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

THỰC TẬP TỐT NGHIỆP

Đề tài:

XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG TRÊN NỀN TẢNG OPENSTACK

Sinh viên thực hiện: ĐỖ XUÂN SƠN
LỚP ĐTTT 06 – K58
Giảng viên hướng dẫn: PGS.TS. NGUYỄN HỮU THANH

Hà Nội, 3-2019

LỜI NÓI ĐẦU

Hiện nay, 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. Điện toán đám mây ra đời cùng với việc tích hợp với các công nghệ nổi bật như NFV từng bước trở thành giải pháp chủ đạo cho các nhà cung cấp giúp thay đổi phương thức mang dịch vụ đến người dùng cuối với nhiều mô hình dịch vụ khác nhau.

Trong quá trình làm đồ án tốt nghiệp, em thực hiện tìm hiểu về kiến trúc của điện toán đám mây, nền tảng mã nguồn mở OpenStack cùng với các dự án liên quan để xây dựng nên hệ thống điện toán đám mây ở mức cơ bản đứng về góc nhìn của nhà cung cấp dịch vụ, nhằm tạo ra môi trường linh hoạt tận dụng lợi thế của NFV.

Em xin chân thành cảm ơn PGS.TS. Nguyễn Hữu Thanh đã tận tình giúp đỡ, tạo điều kiện để em thực hiện thực tập tốt nghiệp. Ngoài ra em cũng xin chân thành cảm ơn các thành viên Future Internet Lab đặc biệt là các bạn trong nhóm Network Function Virtualization đã giúp đỡ, chia sẻ trong suốt thời gian qua.

MỤC LỤC

DANH MỤC HÌNH VẼ

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	Firber 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. GIỚI THIỆU

1.1. Đặt vấn đề

Sự gia tăng yêu cầu về kết nối trong thế giới hiện đại đặt ra những yêu cầu mới trong việc cung cấp dịch vụ, đặc biệt là với sự xuất hiện của khái niệm IoT (Internet of Things – Internet của vạn vật). Các nhà cung cấp dịch vụ, những người chịu trách nhiệm cung cấp các kết nối này đang phải đối mặt với sự gia tăng về lưu lượng mạng và số lượng thuê bao trong hệ thống mạng của họ. Những thách thức mà chọ phải đối mặt có thể kể tới như:

- Sự bùng nổ về số lượng yêu cầu
- Không đủ khả năng cung cấp dịch vụ mới với sự gia tăng nhanh chóng và thay đổi liên tục về số lượng người dùng.
- Chi phí đầu tư và chi phí vận hành ngày càng tăng làm giảm doanh thu.

Các nhà cung cấp dịch vụ nói chung chưa thể tối ưu hóa tài nguyên sử dụng bởi họ trước hết phải đảm bảo phục vụ được tốc độ lưu lượng mạng ở mức đỉnh. Yêu cầu mới đặt ra là họ phải cải thiện hệ thống để quản lý việc cung cấp mạng lưới gồm rất nhiều ứng dụng đồng thời tối đa hóa được lượng tài nguyên sử dụng tránh lãng phí bằng việc triển khai cơ sở hạ tầng ảo hóa.

Công nghệ "Ảo hóa các chức năng mạng" (NFV) áp dụng thành tựu của ảo hóa cho các thiết bị mạng truyền thống để giảm thiểu chi phí đầu tư và vận hành đồng thời cho phép cải thiện thời gian đưa dịch vụ vào thị trường. NFV đang thay đổi cách mà nhà cung cấp dịch vụ thiết kế và triển khai hệ thống mạng của họ.

Trong đồ án này, em thực hiện tiến hành tìm hiểu các vấn đề liên quan tới NFV và thiết kế hệ thống SFC (Service Function Chainning) là một user case cụ thể của NFV.

1.2. Định hướng giải pháp

NFV ra đời thu hút được sự quan tâm đông đảo từ phía các nhà cung cấp dịch vụ công nghệ thông tin và viễn thông. Cùng với đó là sự ra đời của nhiều dự án công nghệ thông tin mã nguồn mở để hỗ trợ và đáp ứng được từng phần tử trong kiến trúc NFV. Đồ án này lựa chọn những dự án nổi bật và trưởng thành nhất trong thế giới

của NFV, tích hợp với nhau thỏa mãn kiến trúc NFV chuẩn mực do ETSI đề ra.

Những công việc chính trong đồ án:

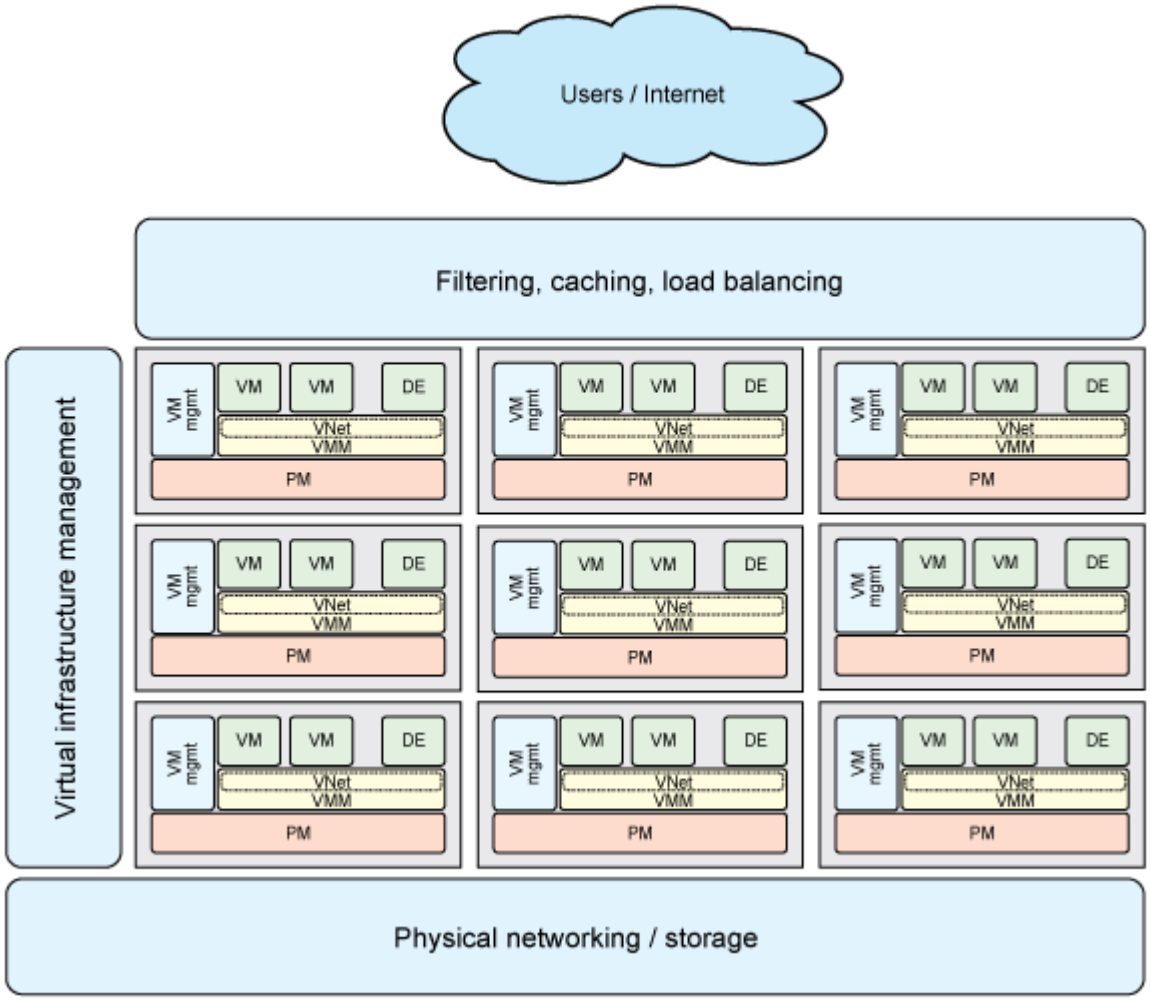
- Tìm hiểu kiến trúc của NFV và các khía cạnh liên quan
- Tìm hiểu các công nghệ phù hợp với kiến trúc của NFV và tích hợp các công nghệ đó lại để có hệ thống kiểm thử.
- Tiến hành xây dựng chuỗi chức năng mạng (SFC).

