ietf.org/html/draft-liu-sfc-use-cases-08>
- [5] Deval Bhamare, Raj Jain, Mohammed Samaka, Aiman Erbad, "A Survey on Service Function Chaining", *Journal of Network and Computer Applications*, [Online], Volume 75, Pages 138-155. Available:
<https://www.sciencedirect.com/science/article/pii/S1084804516301989#s0010>
- [6] <https://docs.openstack.org/pike/> truy cập cuối cùng ngày 06/06/2018
- [7] <https://docs.openstack.org/ocata/networking-guide/config-sfc.html> truy cập lần cuối ngày 06/06/2018.
- [8] <http://www.faqs.org/docs/iptables/> truy cập lần cuối ngày 06/06/2018.
- [9] OISF, Suricata User Guide, [Online], Available :
<https://media.readthedocs.org/pdf/suricata/latest/suricata.pdf>
- [10] Bonesi. [Online]. Available: <https://github.com/Markus-Go/bonesi>
- [11] <https://docs.openstack.org/install-guide/> truy cập lần cuối ngày 01/06/2018.