

Minggu 9

Layer Aplikasi dan Domain Name System

Layer Aplikasi

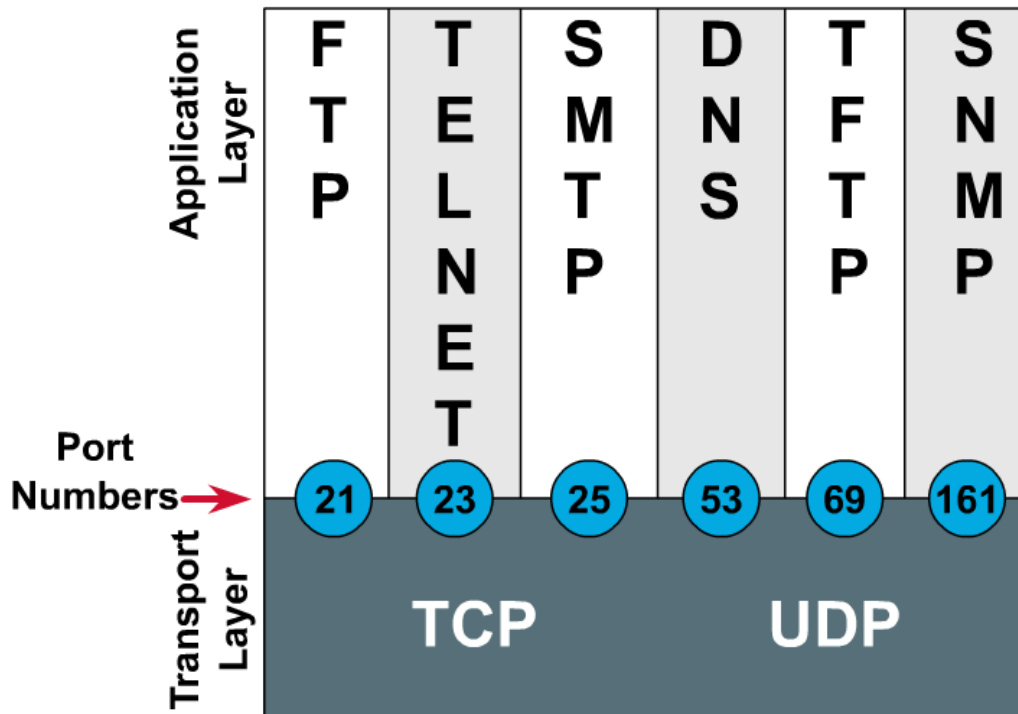
- Layer ini berurusan dengan program komputer yang digunakan oleh user.
- Program komputer yang berhubungan hanya program yang melakukan akses jaringan, bukan program standalone, yaitu program yang berhubungan dengan OSI.
- Contoh: Aplikasi word processing, aplikasi ini digunakan untuk pengolahan text sehingga program ini tidak berhubungan dengan OSI.
- Tetapi bila program tersebut ditambahkan fungsi jaringan misal pengiriman email, maka bisa dikatakan berhubungan dengan OSI.

Pendahuluan

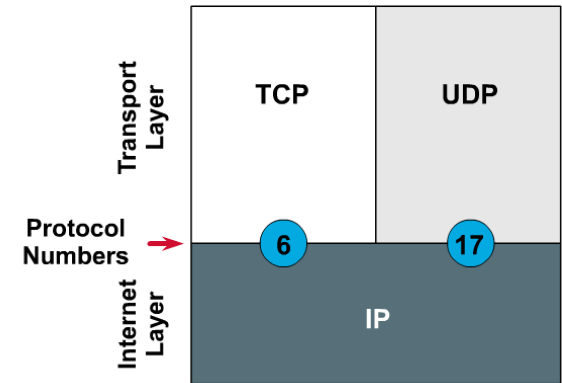
- Transfer data tergantung pada Protokol Layer Transport yang digunakan, bisa menggunakan UDP dan TCP.
- Aplikasi yang menggunakan Protokol UDP:
 - Dynamic Host Configuration Protocol (DHCP)
 - Simple Network Management Protocol (SNMP)
 - Trivial File Transfer Protocol (TFTP)
- Aplikasi yang menggunakan Protokol TCP dalam melakukan transmisi data:
 - TELNET
 - HTTP
 - Simple Mail Transfer Protocol (SMTP)
 - And the File Transfer Protocol (FTP)
- DNS menggunakan keduanya UDP dan TCP

Hubungan Aplikasi dan Transport

Port Numbers



The Protocol Field



Intro to DNS (Domain Name System)

- Merupakan sistem berbentuk database terdistribusi
- Memetakan/mengkonversikan nama host/mesin/domain ke alamat IP (Internet Protocol) dan sebaliknya dari alamat IP ke nama host yang disebut dengan reverse-mapping.

