ĐÁNH GIÁ HIỆU NĂNG MẠNG TRÊN ĐIỆN TOÁN Đám Mây OPENSTACK

Võ Nhân Văn^a, Lê Minh Chí^b, Nguyễn Quốc Long^a, Lê Đắc Như^{c*}

^aKhoa Công nghệ Thông tin, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam

^bSở thông tin và Truyền thông, Đà Nẵng, Việt Nam

^cKhoa Công nghệ Thông tin, Trường Đại học Hải Phòng, Hải Phòng, Việt Nam

Nhận ngày 04 tháng 01 năm 2016

Chỉnh sửa ngày 19 tháng 03 năm 2016 | Chấp nhận đăng ngày 31 tháng 03 năm 2016

Tóm tắt

Điện toán đám mây đã bắt đầu phổ biến nhờ tính linh hoạt, có khả năng mở rộng, dễ dàng kích hoạt và cung cấp các dịch vụ để xây dựng tài nguyên mà khách hàng có thể truy cập theo yêu cầu của họ. Ngoài ra dịch vụ điện toán đám mây có khả năng chia sẻ tài nguyên hiệu quả giữa các người dùng với nhau. OpenStack, một nền tảng điện toán đám mây mã nguồn mở tương đối mới, tập trung vào việc cung cấp mạng như một dịch vụ (NaaS) sử dụng công nghệ ảo hóa và OpenStack hứa hẹn là một nền tảng điện toán đám mây sẽ được mở rộng ở ạt trong các ứng dụng của doanh nghiệp. Từ đó, mục đích của bài báo là nghiên cứu hiệu năng mạng của OpenStack triển khai trên phiên bản Neutron. Các thông số về hiệu năng như thông lượng, vấn đề mất gói tin, thời gian trễ gói tin sẽ được đánh giá theo giao thức UDP dựa vào công cụ IPERF.

Từ khóa: Điện toán đám mây; Hiệu năng mạng; OpenStack.

---

1. GIỚI THIỆU CHUNG

Điện toán đám mây là một vấn đề nóng hiện nay và các nhà nghiên cứu đã đưa ra nhiều ý tưởng trong lĩnh vực này. Cho đến nay có rất nhiều công nghệ được sử dụng cho điện toán đám mây, một trong số đó là EC2, OpenStack, OpenNebula, CloudStack... Điện toán đám mây cung cấp một số dịch vụ như IaaS (Infrastructure as a Service), PaaS (Platform as a Service), DaaS (Data as a Service) và SaaS (Software as a Service). Do nhu cầu của điện toán đám mây phát triển rất nhanh chóng, hiệu quả tốt, đủ để đáp ứng nhu cầu cho người sử dụng. Tuy nhiên, một trong những mối quan tâm quan trọng nhất của điện toán đám mây là phải đạt một hiệu suất mạng tốt, nếu không thì

---

* Tác giả liên hệ: Email: nhuongld@hus.edu.vn

không thể tạo được một hệ thống mạng chất lượng. Do đó, bài báo này tập trung phân tích đánh giá hiệu năng điện toán đám mây dựa trên OpenStack.

1.1. Điện toán đám mây

Điện toán đám mây có thể được định nghĩa [1] là một kiểu máy tính mới trong đó các nguồn lực có thể được mở rộng một cách tự động và thông thường được ảo hóa để cung cấp như một dịch vụ trên Internet. Điện toán đám mây đã và đang trở thành một xu hướng công nghệ trọng yếu và nhiều chuyên gia mong đợi rằng điện toán đám mây sẽ thay đổi hình dáng của các quy trình và thị trường của công nghệ thông tin. Với công nghệ điện toán đám mây hiện nay, người dùng sử dụng một loạt thiết bị như máy tính, máy tính xách tay, điện thoại PDA để truy cập các chương trình, hệ thống lưu trữ hay các nền tảng phát triển hệ thống thông qua Internet bởi các ứng dụng được cung cấp bởi các nhà cung cấp dịch vụ điện toán đám mây. Ưu điểm của công nghệ này là tiết kiệm chi phí, tính sẵn sàng cao và khả năng mở rộng dễ dàng.