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1. Tổng quan về công nghệ điện toán đám mây

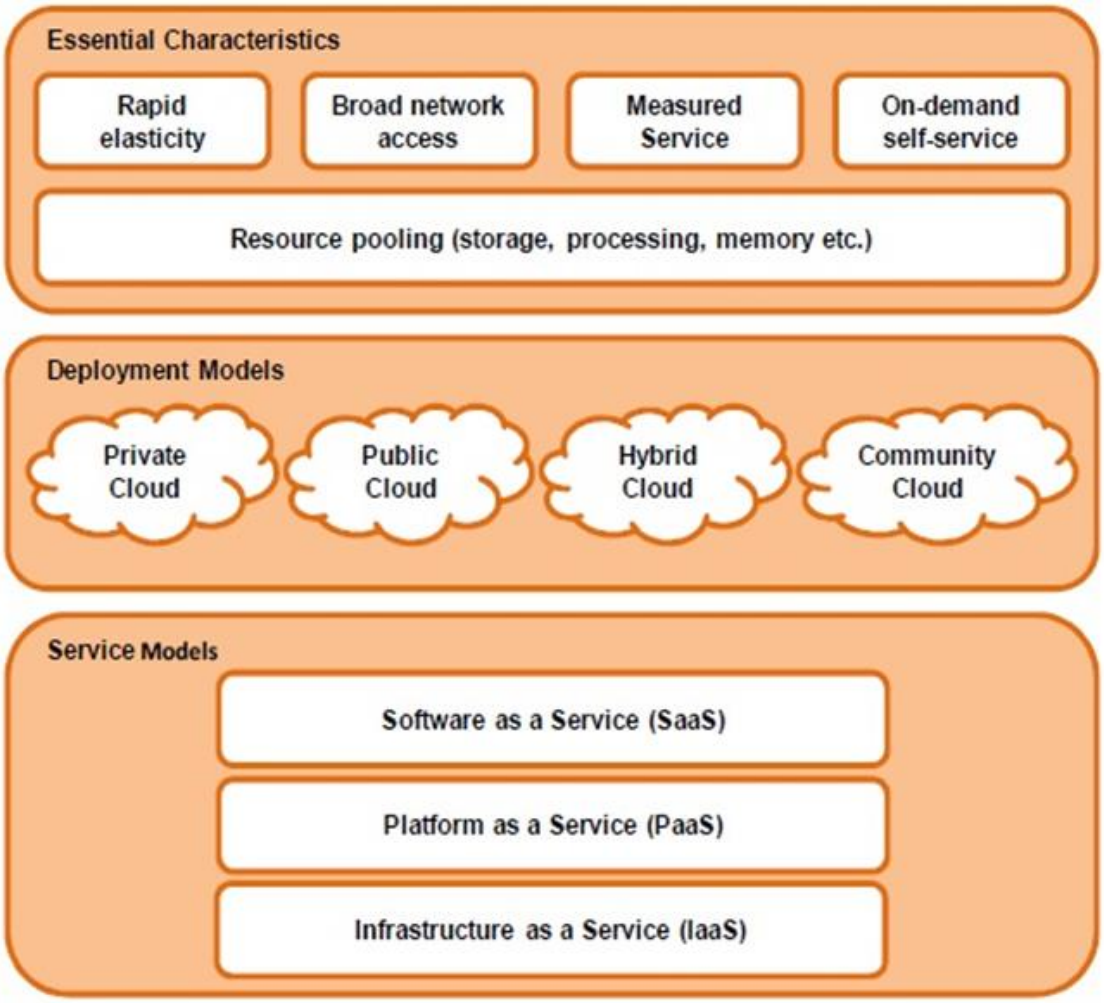
Cloud Computing hay còn gọi là điện toán đám mây là một thuật ngữ xuất hiện vào năm 2007. Thuật ngữ “cloud” là cách nói trừu tượng để chỉ các máy chủ, máy chủ có kết nối Internet, được sử dụng vào mục đích lưu trữ, tính toán, triển khai dịch vụ. Mô hình điện toán (computing) sử dụng các công nghệ máy tính, phát triển dựa vào internet. Như vậy có thể hiểu Cloud Computing là mô hình cho phép truy cập qua mạng để lựa chọn và sử dụng tài nguyên có thể được tính toán (ví dụ: mạng, máy chủ, lưu trữ, ứng dụng và dịch vụ) theo nhu cầu một cách thuận tiện và nhanh chóng đồng thời cho phép kết thúc sử dụng dịch vụ, giải phóng tài nguyên dễ dàng, giảm thiểu các giao tiếp với nhà cung cấp.

Đi sâu hơn vào kiến trúc, mỗi nút trong hệ thống Cloud chính là một máy chủ sử dụng công nghệ ảo hóa. Khi kết hợp các máy chủ đó trong một mạng vật lý kết hợp với lưu trữ có chia sẻ, phối hợp quản lý trên toàn bộ cơ sở hạ tầng, cung cấp cân bằng tải ban đầu của các kết nối đến, ... ta có được một cơ sở hạ tầng ảo gọi là "cloud" như hình 2.1.



Hình 2.1 Kiến trúc Cloud Computing

Cloud Computing đặc trưng bởi khái niệm 5-4-3:



Hình 2.2 Đặc trưng của Cloud Computing

- **5 đặc tính:**
  - **On-demand self-service**: Người dùng có khả năng tự phục vụ, tự dự phòng được khả năng tính toán, như thời gian phục vụ và mạng lưu trữ, chủ động khởi tạo, tạm dừng dịch vụ mà không phải tương tác, phụ thuộc nhiều vào nhà cung cấp dịch vụ.
  - **Broad network access**: khả năng hỗ trợ nhiều chuẩn mạng, hỗ trợ truy cập dịch vụ từ nhiều nền tảng thiết bị, nhiều hạ tầng vật lý (như: điện thoại di động, máy tính bảng, laptop, máy trạm, etc.)
  - **Resource pooling**: Các tài nguyên tính toán của nhà cung cấp được gộp lại để cấp phát, chia sẻ tự động cho nhiều người dùng dựa theo nhu cầu.

- **Rapid elasticity**: Khả năng thu hồi và cấp phát tài nguyên nhanh chóng dựa theo nhu cầu người dùng.
- **Measured service**: Hệ thống cloud tự động hóa việc điều khiển và tối ưu tài nguyên được sử dụng bằng việc tận dụng khả năng đo lường, tính toán mức độ sử dụng dịch vụ, kiểm soát thời gian phục vụ, giám sát, điều khiển, báo cáo, etc. 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 cho nội bộ một tổ chức.
  - **Community cloud**: Nền tảng cloud cung cấp cho một nhóm các tổ chức có cùng chung mục đích, nhiệm vụ, chính sách, etc. 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**: là hệ thống cloud cumg cấp dịch vụ cho khách hàng sử dụng qua internet, có tính chất thương mại. Ví dụ: AWS của Amazon, Azure của Microsoft, Bluemix của IBM, etc.
  - **Hybrid cloud**: là nền tảng kết hợp giữa hai hay nhiều kiến trúc cloud (public, private hoặc community cloud)

- **3 mô hình dịch vụ:**
  - **Infrastructure-as-a-Service (IaaS):** là các dịch vụ Cloud dành cho cấp độ hạ tầng như: máy chủ, thiết bị lưu trữ, mạng, bảo mật… Với IaaS, các tài nguyên hạ tầng này luôn sẵn sàng tại nhà cung cấp. Người dùng không cần mua sắm thiết bị, triển khai hạ tầng cũng như nhân sự quản trị, nhà cung cấp sẽ đảm bảo chất lượng dịch vụ đã cam kết với khách hàng. Hầu hết các dịch vụ đều cung cấp Self-service Portal cho phép người dùng có thể trực tiếp theo dõi, quản lý, tối ưu hệ thống. IaaS trong một số trường hợp còn được gọi là Hardware-as-a-Service (HaaS).

- **Platform-as-a-Service (PaaS):** là mô hình dịch vụ dựa trên nền tảng IaaS với sự bổ sung các thành phần OS, Middleware để tạo môi trường tính toán giúp người dùng dễ dàng triển khai dịch vụ. Với PaaS, người dùng có thể dễ dàng triển khai môi trường lập trình, kiểm thử phần mềm mà không cần quan tâm đến hạ tầng phức tạp bên dưới. Hiện nay trên thế giới có rất nhiều nền tảng PaaS như Microsoft Azure, Google App Engine…
- **Software-as-a-Service (SaaS):** là các dịch vụ phần mềm được triển khai trên hệ thống của nhà cung cấp. Người dùng có thể truy cập và sử dụng mọi lúc mọi nơi thông qua trình duyệt web hay các ứng dụng di động. Với SaaS, người dùng không cần bận tâm việc cài đặt, lưu trữ phần mềm mà chỉ cần chọn phần mềm phù hợp với yêu cầu công việc.