- Untuk memetakan nama mesin misal www.eepis-its.edu ke alamat IP misal 202.154.187.7
- Untuk routing e-mail, telnet, ftp, web, dan lain-lain.

Intro to DNS

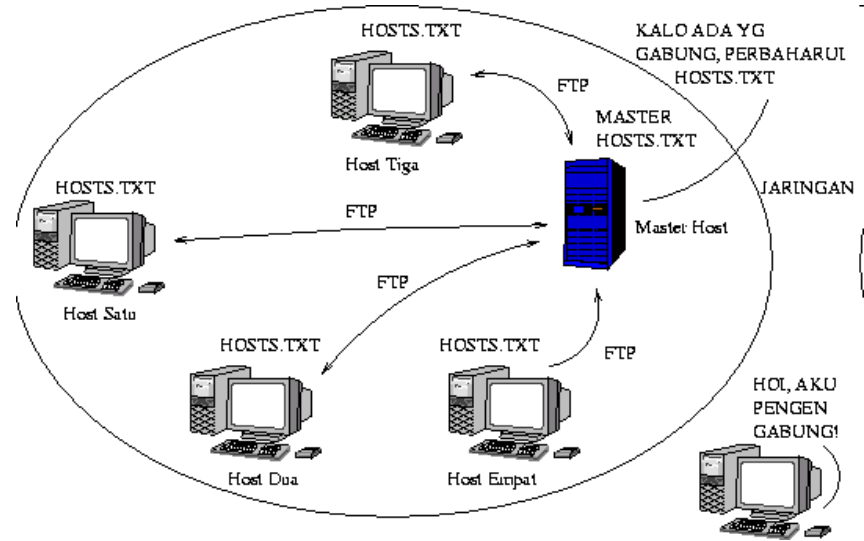
- DNS sebagai jembatan

- Manusia lebih mudah untuk mengingat nama daripada alamat IP dengan panjang 32 bit itu.
- Komputer menggunakan alamat IP untuk berkomunikasi dan berinteraksi.

- DNS tidak diperlukan jika kita bisa mengingat ratusan, ribuan, bahkan jutaan alamat IP di Internet.

History

- Sebelum adanya DNS, tahun 1970-an ARPAnet menggunakan pemetaan dengan bentuk tabel host pada berkas HOSTS.TXT
- HOSTS.TXT berisi nama host dan alamat IP serta pemetaannya dari seluruh mesin/komputer yang terhubung dalam jaringan.
- Ketika ada komputer lain yang terhubung ke jaringan ARPAnet maka masing-masing komputer dalam jaringan tersebut harus memperbaharui berkas HOSTS.TXT-nya.
- Cara meng-update berkas HOSTS.TXT dengan menggunakan ftp setiap satu atau dua minggu sekali.
- Masalah ketika jaringan menjadi semakin besar. Kesulitan meng-update isi berkas HOSTS.TXT karena jumlah nama mesin/komputer yang dituliskan sudah terlalu besar dan tidak efisien.



History

Muncul ide untuk membuat sistem database terdistribusi yang mempunyai data mengenai pemetaan nama host ke alamat IP dan sebaliknya.

