

BÀI THỰC HÀNH SỐ 3

CÁC HỆ THỐNG PHÂN TÁN VÀ ỨNG DỤNG

CHƯƠNG 3: ĐỊNH DANH TRONG HỆ PHÂN TÁN

Họ và tên: Nguyễn Ngọc Phúc

Mã lớp: 157542

MSSV: 20220041

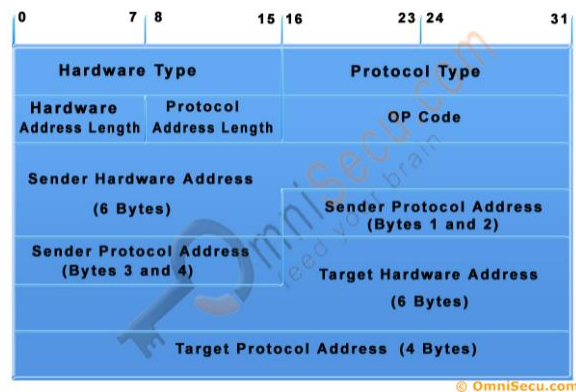
Mã học phần: IT4611

1. Giao thức ARP:

Đầu tiên, chúng ta hãy quan sát hình sau để hiểu về cấu trúc của một thông điệp ARP.

Câu hỏi 1: Giải thích ý nghĩa các trường trong thông điệp ARP trên.

Thông điệp ARP (Address Resolution Protocol) là một giao thức mạng được sử dụng để ánh xạ địa chỉ IP và địa chỉ MAC của máy tính trong cùng một mạng LAN



Ý nghĩa của các trường trong thông điệp ARP là:

- **Hardware Type (2 bytes):** Xác định loại phần cứng mạng của máy gửi cần biết với giá trị 1 cho Ethernets
- **Protocol Type (2 bytes):** Xác định loại giao thức mạng (VD 0x0800 : IPv4)
- **Hardware Address Length (1 byte):** Độ dài địa chỉ phần cứng (MAC) vật lý
- **Protocol Address Length (1 byte):** Độ dài tính bằng byte của địa chỉ logic.
- **OP Code (2 byte):** Xác định loại gói ARP: 1 = ARP Request, 2 = ARP Reply.
- **Sender Hardware Address (6 bytes):** Địa chỉ MAC của máy gửi yêu cầu ARP (máy nguồn)
- **Sender Protocol Address (4 bytes):** Địa chỉ IP của máy gửi yêu cầu ARP.
- **Target Hardware Address (6 bytes):** Địa chỉ vật lý máy đích
- **Target Protocol Address (4 bytes):** Địa chỉ IP của máy nhận, là IP mà người gửi muốn tìm địa chỉ MAC tương ứng.

Hãy kết nối một số máy trong một mạng LAN và kiểm tra xem chúng có được liên kết với nhau chưa.

Cài đặt Wireshark trên mỗi máy: <https://www.wireshark.org/#download>

Chạy lệnh sau để xem thông tin bảng ARP ở máy A:

```
>arp -a
```

```
C:\Windows\System32>arp -a

Interface: 192.168.56.1 --- 0x8
    Internet Address      Physical Address      Type
    192.168.56.255        ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static

Interface: 192.168.1.3 --- 0x13
    Internet Address      Physical Address      Type
    192.168.1.1           cc-71-90-35-f3-e0    dynamic
    192.168.1.255         ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static
```

Hãy chắc chắn là không có bản ghi nào trong ARP table. Nếu có, hãy chạy lệnh sau để xóa hết các bản ghi đó:

```
>arp -d -a
```

```
C:\Windows\System32>arp -d -a

Interface: 192.168.56.1 --- 0x8
    Internet Address      Physical Address      Type
    192.168.56.255        ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static

Interface: 192.168.1.3 --- 0x13
    Internet Address      Physical Address      Type
    192.168.1.1           cc-71-90-35-f3-e0    dynamic
    192.168.1.255         ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static
```

Chạy Wireshark trên tất cả các máy trong mạng của bạn. Bây giờ hãy thực hiện một lệnh ping đến IP máy B:

```
>ping IP_of_machine_B
```

Chúng ta biết rằng lệnh ping hoạt động bằng cách sử dụng ICMP. Thông điệp ICMP được gói gọn trong IP datagram và IP datagram được gói gọn trong Ethernet Frame. Chúng ta cần Địa chỉ IP nguồn (Địa chỉ IP của máy A), địa chỉ IP đích (IP của máy B), địa chỉ MAC nguồn (địa chỉ MAC của máy A) và địa chỉ MAC đích để tạo Ethernet frame cho thông báo ICMP. Địa chỉ IP nguồn, địa chỉ IP đích, Địa chỉ MAC nguồn là những thông tin đã biết trong trường hợp này, nhưng địa chỉ MAC đích không xác định trong trường hợp này. Quan sát cửa sổ Wireshark (máy B, hoặc các máy khác máy A), ấn chọn vào thông điệp ARP request và phân tích nó ở khung cửa sổ phía dưới.

