Công ty An ninh mạng Viettel

Báo cáo về Linux Network

Sinh viên: Nguyễn Đan Trường

*MỤC LỤC*

[Chương 1. Các thao tác với card mạng 3](#_Toc153266912)

[1. Cấu hình network interface 3](#_Toc153266913)

[2. Các service quản lý network trên Linux 9](#_Toc153266914)

[Chương 2. DNS – Domain Name System 10](#_Toc153266915)

[Chương 3. Routing trên Linux 19](#_Toc153266916)

[Chương 4. NTP – Network Time Protocol 20](#_Toc153266917)

# Các thao tác với card mạng

## Cấu hình network interface

**Tắt bật interface**

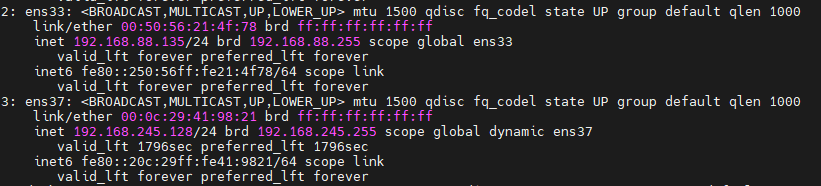
Sử dụng ifconfig



Sử dụng ip



**Các thao tác với địa chỉ IP**

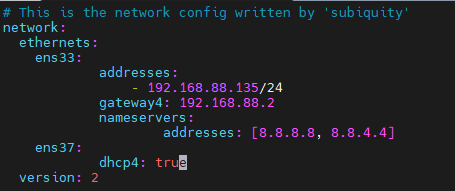
****

Có 2 cách cấu hình địa chỉ IP cho interface: ip static và ip dynamic

Nếu Ip được cấu hình theo kiểu dynamic, sử dụng câu lênh “ip a” sẽ thấy được interface ở chết độ dynamic. Nếu không thì interface được đặt ở chết độ static ip.

**Cấu hình địa chỉ ip static:**

Để cấu hình ip static cho interface ta chỉnh sửa file */etc/netplan/00-installer-config.yaml*



Với cấu hình này ta thấy interface ens33 được cấu hình IP static bao gồm các thông số:

* Ip address: 192.168.88.135
* Subnet mask: /24
* Ip gateway: 192.168.88.2
* DNS server: 8.8.8.8, 8.8.4.4

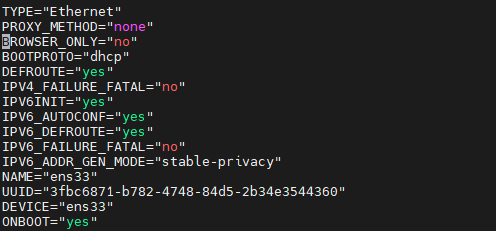
**Cấu hình địa chỉ ip dynamic:**

Để cấu hình địa chỉ dynamic ta khai báo thông số: “dhcp4: true”. Lúc này interface sẽ gửi ra bản tin DHCP để xin cấp địa chỉ IP.

**Cấu hình địa chỉ IP trên CentOS**

Cấu hình interface trên CentOS được đặt trong thư *mục /etc/sysconfig/network-scripts/*

Thực hiện cấu hình IP dynamic cho một interface:

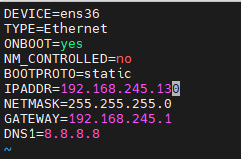


Đây là cấu hình IP dynamic với thông số:

* BOOTPROTO="dhcp": Xác định phương thức cấp địa chỉ IP. Ở đây, địa chỉ IP được cấp động tự động thông qua DHCP.
* ONBOOT="yes": Xác định xem interface có được kích hoạt khi khởi động hệ thống hay không. Ở đây, interface được kích hoạt khi khởi động (yes).

Cấu hình IP static cho interface:





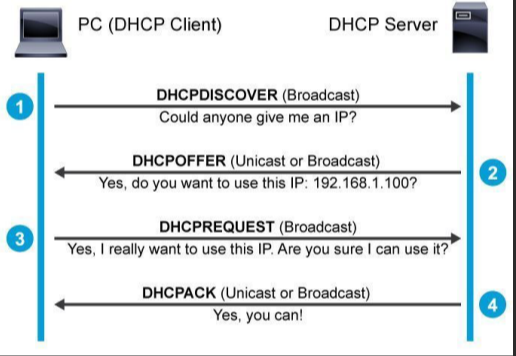


**Giao thức DHCP**

**Các thành phần của DHCP**

* **DHCP server:** Một thiết bị mạng chạy dịch vụ DHCP chứa địa chỉ IP và thông tin cấu hình liên quan.
* **DHCP client**: Thiết bị nhận thông tin cấu hình từ máy chủ DHCP.
* **IP address pool**: Dãy địa chỉ có sẵn cho client DHCP. Những địa chỉ này thường được truyền tuần tự từ thấp nhất đến cao nhất.
* **Subnet**: Mạng IP có thể được phân thành các phân đoạn được gọi là subnet (mạng con)
* **Lease**: Khoảng thời gian client DHCP giữ thông tin địa chỉ IP. Khi khoảng thời gian này hết hạn, client phải làm mới nó.
* **DHCP relay**: Router hoặc máy chủ nghe tin nhắn được phát trên mạng đó và sau đó chuyển chúng đến một máy chủ được cấu hình. Máy chủ này sau đó phản hồi lại relay agent để truyền chúng đến client.