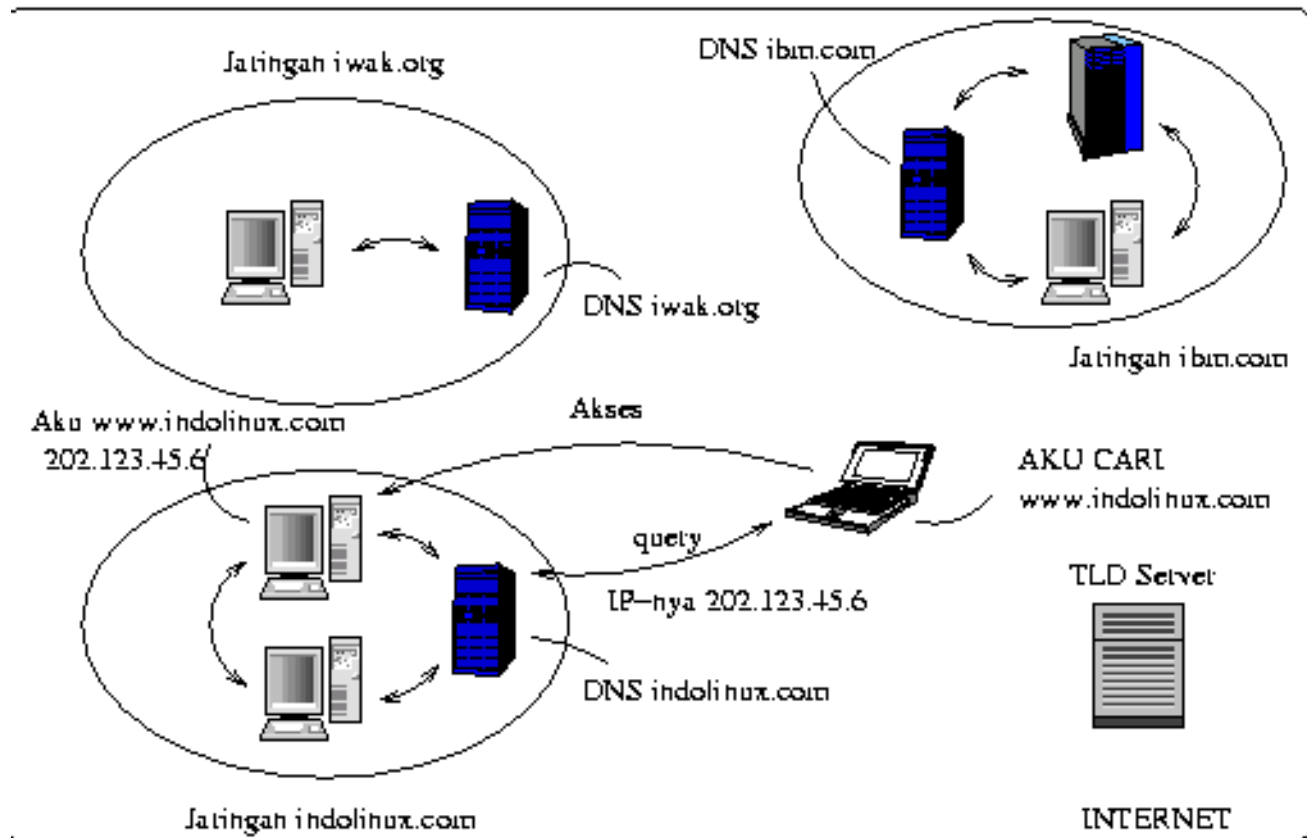
Dengan adanya pendistribusian database nama host dan alamat IP, maka tiap organisasi yang memiliki jaringan di dalam domain tertentu hanya bertanggung jawab terhadap database informasi pemetaan nama host dan alamat IP pada jaringannya saja yang biasa disebut zone.

Administrasi domain tersebut dilakukan secara lokal tetapi informasi itu dapat diakses oleh semua komputer di Internet.

Karena sifat database yang terdistribusi ini, maka dibutuhkan suatu mekanisme pengaksesan informasi bagi host lain pada database yang terdistribusi untuk menemukan informasi host atau jaringan yang dipunyai oleh suatu organisasi.

