

LAPORAN JARINGAN KOMPUTER II

SETTING DNS



Disusun Oleh :

Adham Hayukalbu | IK-2B | 3.34.12.1.01

Jurusan Elektro

Teknik Informatika

Politeknik Negeri Semarang

2014

I. Tujuan

Setelah menyelesaikan praktek ini, mahasiswa dapat :

- a. menjelaskan Konsep DNS
- b. menyebutkan Paket DNS
- c. mengkonfigurasi Server DNS

II. Dasar Teori

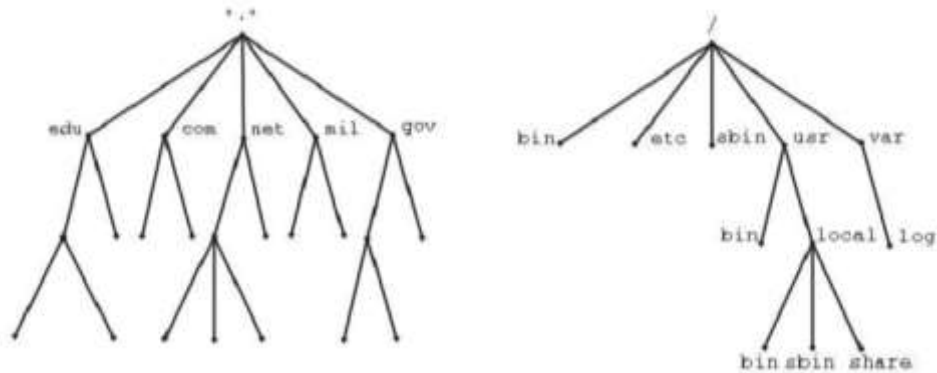
DNS merupakan sistem berbentuk database terdistribusi yang akan memetakan/mengkonversikan nama host /mesin/domain ke alamat IP (Internet Protocol) dan sebaliknya dari alamat IP ke nama host yang disebut dengan reverse-mapping . DNS sangat berguna sekali dalam jaringan terutama Internet. Sistem ini akan memetakan nama mesin, misalnya elektro.polines.com ke alamat IP, misalnya 192.168.45.16. Bila tidak ada DNS, kepala kita bisa pusing, jika harus mengingat ratusan, ribuan, bahkan jutaan alamat IP di Internet. Manusia lebih mudah untuk mengingat nama daripada alamat IP dengan panjang 32 bit itu. Komputer menggunakan alamat IP untuk berkomunikasi dan berinteraksi. Disitulah gunanya DNS sebagai jembatan antara keduanya. Implementasi DNS pada sistem operasi Linux yang sering digunakan adalah BIND meskipun ada juga salah satu implementasi yang cukup baru dan juga banyak dipakai yaitu *djbdns*.

1. Struktur dan Cara Kerja DNS

Struktur database DNS sangat mirip dengan sistem-berkas/ filesystem UNIX yaitu berbentuk hierarki atau pohon. Tingkat teratas pada DNS adalah root yang disimbolkan dengan titik/ dot (.) sedangkan pada sistem berkas UNIX, root disimbolkan dengan slash (/). Setiap titik cabang mempunyai label yang mengidentifikasikannya relatif terhadap root (.). Tiap titik cabang merupakan root bagi sub-tree /tingkat bawahnya. Tiap sub-tree merupakan domain dan dibawah domain terdapat sub-tree lagi bernama subdomain.

Setiap domain mempunyai nama yang unik dan menunjukkan posisinya pada pohon DNS, pengurutan/penyebutan nama domain secara penuh dimulai dari domain paling bawah menuju ke root (.). Masing-masing nama yang membentuk suatu domain dipisahkan dengan titik/ dot (.) dan diakhiri dengan titik yang merupakan nama absolut relatif terhadap root (.). Contoh:

raichu.cs.mit.edu . "." merupakan root, domain edu merupakan Top Level Domain, mit merupakan Second Level Domain, cs merupakan Third Level Domain, raichu merupakan nama komputer/mesin yang bersangkutan. Sistem penulisan nama secara absolut dan lengkap ini disebut FQDN (Fully Qualified Domain Name). Gambar berikut memperlihatkan hirarkhi akar pohon DNS dan berkas Unix.



Gambar 1: Hierarki akar-pohon DNS dan sistem berkas UNIX

2. Top Level Domain (TLD)

Top Level Domain adalah domain pada level teratas di bawah root (.). Ada tiga pengelompokan Top Level Domain:

a. Domain Generik

Terdiri atas 7 domain yaitu

1) com

Untuk organisasi komersial. contoh: ibm.com , sun.com .

2) net

Untuk organisasi/perusahaan penyedia layanan jaringan/Internet. contoh: internic.net , nsf.net .

3) gov

Untuk lembaga/organisasi pemerintahan. contoh: whitehouse.gov , nasa.gov.

4) mil

Untuk badan/organisasi militer. contoh: army.mil .

5) org

Untuk organisasi non-komersial. contoh: linux.org .

6) edu

Untuk lembaga pendidikan. contoh: mit.edu, berkeley.edu.

7) int

Untuk organisasi Internasional. contoh: nato.int.