**Luồng hoạt động DHCP**

****

Bước 1: Client gửi ra bản tin DHCP discovery (broadcast) để tìm kiếm DHCP server. Bản tin này có địa chỉ nguồn là 0.0.0.0 và địa chỉ đíc là 255.255.255.255

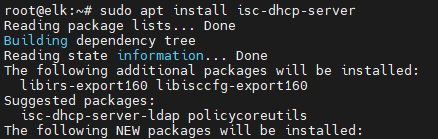
Bước 2: Server nhận được gói tin DHCP discovery (broadcast hoặc unicast) sẽ gửi lại một bản tin DHCP offer chứa IP được cấp phát và các thông tin cấu hình cho client.

Bước 3: client gửi ra bản tin DHCP request (Broadcast) yêu cầu xác nhận địa chỉ IP được cấp từ máy chủ DHCP

Bước 4: Server gửi ra bản tin DHCP ACK (unicast hoặc broadcast) xác nhận địa chỉ IP của client.

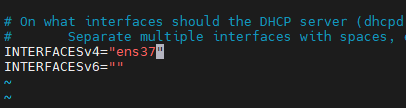
**Cấu hình DHCP**

Bước 1: cài đặt DHCP server trên ubuntu



Cấu hình xác định interface phục vụ yêu cầu DHCP:





Bước 2: Định cấu hình máy chủ DHCP

Tệp cấu hình DHCP chính là */etc/dhcp/dhcpd.conf*, bạn phải thêm tất cả thông tin mạng của mình để được gửi đến các máy khách tại đây.

Và, có hai loại câu lệnh được định nghĩa trong tệp cấu hình DHCP, đó là:

* tham số - chỉ định cách thực hiện một tác vụ, có thực hiện một tác vụ hay không hoặc những tùy chọn cấu hình mạng nào để gửi đến máy khách DHCP.
* khai báo - xác định cấu trúc liên kết mạng, nêu các máy khách, cung cấp địa chỉ cho máy khách hoặc áp dụng một nhóm tham số cho một nhóm khai báo.

*$ sudo vi /etc/dhcp/dhcpd.conf*

Đặt các tham số toàn cục sau ở đầu tệp, chúng sẽ áp dụng cho tất cả các khai báo bên dưới (chỉ định các giá trị áp dụng cho kịch bản của bạn):

*option domain-name "****tecmint.lan****";*

*option domain-name-servers* ***8.8.8.8****;*

*default-lease-time 3600;*

*max-lease-time 7200;*

*authoritative;*

Bây giờ, xác định một mạng con; tại đây, chúng tôi sẽ thiết lập DHCP cho mạng LAN 192.168.10.0/24 (sử dụng các tham số áp dụng cho trường hợp của bạn).

subnet 192.168.10.0 netmask 255.255.255.0 {

option routers 192.168.10.1;

option subnet-mask 255.255.255.0;

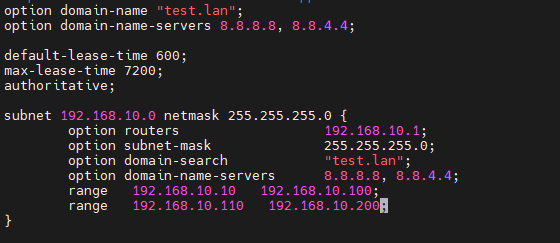
option domain-search "**tecmint.lan**";

option domain-name-servers 192.168.10.1;

range 192.168.10.10 192.168.10.100;

range 192.168.10.110 192.168.10.200;

}



Có thể gán IP cố định cho một thiết bị bằng cách:

host centos-node {

hardware ethernet 00:f0:m4:6y:89:0g;

fixed-address 192.168.10.105;

}

host fedora-node {

hardware ethernet 00:4g:8h:13:8h:3a;

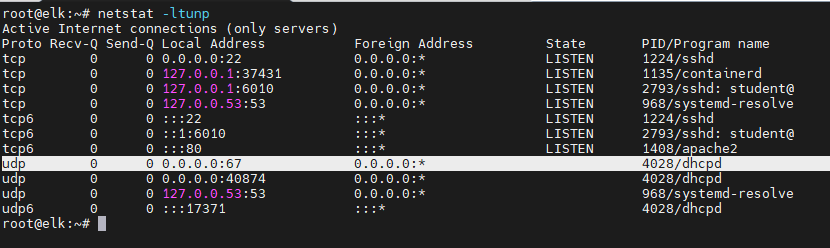
fixed-address 192.168.10.106;

}

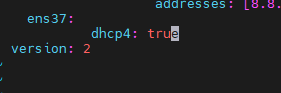
Lưu file và khởi chạy dịch vụ:

*$ sudo systemctl start isc-dhcp-server.service*

*$ sudo systemctl enable isc-dhcp-server.service*

**

Bước 3: Cấu hình trên Client



ta chỉ cấu hình cho cổng ens37 với thông số “dhcp4: true”

## Các service quản lý network trên Linux

**Trên CentOS:**

**NetworkManager:** là một dịch vụ quản lý mạng chung trên CentOS. Nó giúp quản lý kết nối mạng dễ dàng, bao gồm cả kết nối có dây và không dây.