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.

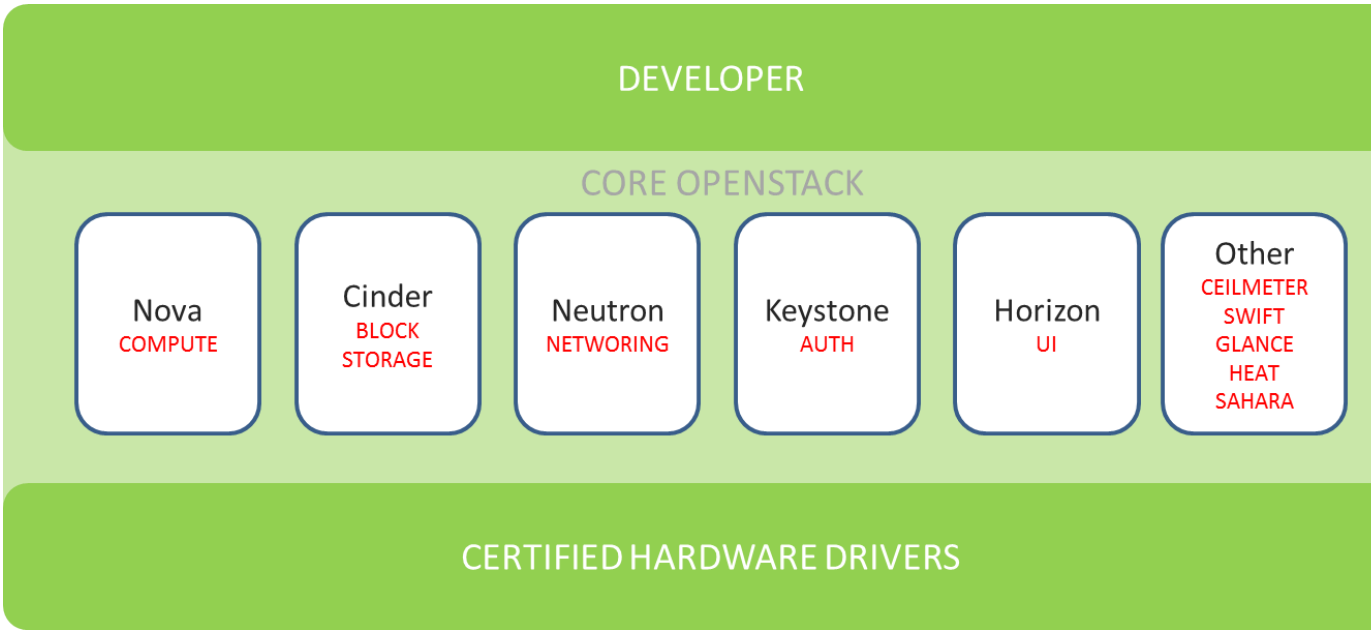
OpenStack hoạt động theo hướng mở: công khai lộ trình phát triển, công khai mã nguồn mở… OpenStack được phát triển và phát hành phiên bản mới trong vòng 6 tháng, hiện tại đã có 13 phiên bản OpenStack. Tên các phiên bản được đặt theo thứ tự chữ cái A, B, C, … phiên bản hiện tại là Rockey.

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): dịch vụ xác thực, ủy quyền người dùng và các dịch vụ khác của OpenStack.

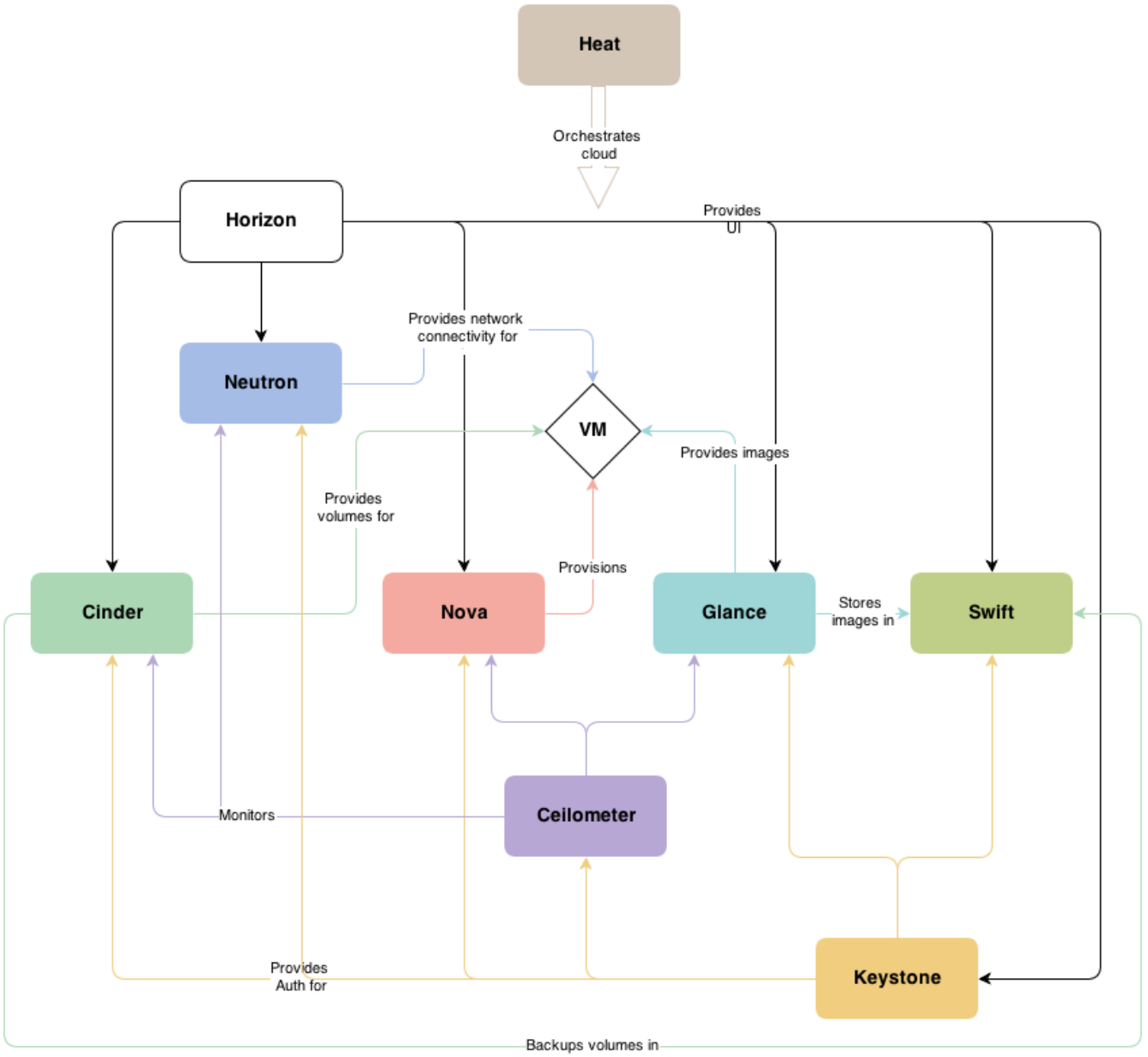
- GLANCE (Images Service): dịch vụ cung cấp các image cho máy ảo của OpenStack.
- NOVA (Compute Service): dịch vụ tính toán, quản lý vòng đời máy ảo bên trong OpenStack.
- NEUTRON (Network Service): dịch vụ cung cấp kết nối mạng cho các thành phần của OpenStack.
- CINDER (Block Service): dịch vụ lưu trữ volume cho các máy ảo.
- HORIZON(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.