Các tính năng chính [2]:

- **Rapid elasticity:** nhà cung cấp CC dễ dàng chỉ định cũng như thu hồi tài nguyên người dùng rất nhanh chóng. Về phía người dùng được phép yêu cầu một tài nguyên “không giới hạn” và chỉ việc chi trả theo tiền.
- **Broad network access:** truy cập vào các tài nguyên máy tính dễ dàng thông qua các cơ chế network tiêu chuẩn.
- **Measured service:** provider đảm bảo việc tính toán lượng tiêu dùng của khách hàng. Mô hình hướng đến là “pay as you go”.
- **On-demand self-service:** cho phép khách hàng tùy chỉnh tài nguyên sử dụng mà không cần phải thông báo hay qua bất kỳ sự can thiệp nào của provider.
- **Resource pooling:** các loại tài nguyên vật lý và ảo của CC được chia sẻ với nhau và tự động cấp cho các users.

Các mô hình triển khai

- Có 3 mô hình triển khai điện toán đám mây chính là public (công cộng), private (riêng), và hybrid (“lai” giữa đám mây công cộng và riêng).
- Đám mây công cộng là mô hình đám mây mà trên đó, các nhà cung cấp đám mây cung cấp các dịch vụ như tài nguyên, platform, hay các ứng dụng lưu trữ trên đám mây và public ra bên ngoài. Các dịch vụ trên public cloud có thể miễn phí hoặc có phí [3].
- Đám mây riêng thì các dịch vụ được cung cấp nội bộ và thường là các dịch vụ kinh doanh, mục đích nhắm đến cung cấp dịch vụ cho một nhóm người và đứng đằng sau firewall.
- Đám mây “lai” là môi trường đám mây mà kết hợp cung cấp các dịch vụ công cộng và riêng [3].
- Ngoài ra còn có “community cloud” là đám mây giữa các nhà cung cấp dịch vụ đám mây.

Các mô hình dịch vụ: Về mô hình cung cấp dịch vụ có 3 loại chính là IaaS – cung cấp hạ tầng như một server, PaaS – cung cấp Platform như một service, và SaaS – cung cấp software như một service.

1.2. OpenStack

“Sứ mạng của OpenStack: Cung cấp nền tảng điện toán đám mây mã nguồn mở phổ biến để đáp ứng được nhu cầu của bất kể kích cỡ đám mây công cộng hay đám mây riêng, bằng cách đơn giản hóa việc triển khai và khả năng mở rộng cao”. [4]

Đúng như sứ mạng trên, OpenStack là một hệ thống quản lý đám mây, điều khiển lượng lớn các tài nguyên tính toán, lưu trữ và mạng nằm trong một trung tâm dữ liệu. Tất cả các thao tác quản lý được thực thi thông qua một bảng điều khiển nhằm cung cấp cho quản trị viên khả năng kiểm soát hệ thống và cho phép người dùng tương tác với tài nguyên mà họ cần thông qua giao diện web. Openstack hoàn toàn là nguồn

mở, các thành phần của nó được viết trên Python - ngôn ngữ đang được đánh giá rất cao những năm gần đây.

Sau đây là sơ đồ kiến trúc ở mức logic của Openstack [5] (Hình 1):

Hình 1. Các thành phần cơ bản trong OpenStack

Trong đó:

- **Dashboard** (tên mã Horizon): Cung cấp giao diện điều khiển dưới dạng web, cho phép người dùng tương tác với các dịch vụ của OpenStack để khởi tạo một máy ảo, gán địa chỉ IP, thiết lập kiểm soát truy cập...
- **Compute** (tên mã Nova): Cung cấp việc xây dựng và quản lý các máy ảo
- **Networking** (tên mã Neutron): Cung cấp kết nối mạng giữa các máy ảo, giữa máy ảo và Internet.
- **Object Storage** (tên mã swift): Cung cấp dịch vụ lưu trữ file.
- **Block Storage** (tên mã Cinder): Cung cấp thành phần lưu trữ dự phòng cho các máy ảo, tương tự USB, HDD di động,...
- **Identity** (tên mã Keystone): Cung cấp dịch vụ chứng thực và ủy quyền cho các dịch vụ khác trong OpenStack.

- **Image** (tên mã glance): Cung cấp các file chứa hệ điều hành dành cho các máy ảo.
- **Metering/Monitoring** (tên mã Ceilometer): Quản lý và thống kê các thông số thay đổi trong quá trình hoạt động.
- **Orchestration** (tên mã Heat): Cung cấp khả năng xây dựng tự động các dịch vụ như web server, firewall,... cho máy ảo; Điều phối các hoạt động cho đám mây.
- **Database Service** (tên mã trove): Cung cấp khả năng mở rộng và tin cậy của chức năng Cloud Database as a Service cho cả hai công cụ cơ sở dữ liệu tin tưởng và không tin tưởng.