Câu hỏi 2: Hãy cho biết các thông tin sau trong cửa sổ bạn đang quan sát:

```
> Frame 36458: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{FF928B9E-B826-4DC3-A092-57E5BB1C70}
> Ethernet II, Src: LiteonTechno_af:a4:43 (e0:0a:f6:af:a4:43), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
▼ Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: LiteonTechno_af:a4:43 (e0:0a:f6:af:a4:43)
  Sender IP address: 192.168.1.3
  Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
  Target IP address: 192.168.1.6
```

- Destination MAC address: FF:FF:FF:FF:FF:FF
Đây là một địa chỉ Broadcast, gói tin được gửi đến tất cả thiết bị trong mạng LAN.
- Opcode: request (1)
Đây là một gói ARP Request, là máy gửi đang yêu cầu địa chỉ MAC tương ứng với một địa chỉ IP cụ thể
- Target MAC address: 00:00:00:00:00:00
Do máy gửi chưa biết địa chỉ MAC của máy đích.

Câu hỏi 3: Hãy cho biết các thông tin sau trong cửa sổ bạn đang quan sát:

```
> Frame 36459: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface \Device\NPF_{FF928B9E-B826-4DC3-A092-57E5BB1C70}
> Ethernet II, Src: ZhejiangDahu_57:e1:e3 (c0:39:5a:57:e1:e3), Dst: LiteonTechno_af:a4:43 (e0:0a:f6:af:a4:43)
▼ Address Resolution Protocol (reply)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: reply (2)
  Sender MAC address: ZhejiangDahu_57:e1:e3 (c0:39:5a:57:e1:e3)
  Sender IP address: 192.168.1.6
  Target MAC address: LiteonTechno_af:a4:43 (e0:0a:f6:af:a4:43)
  Target IP address: 192.168.1.3
```

Opcode: reply (2)
Sender MAC address: c0:39:5a:57:e1:e3
Sender IP address: 192.168.1.6
Target MAC address: e0:0a:f6:af:a4:43
Target IP address: 192.168.1.3

Bây giờ hãy kiểm tra những thông tin đã được ghi lại ở bảng ARP của máy A:

```
>arp -a
```

```
C:\Windows\System32>arp -a

Interface: 192.168.56.1 --- 0x8
    Internet Address      Physical Address      Type
    192.168.56.255        ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static

Interface: 192.168.1.3 --- 0x13
    Internet Address      Physical Address      Type
    192.168.1.1           cc-71-90-35-f3-e0    dynamic
    192.168.1.6           c0-39-5a-57-e1-e3    dynamic
    192.168.1.255         ff-ff-ff-ff-ff-ff    static
    224.0.0.22            01-00-5e-00-00-16    static
```

Câu hỏi 4: Bạn quan sát được gì và rút ra được kết luận gì?

Khi máy 192.168.1.3 cần giao tiếp với IP 192.168.1.6 mà chưa biết địa chỉ MAC tương ứng, nó sẽ gửi một gói ARP Request để hỏi. Sau khi nhận được ARP Reply từ 192.168.1.6 chứa địa chỉ MAC c0-39-5a-57-e1-e3, máy 192.168.1.3 sẽ lưu ánh xạ này vào bảng ARP dưới dạng dynamic. Sau quá trình này, 2 máy đều biết được địa chỉ MAC của nhau. Điều này cho thấy giao thức ARP hoạt động chính xác, giúp máy tính phân giải địa chỉ IP sang địa chỉ MAC để giao tiếp trong mạng LAN.

2. Tự cài đặt máy chủ DNS:

Cài đặt BIND ở cả 2 máy chủ ns1 và ns2

Ở cả 2 máy chủ ns1 và ns2, hãy thực hiện cập nhật bằng lệnh sau:

```
$sudo apt-get update
```

Tiến hành cài đặt BIND:

```
$sudo apt-get install bind9 bind9utils bind9-doc
```

Trước khi tiếp tục, hãy cùng đặt BIND ở chế độ chạy ở IPv4 vì mạng của chúng ta sử dụng chủ yếu IPv4. Ở cả 2 máy chủ, hãy thay đổi nội dung OPTIONS trong file cài đặt mặc định của bind9 là file: `/etc/default/bind9`

```
OPTIONS="-u bind -4"
```

Khởi động lại BIND để cho các thay đổi có hiệu lực

```
$sudo systemctl restart bind9
```

Cấu hình máy chủ DNS ns1

Ở máy chủ ns1, mở tệp sau để chỉnh sửa: `/etc/bind/named.conf.options` Đầu tiên, hãy thêm block `acl` (access control list) vào. Block này được thêm vào trước block `options`. Block này dùng để lên danh sách các clients mà bạn sẽ cho phép gửi yêu cầu DNS lên server. Ở bài thực hành này, chúng ta sẽ thêm vào `ns1`, `ns2`, `host1`, và `host2` vào danh sách đó

Câu hỏi 5: Vai trò của block `forwarders` trong block `options` là gì?