Các lệnh quản lý NetworkManager:

*sudo systemctl status NetworkManager # Kiểm tra trạng thái*

*sudo systemctl start NetworkManager # Khởi động*

*sudo systemctl stop NetworkManager # Dừng*

Các tệp cấu hình:

/etc/NetworkManager/NetworkManager.conf

/etc/sysconfig/network-scripts/ifcfg-\* (cho các interface)

**Trên Ubuntu**

**Sytemd-networkd:**

Các lệnh quản lý trên systemd-networkd:

*sudo systemctl status systemd-networkd # Kiểm tra trạng thái*

*sudo systemctl start systemd-networkd # Khởi động*

*sudo systemctl stop systemd-networkd # Dừng*

Các tệp cấp hình:

/etc/systemd/network/\* (các tệp cấu hình của interface)

**Netplan**

Lệnh quản lý Netplan: *sudo netplan apply # Áp dụng cấu hình*

Tệp cấu hình: /etc/netplan/\*.yaml

**NetworkManager: (ubuntu 20.04)**

Các lệnh quản lý trên NetworkManager:

*sudo systemctl status NetworkManager # Kiểm tra trạng thái*

*sudo systemctl start NetworkManager # Khởi động*

*sudo systemctl stop NetworkManager # Dừng*

Các tệp cấu hình:

*/etc/NetworkManager/NetworkManager.conf*

*/etc/NetworkManager/system-connections/ (cấu hình của các kết nối)*

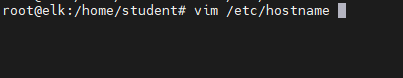
# DNS – Domain Name System

**Cách thiết lập, kiểm tra hostname**

**Ubuntu**

Để thay đổi hostname trên ubuntu ta thực hiện như sai:

Thay đổi hostname trong file /etc/hostname



Cật nhật file /etc/hosts, sửa lại hostname trong dòng 127.0.0.1





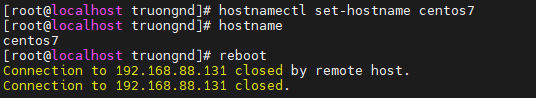
Khởi động lại máy và đăng nhập lại:



Ngoài ta có thể sử dụng lệnh: hostnamectl set-hostname ubuntu1

**CentOS**

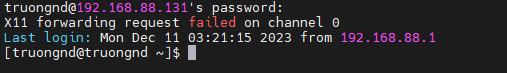
Trên CentOS cũng có thể sử dụng công cụ hostnamectl để đổi hostname:



Chỉnh sửa hostname thông qua file /etc/hostname:



Thực hiện thay đổi sau đó reboot:



**DNS – Domain Name System**

DNS (Domain Name System) là một hệ thống dùng để phân giải tên miền. Nó dùng để thiết lập mối quan hệ giữa tên tên miền và địa chỉ IP.

Cách thức hoạt động của DNS:

* Khi người dùng truy cập một website, máy tính sẽ gửi yêu cầu đến máy chủ DNS cục bộ để tìm địa chỉ IP của website đó. Máy chủ DNS cục bộ sẽ kiểm tra cơ sở dữ liệu của mình xem có chứa địa chỉ IP của website hay không. Nếu có, sẽ trả về địa chỉ IP cho máy tính của người dùng.
* Nếu máy chủ DNS cục bộ không có địa chỉ IP của website, nó sẽ hỏi máy chủ DNS gốc. Máy chủ DNS gốc sẽ trả về địa chỉ IP của máy chủ DNS cấp cao nhất cho website.
* Máy chủ DNS cấp cao nhất sẽ trả về địa chỉ IP của máy chủ DNS quản lý website. Máy chủ DNS quản lý sẽ trả về địa chỉ IP của trang web cho máy chủ DNS cục bộ.
* Cuối cùng, máy chủ DNS cục bộ sẽ trả về địa chỉ IP của trang web cho máy tính của người dùng. Máy tính của người dùng sẽ sử dụng địa chỉ IP này để kết nối với website.