1.3. Kết nối mạng trong OpenStack

OpenStack Networking cung cấp một lượng API phong phú để quản lý kết nối và xác định kết nối trong hạ tầng cloud. Nó hỗ trợ nhiều công nghệ mạng khác nhau để kết nối vào hạ tầng cloud. Chúng ta sẽ tìm hiểu về mô hình triển khai phần Network, và một số dịch vụ cơ bản của network cung cấp cho hạ tầng cloud. Neutron bao gồm các tài nguyên trừu tượng là Network, Subnet, Port.

- **Network**: quản lý cô lập mạng ở Layer 2, giống VLAN trong mạng vật lý.
- **Subnet**: nhóm địa chỉ IP v4/v6 được phân cấp và cấu hình cho hệ thống.
- **Port**: một port sẽ được gán với một thiết bị ảo trong hạ tầng.

Người dùng có thể tạo ra mô hình mạng riêng của mình, bằng cách tạo ra Network và subnet và sau đó hướng dẫn cho các dịch vụ của khác, như Compute để nó có thể gán thiết bị ảo đến port trên network đó. Đặc biệt là Neutron cho phép tạo ra mạng riêng (Private network) và gán cho các tenant, một tenant có thể có nhiều private network và các private network có thể trùng nhau cho mỗi tenant. Các dịch vụ của Neutron cung cấp bao gồm:

- Tạo ra một topology riêng cho hệ thống của riêng mình. Ví dụ như tạo ra hệ thống webserver nhiều tầng, để backup hoặc cân bằng tải.
- Cung cấp sự linh hoạt trong quản trị hạ tầng mạng, cho phép tùy chỉnh hạ tầng cao.
- Cung cấp API mở rộng, cho phép phát triển và tích hợp Neutron vào nhiều hệ thống khác nhau.

Neutron hỗ trợ nhiều công nghệ mạng khác nhau để tạo ra nhiều chức năng hơn, ngoài việc chia VLAN, Subnet, còn có thể làm switch ảo qua công nghệ của vSwitch, các chức năng khác như Firewall, DHCP, VPN, Load balancing.

2. MÔ HÌNH THỰC NGHIỆM

Dịch vụ điện toán đám mây sẽ cung cấp các máy ảo với VCPU, RAM, khả năng lưu trữ khác nhau. Trong OpenStack, các thành phần liên quan đến mạng và chuyển mạch cũng được ảo hóa thông qua Neutron. Neutron – phần mềm để định nghĩa các chức năng của một hệ thống mạng, từ lớp 2 cho đến lớp 3, từ switch cho đến router và gắn vào hoạt động của các instances. Việc truyền thông của các máy ảo trong OpenStack sẽ không giống như việc truyền dẫn trên các hệ thống mạng thông thường vì các thành phần của mạng được định hướng theo mô hình SDN (Software Define Network).

Trong OpenStack, việc triển khai Neutron được hướng đến theo 2 hình thức: sử dụng GRE tunnel hoặc theo mô hình VLAN. GRE tunnel với ưu thế linh hoạt và dễ cấu hình tuy nhiên hiệu suất thấp, mô hình VLAN có nhiều ưu điểm trong việc triển khai các hệ thống lớn tuy nhiên cần đòi hỏi số lượng thiết bị, các thiết kế và cấu hình cao hơn. Để phục vụ cho mục đích thử nghiệm, mô hình GRE tunnel được sử dụng để hạn chế việc triển khai phức tạp đối với hạ tầng vật lý.

Do đó, mô hình thử nghiệm sử dụng GRE tunnel sẽ được thực hiện trong 4 kịch bản nhằm mục đích nghiên cứu hiệu suất mạng về thông lượng, thời gian trễ và vấn đề mất gói tin UDP giữa các máy ảo.