- Block forwarders trong block options trong tập tin cấu hình của BIND server là để xác định danh sách các máy chủ DNS khác để chuyển tiếp các truy vấn DNS mà không tìm thấy trong file zone cục bộ của máy chủ DNS.
- Khi máy chủ DNS nhận được một truy vấn mà nó không thể giải quyết được bằng cách sử dụng các khu vực DNS mà nó quản lý, nó sẽ liên hệ với các máy chủ DNS được liệt kê trong phần **forwarders** để giải quyết truy vấn đó. Nếu không có bất kỳ máy chủ DNS nào được cấu hình trong phần **forwarders**, máy chủ DNS sẽ tìm kiếm các máy chủ DNS gốc để giải quyết truy vấn.

Khởi forwarders trong phần options của cấu hình BIND9 đóng vai trò quan trọng trong việc định tuyến các truy vấn DNS mà máy chủ không thể giải quyết trực tiếp.

Câu hỏi 6: Giải thích yêu cầu tìm kiếm *forward* và *reverse* trong DNS là gì?

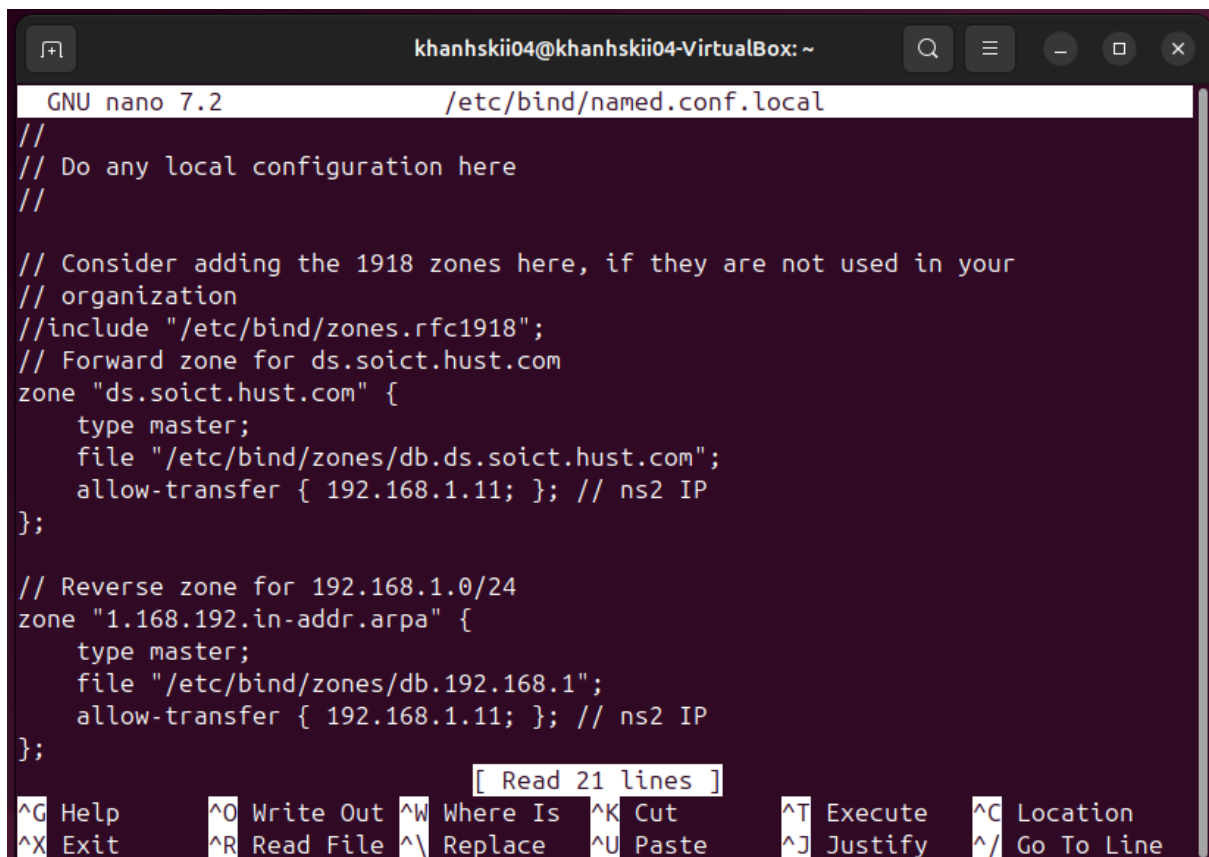
- Yêu cầu tìm kiếm forward trong DNS là quá trình dịch tên miền sang địa chỉ IP. Khi nhập tên miền vào trình duyệt web, trình duyệt sẽ gửi một yêu cầu đến máy chủ DNS để giải quyết địa chỉ IP của tên miền đó. Khi máy chủ DNS giải quyết thành công, nó sẽ trả lại địa chỉ IP tương ứng cho trình duyệt web để tiếp tục kết nối đến tên miền đó.
- Reverse lookup là quá trình chuyển đổi địa chỉ IP thành tên miền (hostname) tương ứng. Reverse lookup sử dụng các bản ghi PTR trong vùng tra cứu ngược (reverse lookup zone) để thực hiện quá trình này.