Các loại bản ghi trong DNS thường sử dụng

* A Record: Ánh xạ một tên miền đến một địa chỉ Ipv4

Vd: example.com IN A 192.168.1.1

* AAA Record: Ánh xạ một tên miền đến một địa chỉ Ipv6

Vd: example.com IN AAAA 2001:0db8:85a3:0000:0000:8a2e:0370:7334

* CNAME Record: Tạo một tên miền con (alias) đến một tên miền khác

Vd: www IN CNAME example.com

* MX (Mail change) Record: Xác định máy chủ mail cho tên miền

Vd: example.com IN MX 10 mailserver.example.com

* PTR Record: thường được sử dụng cho quá trình reverse DNS để ánh xạ địa chỉ IP đến tên miền.

Vd: 1.1.168.192.in-addr.arpa IN PTR host.example.com

* SOA Record: Chứa thông tin quản lý tên miền và máy chủ DNS chịu trách nhiệm

Vd: example.com IN SOA ns1.example.com admin.example.com (

2022121101 ; Serial

3600 ; Refresh

1800 ; Retry

1209600 ; Expire

3600 ) ; Minimum TTL

* TXT Record: Chứa thông tin văn bản tùy chọn, thường được sử dụng cho việc xác minh tên miền và triển khai các chính sách bảo mật

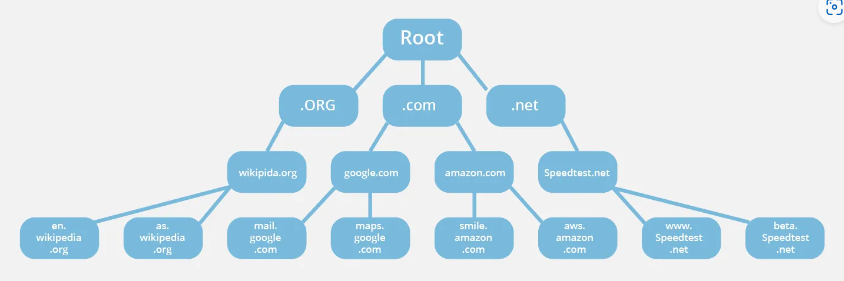
Vd: example.com IN TXT "v=spf1 include:\_spf.example.com ~all"

**Các loại DNS Server**

**Root Namer Server**

Root Name Server là một dịch vụ phân giải tên miền gốc và trên thế giới có khoảng 12 DNS root Server.

DNS root Server quản lý tất cả các tên miền Top-level. Khi có yêu cầu phân giải một Domain Name thành một địa chỉ IP, client sẽ gửi yêu cầu đến DNS gần nhất (DNS ISP). DNS ISP sẽ kết nối tới DNS root Server để hỏi địa chỉ của Domain Name.

****

**Local Name Server**

DNS Server này dùng để chứa thông tin để truy xuất và tìm kiếm máy chủ tên miền. Và thường được duy trì và phát triển bởi các doanh nghiệp hay các nhà cung cấp dịch vụ Internet.

**DBS Recursor**

DNS Recursor là một máy chủ có nhiệm vụ chuyển đổi tên miền thành địa chỉ IP. Được hoạt động như một thư viện, lưu trữ thông tin về địa chỉ IP của các trang web và ứng dụng. Khi người dùng nhập một tên miền, DNS Recursor sẽ tìm kiếm thông tin đó trong cơ sở dữ liệu của mình. Nếu không tìm thấy, nó sẽ liên hệ với các DNS Recursor khác để tìm kiếm. DNS Recursor cũng sử dụng bộ nhớ đệm để lưu trữ thông tin về các tên miền mà nó đã từng truy cập. Điều này giúp tăng tốc độ phản hồi cho các truy vấn trong tương lai.

**TLD Name Server**

TLD Name Server là máy chủ tên miền cấp cao nhất, chịu trách nhiệm lưu trữ thông tin về các tên miền có phần mở rộng chung (gTLD), chẳng hạn như .com, .org, .net. Trong quá trình tìm kiếm địa chỉ IP, máy chủ định danh sẽ liên hệ với TLD Name Server để lấy thông tin về tên miền.

Ví dụ, khi truy cập google.com, máy chủ định danh sẽ liên hệ với TLD Name Server để lấy thông tin về tên miền .com. Sau đó, TLD Name Server sẽ trả về địa chỉ IP của máy chủ tên miền cấp thấp hơn, là nơi lưu trữ thông tin về tên miền google.com

**Authoritative Name Server**

Authoritative Name Server lưu trữ thông tin về tên miền và địa chỉ IP tương ứng. Là điểm cuối của quá trình truy vấn và phân giải địa chỉ IP cần thiết cho DNS Recursor.

**Ví dụ dưới đây phác thảo tất cả 8 bước khi không có cache:**

**Bước 1:** Một người dùng nhập “example.com” vào trình duyệt web và truy vấn trên Internet. Và được nhận bởi DNS Recursive Resolver.

**Bước 2:** Resolver sau đó truy vấn một root nameserver DNS (.).

**Bước 3:** Sau đó, Root Nameserver phản hồi resolver bằng địa chỉ của máy chủ DNS tên miền cấp cao (TLD) (chẳng hạn như .com hoặc .net), nơi lưu trữ thông tin cho các tên miền của nó. Khi tìm kiếm example.com, yêu cầu ban đầu là hướng tới TLD.com.