Selain 7 domain di atas ada lagi 7 domain baru dari ICANN (www.icann.org) yaitu

1) aero

Untuk industri atau perusahaan udara.

2) biz

Untuk perusahaan atau lembaga bisnis.

3) coop

Untuk perusahaan atau lembaga kooperatif

4) info

Untuk penggunaan umum.

5) museum

Untuk museum.

6) name

Untuk registrasi bagi penggunaan individual/personal.

7) pro

Untuk para profesional seperti: akuntan, dan lain-lain.

b. Domain Negara

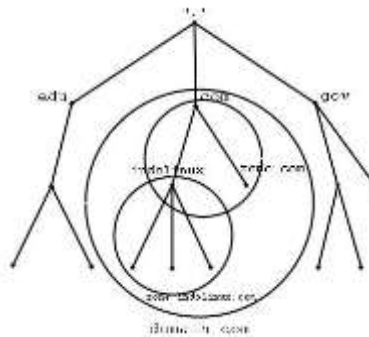
Merupakan standar pembagian geografis berdasarkan kode negara. Contoh: id untuk Indonesia, au untuk Australia, uk untuk Inggris, dan lain-lain. Domain negara ini dapat dan umumnya diturunkan lagi ke level-level di bawahnya yang diatur oleh NIC dari masing-masing negara, untuk Indonesia yaitu IDNIC. Contoh level bawah dari id yaitu net.id, co.id, web.id.

c. Domain Arpa

Merupakan domain untuk jaringan ARPAnet. Tiap domain yang tergabung ke Internet berhak memiliki name-space .in-addr.arpa sesuai dengan alamat IPnya.

3. Name-Server dan Zone

Name-server adalah program server yang memiliki informasi mengenai host di bawah domain tertentu. Name-server mempunyai bagian-bagian informasi mengenai domain name-space yang dikenal dengan zone. Zone ini akan dimuat dari berkas yang disimpannya atau dari server DNS lain. Sebagai contoh domain id bisa dibagi menjadi beberapa zone yaitu ac.id, net.id dan dari zone tersebut bisa dibagi lagi menjadi zone- zone yang lebih kecil misal undip.ac.id,dst. Disini yang bertanggung jawab ialah organisasi/lembaga yang memiliki domain tersebut.



Gambar 2: Pembagian zone

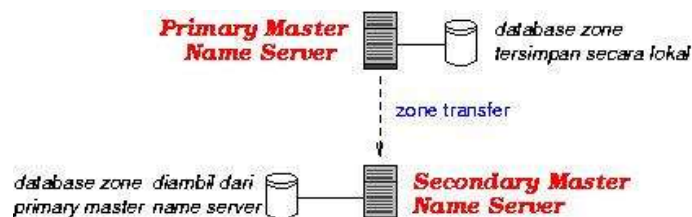
Ada dua tipe dari name-server yaitu :

a. Primary Master

Primary Master name-server bertanggung jawab terhadap suatu zone dengan memuat informasi dari berkas database pada dirinya sendiri.

b. Secondary Master (Slave)

Secondary Master name-server memuat informasi zone dari server lain yang otoritatif pada suatu zone yang disebut Master Server. Secondary Master akan melakukan zone-transfer dari Master Server untuk mendapatkan data/informasi pada suatu zone yang dikelolanya.



Gambar 3: Primary Master dan Secondary Master server DNS

Untuk dapat bekerja name-server membutuhkan resource-record untuk domain yang dikelola. Resource-record adalah data yang mengacu/menunjukkan pada informasi mengenai host pada jaringan atau domain tertentu.

4. Komponen DNS

DNS sebenarnya merupakan suatu sistem server-client, jadi ada suatu mekanisme dari client untuk meminta informasi dari server yang akan memberikan informasi yang diminta sang client. Seperti yang disebutkan di atas program pada server tersebut sering disebut dengan name-server. Pada client sering disebut dengan resolver.

DNS mempunyai beberapa komponen yaitu :

a) Resolver

Resolver yaitu suatu rutin pustaka yang akan membuat suatu permintaan/query dan mengirimkannya lewat jaringan ke sebuah name-server. Program tersebut berjalan pada host yang menginginkan informasi mengenai suatu host di Internet. Resolver juga menginterpretasikan respon dari name-server apakah informasi yang diminta merupakan record ataupun kesalahan.

b) Resolution

Resolution yaitu proses pencarian name-server yang mempunyai tanggung jawab terhadap suatu domain yang akan diminta. Setelah name-server yang dicari ditemukan maka server akan memberikan informasi name-server yang bersangkutan kepada pemintanya.

c) Caching

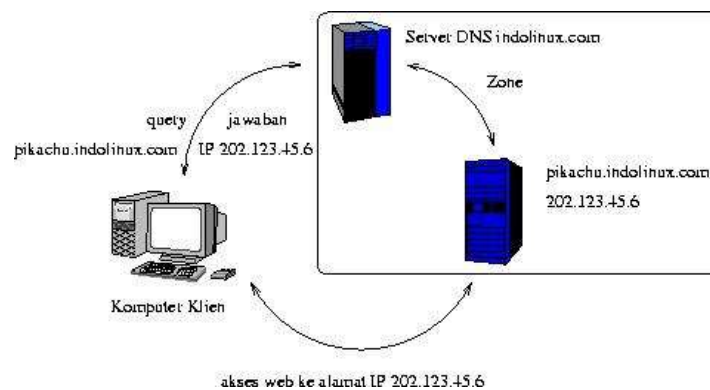
Caching yaitu suatu rutin yang akan menyimpan hasil pencarian domain dalam database dari name-server yang pernah diminta. Time To Live (TTL) merupakan batas waktu dimana server DNS dapat menyimpan/caching informasi yang pernah dicari.