Mở tệp `/etc/bind/named.conf.local` và thêm vào zone để forward như sau:

```
zone "ds.soict.hust.com"
{
    type master;
    file "/etc/bind/zones/db.ds.soict.hust.com"; # zone file path
    allow-transfer { 192.168.1.21; }; # ns2 IP
};
```

Coi như là địa chỉ mạng con là 192.168.1.0/24, thêm vào zone để reverse như sau:

```
zone "1.168.192.in-addr.arpa"
{
    type master;
    file "/etc/bind/zones/db.192.168.1"; # 192.168.1.0/24 subnet
    allow-transfer { 192.168.1.21; }; # ns2 IP
}
```



```
GNU nano 7.2 /etc/bind/named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
// Forward zone for ds.soict.hust.com
zone "ds.soict.hust.com" {
    type master;
    file "/etc/bind/zones/db.ds.soict.hust.com";
    allow-transfer { 192.168.1.11; }; // ns2 IP
};

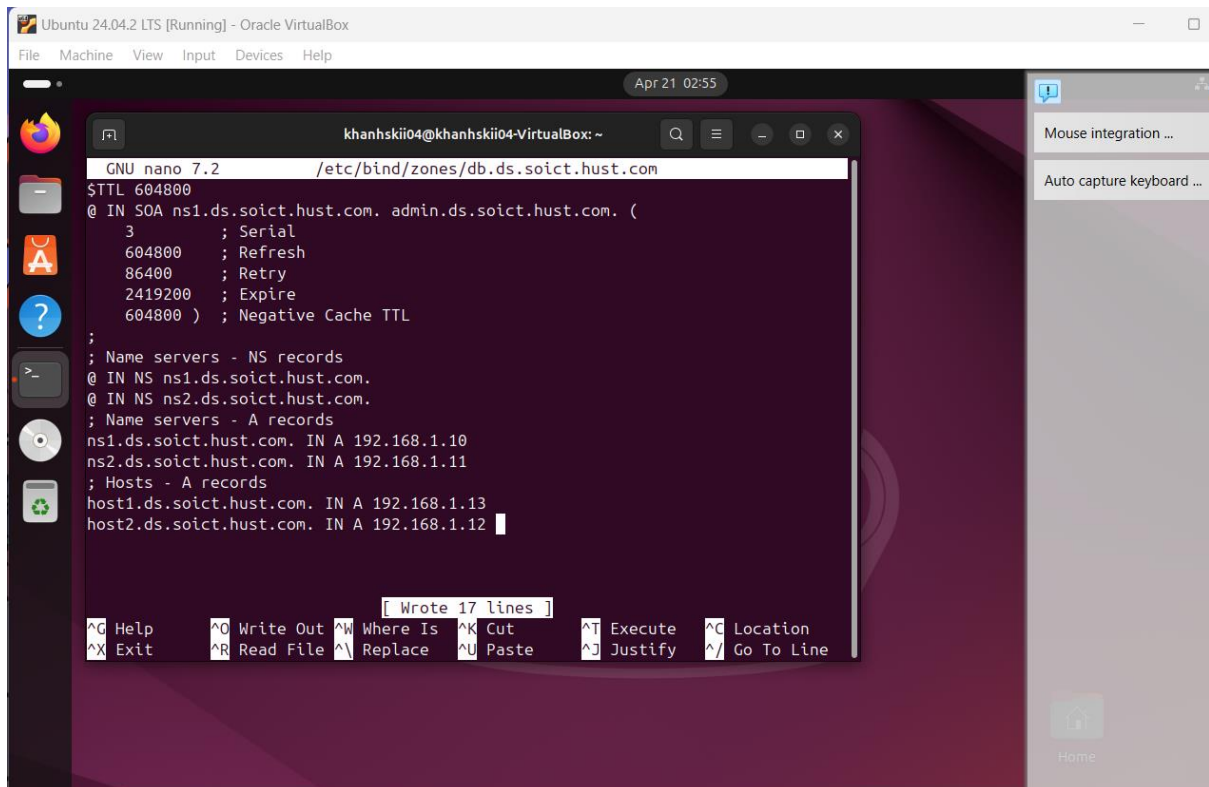
// Reverse zone for 192.168.1.0/24
zone "1.168.192.in-addr.arpa" {
    type master;
    file "/etc/bind/zones/db.192.168.1";
    allow-transfer { 192.168.1.11; }; // ns2 IP
};

[ Read 21 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Câu hỏi 7: 2 tệp *db.ds.soict.hust.com* và *db.192.168.1* dùng để làm gì?

Hai tệp *db.ds.soict.hust.com* và *db.192.168.1* đều là các tệp zone files được sử dụng bởi máy chủ DNS để định nghĩa các bản ghi DNS cho các zone forward và reverse tương ứng.

- Tệp *db.ds.soict.hust.com* chứa các bản ghi DNS cho zone forward *ds.soict.hust.com*, được sử dụng để xác định các máy chủ và dịch vụ cho tên miền *ds.soict.hust.com*
- Tệp *db.192.168.1* chứa các bản ghi DNS cho zone reverse *1.168.192.in-addr.arpa*, để ánh xạ địa chỉ IP của các máy tính trong dải 192.168.1.0/24 về tên miền tương ứng.



Câu hỏi 8: Hãy giải thích 3 kiểu bản ghi của DNS: SOA, NS và A

- SOA (Start of Authority): Đây là bản ghi xác định máy chủ DNS chính chịu trách nhiệm quản lý 1 zone cụ thể. Bản ghi SOA định nghĩa các thông tin về tên miền, bao gồm tên miền chính, email của quản trị viên, số serial của zone, và các tham số về thời gian như refresh, retry, expire và TTL. Nó giúp các máy chủ phụ biết khi nào cần cập nhật thông tin từ máy chủ chính
- NS (Name Server): Record tiếp theo cần có trong zone là NS (name server) record. Mỗi name server cho zone sẽ có một NS record. Chứa địa chỉ IP của DNS Server cùng với các thông tin về domain đó. Khi một truy vấn DNS được thực hiện, NS record giúp xác định máy chủ nào sẽ trả lời truy vấn đó
- A (Address): Là một record căn bản và quan trọng, dùng để ánh xạ từ một domain thành địa chỉ IP cho phép có thể truy cập website. Khi người dùng nhập một tên miền vào trình duyệt, A record giúp xác định địa chỉ IP của máy chủ để kết nối.