**Bước 4:** Resovler sau đó thực hiện một yêu cầu tới TLD.com.

**Bước 5:** Sau đó, máy chủ TLD phản hồi với địa chỉ IP nameserver của domain example.com.

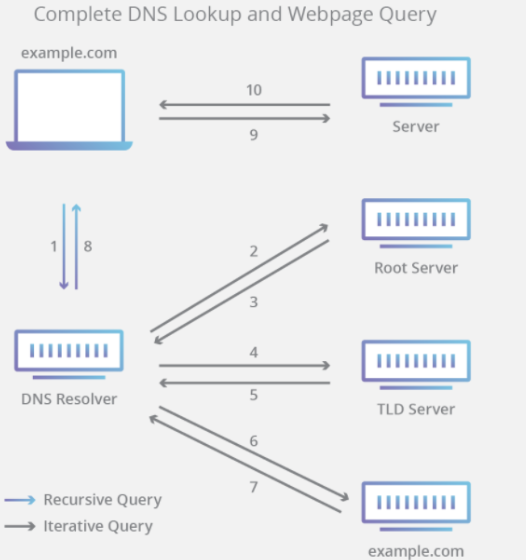
**Bước 6:** Cuối cùng, recursive resolver gửi một truy vấn đến nameserver của tên miền.

**Bước 7:** Địa chỉ IP cho example.com sau đó được trả về từ nameserver.

**Bước 8:** DNS Resovler sau đó trả lời trình duyệt web bằng địa chỉ IP của tên miền được yêu cầu ban đầu.

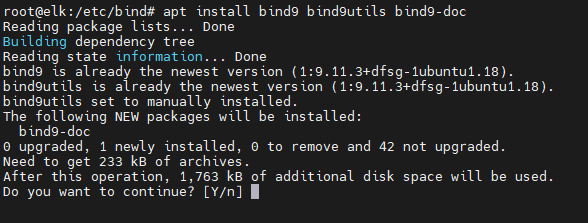
**Bước 9:**Khi 8 bước tra cứu DNS đã trả về địa chỉ IP cho example.com. Trình duyệt có thể đưa ra yêu cầu cho trang web. Trình duyệt tạo một yêu cầu HTTP đến địa chỉ IP.

**Bước 10:** Máy chủ tại IP đó trả về trang web sẽ được hiển thị trong trình duyệt (bước 10).

****

.

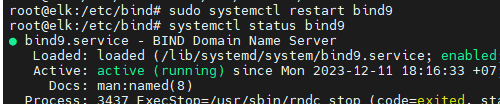
**Cấu hình DNS Server**



Hãy thiết lập Bind sang chế độ Ipv4. Trên server chỉnh sửa tệp cài đặt mặc định named: vim /etc/default/named

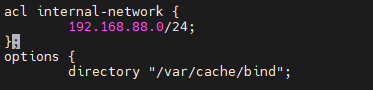


Khởi động lại bind9 service:

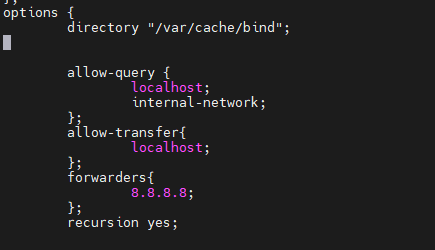


**Hiệu chỉnh DNS server**

Cấu hình của BIND bao gồm nhiều tệp như tệp cấu hình chính và named.conf. Tên các tệp này bắt đầu bằng named.conf.options vì đó là tên của tiến trình mà BIND chạy: vim /etc/bind/named.conf.options



Bên dưới directory, thêm các dòng cấu hình sau:



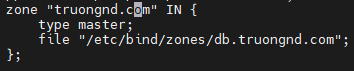
Lưu ý răng khối forwarder sẽ bao gồm địa chỉ: 8.8.8.8, 8.8.4.4. Khối này xác định forwarder, một cơ chế đặc biệ mà BIND sử dụng để giảm lưu lượng liên kết tới các server bên ngoài. Hai địa chỉ IP trong khối này đại diện cho Public DNS Server của Google, nhưng địa chỉ IP của bất kỳ Public DNS Server nào khác cũng có thể hoạt động ở đây. Ví dụ, bạn có thể sử dụng địa chỉ IP của server DNS của Cloudflare (1.1.1.1) để thay thế.

Sau khi hoàn thành cấu hình tệp named.conf.options. Lúc này những server trusted mới cớ thể truy vấn tới DNS server.

