BÀI THỰC HÀNH
MÔN HỌC: HỆ PHÂN TÁN
CHƯƠNG 3: ĐỊNH DANH

1. Giao thức ARP

1.1. Contents

Trong buổi học lý thuyết, chúng ta đã học được một trong những cơ chế phân giải tên cho không gian tên phẳng là *Quảng bá*. Ví dụ điển hình của cơ chế phát sóng là Giao thức ARP trong mọi mạng LAN. ARP là một trong những giao thức chính trong TCP / IP và mục đích của Giao thức phân giải địa chỉ (ARP) là phân giải địa chỉ IPv4 (32 bit) thành địa chỉ vật lý (Địa chỉ MAC 48 bit). Các ứng dụng mạng tại tầng ứng dụng sử dụng Địa chỉ IPv4 để liên lạc với thiết bị khác. Nhưng ở tầng Datalink, địa chỉ là địa chỉ MAC (Địa chỉ vật lý 48 bit) và địa chỉ này được ghi vào card mạng vĩnh viễn. Đó là lý do tại sao chúng ta phải sử dụng giao thức ARP. Trong phần đầu tiên của bài thực hành này, chúng ta sẽ xem ARP hoạt động như thế nào.

1.2. Yêu cầu

1.2.1. Lý thuyết

- ARP protocol

1.2.2. Phần cứng

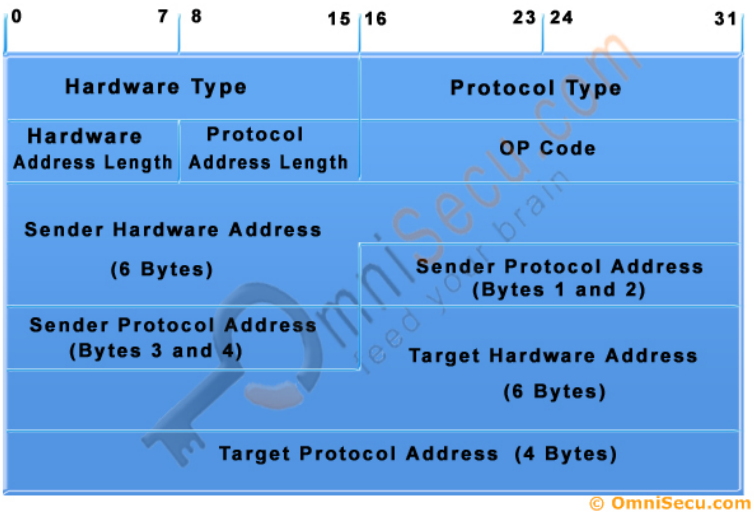
- Laptop on Windows/Linux

1.2.3. Phần mềm

- Wireshark

1.3. Các bước thực hành

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.

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
```

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
```

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 Wireshare (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:
- Destination MAC address
- Opcode
- Target MAC address

Bây giờ hãy quan sát cửa sổ thông tin của Wireshark trên máy A bằng cách chọn thông điệp ARP reply:

Câu hỏi 3: Hãy cho biết các thông tin sau trong cửa sổ bạn đang quan sát:
- Opcode
- Sender MAC address
- Sender IP address
- Target MAC address
- Target IP address

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

```
>arp -a
```

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

2. Tự cài đặt máy chủ DNS

2.1. Nội dung

Quá trình phân giải tên miền DNS bao gồm quá trình chuyển đổi hostname (ví dụ như example.com) sang địa chỉ IP (ví dụ 192.168.2.1). Trong bài thực hành này, các bạn sẽ tự cài đặt máy chủ DNS cho mạng cục bộ riêng của các bạn. Đây là cách để quản lý máy chủ của bạn tốt hơn. Cụ thể, bạn sẽ cài đặt máy chủ DNS cục bộ với việc sử dụng phần mềm BIND (BIND9) trên Ubuntu18.04 để phân giải các hostname và địa chỉ IP trong mạng của bạn.