```

khanhskii04@khanhskii04-VirtualBox:~$ sudo named-checkzone ds.soict.hust.com /etc/bind/zones/db.ds.soict.hust.com
zone ds.soict.hust.com/IN: loaded serial 3
OK
khanhskii04@khanhskii04-VirtualBox:~$ sudo named-checkzone 1.168.192.in-addr.arpa /etc/bind/zones/db.192.168.1
zone 1.168.192.in-addr.arpa/IN: loaded serial 3
OK

```

Câu hỏi 9: Lệnh trên sẽ đưa ra kết quả gì? Giải thích!

Kết quả thu được cho biết file zone đã được tải thành công và số serial là 3, serial number

xác định giá trị phiên bản của file zone. OK cho thấy file zone hợp lệ, không có lỗi cú pháp hoặc định dạng. Điều này xác nhận rằng zone file đã được cấu hình chính xác và có thể được sử dụng bởi BIND để phục vụ truy vấn DNS.

```
khanhskii04@khanhskii04-VirtualBox: ~  
^C  
khanhskii04@khanhskii04-VirtualBox:~$ sudo named-checkzone ds.soict.hust.com /etc/bind/zones/db.ds.soict.hust.com  
zone ds.soict.hust.com/IN: loaded serial 3  
OK  
khanhskii04@khanhskii04-VirtualBox:~$ sudo named-checkzone 1.168.192.in-addr.arpa /etc/bind/zones/db.192.168.1  
zone 1.168.192.in-addr.arpa/IN: loaded serial 3  
OK  
khanhskii04@khanhskii04-VirtualBox:~$ sudo systemctl restart bind9  
khanhskii04@khanhskii04-VirtualBox:~$ sudo ufw allow bind9  
Rules updated  
Rules updated (v6)  
khanhskii04@khanhskii04-VirtualBox:~$ sudo systemctl status bind9  
● named.service - BIND Domain Name Server  
   Loaded: loaded (/usr/lib/systemd/system/named.service; enabled; preset: en>  
   Active: active (running) since Mon 2025-04-21 03:13:00 +07; 7min ago  
     Docs: man:named(8)  
  Main PID: 4578 (named)  
    Status: "running"  
    Tasks: 5 (limit: 2271)  
  Memory: 5.2M (peak: 5.5M)  
     CPU: 138ms  
   CGroup: /system.slice/named.service  
           └─4578 /usr/sbin/named -f -u bind
```

Câu hỏi 10: Bạn dùng lệnh nào để chắc chắn là *bind9* đang chạy?

Để kiểm tra trạng thái hoạt động của dịch vụ *bind9*, sử dụng câu lệnh:

```
sudo systemctl status bind9
```

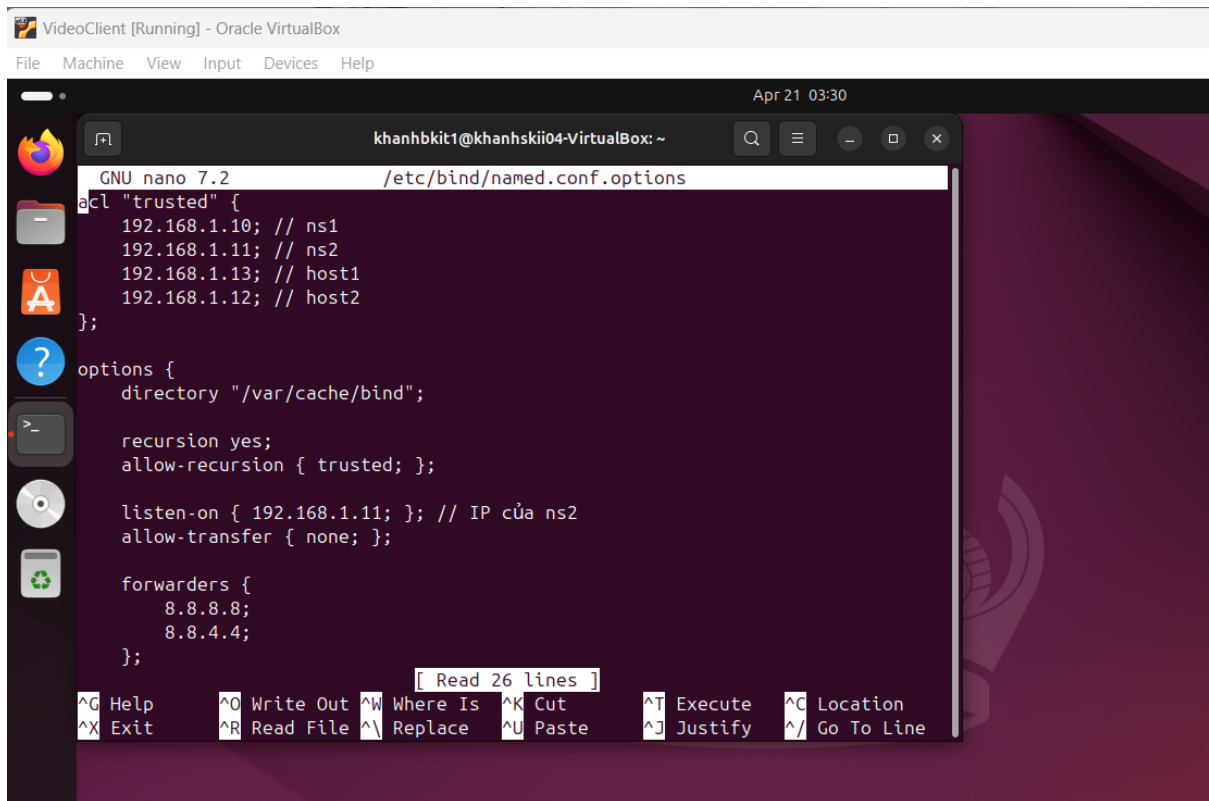
Bây giờ máy chủ DNS chính đã được cấu hình chạy và sẵn sàng trả lời các yêu cầu DNS.

Hãy cùng nhau cấu hình máy chủ thứ 2.

Cấu hình máy chủ DNS thứ 2 *ns2*

Thông thường, người ta thường phải xây dựng thêm một máy chủ DNS thứ 2 để có thể trả lời các yêu cầu từ client khi mà máy chủ chính không sẵn sàng trả lời.

Chỉnh sửa tệp sau */etc/bind/named.conf.options*



```
GNU nano 7.2 /etc/bind/named.conf.options
acl "trusted" {
    192.168.1.10; // ns1
    192.168.1.11; // ns2
    192.168.1.13; // host1
    192.168.1.12; // host2
};

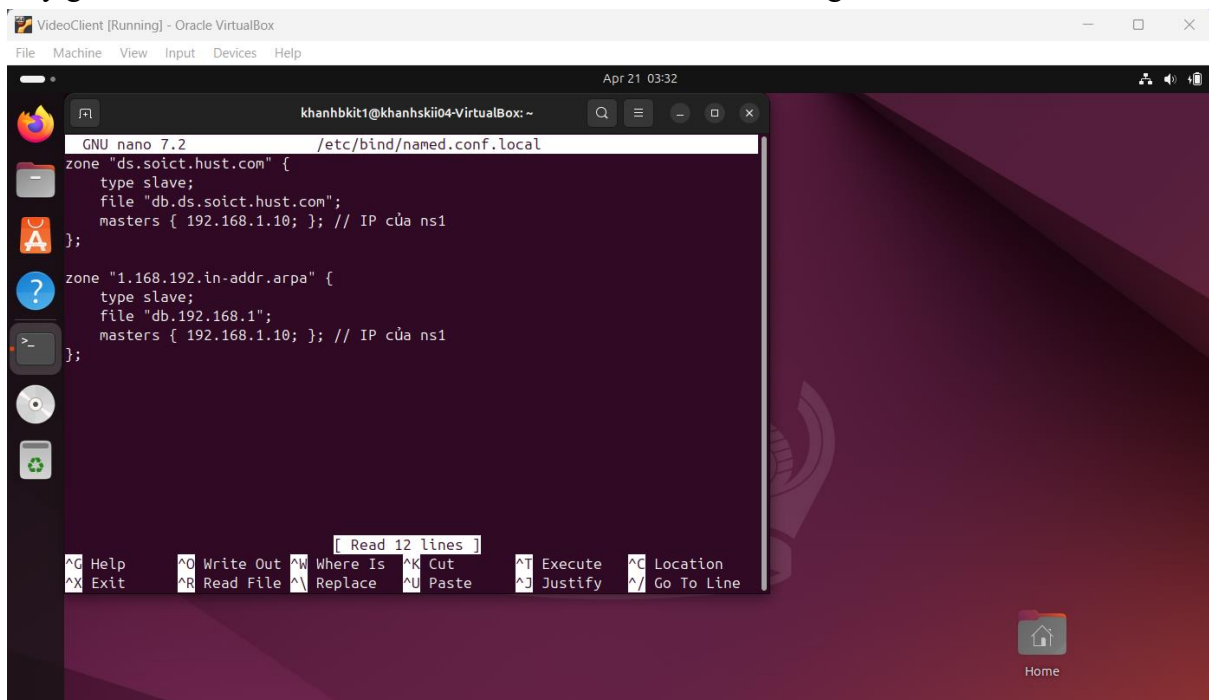
options {
    directory "/var/cache/bind";