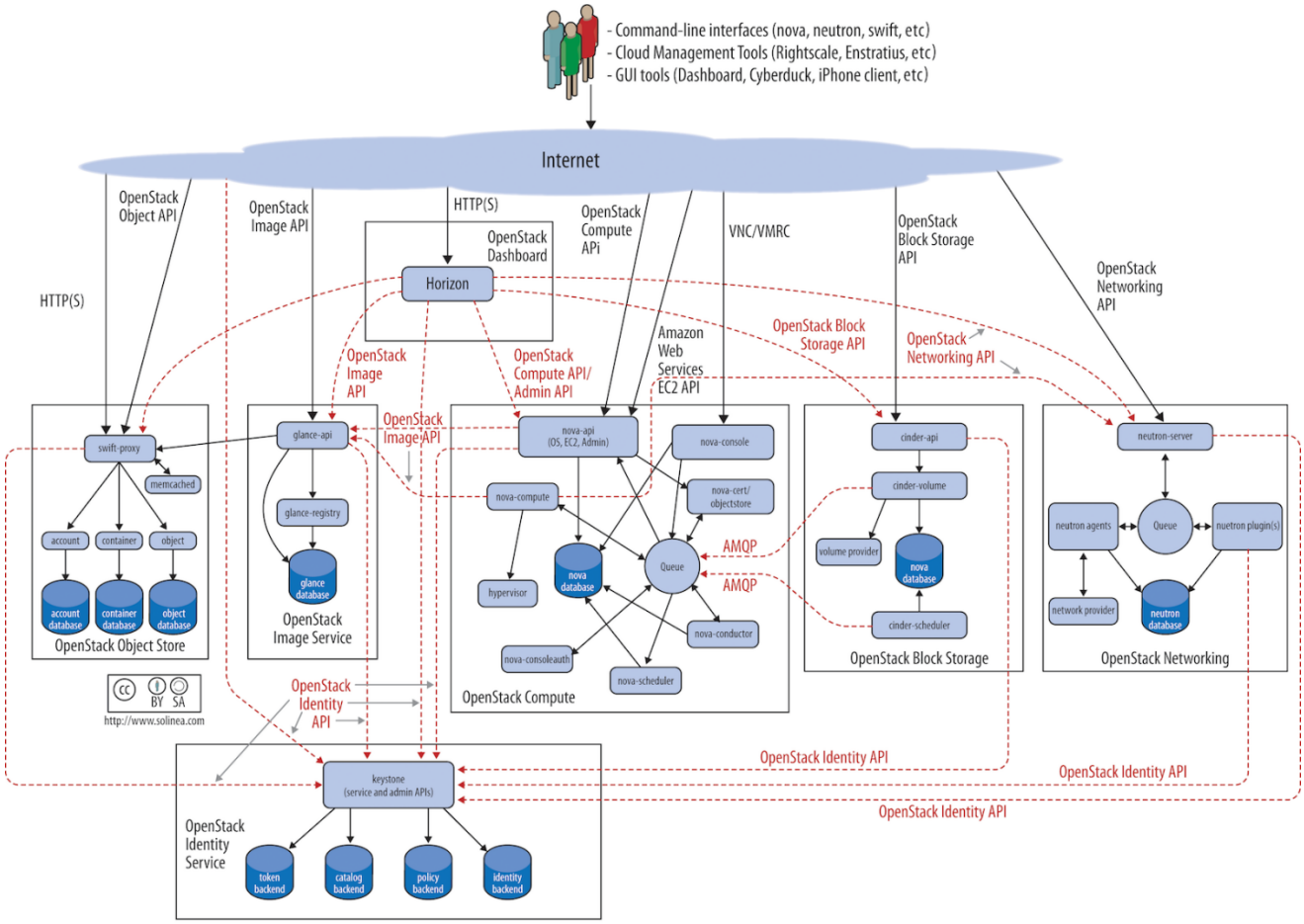
Hình 2.3 Mô hình triển khai OpenStack

Kiến trúc OpenStack mức khái niệm:



Hình 2.4 Kiến trúc OpenStack mức khái niệm

Kiến trúc OpenStack mức logic:



Hình 2.5 Kiến trúc OpenStack mức logic

2.3. Tổng quan về công nghệ ảo hóa các chức năng mạng - NFV

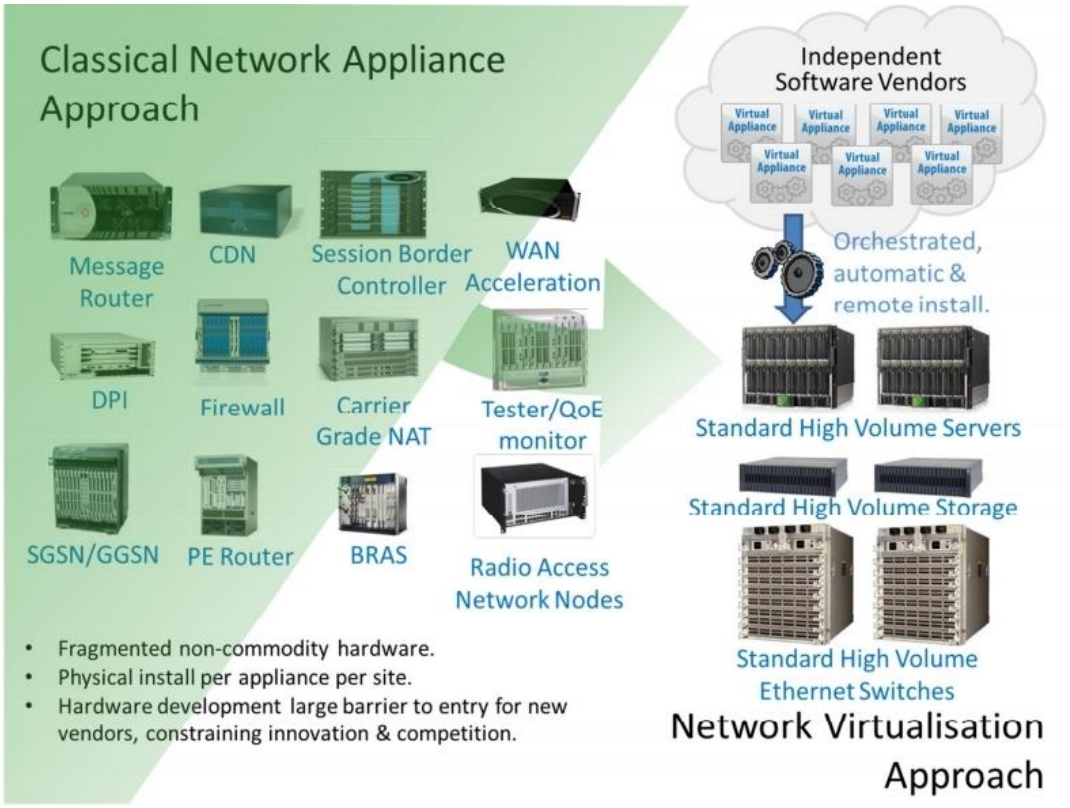
Hệ thống mạng viễn thông hiện tại được vận hành sử dụng các thiết bị phần cứng độc quyền của nhiều nhà cung cấp khác nhau. Việc vận hành dịch vụ mạng mới đồng nghĩa với việc sử dụng thêm nhiều thiết bị hơn, đòi hỏi phải mở rộng không gian để triển khai, đặt ra vấn đề về việc chi phí năng lượng ngày càng tăng, thách thức về vốn đầu tư, yêu cầu các kỹ năng cần thiết để thiết kế, tích hợp và vận hành các thiết bị mạng vật lý càng ngày càng phức tạp. Ngoài ra vòng đời các thiết bị phần cứng cũng không dài, yêu cầu có kế hoạch về chu kì thiết kế - tích hợp - triển khai phù hợp. Tệ hơn, vòng đời của

phần cứng đang ngày một ngắn dần do sự phát triển nhanh chóng của các dịch vụ và công nghệ, gây khó khăn cho việc triển khai các network services mới để thu về lợi nhuận, hạn chế sự đổi mới bởi vì xu hướng hiện tại là hướng về các giải pháp mạng lưới tập trung.

Network Functions Virtualization (NFV) ra đời mang đến cách thức mới để thiết kế, triển khai và quản lý các dịch vụ mạng, sử dụng các công nghệ ảo hóa tiêu chuẩn hiện tại để hợp nhất nhiều loại thiết bị mạng trên các máy chủ, switches và storages theo tiêu chuẩn công nghiệp được đặt trong các trung tâm dữ liệu, các nút mạng và tại nhà của người dùng cuối. NFV tách biệt các chức năng mạng (NAT, firewalling, intrusion detection, DNS, caching) khỏi các thiết bị vật lý và triển khai dưới hình thức phần mềm và có thể chạy trên các máy chủ vật lý truyền thống, đồng thời có thể di trú hoặc được khởi tạo trên nhiều vị trí trong hệ thống mạng theo yêu cầu mà không cần phải triển khai thiết bị mới như trước đây.