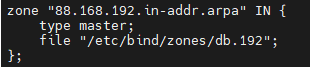
**Cấu hình tệp local**

Vim /etc/bind/named.conf.local

Ở đây bạn sẽ chỉ định các forward zones và reverse zones của mình. Thêm forward zone với các dòng sau:



Hãy thêm reverse zone bằng các dòng sau (lưu ý rằng tên reverse zone của bạn bắt đầu bằng 188.168 là sự đảo ngược của octet 168.88):

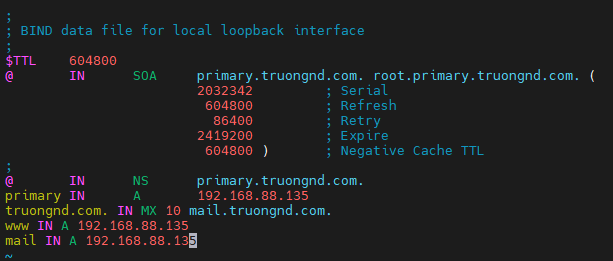


**Khởi tạo Forward Zone**

*sudo mkdir /etc/bind/zones*

*sudo cp /etc/bind/db.local /etc/bind/zones/db.nyc3.example.com*

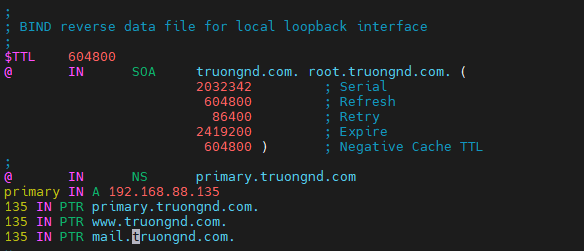
*sudo vim/etc/bind/zones/db.truongnd.com*

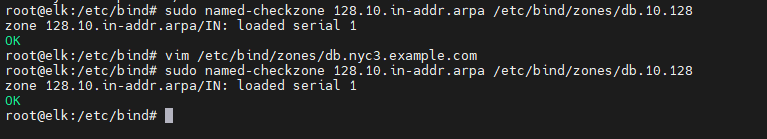
**

**Khởi tạo reverse zone file**

*sudo cp /etc/bind/db.127 /etc/bind/zones/db.192*

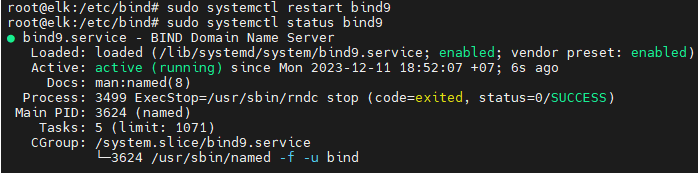
*sudo vim/etc/bind/zones/db.192*

**

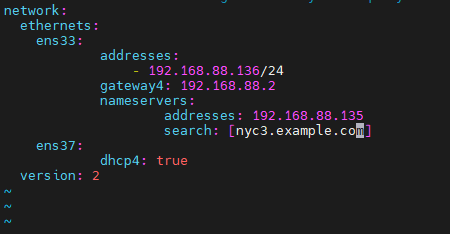
**

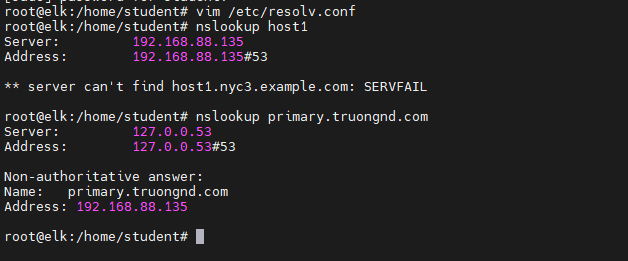
**Khởi động lại Bind**

*sudo systemctl restart bind9*

****

**Hiệu chỉnh DNS Client**

****

****

# Routing trên Linux

Routing (định tuyến) là một khái niệm quan trọng trong mạng máy tính và Internet. Nó đề cập đến quá trình xác định cách thông tin (dữ liệu) di chuyển từ một điểm đến một điểm khác trong mạng. Dưới đây là một số lý do quan trọng tại sao cần routing:

* Routing là cầu nối giữa các mạng khác nhau.
* Khi có nhiều đường đi giữa 2 điểm, routing có thể giúp chia đều tải lưu lượng trên các đường đi
* Tăng cường bảo mật, quản trị viên có thể kiểm soát được cách dữ liệu truyền tải trên mạng có đúng đường hay không.

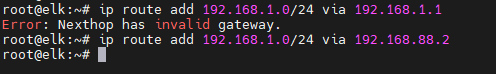
**Các cấu hình routing trên Linux**

Thêm một route:

*sudo ip route add [destination\_network] via [gateway\_ip]*

*sudo route add -net [destination\_network] gw [gateway\_ip]*

*sudo ip route add default via [gateway\_ip]*

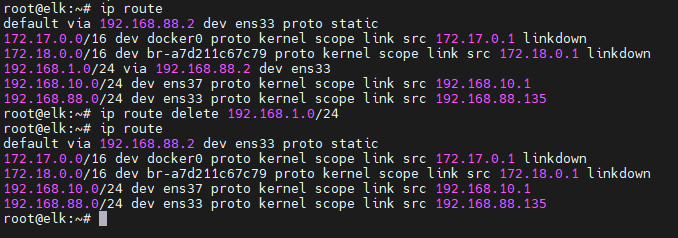
**

Xóa một route:

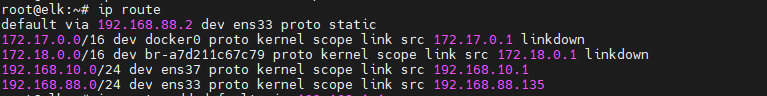
*sudo ip route delete [destination\_network]*

*sudo route del -net [destination\_network]*

*sudo ip route del default*

**

**Bảng định tuyến**

****

Các trường giải thích:

**via**: Địa chỉ IP của gateway hoặc router.

**dev**: Thiết bị mạng mà gói tin sẽ được chuyển qua.

**proto**: Giao thức sử dụng để định tuyến, ở đây là static, kernel (tự động tạo bởi kernel), và scope link (định tuyến trên cùng một giao diện).

**scope**: Phạm vi của route, ở đây là link (cùng một mạng con).

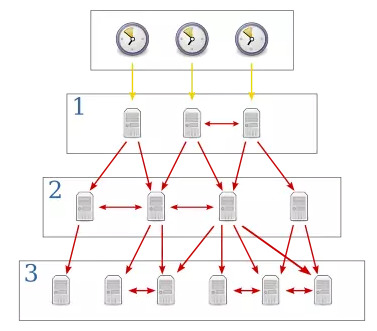
**src**: Địa chỉ IP của thiết bị nguồn (source).

# NTP – Network Time Protocol

Giao thức NTP (Network Time Protocol) là một giao thức để đồng bộ đồng hồ của các hệ thống máy tính thông qua mạng dữ liệu chuyển mạch gói với độ trễ biến đổi.

Để một hệ thống hoạt động và có thể phối hợp được với nhau điều kiện tiên quyết đầu tiên là giữa các hệ thống đó phải đồng bộ được về mặt thời gian. Mục tiêu chính của NTP là đảm bảo rằng các máy tính và thiết bị trong mạng đều giữ thời gian chính xác và đồng bộ.

**Cách NTP hoạt động**

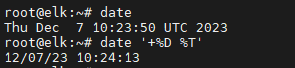


* Server NTP: Các máy chủ NTP là những thiết bị chính xác thời gian, thường được kết nối với các nguồn thời gian chính xác như atomic clocks hoặc máy chủ thời gian quốc gia. Những máy chủ này được coi là đáng tin cậy và chính xác.
* Client NTP: Các máy tính và thiết bị khác trong mạng là các máy khách NTP. Chúng sử dụng giao thức NTP để liên lạc với máy chủ NTP và điều chỉnh thời gian của chúng theo thời gian được cung cấp bởi máy chủ.
* Giao thức NTP: sử dụng mô hình đồng bộ động (dynamic synchronization) để điều chỉnh thời gian. Mỗi máy khách NTP đều cố gắng điều chỉnh thời gian của mình để đồng bộ với máy chủ. Giao thức này cũng chứa các thuật toán và cơ chế để giảm ảnh hưởng của độ trễ mạng và đảm bảo độ chính xác cao.
* Stratum levels: Hệ thống NTP sử dụng các cấp độ "stratum" để xác định độ chính xác của các máy chủ NTP. Stratum 0 đại diện cho các nguồn thời gian chính xác nhất (như atomic clocks), còn stratum 1 là các máy chủ NTP trực tiếp kết nối với stratum 0. Stratum 2 là các máy chủ NTP kết nối với stratum 1,…

**Kiến trúc cơ bản**

* Nguồn thời gian chính xác (Stratum 0): Atomic clocks, GPS receivers, các máy chủ thời gian quốc gia.
* Máy chủ NTP (Stratum 1): Kết nối với nguồn thời gian chính xác và cung cấp thời gian cho các máy chủ NTP cấp thấp hơn.
* Máy chủ NTP (Stratum 2, Stratum 3, ...): Kết nối với máy chủ NTP ở cấp trên và cung cấp thời gian cho các máy khách NTP trong mạng.

**Các câu lệnh NTP**



Có thể đổi thời gian của hệ thống bằng lệnh: *sudo date MMDDhhmm[[CC]YY][.ss]*

MM: Tháng.

DD: Ngày.

hh: Giờ.

mm: Phút.

CC: Thế kỷ (nếu không có, mặc định là 19).

YY: Năm (00-99).

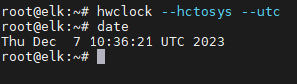
.ss: Giây.



Hwclock là câu lệnh dùng để kiểm soát đồng hồ phần cứng do BIOS quản lý. Khi dùng date để thay đổi giờ thì đồng hồ phần cứng sẽ không bị ảnh hưởng, do đó tại lần khởi động kết tiếp hệ thống sẽ lấy thông tin thời gian từ đồng hồ phần cứng.