    recursion yes;
    allow-recursion { trusted; };

    listen-on { 192.168.1.11; }; // IP của ns2
    allow-transfer { none; };

    forwarders {
        8.8.8.8;
        8.8.4.4;
    };
};
```

Bây giờ mở file /etc/bind/named.conf.local và thêm vào nội dung sau:



```
GNU nano 7.2 /etc/bind/named.conf.local
zone "ds.soict.hust.com" {
    type slave;
    file "db.ds.soict.hust.com";
    masters { 192.168.1.10; }; // IP của ns1
};

zone "1.168.192.in-addr.arpa" {
    type slave;
    file "db.192.168.1";
    masters { 192.168.1.10; }; // IP của ns1
};
```

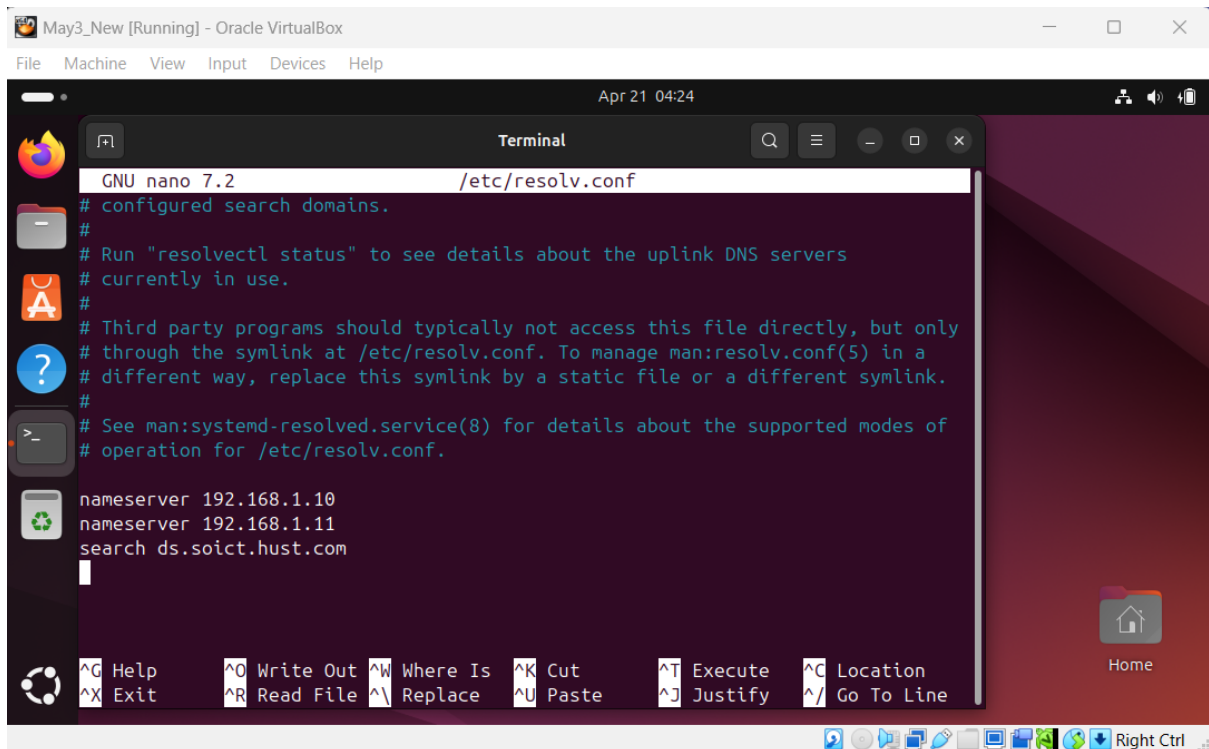
Chạy lệnh sau để kiểm tra cú pháp file cấu hình:

```
$sudo named-checkconf
```

Khi đã kiểm tra không có lỗi, hãy khởi động lại BIND:

```
$sudo systemctl restart bind9
```

```
$sudo ufw allow Bind9
```



Thao tác trên Client host1 và host1

Bây giờ hãy thực hiện các thao tác trên 2 máy client host1 và host2.

Đầu tiên, hãy tìm các thiết bị kết nối với mạng riêng của bạn bằng cách dùng lệnh *ip command*:

```
$ip address show to 192.168.1.0/24
```

Các bạn sẽ thấy xuất hiện nội dung đại loại như sau:

```
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
```

... Vậy với ví dụ này thì private interface là *eth1*. (chú ý là trường hợp của bạn có thể khác).

Bây giờ hãy dùng lệnh *nslookup* để thực hiện gửi yêu cầu tìm kiếm đến cho những DNS server mà bạn đã cấu hình ở trên:

```
$nslookup host1
```

hoặc

```
$nslookup host2
```

```
khanh03@khanhskii04-VirtualBox:~$ nslookup host1
Server:      192.168.1.10
Address:     192.168.1.10#53

Name:   host1.ds.soict.hust.com
Address: 192.168.1.13
```

```
khanh03@khanhskii04-VirtualBox:~$ sudo nano /etc/resolv.conf
khanh03@khanhskii04-VirtualBox:~$ nslookup host2
Server:          192.168.1.10
Address:         192.168.1.10#53

Name:   host2.ds.soict.hust.com
Address: 192.168.1.12
```

```
khanh04@khanhskii04-VirtualBox:~$ sudo nano /etc/resolv.conf
khanh04@khanhskii04-VirtualBox:~$ nslookup host2
Server:          192.168.1.10
Address:         192.168.1.10#53

Name:   host2.ds.soict.hust.com
Address: 192.168.1.12
```

Câu hỏi 11: Bạn nhận được kết quả gì sau 2 lệnh ở trên? Hãy giải thích cơ chế hoạt động của nó.

Khi thực hiện `nslookup host2`, thu được:

```
Name:      host2.ds.soict.hust.com
Address: 192.168.1.12
```

Khi thực hiện `nslookup host2`, trình phân giải kiểm tra trong `/etc/resolv.conf` và thấy dòng `search ds.soict.hust.com`, nó tự động mở rộng `host2` thành `host2.ds.soict.hust.com` rồi thực hiện truy vấn. Trình phân giải DNS trên máy gửi yêu cầu đến máy chủ DNS chính (192.168.1.10) đã khai báo trong `/etc/resolv.conf`. Máy chủ DNS tra cứu file zone `/etc/bind/zones/db.ds.soict.hust.com`, trả về địa chỉ IP

Tương tự, hãy thực hiện truy vấn tìm kiếm reverse:

```
$nslookup 192.168.1.100
```

hoặc

```
$nslookup 192.168.1.101
```

```
khanh04@khanhskii04-VirtualBox:~$ nslookup 192.168.1.12
12.1.168.192.in-addr.arpa      name = host2.ds.soict.hust.com.

khanh04@khanhskii04-VirtualBox:~$ nslookup 192.168.1.13
13.1.168.192.in-addr.arpa      name = host1.ds.soict.hust.com.
```

Câu hỏi 12: Bạn thu được nội dung gì sau khi gõ 2 lệnh trên? Giải thích.

Khi thực hiện truy vấn tìm kiếm reverse với lệnh `nslookup` trên hai địa chỉ IP 192.168.1.12 và 192.168.1.13, kết quả trả về sẽ là tên miền DNS của các địa chỉ IP này

Trình phân giải DNS chuyển địa chỉ IP thành tên miền bằng cách:

- Đảo ngược địa chỉ IP: 192.168.1.12 → 12.1.168.192.in-addr.arpa
- Gửi truy vấn đến DNS Server (192.168.1.10) để tra trong file reverse zone

Câu hỏi 13: Bây giờ giả sử bạn muốn thêm 1 host vào mạng của bạn, và bạn cũng muốn thêm nó vào dịch vụ DNS. Chỉ ra lần lượt các bước mà bạn phải làm/cấu hình.

Bước 1: Xác định tên miền và địa chỉ IP cho máy chủ mới

Bước 2: Khai báo IP trong named.conf.local ứng với server DNS trực tiếp của zone muốn tham gia để được cho phép gửi yêu cầu DNS, cập nhật file cấu hình trong zone forward.

Bước 3: Cập nhật file cấu hình zone reverse

Bước 4: Kiểm tra cấu hình và khởi động lại BIND

Bước 5: Cấu hình DNS trên máy chủ mới: Mở file cấu hình và thêm khai báo tên miền muốn được nhận và địa chỉ IP.