5. BIND (Berkeley Internet Name Domain)

BIND merupakan salah satu implementasi dari DNS yang paling banyak digunakan pada server di Internet. Implementasi DNS pertama adalah JEEVES buatan Paul Mockapetris. BIND dibuat untuk sistem operasi BSD UNIX 4.3 oleh Kevin Dunlap, tapi kemudian banyak di-porting ke banyak turunan

UNIX termasuk Linux. BIND sampai sekarang masih dikoordinasi oleh Internet Software Consortium. Program utama dari BIND adalah bernama `named` yaitu sebuah daemon yang bila dijalankan akan menunggu koneksi pada port 53 (default). Koneksi pada port 53 ini adalah koneksi permintaan informasi pemetaan dari nama domain/mesin ke alamat IP dan sebaliknya. Jika sebuah server DNS mempunyai otorisasi terhadap suatu domain maka si server DNS tersebut akan memberikan informasi mengenai nama-nama mesin/domain yang berada di bawah domain yang dipegangnya. Misal: server DNS mempunyai otorisasi terhadap domain `indolinux.com`, maka ketika ada sebuah komputer dari Internet ingin mengakses `pikachu.indolinux.com`, maka sang komputer itu akan menghubungi server DNS penulis untuk mengetahui informasi alamat IP dari `pikachu.indolinux.com`. Server DNS akan menjawab permintaan/query dari komputer peminta tersebut dengan alamat.

IP yang sesuai kemudian komputer tersebut dapat mengakses `pikachu.indolinux.com` melalui alamat IP yang diberikan oleh name-server.



Gambar 5: Pencarian host

Sebagian besar distribusi Linux menyertakan paket BIND. Jadi anda tinggal konfigurasi dan aktifkan saja jika sudah terinstal. BIND secara umum terdiri dari beberapa file yang mendukung yaitu :

a. `/etc/resolv.conf`

Yaitu file konfigurasi berisi domain atau alamat IP name-server yang pertama dicari oleh resolver ketika sebuah domain/nama mesin diminta untuk dipetakan ke alamat IP.

b. `/etc/nsswitch.conf`

Yaitu file konfigurasi sistem untuk melakukan mekanisme switch sistem database dan name- service. Switch dapat melalui file, name-server/ NIS server.

c. `/etc/named.conf`

Yaitu file konfigurasi dari BIND yang utama, berisi informasi mengenai bagaimana klien DNS mengakses port 53, letak dan jenis file database yang diperlukan. Umumnya berisi letak file konfigurasi name-server root, domain, localhost/loopback, dan reverse-mapping.

d. `/var/named/named.ca`

Yaitu file database name-server root yang bertanggung jawab terhadap Top Level Domain di Internet. Digunakan untuk mencari domain di luar domain lokal. Nama file ini dapat didefinisikan sendiri tetapi tetap harus mengacu ke `named.conf` sebagai berkas konfigurasi utama dari BIND.

e. `/var/named/named.local`

Yaitu file database name-server untuk alamat loopback/host lokal/alamat diri sendiri. Nama file ini dapat didefinisikan sendiri tetapi tetap harus mengacu ke `named.conf` sebagai berkas konfigurasi utama dari BIND.

f. `/var/named/db.domain-kita.com`

Yaitu file database name-server untuk domain `domain-kita.com` yang berisi resource record, informasi nama host dan alamat IP yang berada di bawah domain `domain-kita.com`. File ini bisa lebih dari satu tergantung jumlah domain yang dikelola. Nama file ini dapat didefinisikan sendiri tetapi tetap harus mengacu ke `named.conf` sebagai berkas konfigurasi utama dari BIND.

g. `/etc/rndc.conf`

Berkas konfigurasi program `rndc` yaitu suatu program untuk administrasi dan kontrol operasi name-server BIND.

6. Paket yang diperlukan

Sebelum membuat server DNS, terlebih dahulu perlu dilakukan instalasi aplikasi server DNS, yaitu BIND, kecuali bila saat melakukan instalasi Linux paket-paket untuk server DNS ini telah dipilih untuk diinstalasi. Ada tiga paket yang perlu diinstalasi, yaitu :

- `bind-[versi].rpm`,
- `bind-utils-[versi].rpm`, dan
- `caching-nameserver-[versi].rpm`.