Trong bài báo này chúng tôi đưa ra 4 kịch bản thử nghiệm như sau: case 1 - Cùng mạng cùng compute node, case 2 - Khác mạng cùng compute node, case 3- Cùng mạng khác compute node, case 4 - Khác mạng khác compute node. Bởi vì Network trong OpenStack được xem như là một tài nguyên và được ảo hóa toàn bộ để hướng đến việc thao tác và quản trị đơn giản. Về lý thuyết, hoạt động của hệ thống network “ảo hóa” này không khác gì so với hệ thống network vật lý. Trong OpenStack cũng hình thành nên các port ảo tương đương với các kết nối vật lý giữa các thiết bị, các router ảo tương đương với việc phân chia các dải mạng với nhau, các khái niệm như VLAN hoặc access control list, ... cũng được thể hiện tương tự. Vì vậy để đánh giá hoạt động của lớp ảo hóa Network này thì cần đánh giá dựa trên các tiêu chí sau: - Xem xét việc kết nối mạng trong hệ thống network ảo tại các máy compute node - Xem xét việc kết nối mạng trong hệ thống network ảo giữa các máy compute node với nhau. Từ kết quả đánh giá đó (chính là 4 test cases được thực hiện) mới xem xét đến việc sử dụng OpenStack Network trong thực tế có bảo đảm được như các mô hình network vật lý thông thường.

Hình 2. Mô hình triển khai

Mô hình triển khai trên hệ thống 3 máy chủ, bao gồm:

- 01 máy chủ làm Controller và Network, chạy các dịch vụ quản trị Controller của OpenStack như Keystone, MySQL, Neutron, Cinder, và các dịch vụ liên quan khác.
- 02 máy chủ làm Compute nodes cài đặt Neutron L2 Agent và nền tảng ảo hóa Hypervisor KVM cho việc cung cấp năng lực tính toán dành cho các máy chủ ảo.

Hệ thống mạng của mô hình triển khai bao gồm 4 mạng như sau:

- Public và Floating IP (trong mô hình là VM Network): 10.196.205.0/24
- Management network: 10.196.202.0/24
- Storage network: 10.196.203.0/24. Trong mô hình triển khai Storage network nằm chung kết nối vật lý với Management Network. Storage network không nằm trong giới hạn nghiên cứu của bài báo.
- Internal Network – network dành riêng cho các tenants: 192.168.111.0/24 và 192.168.112.0/24
- Ngoài ra còn có 1 mạng PXE phục vụ cho việc quản trị và cài đặt tự động các máy chủ trong hệ thống.

Case 1. Mô hình thử nghiệm này sẽ thực hiện nhằm nghiên cứu thông lượng, độ trễ, tỉ lệ mất gói tin UDP giữa hai máy ảo cùng node compute và cùng mạng.

Hình 3. Case 1 được thực hiện với 2 máy ảo VM1 và VM2 được đặt cùng mạng 192.168.111.0/24 và cùng node 18

Case 2. Mô hình thử nghiệm này sẽ thực hiện nhằm nghiên cứu thông lượng, thời gian trễ và tỉ lệ mất gói tin UDP giữa hai máy ảo có cùng node compute nhưng nằm ở hai mạng khác nhau. Hai mạng khác nhau có thể nằm trên cùng một router hoặc hai router khác nhau.

Hình 4. Case 2 được thực hiện với 2 máy ảo cùng chung 1 node 18 nhưng khác mạng với nhau (192.168.111.9/24 và 192.168.112.103/24)

Case 3. Mô hình thử nghiệm này sẽ thực hiện nhằm nghiên cứu thông lượng, thời gian trễ và tỉ lệ mất gói tin UDP giữa hai máy ảo có node compute khác nhau nhưng cùng một mạng với nhau.

Hình 5. Case 3 được thực hiện với 2 máy ảo cùng chung 1 mạng (192.168.111.0/24) nhưng ở 2 node khác nhau (node 17 và node 18)

Case 4. Mô hình thử nghiệm này sẽ thực hiện nhằm nghiên cứu thông lượng, thời gian trễ, tỉ lệ mất gói tin UDP giữa hai máy ảo có node compute khác nhau và khác mạng với nhau.

Hình 6. Case 4 được thực hiện với 2 máy ảo khác mạng (192.168.112.103 và 192.168.111.10) cũng như khác node nhau (node 17 và node 18)

3. PHÂN TÍCH VÀ ĐÁNH GIÁ

Dịch vụ network trong OpenStack chính là Neutron, và Neutron tạo ra các môi trường chuyển mạch (switch), định tuyến (router) thông qua các dịch vụ Neutron L2 Agent, Neutron L3 Agent. Các mô hình thử nghiệm về mạng được mô tả ở trên do module Neutron trong OpenStack tạo ra.