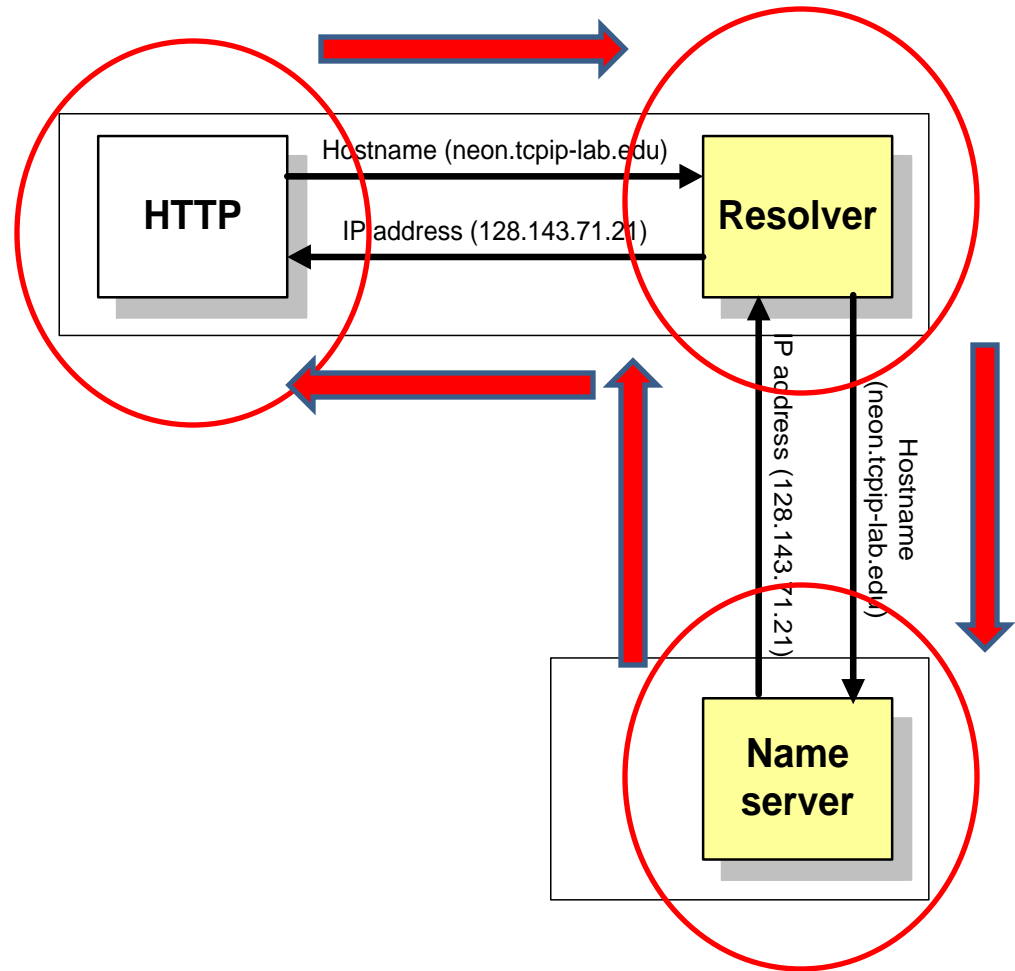
Dan pada tahun 1984, Paul Mockapetris mengusulkan sistem database terdistribusi ini dengan Domain Name System (DNS) yang dideskripsikan dalam RFC 882 dan 883. Sistem ini digunakan sampai sekarang pada jaringan khususnya Internet.