Pada RHEL4 ini, versi `bind` yang digunakan adalah 9.2.4-2, sedangkan versi `caching-nameserver` yang digunakan adalah 7.3-3. Paket `bind-9.2.4-2.i386.rpm` adalah paket yang berisi program server DNS (daemon), yaitu `/usr/sbin/named` yang bertanggung jawab dalam merespon pertanyaan

klien DNS. Paket ini juga berisi utilitas untuk verifikasi server DNS, dan paket `bind-utils-9.2.4-2.i386.rpm` berisi sekumpulan utilitas untuk proses query ke server DNS. Sedangkan paket `caching-nameserver-7.3-3.noarch.rpm` berisi file-file konfigurasi BIND yang akan membuat server DNS berlaku sebagai caching nameserver. Setelah ketiga paket tersebut diinstalasi, maka telah terinstal program dan file-file konfigurasi ke dalam server DNS.

BIND meletakkan file-file konfigurasi ke dalam dua lokasi berbeda, yaitu :

- **`/etc/named.conf`**, yaitu file konfigurasi untuk daemon named
- **`/var/named`**, yaitu nama direktori kerja daemon named yang akan menyimpan file- file zone, statistics dan cache.

- File **`/etc/named.conf`**

File ini berisi sekumpulan statemen yang menggunakan opsi-opsi yang bersarang dan diapit oleh tanda kurung pembuka dan penutup { }. Perlu kehati-hatian dalam mengedit file ini untuk mencegah adanya syntax error yang menyebabkan gagalnya service named berjalan.

- File-file zone

File-file zone berisi informasi mengenai bagian tertentu dari namespace dan tersimpan di direktori kerja `/var/named`. Tiap file zone diberi nama sesuai dengan nilai dari option file yang ada dalam pernyataan zone di file `/etc/named.conf`. Biasanya dalam penggunaannya berhubungan dengan nama domain atau zone yang ditanyakan dan mengidentifikasi file sebagaimana berisi data zone tersebut, misalnya `linuxpolines.com.zone`. Setiap file zone berisi direktif-direktif dan resource records. Direktif-direktif ini bertujuan agar nameserver menampilkan tugas-tugas atau menerapkan konfigurasi khusus pada zone. Resource records mendefinisikan parameter-parameter zone dan menentukan identitas masing-masing host (komputer). Direktif sifatnya optional, tetapi resource records harus ada dalam menyediakan nameservice suatu zone.

III. Peralatan yang Digunakan

- 1) PC Komputer sebagai Server
- 2) PC Komputer sebagai Client/workstation
- 3) Alat penghubung Switch/hub
- 4) Kabel UTP

IV. Langkah Kerja

1. Instal Aplikasi DNS yaitu Bind9 pada distro Ubuntu.
2. Ganti media default pada ETH0 di file konfigurasi **/etc/default/isc-dhcp-server**.
3. Masuk dan berikan setting IP pada media eth0 di file konfigurasi **/etc/network/interfaces**.
4. Masuk dan berikan nameserver pada file konfigurasi **/etc/resolv.conf**.
5. Berikan IP ISP untuk DNS anda pada file konfigurasi di **/etc/bind/named.conf.options**.
6. Masuk dan memberikan settingan zone master sebagai forward dan reverse pada file konfigurasi **/etc/bind/named.conf.local**.
7. Berikan nama file forward nya yaitu : **db.adham.net**.
8. Berikan nama file reverse nya yaitu : **db.192**.
9. Copy isi file konfigurasi **db.local** ke **db.adham.net**.
10. Copy isi file konfigurasi **db.127** ke **db.192**.
11. Restart DNS setelah itu lakukan cek IP dan lakukan PING pada nama DNS yang telah kita buat dengan memberikan nameserver yaitu adham.net
12. Lakukan nslookup pada setting DNS yang telah kita buat.

Review Langkah :

- **Masuk Sebagai Root User**

```
root@ubuntu: /home/sojoyenjoy
sojoyenjoy@ubuntu:~$ sudo su
[sudo] password for sojoyenjoy:
root@ubuntu: /home/sojoyenjoy#
```

- **Install Aplikasi Bind9**

```
root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# apt-get install bind9
Reading package lists... Done
Building dependency tree
Reading state information... Done
bind9 is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@ubuntu:/home/sojoyenjoy#
```

- Ubah Media default to ETH0

```
root@ubuntu: /home/sojoyenjoy
GNU nano 2.2.6      File: /etc/default/isc-dhcp-server      Modified

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPD_PID=/var/run/dhcpd.pid

# Additional options to start dhcpd with.
#       Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACES="eth0"

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

- Berikan Setting IP pada ETH0

```
root@ubuntu: /home/sojoyenjoy
GNU nano 2.2.6      File: /etc/network/interfaces      Modified