Những lợi ích của NFV có thể kể tới như:

- Giảm chi phí vốn: giảm chi phí để mua những phần cứng chuyên dụng để triển khai các chức năng mạng, hỗ trợ mô hình pay-as-you-grow (chi trả theo mức độ mở rộng), bớt lãng phí dự phòng không cần thiết.
- Giảm chi phí vận hành: thu hẹp không gian, chi phí về năng lượng và làm mát cho các thiết bị, đơn giản hóa việc quản lý các dịch vụ mạng.
- Cung cấp nhanh chóng và linh hoạt: NFV cho phép nhanh chóng mở rộng hoặc thu hẹp quy mô dịch vụ để giải quyết những thay đổi trong yêu cầu của khách hàng bằng việc triển khai trong các máy chủ tiêu chuẩn công nghiệp.
- Tăng tốc đưa dịch vụ mới vào thương mại: giảm bớt thời gian triển khai dịch vụ mạng mới đáp ứng thay đổi của doanh nghiệp, nắm bắt những cơ hội thị trường mới và tăng lợi nhuận khi đầu tư vào các dịch vụ mới đó. Thêm vào đó, nhà cung cấp có thể thử nghiệm các dịch vụ mới mà ít gặp rủi ro hơn và đáp ứng tốt hơn nhu cầu ngày một thay đổi của khách hàng.

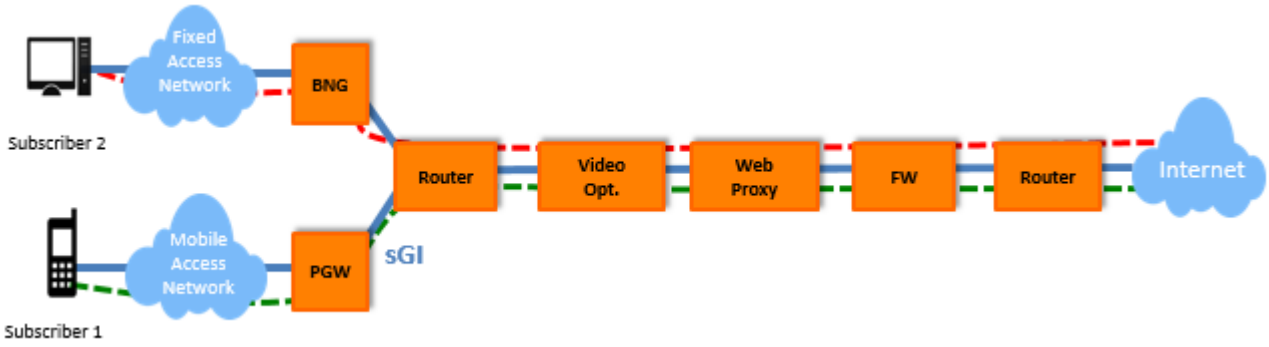


Hình 2.6 NFV

2.4. Chuỗi các chức năng mạng (Service Function Chaining – SFC)

Đối với nhà cung cấp dịch vụ, để triển khai và cung cấp hoàn chỉnh một dịch vụ đầu cuối cho khách hàng thường yêu cầu sử dụng nhiều thiết bị cung cấp các dịch vụ mạng. Các thiết bị này đặt ở nhiều nơi trong hệ thống mạng của nhà cung cấp, trong trung tâm dữ liệu hoặc giữa các trung tâm dữ liệu của nhà cung cấp với nhau. Trong đó bao gồm hai loại: thiết bị chuyển tiếp (forwarding device) như router, switch và thiết bị biến đổi, điều tra, lọc và xử lý lưu lượng (middlebox) như NAT (Network Address Translators), firewall (tường lửa), DPI (Deep Packet Inspection – Kiểm soát gói mức độ sâu), IDS/IPS (Intrusion Detection System/ Intrusion Prevention System – Hệ thống phát hiện / ngăn chặn xâm nhập). Service Function Chaining (SFC) hay chuỗi các dịch vụ

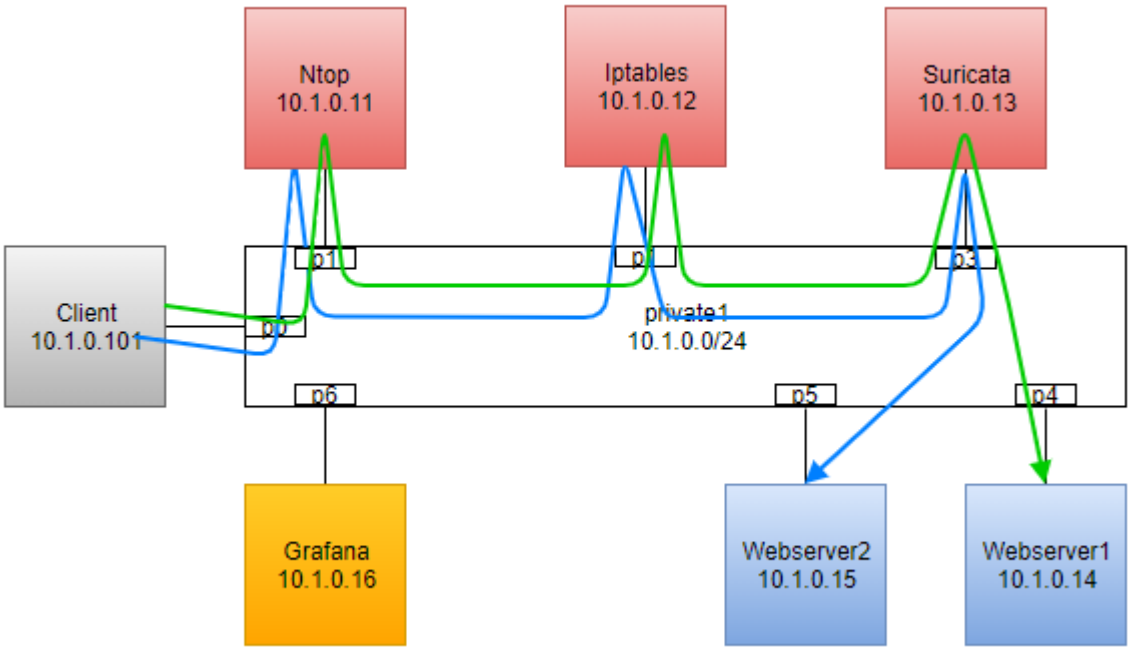
mạng là tập hợp các dịch vụ mạng theo trật tự nhất định và điều khiển lưu lượng đi qua chuỗi dịch vụ đó.



Hình 2.6 Chuỗi chức năng mạng

CHƯƠNG 3. XÂY DỰNG CHUỖI CHỨC NĂNG MẠNG

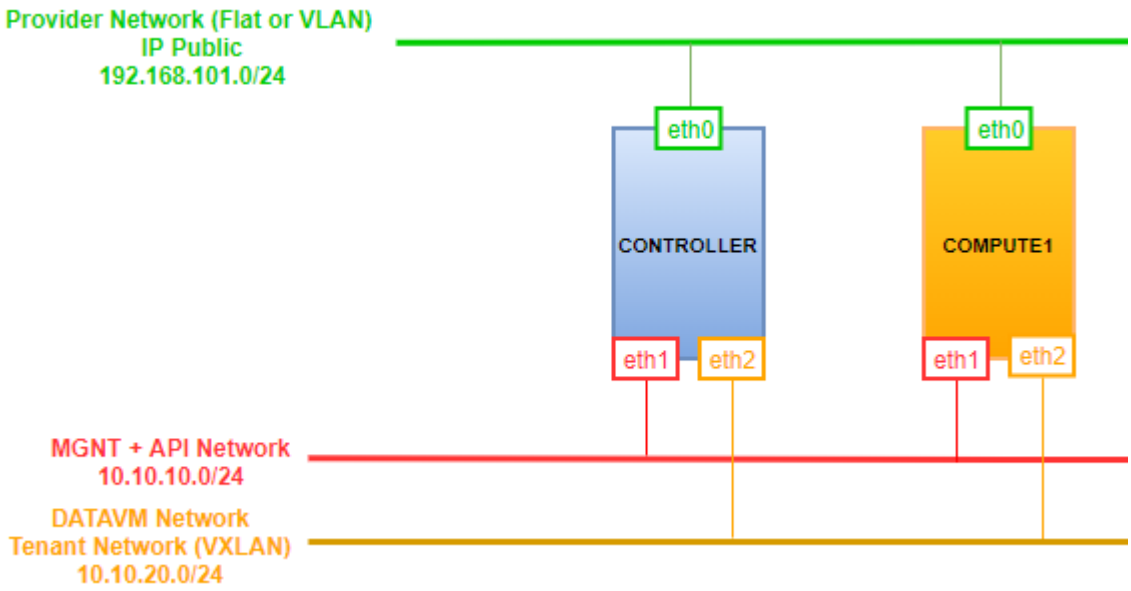
Chương này mô tả phần xây dựng testbed chuỗi chức năng mạng đã đề xuất ở chương 2. Xây dựng mô hình chuỗi chức năng mạng trên nền tảng OpenStack. Các chức năng mạng đều là máy ảo được tạo bởi OpenStack.