2.2. Yêu cầu

2.2.1. Lý thuyết

- DNS
- Naming System in Distributed Systems

2.2.2. Phần cứng

- Laptop/PC on Ubuntu 18.04

2.2.3. Phần mềm

- Ubuntu OS 18.04
- bind9

2.3. Các bước thực hành

Trong bài thực hành này chúng ta sẽ làm việc với 4 máy cài Ubuntu 18.04 được mô tả như bảng sau (các bạn có thể sử dụng máy ảo nếu không đủ máy thật).

Tên máy	Vai trò	Private FQDN	Private IP Address
ns1	Primary DNS Server	ns1.ds.soict.hust.com	192.168.1.20
ns2	Secondary DNS Server	ns2.ds.soict.hust.com	192.168.1.21
host1	Generic Host 1	host1.ds.soict.hust.com	192.168.1.100
host2	Generic Host 2	host2.ds.soict.hust.com	192.168.1.101

Trong bài thực hành này, ns1 và ns2 là 2 máy chủ DNS. host1 và host2 là 2 client sẽ sử dụng dịch vụ DNS mà bạn vừa tạo ra. Bạn có thể thêm nhiều hơn 2 host client, tuy nhiên trong bài thực hành này chúng ta sẽ làm việc với 2 client. Chúng ta giả định là các máy chủ cùng nằm trong cùng 1 datacenter tên là *ds*. Bạn sẽ sử dụng một name scheme sử dụng "*ds.soict.hust.com*" để trỏ đến mạng con riêng hoặc là zone.
Địa chỉ IP có thể không nhất thiết phải y hệt như bảng trên, tuy nhiên nếu bạn thay đổi nó thì bạn nhớ phải thay đổi ở tất cả các file cấu hình khác trong bài thực hành này.

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:

```
acl "trusted" {
        192.168.1.20;    # ns1
        192.168.1.21;    # ns2
        192.168.1.100;  # host1
        192.168.1.101;  # host2
};

options {

        . . .
```

Bây giờ bạn hãy chỉnh sửa block *options* như sau:

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

      recursion yes;                 # enables recursive queries
      allow-recursion { trusted; };  # allows recursive queries
      listen-on { 192.168.1.20; };   # ns1 private IP address
      allow-transfer { none; };   # disable zone transfers by default

      forwarders {
              8.8.8.8;
              8.8.4.4;
      };

        . . .
};
```

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

Các thông tin cấu hình trên ghi rõ chỉ các máy mà bạn tin tưởng mới được gửi yêu cầu phân giải DNS cho các miền bên ngoài.

Bây giờ, bạn sẽ cấu hình file local (*/etc/bind/named.conf.local*), để chỉ rõ các zone phục vụ yêu cầu forward hay reverse. Vùng DNS chỉ rõ mục đích chính để quản lý và định nghĩa các bản ghi DNS. Vì các domains của bạn đều nằm trong domain con *ds.soict.hust.com*, bạn sẽ sử dụng nó như zone để forward. Vì các địa chỉ IP của các máy chủ của bạn đều năm trong dải 192.168.1.0/24, nên bạn sẽ cài đặt zone cho reverse sao cho bạn có thể định nghĩa việc tìm kiếm reverse trong vùng đó.

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

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
};
```

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

Bây giờ hãy tạo tệp cho zone để forward.
Bạn sẽ tạo 1 thư mục tên là zones và tạo 1 tệp cho zone để forward bằng cách copy từ tệp *db.local* như sau:

```
$sudo mkdir /etc/bind/zones
$sudo cp /etc/bind/db.local /etc/bind/zones/db.ds.soict.hust.com
```

Bây giờ hãy mở tệp *db.ds.soict.hust.com*, hãy xóa hết nội dung của nó và thay bằng nội dung sau:

```
$TTL    604800
@       IN      SOA     ns1.ds.soict.hust.com. admin.ds.soict.hust.com. (
                              3         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
;
; name servers - NS records
     IN      NS      ns1.ds.soict.hust.com.
     IN      NS      ns2.ds.soict.hust.com.

; name servers - A records
ns1.ds.soict.hust.com.          IN      A       192.168.1.20
ns2.ds.soict.hust.com.          IN      A       192.168.1.21

; 192.168.1.0/24 - A records
host1.ds.soict.hust.com.        IN      A       192.168.1.100
host2.ds.soict.hust.com.        IN      A       192.168.1.101
```