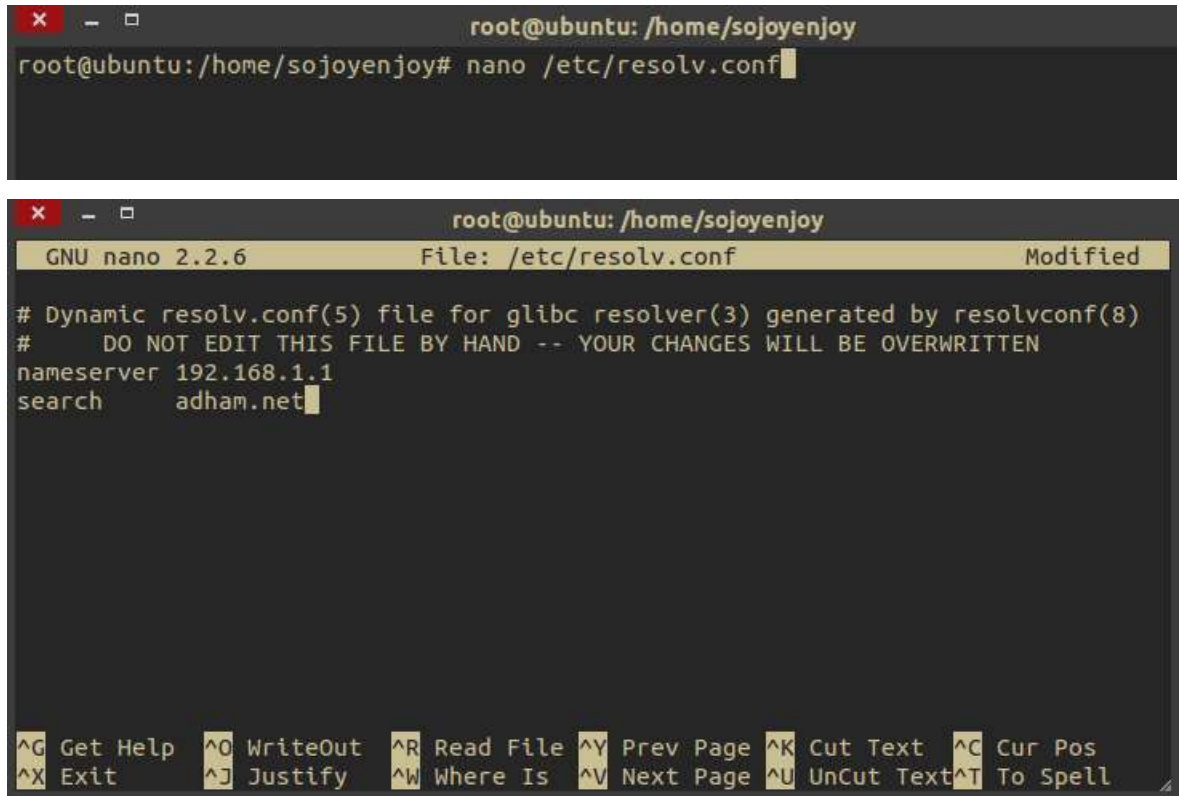
# interfaces(5) file used by ifup(8) and ifdown(8)
# auto lo
# iface lo inet loopback
auto eth0
iface eth0 inet static

address 192.168.1.1
netmask 255.255.255.0
gateway 192.168.1.1
broadcast 192.168.1.255
network 192.168.1.0

# dns-search adham.net
# dns-nameserver 192.168.1.1

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

- Masuk dan Berikan Nameserver pada file konfigurasi **/etc/resolv.conf**



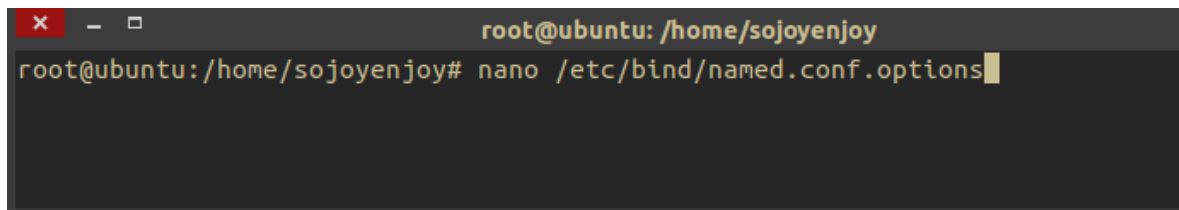
The first screenshot shows a terminal window with the prompt `root@ubuntu: /home/sojoyenjoy` and the command `root@ubuntu:/home/sojoyenjoy# nano /etc/resolv.conf` being entered. The second screenshot shows the nano editor interface for `/etc/resolv.conf`. The status bar at the top indicates `GNU nano 2.2.6`, `File: /etc/resolv.conf`, and `Modified`. The main text area contains the following content:

```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.1.1
search      adham.net
```

The bottom status bar of the nano editor shows various keyboard shortcuts: `^G Get Help`, `^O WriteOut`, `^R Read File`, `^Y Prev Page`, `^K Cut Text`, `^C Cur Pos`, `^X Exit`, `^J Justify`, `^W Where Is`, `^V Next Page`, `^U UnCut Text`, and `^T To Spell`.

Keterangan : gambar diatas adalah memberikan nama ns yang berada pada ip 192.168.1.1

- Berikan ISP jika anda butuhkan atur pada file konfigurasi **/etc/bind/named.conf.options**.