Hình 3.1 Mô hình logic testbed

Hình 3.1 mô tả đồ hình testbed mức logic. Chuỗi chức năng mạng triển khai gồm chức năng giám sát lưu lượng mạng (Ntop), 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. Triển khai hệ thống OpenStack



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.
  - Dải DATAVM Network: cung cấp kết nối giữa các VM bên trong hệ thống.

- 3 dải này có thể được cấu hình trên 3 VLAN khác nhau của một switch vật lý để tiết kiệm chi phí.

3.2. Xây dựng luồng đi chuỗi chức năng mạng SFC

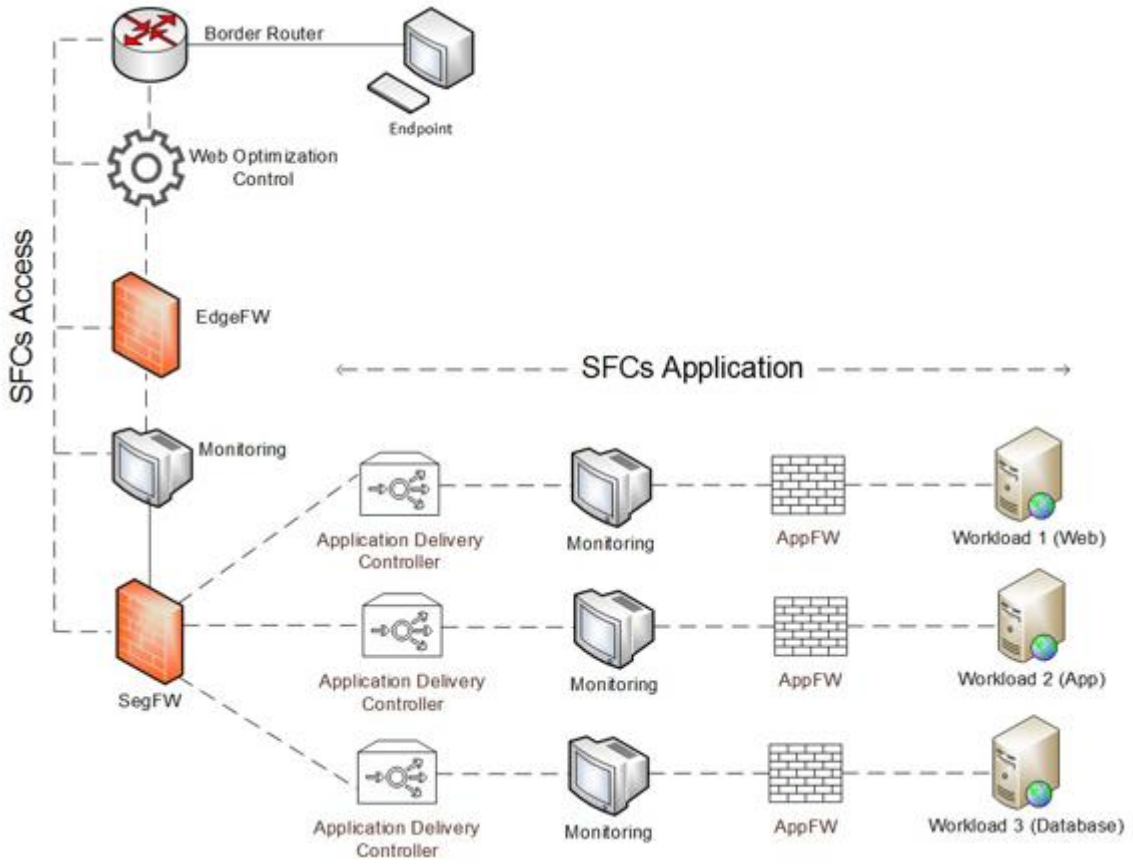
3.2.1. SFC trong trung tâm dữ liệu

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 chống 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 exgress 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.

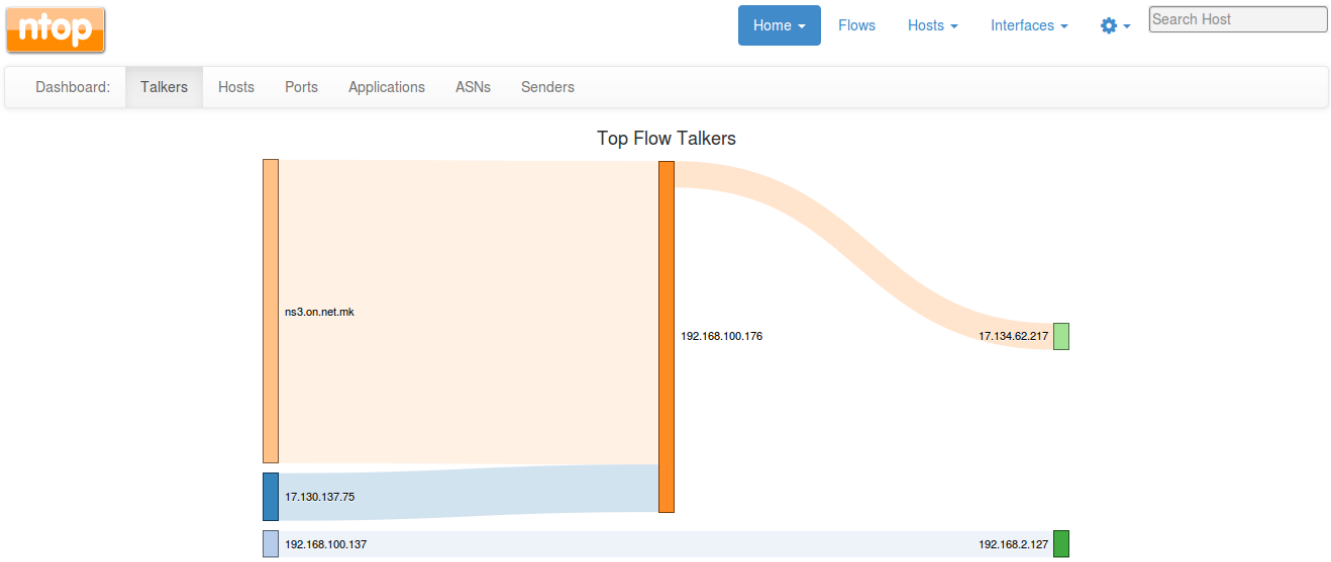
Một chuỗi các port (port chain) được coi là một service chain vô hướng. Một SFC bao gồm hai port chain vô hướng là SFC hai chiều.

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.

3.3. Các công nghệ sử dụng làm các chức năng mạng

3.3.1. Chức năng mạng giám sát - Ntop

Ntop là công cụ mã nguồn mở được sử dụng để giám sát các giao thức mạng khác nhau trên server của bạn. Ntop cung cấp giao diện web thân thiện với người dùng để lấy thông tin về lưu lượng và trạng thái hệ thống mạng.