History



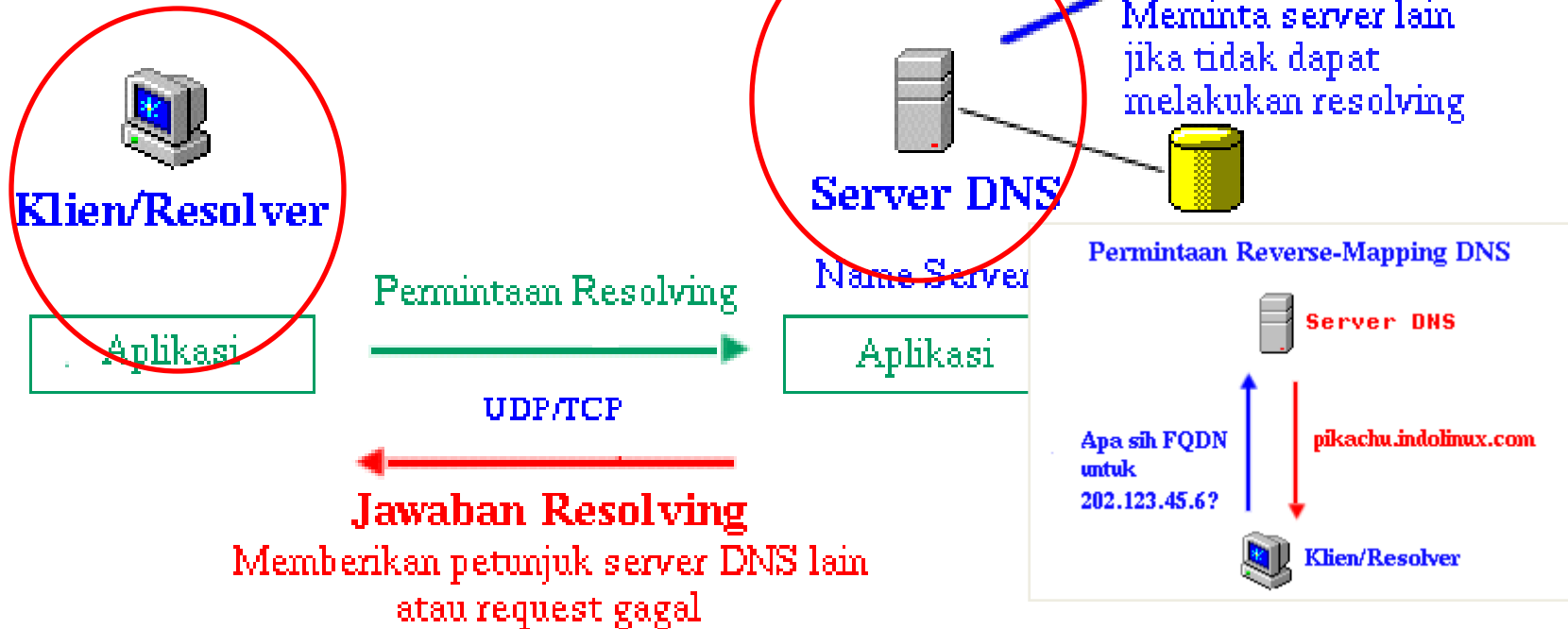
Resolver and name server

1. Sebuah program aplikasi pada host yang mengakses domain system disebut sebagai **resolver**
 2. Resolver mengontak DNS server, yang biasa disebut name server
 3. DNS server mengembalikan IP address ke resolver yang meneruskan ke aplikasi yang membutuhkan IP address
- ✘ Reverse lookups are also possible, i.e., find the hostname given an IP address