The screenshot shows a terminal window with the prompt `root@ubuntu: /home/sojoyenjoy` and the command `root@ubuntu:/home/sojoyenjoy# nano /etc/bind/named.conf.options` being entered.

```

root@ubuntu: /home/sojoyenjoy
GNU nano 2.2.6      File: /etc/bind/named.conf.options      Modified

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

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        208.67.222.222; //OPEN DNS
        8.8.4.4; //Google DNS
    };
    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys.  See https://www.isc.org/bind-keys

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell

```

- Berikan Setting Zone Master untuk Forward dan Reverse pada file konfigurasi berikut :

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu: /home/sojoyenjoy# nano /etc/bind/named.conf.local

```

```

root@ubuntu: /home/sojoyenjoy
GNU nano 2.2.6      File: /etc/bind/named.conf.local

zone "adham.net"
{
    type master;
    file "/etc/bind/db.adham.net";
};

zone "1.168.192.in-addr.arpa"
{
    type master;
    notify no;
    file "/etc/bind/db.192";
};

^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell

```


- Copy file konfigurasi db.local pada db.adham.net dan ubah setting pada db.adham.net

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# cp /etc/bind/db.local /etc/bind/db.adham.net

```

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# nano /etc/bind/db.adham.net

```

```

GNU nano 2.2.6 File: /etc/bind/db.adham.net

;
; BIND data file for local loopback interface
;
$TTL      604800
@          IN      SOA      ns.adham.net. root.adham.net. (
                        2      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800  ) ; Negative Cache TTL
;
@          IN      NS       ns.adham.net.
@          IN      A        192.168.1.1
ns         IN      A        192.168.1.1

[ Read 15 lines ]
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell

```

- Copy file konfigurasi db.127 pada db.192 dan ubah setting file db.192

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# cp /etc/bind/db.127 /etc/bind/db.192

```

```

root@ubuntu: /home/sojoyenjoy
GNU nano 2.2.6 File: /etc/bind/db.192 Modified

;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@        IN      SOA      ns.adham.net. root.adham.net. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@        IN      NS       ns.
1        IN      PTR      192.168.1.1.

[ Cancelled ]
^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text ^T To Spell

```

- Lakukan Restart bind9

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# /etc/init.d/bind9 restart
* Stopping domain name service... bind9
waiting for pid 1079 to die

* Starting domain name service... bind9
root@ubuntu:/home/sojoyenjoy#
[ OK ]
[ OK ]

```

- Ping IP dan NameServer

```

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.042 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=0.053 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.051 ms
64 bytes from 192.168.1.1: icmp_seq=4 ttl=64 time=0.060 ms
64 bytes from 192.168.1.1: icmp_seq=5 ttl=64 time=0.062 ms

root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# ping adham.net
PING adham.net (192.168.1.1) 56(84) bytes of data.
64 bytes from adham.net (192.168.1.1): icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from adham.net (192.168.1.1): icmp_seq=2 ttl=64 time=0.054 ms
64 bytes from adham.net (192.168.1.1): icmp_seq=3 ttl=64 time=0.060 ms
64 bytes from adham.net (192.168.1.1): icmp_seq=4 ttl=64 time=0.061 ms
64 bytes from adham.net (192.168.1.1): icmp_seq=5 ttl=64 time=0.060 ms

```

- Lakukan NSLookup pada Nameserver

```
root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# nslookup adham.net
Server:      192.168.1.1
Address:     192.168.1.1#53

Name:   adham.net
Address: 192.168.1.1

root@ubuntu:/home/sojoyenjoy#
```

V. Tugas

1. Lakukan uji koneksi antar PC dengan menggunakan seting DNS dengan nomor IP address yang anda set-kan

Jawab :

- a. Untuk melakukan uji koneksi pada DNS harus mengaktifkan DHCP terlebih dahulu, sehingga saling berkaitan.

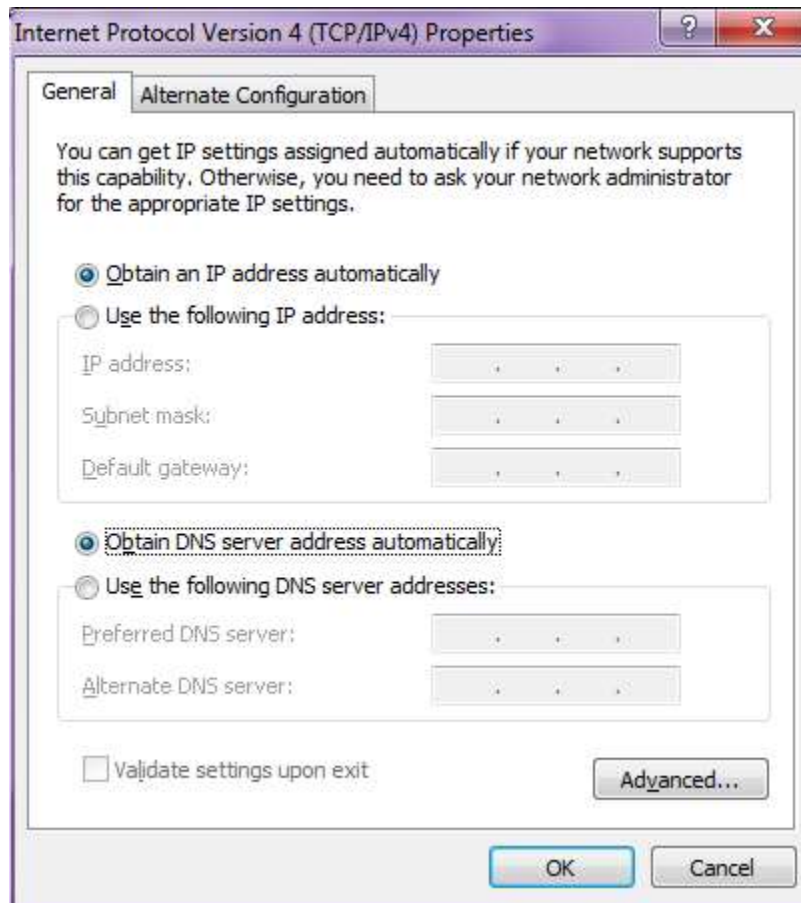
```
root@ubuntu: /home/sojoyenjoy
root@ubuntu:/home/sojoyenjoy# nano /etc/default/isc-dhcp-server
```