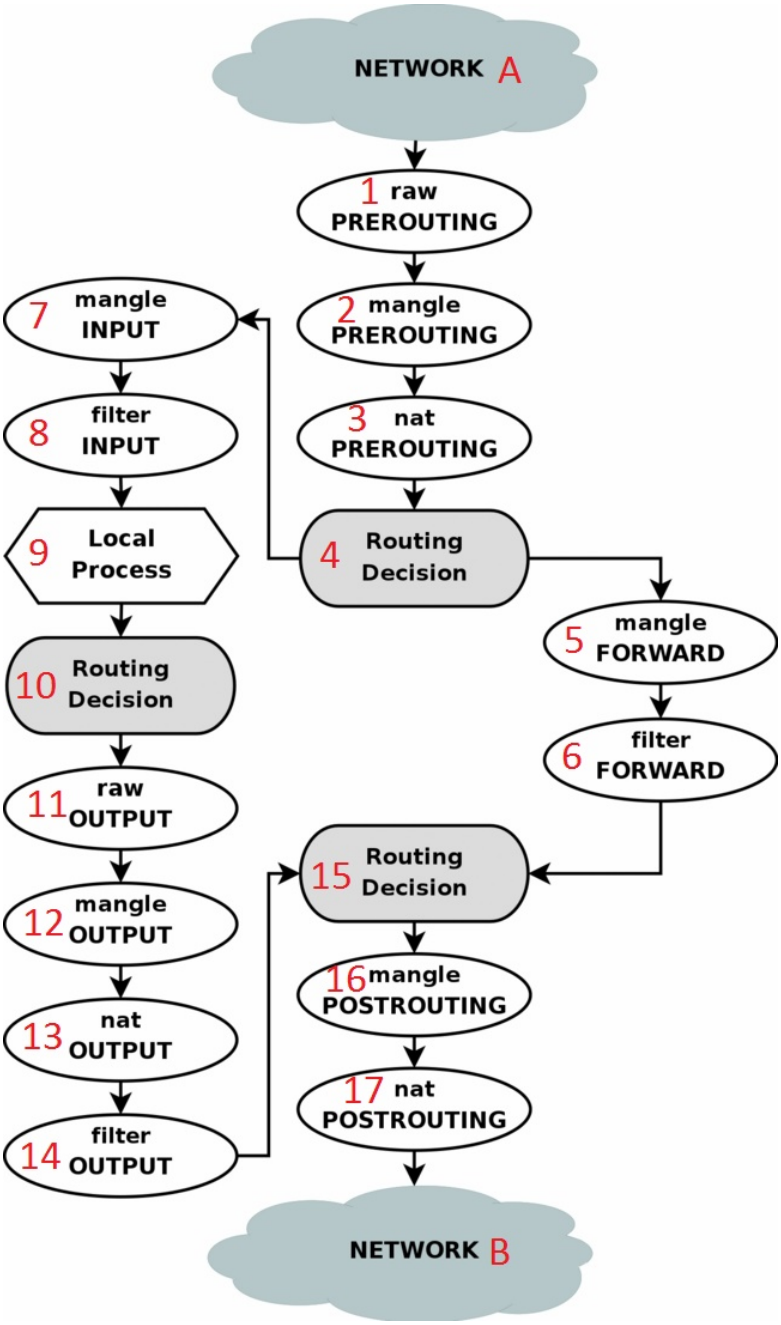


Hình 3.4 Giám sát mạng với Ntop

3.3.2. 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 doạ hàng đầu đối vói 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.

3.3.3. 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 trong trạng thái bình thường để tìm ra các dấu hiệu bất thường.

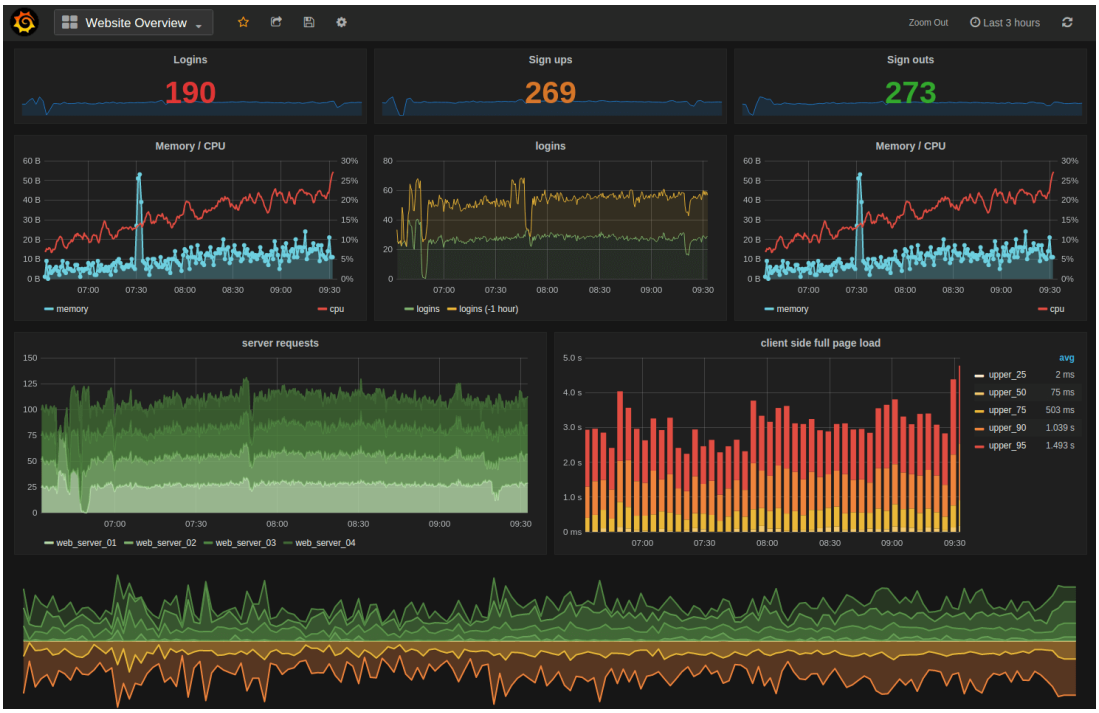
Hệ thống IDS được chia thành hai loại cơ bản:

- 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.

3.3.4. 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 3.6 Giám sát hệ thống với 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 daaashboard 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.

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 bảo 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 Ntop, Iptables, Suricata rồi mới tới Web server bên trong.

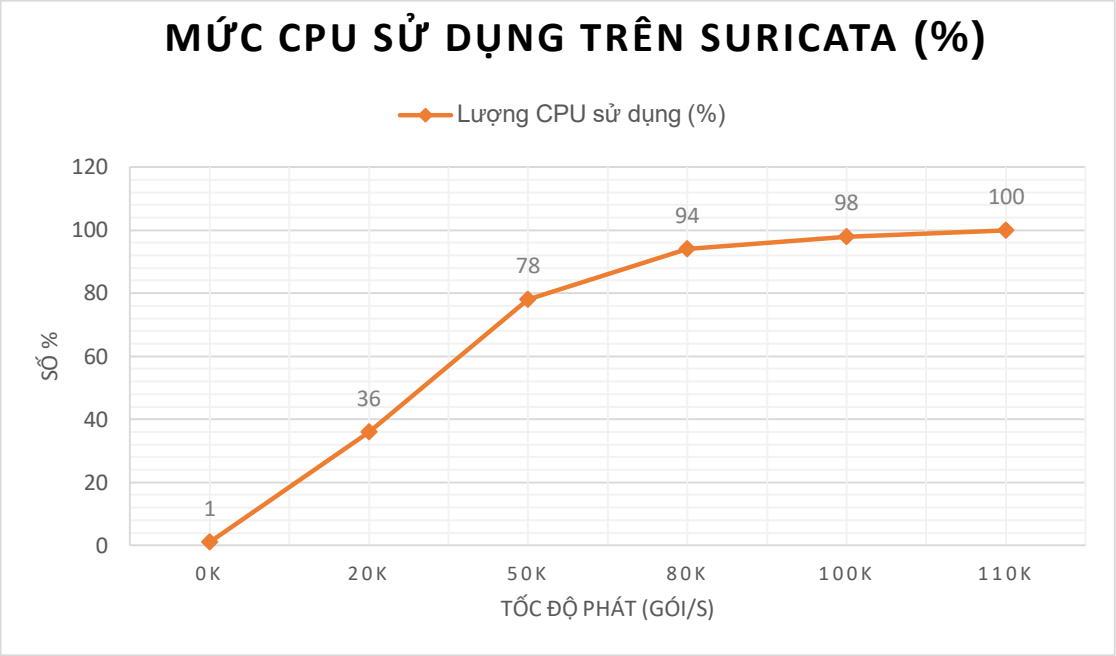
Từ bên ngoài, thực hiện ping vào địa chỉ IP public của máy chủ Webserver, ta kiểm tra tại các VNF sẽ thấy lưu lượng mạng đi qua chúng.