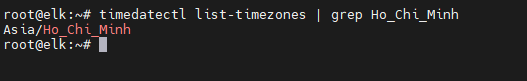


Đồng bộ thời gian từ hwclock vào system time:

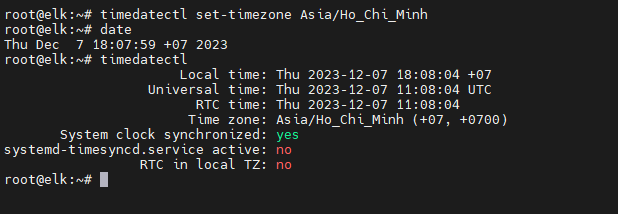


**Kiểm tra và cài đặt NTP**

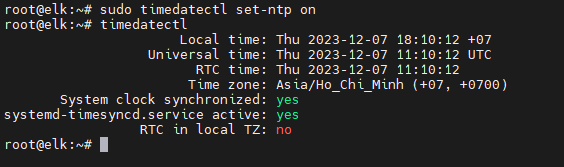
Liệt kê time-zone đang có trên hệ thống



Bạn có thể set time-zone lại bằng lệnh



systemd-timesyncd.service active: no – tương đương với việc đồng bộ thời gian đang không chạy, ta cần bật lại lên bằng lệnh:

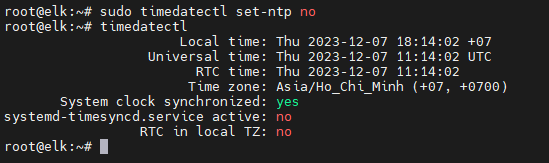


**Sử dụng ntpd**

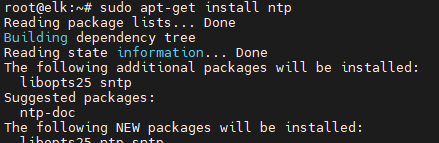
Timesyncd sẽ hoạt động trong hầu hết trường hợp. Tuy nhiên, có những trường khi một ứng dụng nhay cảm với bất kì sự xáo trộn theo thời gian. Trong trường hợp này, ntpd là dịch vụ thời gian thay thế có thể sử dụng.

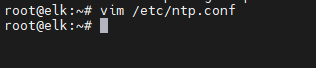
Ntpd sử dụng các kỹ thuật phức tạp để liên tục và dần dần giữ cho thời gian của hệ thống đi đúng hướng.

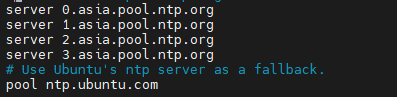
Trước khi cài đặt ntpd, bạn cần tắt timesyncd để tránh hai dịch vụ xung đột với nhau. Bạn có thể thực hiện việc này bằng cách tắt đồng bộ hóa thời gian mạng bằng lệnh sau:



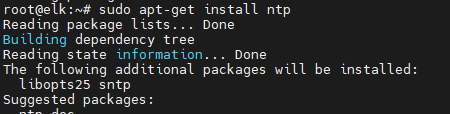
Cài đặt ntpd cho server







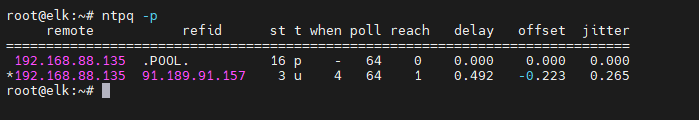
Thực hiện cài đặt NTP cho client







Ntpq là công cụ truy vấn cho ntpd. Cờ -p yêu cầu thông tin về máy chủ NTP ntpd được kết nối tới.



**remote:** Tên hoặc địa chỉ IP của máy chủ NTP từ nguồn dữ liệu.

**refid:** Một số duy nhất hoặc địa chỉ IP của máy chủ NTP mà máy chủ hiện tại đang sử dụng làm nguồn thời gian.

**st:** Stratum level, cấp độ stratum của máy chủ NTP. Stratum 0 là các nguồn thời gian chính xác nhất, như atomic clocks, Stratum 1 là máy chủ được đồng bộ trực tiếp với Stratum 0, và tiếp tục như vậy.

**t:** Loại máy chủ: 'u' cho máy chủ đang hoạt động bình thường.

**when:** Thời gian tính bằng giây kể từ lần cuối cùng máy chủ được truy vấn.

**poll:** Chu kỳ truy vấn, tức là khoảng thời gian giữa các lần truy vấn máy chủ NTP.

**reach:** Trạng thái kết nối, số này là một biểu diễn nhị phân của 8 bit, trong đó mỗi bit đại diện cho một lần thành công (1) hoặc thất bại (0) trong việc kết nối gần đây.

**delay:** Thời gian mà một gói tin mất để đi từ máy chủ NTP đến máy chủ hiện tại, tính bằng mili giây.

**offset:** Độ chênh lệch thời gian giữa máy chủ NTP và máy chủ hiện tại, tính bằng mili giây.

**jitter:** Độ biến động của độ chênh lệch thời gian giữa các gói tin NTP, tính bằng mili giây.