```
GNU nano 2.2.6      File: /etc/dhcp/dhcpd.conf      Modified

# option domain-name "internal.example.org";
# option routers 10.5.5.1;
# option broadcast-address 10.5.5.31;
# default-lease-time 600;
# max-lease-time 7200;
#}

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.2 192.168.1.30;
    option domain-name-servers adham.net;
    option domain-name "adham.net";
    option routers 192.168.1.1;
    option broadcast-address 192.168.1.255;
    default-lease-time 600;
    max-lease-time 7200;
}
```

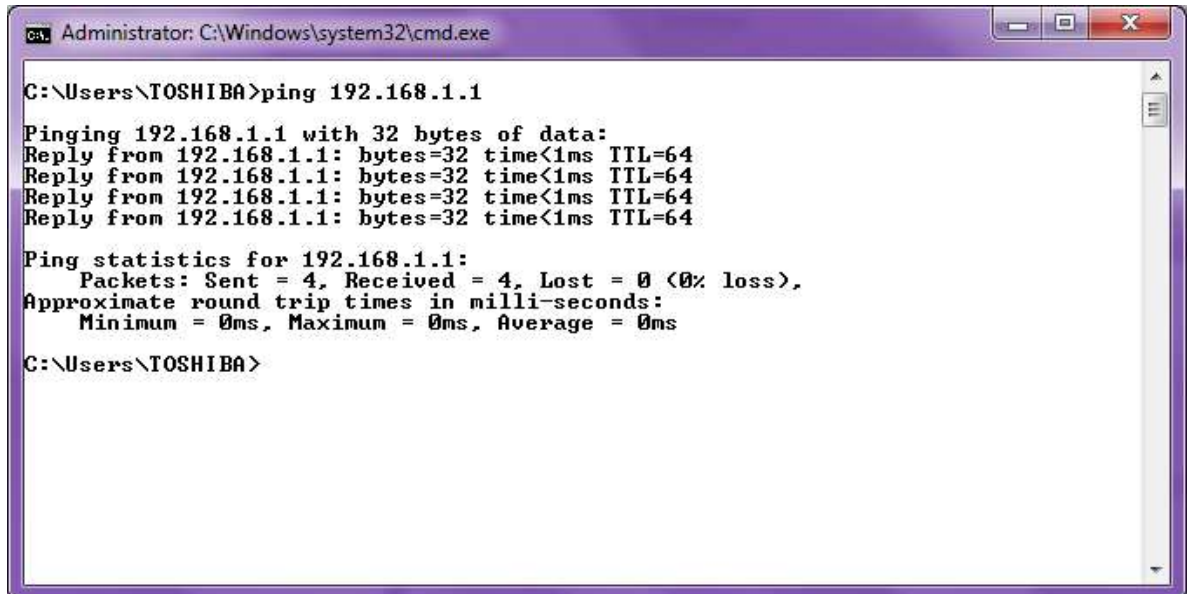

- b. Ubah IP Client menjadi automatic agar dapat menerima DHCP yang diberikan oleh server



- c. Liat Report yang diberikan apakah IP telah diterima oleh client. Berikut gambar IP yang diterima dan IP server berhasil ditangkap.

Network Connection Details:	
Property	Value
Connection-specific DN...	
Description	Atheros AR8151 PCI-E Gigabit Ethernet C
Physical Address	5C-26-0A-6A-A0-E6
DHCP Enabled	No
IPv4 Address	192.168.1.2
IPv4 Subnet Mask	255.255.255.0
IPv4 Default Gateway	192.168.1.1
IPv4 DNS Server	192.168.1.1

- d. Lakukan Ping IP dan Nameserver pada Client to Server



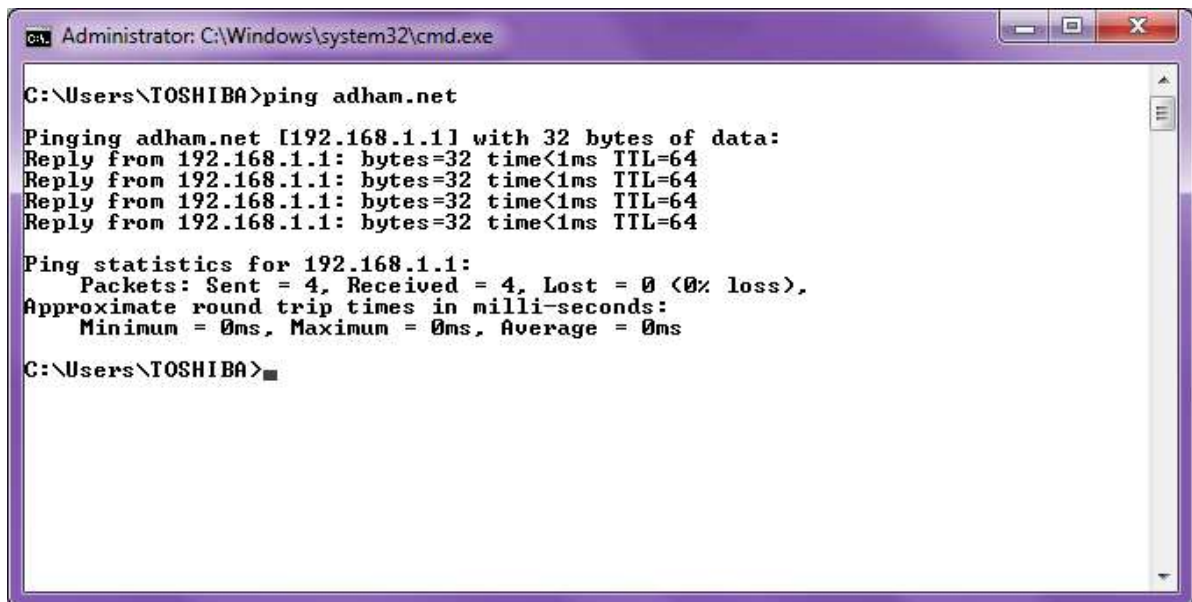
```
Administrator: C:\Windows\system32\cmd.exe