Kerja DNS

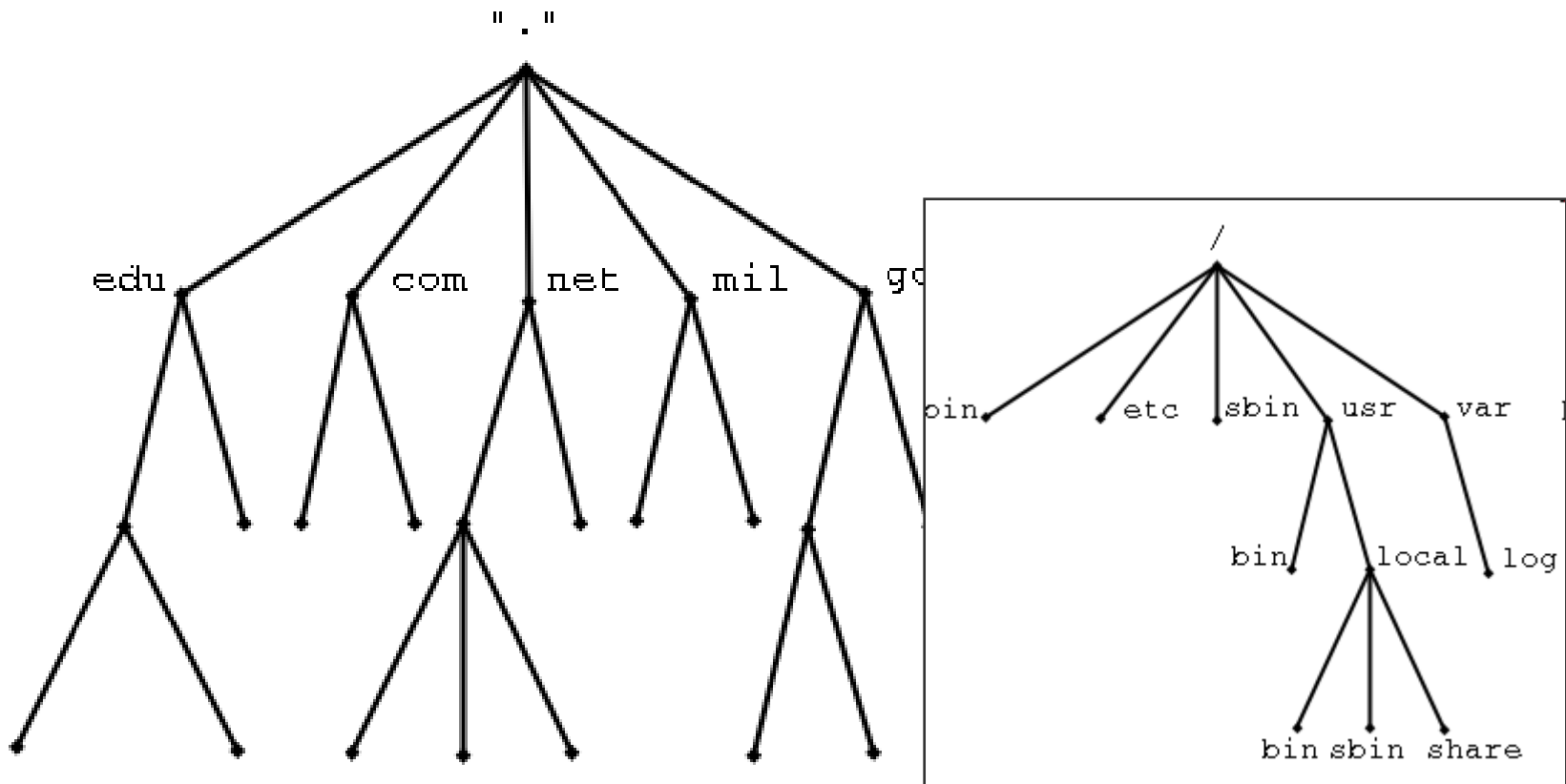
Bagaimana DNS Bekerja



Struktur

- Struktur database DNS 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 (.).

Struktur



Struktur

- Contoh: `www.its.ac.id`.
- "." merupakan root domain
- `id` merupakan Top Level Domain
- `ac` merupakan Second Level Domain
- `its` merupakan Third Level Domain
- `www` merupakan nama komputer/mesin yang bersangkutan
- Sistem penulisan nama secara absolut dan lengkap ini disebut FQDN (Fully Qualified Domain Name) - `www.its.ac.id`.

Hirarki

- Tiap organisasi yang telah mendaftar ke Network Information Center(NIC) akan mendapatkan nama domain sesuai dengan organisasi tersebut.
- Nama domain tersebut bisa dibagi menjadi subdomain sesuai kebutuhan organisasi.
 - Contoh: InterNIC mempunyai semua Top Level Domain termasuk edu,
 - Lembaga pendidikan PENS akan mendaftarkan nama domain eepis-its.edu (education), maka PENS diberikan/didelegasikan oleh InterNIC untuk mengelola domain eepis-its.edu yang merupakan sub domain dari edu.
 - PENS dapat membagi lagi domain eepis-its.edu ke beberapa sub domain misal `www.eepis-its.edu` , `ies.eepis-its.edu`, `eis.eepis-its.edu`, `elearning.eepis-its.edu`.

Hirarki

- Dengan adanya sistem berbentuk hierarki/pohon ini maka tidak ada nama host yang sama pada domain/subdomain yang sama, karena masing-masing dari node/titik-cabang mempunyai nama unik dan tidak boleh ada yang menyamainya kecuali berbeda sub-tree/sub pohon.
- Tidak akan ada konflik antar organisasi karena masing-masing organisasi mempunyai domain yang berbeda-beda dan ini diatur oleh InterNIC untuk TLD (Top Level Domain).
- Kedalaman pohon dibatasi sampai level 127

Top Level Domain (TLD)

- Domain Generik
 - com , net , gov , mil , org , edu , int
 - Selain 7 domain di atas ada lagi 7 domain baru dari ICANN (www.icann.org) yaitu: aero, biz , coop , info , museum , name , pro
- Domain 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.
- Domain Arpa
 - Merupakan domain untuk jaringan ARPAnet. Tiap domain yang tergabung ke Internet berhak memiliki name-space .in-addr.arpa sesuai dengan alamat IP-nya.