4.2. Kết quả đo lượng 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

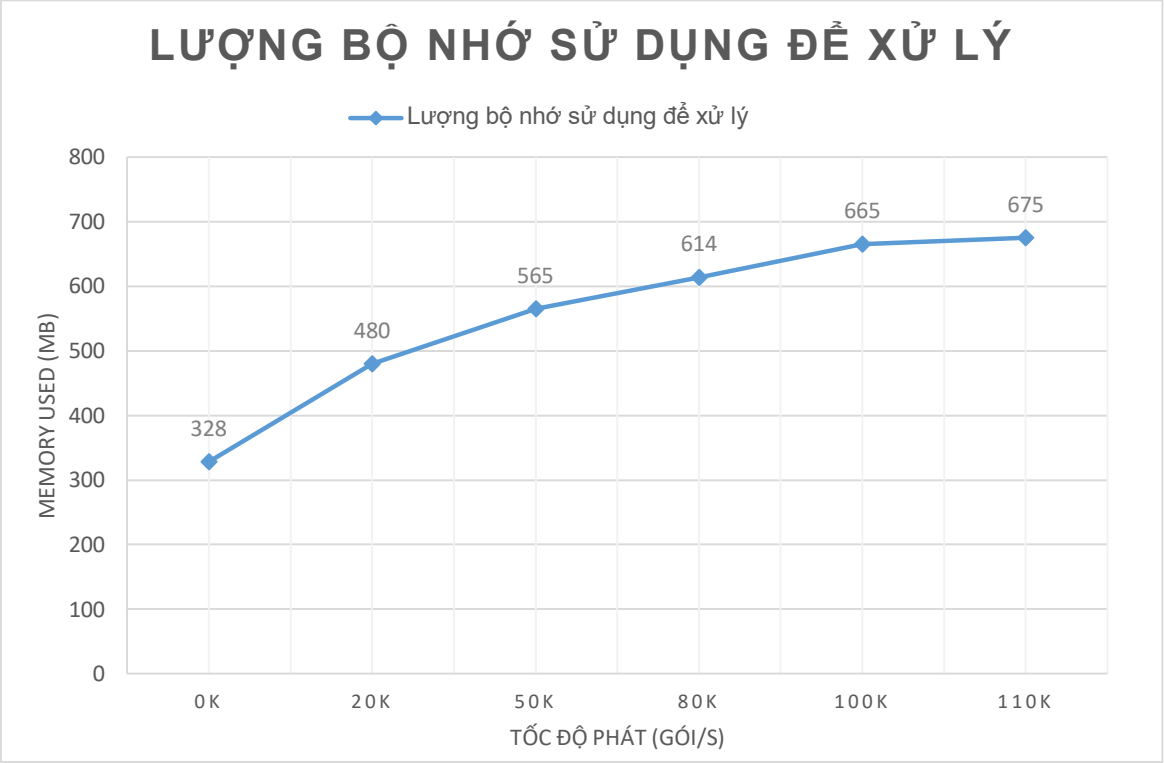
Trên Suricata, Iptables và máy chủ Webserver, cài đặt thêm công cụ collectd để thu thập tham số hệ thống, gửi về máy Grafana (có cài đặt Graphite). Số liệu được hiển thị trên Grafana cho phép giám sát các tài nguyên của các máy ảo.

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). Kết quả thu được như sau:



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

Biểu đồ 4.1 cho thấy mức sử dụng CPU của Suricata khi tải đầu vào tăng dần. Ban đầu, khi chưa đẩy tải vào chuỗi chức năng mạng, mức sử dụng CPU của Suricata rất ít. Sau đó, tăng tải từ từ, mức sử dụng CPU tăng dần, đạt đến mức ngưỡng là hơn 100000 gói tin/s (gói TCP).

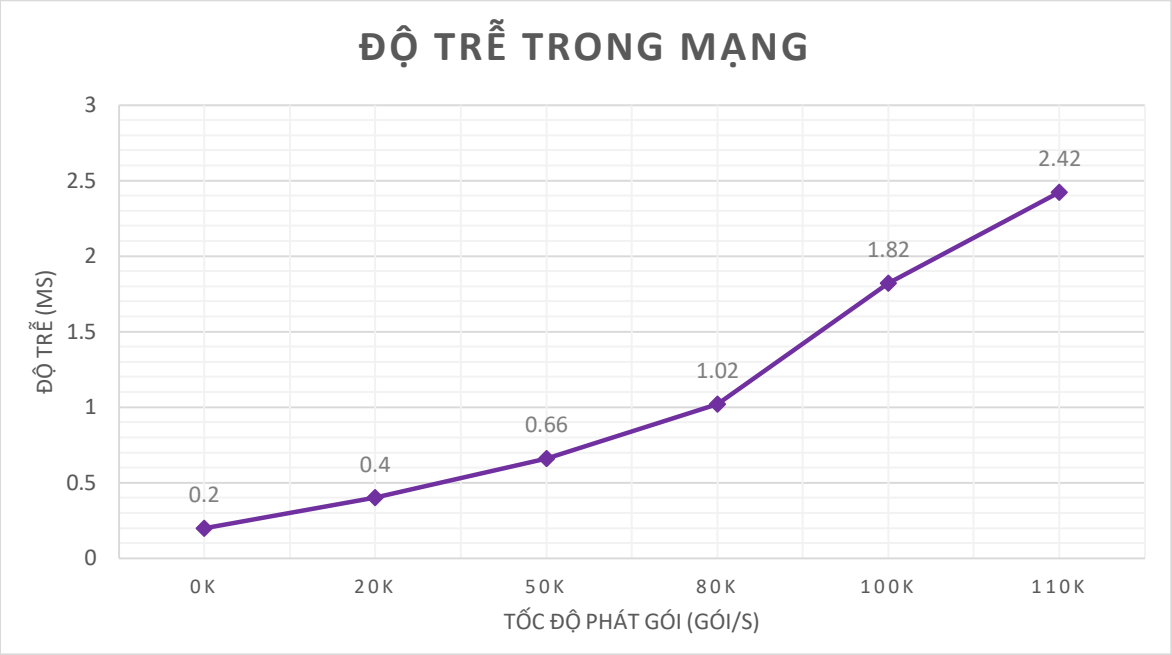


Biểu đồ 4.2. Lượng bộ nhớ sử dụng để xử lý các gói tin trên Suricata khi tải tăng dần

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

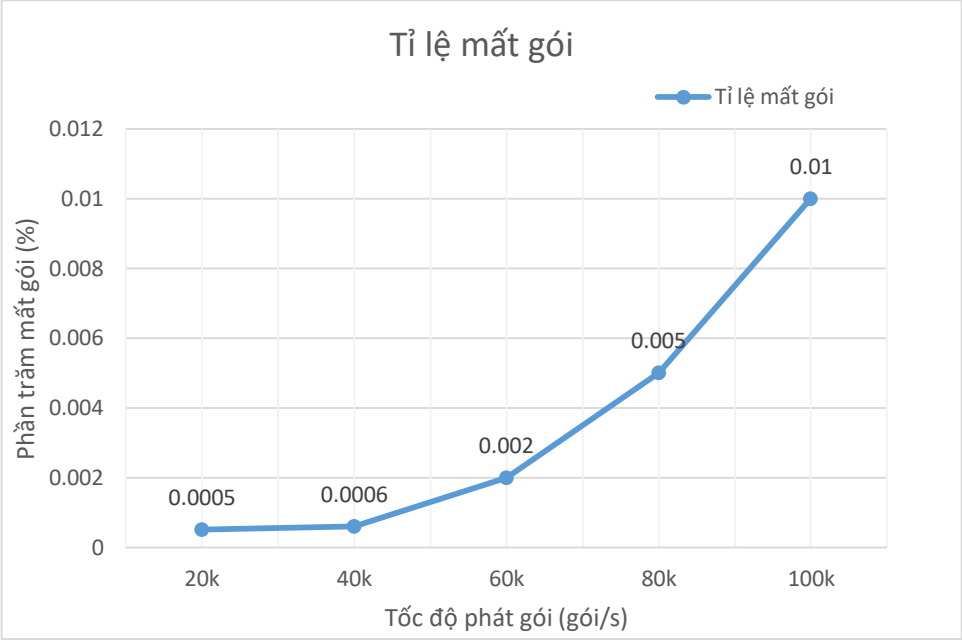
Phát lần lượt số lượng các gói tin TCP tăng dần vào chuỗi SFC. Thực hiện kích hoạt plugin ping (của collectd) trên máy client từ bên ngoài mạng. Cấu hình client cứ mỗi giây ping tới web sererver một lần để đo thông số thời gian round trip time của gói tin trả về. Độ trễ gói tin trả về tăng dần khi tải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

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:

- 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

Qua quá trình thực hiện đồ án, em đã có được thêm nhiều hiểu biết về công nghệ ảo hóa chức năng mạng và chuỗi các dịch vụ mạng.

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.

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 Funtions 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]. Availble: 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], Volume75, Pages 138-155. Availble: https://www.sciencedirect.com/science/article/pii/S1084804516301989#s0010

[6] https://docs.openStack.org/pike/ truy cập cuối cùng ngày 06/02/2019

[7] https://docs.openStack.org/ocata/networking-guide/config-sfc.html truy cập lần cuối ngày 06/02/2019.

[8] http://www.faqs.org/docs/iptables/ truy cập lần cuối ngày 06/02/2019.

[9] OISF, Suricata User Guide, [Online], Availble : https://media.readthedocs.org/pdf/suricata/latest/suricata.pdf

[10] Bonesi. [Online]. Availble: https://github.com/Markus-Go/bonesi

[11] https://docs.openstack.org/install-guide/ truy cập lần cuối ngày 01/02/2019.

[12] Grafana. [Online]. Availble: https://grafana.com/

[13] Collectd. [Online]. Availble: https://collectd.org/

[14] Graphite. [Online]. Availble: https://graphiteapp.org/