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

Bây giờ bạn cần phải tạo tệp cho zone để reverse. Tệp zone để reverse là nơi mà chúng ta định nghĩa các bản ghi DNS PTR để cho các yêu cầu tìm kiếm reverse DNS. Cụ thể đó là khi máy chủ DNS nhận được yêu cầu phân giải từ 1 địa chỉ IP sang 1 địa chỉ tên miền.
Hãy tạo 1 tệp zone để reverse trong 1 được copy từ file *db.127*

```
$sudo cp /etc/bind/db.127 /etc/bind/zones/db.192.168.1
```

Mở file *db.192.168.1* vừa tạo, xóa hết nội dung đi và đưa nội dung sau vào:

```
$TTL    604800
@       IN      SOA     ds.soict.hust.com. admin.ds.soict.hust.com. (
                              3         ; Serial
                         604800         ; Refresh
```

```
                          86400         ; Retry
                        2419200         ; Expire
                         604800 )       ; Negative Cache TTL
; name servers
      IN      NS      ns1.ds.soict.hust.com.
      IN      NS      ns2.ds.soict.hust.com.

; PTR Records
20    IN      PTR     ns1.ds.soict.hust.com.    ; 192.168.1.20
21    IN      PTR     ns2.ds.soict.hust.com.    ; 192.168.1.21
100   IN      PTR     host1.ds.soict.hust.com.  ; 192.168.1.100
101   IN      PTR     host2.ds.soict.hust.com.  ; 192.168.1.101
```

Sau đó chạy lệnh sau để kiểm tra cú pháp của các tệp *named.conf**

```
$sudo named-checkconf
```

Nếu các tệp cấu hình tên của bạn không có lỗi nào thì bạn sẽ không nhận được kết quả báo lỗi gì sau khi gõ lệnh trên.

Bạn sẽ dụng lệnh *named-checkzone* để kiểm tra cấu hình zone để forward "*ds.soict.hust.com*":

```
$sudo named-checkzone ds.soict.hust.com
/etc/bind/zones/db.ds.soict.hust.com
```

Và bạn cũng kiểm tra cả file cấu hình zone để reverse:

```
$sudo named-checkzone 1.168.192.in-addr.arpa
/etc/bind/zones/db.192.168.1
```

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

Khi tất cả các file cấu hình đều không bị lỗi gì thì bạn cần phải khởi động lại dịch vụ BIND với các lệnh sau:

```
$sudo systemctl restart bind9
$sudo ufw allow bind9
```

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

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

```
acl "trusted" {
        192.168.1.20;    # ns1
        192.168.1.21;    # ns2
        192.168.1.100;   # host1
        192.168.1.101;   # host2
};

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

      recursion yes;                 # enables resursive queries
      allow-recursion { trusted; };  # allows recursive queries
      listen-on { 192.168.1.21; };   # ns2 private IP address
      allow-transfer { none; };      # disable zone transfers by
      default

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

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

```
zone "ds.soict.hust.com" {
    type slave;
    file "db.ds.soict.hust.com";
    masters { 192.168.1.20; };  # ns1 private IP
};

zone "1.168.192.in-addr.arpa" {
    type slave;
    file "db.192.168.1";
    masters { 192.168.1.20; };  # ns1 private IP
};
```

Chạy lệnh sau để kiểm tra cú phép 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 cấu hình DNS server trong tệp /etc/resolv.conf và thêm các dòng sau vào:

```
nameserver 192.168.1.20
nameserver 192.168.1.21
search ds.soict.hust.com
```

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
```

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

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

```
$nslookup 192.168.1.100
```

hoặc

```
$nslookup 192.168.1.101
```

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

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.