C:\Users\TOSHIBA>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\TOSHIBA>
```



```
Administrator: C:\Windows\system32\cmd.exe

C:\Users\TOSHIBA>ping adham.net

Pinging adham.net [192.168.1.1] with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64
Reply from 192.168.1.1: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\TOSHIBA>
```

- e. Lakukan nslookup pada client untuk melihat keterangan nameserver dari setting dns yang anda lakukan.



```

Administrator: C:\Windows\system32\cmd.exe

C:\Users\TOSHIBA>nslookup adham.net
Server: 192.168.1.1
Address: 192.168.1.1

Name: adham.net
Address: 192.168.1.1

C:\Users\TOSHIBA>nslookup 192.168.1.1
Server: 192.168.1.1
Address: 192.168.1.1

Name: 192.168.1.1
Address: 192.168.1.1

C:\Users\TOSHIBA>
  
```

Keterangan : PC Client adalah Toshiba , client dapat menerima ip automatic dhcp dari server sojoyenjoy dan dapat melakukan uji koneksi ping pada DNS Server baik dari nameserver maupun ip server.

VI. Pertanyaan

1. Dapatkah DNS dilakukan dengan menggunakan kelas jaringan yang berbeda, jelaskan alasannya?

Jawab :

Karena kelas jaringan didunia menggunakan setting IP Address di bawah ini, maka tidak mungkin bisa jika menggunakan kelas jaringan yang berbeda.

\$ gedit /etc/bind/named.conf.options

// Menambah IP DNS ISP Anda

forwarders{

202.134.1.10;

202.134.0.155;

};

Setting pada DNS diawali dengan setting DHCP terlebih dahulu, sehingga jika mengaktifkan DNS saja, maka tidak mungkin bisa tersambung ke client.

Jika menggunakan kelas jaringan yang berbeda, itu tidak mungkin bisa. Karena harus menggunakan kelas jaringan yang sama.

2. Jelaskan mengapa setting DNS, lebih sulit dikenali dibandingkan dengan IP address?

Jawab :

DNS dapat disamakan fungsinya dengan buku telepon. Dimana setiap komputer di jaringan Internet memiliki host name (nama komputer) dan Internet Protocol (IP) address. Secara umum, setiap client yang akan mengkoneksikan komputer yang satu ke komputer yang lain, akan menggunakan host name.

Lalu komputer anda akan menghubungi DNS server untuk mengecek host name yang anda minta tersebut berapa IP address-nya. IP address ini yang digunakan untuk mengkoneksikan komputer anda dengan komputer lainnya.

Jadi lebih mudah dengan IP Address yang biasanya, tapi jika setting DNS sebaiknya diberikan host name, karena manusia tidak mudah mengingat IP Address dari komputer lain (client).

VII. Kesimpulan

1. Diperoleh bahwa Implementasi DNS pada sistem operasi Linux yang sering digunakan adalah BIND meskipun ada juga salah satu implementasi yang cukup baru dan juga banyak dipakai yaitu djbdns. Yang merupakan sistem berbentuk database terdistribusi yang akan memetakan/mengkonversikan nama host /mesin/domain ke alamat IP (Internet Protocol) dan sebaliknya dari alamat IP ke nama host yang disebut dengan reverse-mapping .
2. Diperhatikan bahwa sebelum membuat server DNS, terlebih dahulu perlu dilakukan instalasi aplikasi server DNS, yaitu BIND, kecuali bila saat melakukan instalasi Linux paket-paket untuk server DNS ini telah dipilih untuk diinstalasi. Ada tiga paket yang perlu diinstalasi, yaitu :