Kết quả thử nghiệm tập trung vào việc phân tích các tham số liên quan đến việc truyền nhận gói tin trong hệ thống các máy ảo của OpenStack, các thông số được phân tích bao gồm thông lượng, tỉ lệ mất gói tin trên đường truyền, và thời gian trễ của gói tin.

3.1. Thông lượng TCP

Hình 7a cho thấy giá trị của TCP được thu thập mỗi 5s một lần trong vòng 5 phút. Từ sơ đồ này cho ta thấy các thông lượng của máy ảo trên cùng một node compute và cùng địa chỉ mạng thì dao động xung quanh giá trị 890Mbps, trong khi các máy ảo trên cùng một node compute nhưng khác địa chỉ mạng khoảng 500Mbps.

Hình 7a. Thông lượng TCP trong 4 trường hợp

Từ sơ đồ Hình 7b ta có thể thấy thông lượng cho các máy ảo trên cùng một node compute và cùng địa chỉ mạng là cao hơn nhiều so với các trường hợp còn lại. Điều này lý giải là do khi 1 máy khác địa chỉ mạng thì thông lượng cần phải được gửi tới Router do OpenStack tạo ra.

Hình 7b. Thông lượng TCP trung bình trong 4 trường hợp

3.2. Thông lượng và tỉ lệ mất gói tin UDP

Để đo lường thông lượng, thời gian trễ và tỉ lệ mất gói tin UDP, IPERF được thực thi trong vòng 5 phút và dữ liệu sẽ được lấy định kỳ mỗi 5 giây 1 lần. Trong khi dữ liệu được lấy ở một mô hình thì các mô hình thử nghiệm khác sẽ bị dừng lại.

Từ những kết quả trong Hình 8 và Hình 9, có thể nhận thấy rằng khi 2 máy ảo được đặt trên cùng node và cùng mạng, chúng thực hiện tốt hơn so với khi chúng ở các node và mạng trong các trường hợp khác. Việc đo lường cho thấy các máy ảo trên cùng node, cùng mạng thì đạt được thông lượng trung bình xấp xỉ 4.2% so với các trường hợp còn lại.

Hình 8. Thông lượng UDP trong 4 case

Hình 9. Tỉ lệ mất gói tin UDP trong 4 case

Tương tự trong trường hợp đo lường tỉ lệ mất gói tin thì các máy ảo ở cùng node nhưng khác mạng hoặc khác node và khác mạng thì có thông số cao hơn so với các máy ảo cùng node và cùng mạng. Đặc biệt trong trường hợp các máy ảo cùng node nhưng khác mạng cho chúng ta thấy tỉ lệ mất gói tin xấp xỉ 40% so với trường hợp đặt các máy ảo cùng node và cùng mạng.

3.3. Thời gian trễ

Trong trường hợp đo lường thời gian trễ gói tin, ta nhận thấy giá trị ở case 1 thấp hơn giá trị ở case 2 và giá trị ở case 3 thấp hơn giá trị ở case 4. Điều này cho chúng ta kết luận các máy ảo trên khác địa chỉ mạng có giá trị cao hơn so với các máy ảo có cùng địa chỉ mạng. Kết quả này là bình thường vì khi máy ảo khác mạng nhau thì lưu lượng đi từ nguồn tới đích phải thông qua nhiều hop hơn khi chúng cùng mạng với nhau.

Hình 10. Thời gian trễ gói tin

Mô hình thử nghiệm thường dựa trên các con số lý tưởng khi các thiết bị cũng như đường truyền không bị ảnh hưởng bởi các tham số tác động bên ngoài. Hệ thống các test cases cũng được xây dựng trên môi trường như vậy. Một đặc điểm quan trọng của việc kiểm tra thông lượng thường tập trung vào các gói tin UDP để đánh giá. Vì vậy nên mô hình thử nghiệm cũng căn cứ trên các cơ sở đó để thực hiện. Trong thực tế, các nút mạng cũng như máy chủ sẽ chịu một tải nhất định do ảnh hưởng của môi trường xung quanh, vì vậy nên giá trị thu nhận được trong môi trường LAB sẽ khác biệt so với thực tế và cần phải được kiểm chứng đối với từng trường hợp cụ thể, ví dụ trong môi trường data center của FTP, hoặc data center của VNPT, ... Tuy nhiên bài báo chỉ giới hạn trong nội dung môi trường LAB và chưa đánh giá ảnh hưởng của môi trường thực tế. Vì thực hiện trong môi trường LAB nên số lượng máy chủ như vậy thì có thể bảo đảm cho việc thu thập các thông số kiểm tra và phân tích. Thực tế hệ thống sẽ lớn hơn

nhiều, có thể lên đến hàng trăm máy chủ và việc đánh giá phải được thực hiện trên diện rộng hơn, nằm ngoài phạm vi của bài báo.

4. KẾT LUẬN

Điện toán đám mây mã nguồn mở và đặc biệt là OpenStack hiện nay là một trong những công nghệ được lựa chọn để triển khai cho các công ty trên toàn thế giới. Trong bài báo này, chúng tôi phân tích và đánh giá hiệu năng mạng dựa trên các thông số thông lượng UDP, thời gian trễ, tỉ lệ mất gói tin trên OpenStack. Từ đó đưa ra kết luận OpenStack Neutron đảm bảo một hiệu suất cao, băng thông hầu như không bị hiện tượng thất cổ chai. Ngoài ra, kết quả cũng cho thấy rằng, vị trí của máy ảo cũng như địa chỉ mạng ảnh hưởng đến hiệu suất, cụ thể các máy ảo cùng node và cùng mạng sẽ cho hiệu năng cao hơn tất cả các trường hợp còn lại bởi vì đường truyền của các máy ảo ngắn hơn.

Tuy nhiên, chúng tôi chỉ mới nghiên cứu gói tin UDP trong mô hình nhỏ, cần mở rộng để cho kết quả và kết luận tốt hơn. Trong thời gian tới, chúng tôi sẽ đo lường các thông số hiệu năng về nhiều loại gói tin (TCP, UDP) với mô hình lớn và nhiều trường hợp hơn để cho kết quả tin cậy cao.

TÀI LIỆU THAM KHẢO

- [1] Borko Furht, Armando Escalante, *Handbook of Cloud Computing*, Springer, (2010).
- [2] P., M. and T. Grance., The NIST Definition of Cloud Computing, (2011).
- [3] Sabahi, F. Cloud computing security threats and responses in Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on. (2011).
- [4] Vo Nhan Van, Le Minh Chi, Nguyen Quoc Long, Nguyen Gia Nhu, and Dac-Nhuong Le, A Performance Analysis of OpenStack Open-source Solution for IaaS Cloud Computing, in proceeding of International Conference on Computer and Communication Technologies (IC3T 2015), Advances in Intelligent Systems and Computing Vol.380, pp.141-150 Springer. (2015)
- [5] https://wiki.openstack.org/wiki/Main_Page
- [6] <http://commons.wikimedia.org/wiki/File:Openstack-conceptual-arch-folsom.jpg>

NETWORK VIRTUALIZATION PERFORMANCE ANALYSIS ON OPENSTACK CLOUD COMPUTING INFRASTRUCTURES

Vo Nhan Van^a, Le Minh Chi^b, Nguyen Quoc Long^a, Le Duc Nhung^{c*}

^aThe Faculty of Information Technology, Duytan University, Danang, Vietnam

^bDanang ICT Infrastructure Development Center, Danang, Vietnam

^cThe Faculty of Information Technology, Haiphong University, Haiphong, Vietnam

*Corresponding author: nhuongld@hus.edu.vn

Article history

Received: January 04th, 2016

Received in revised form: March 19th, 2016

Accepted: March 31st, 2016

Abstract

Cloud computing has become popular in IT technology because of its advantages that focus on flexible, scaling, resources and services which help customers easy to build their own on-demand IT system easily. Moreover, Cloud computing also can assist users to balance, share, and manage IT resources among customers to get better performance of compute and storage. OpenStack, a new open source cloud computing framework which was a built-in modular architecture and focused on IaaS. OpenStack also focuses on NaaS by using network virtualization technology with native opensource intergrated solution – openvswitch and supports other commercial solutions from vendors. OpenStack nowadays has been popularly in business. This paper does a research on network performance on OpenStack network module code named Neutron. The parameters in virtualization network of OpenStack related to network performance such as throughput, package loss, time and delay of data transmission are estimated through UDP protocol using IPERF benchmarking tool. Our research investigated the possible internal traffic flow patterns and evaluated network performance of each pattern on OpenStack cloud computing environment.

Keywords: Cloud computing; Network performance; Openstack.