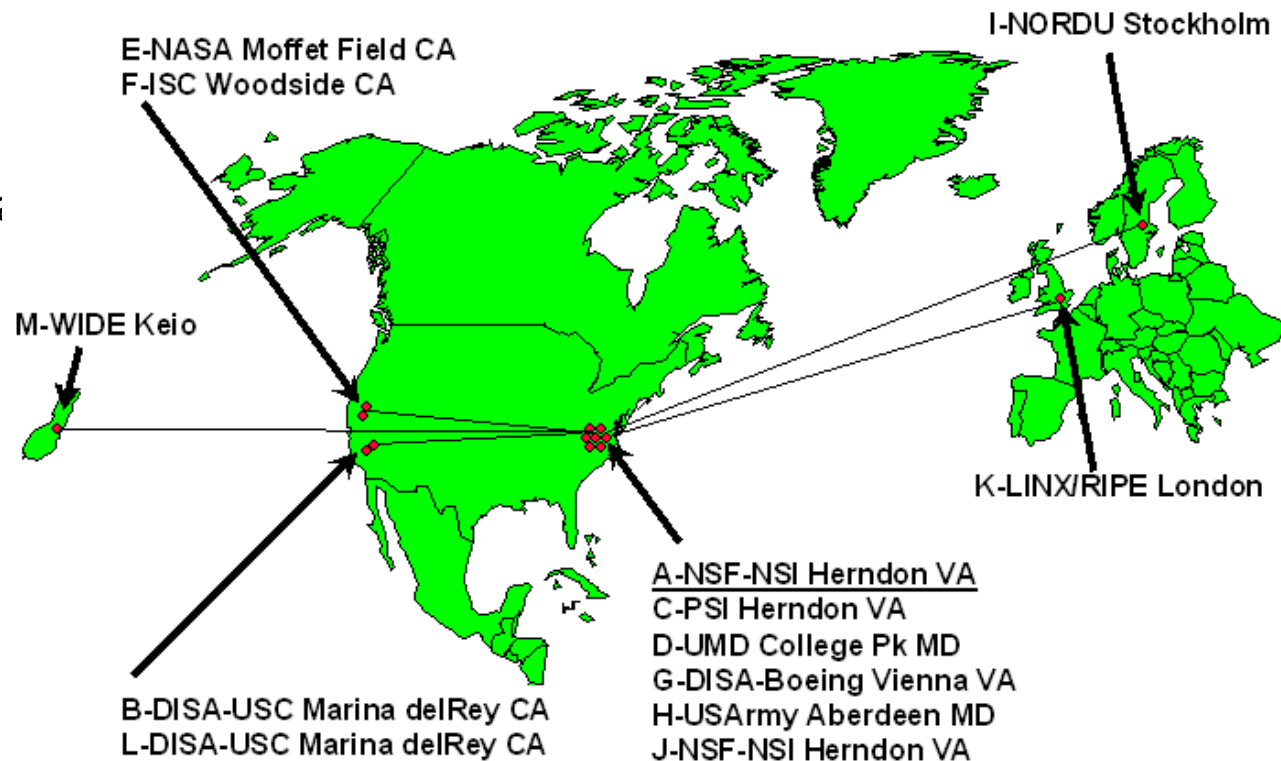
Root name servers

DNS Root Servers

1 Feb 98

Designation, Responsibility, and Locations

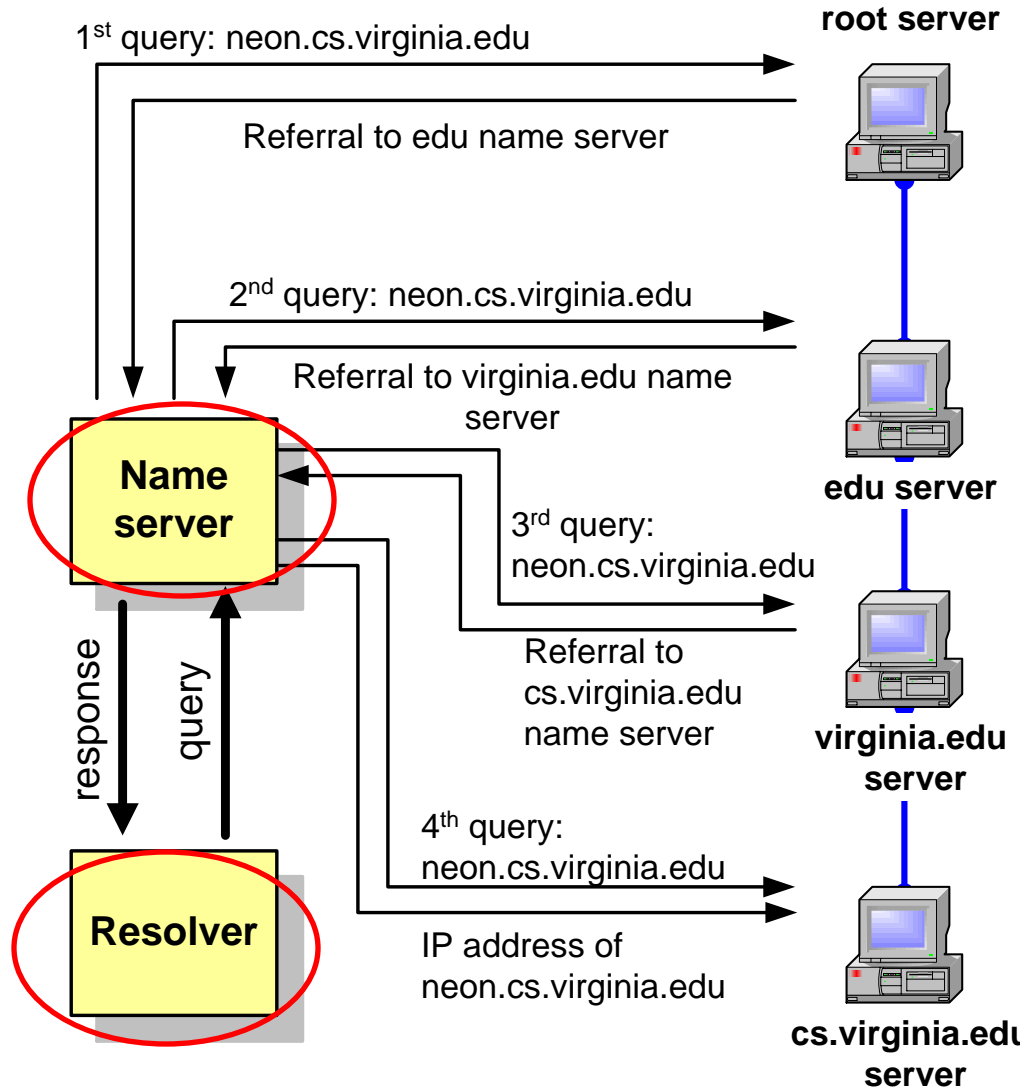
- Server root digunakan untuk menemukan authoritative name servers untuk semua zona top-level.
- Ada 13 server root
- Digunakan untuk name resolution



Address root servers (2004)

A.ROOT-SERVERS.NET.	(VeriSign, Dulles, VA)	198.41.0.4
B.ROOT-SERVERS.NET.	(ISI, Marina Del Rey CA)	192.228.79.201
C.ROOT-SERVERS.NET.	(Cogent Communications)	192.33.4.12
D.ROOT-SERVERS.NET.	(University of Maryland)	128.8.10.90
E.ROOT-SERVERS.NET.	(Nasa Ames Research Center)	192.203.230.10
F.ROOT-SERVERS.NET.	(Internet Systems Consortium)	192.5.5.241
G.ROOT-SERVERS.NET.	(US Department of Defense)	192.112.36.4
H.ROOT-SERVERS.NET.	(US Army Research Lab)	128.63.2.53
I.ROOT-SERVERS.NET.	(Autonomica/NORDUnet)	192.36.148.17
J.ROOT-SERVERS.NET.	(Verisign, multiple cities)	192.58.128.30
K.ROOT-SERVERS.NET.	(RIPE, Europe multiple cities)	193.0.14.129
L.ROOT-SERVERS.NET.	(IANA, Los Angeles)	198.32.64.12
M.ROOT-SERVERS.NET.	(WIDE, Tokyo, Seoul, Paris)	202.12.27.33

Recursive queries



Caching

- Penyimpanan informasi mapping antara IP dan name server untuk mengurangi traffic
- Ketika ada permintaan/ query server tidak perlu lagi menghubungi server lain

Resource Records

- Record database pada Database DNS terdistribusi disebut **resource records (RR)**
- Resource records disimpan pada file konfigurasi (zone files) pada name servers.

Berikut ini contoh sebuah zone Resource record →

```
db.mylab.com

$TTL 86400
mylab.com. IN SOA PC4.mylab.com.
                               hostmaster.mylab.com. (
                               1 ; serial
                               28800 ; refresh
                               7200 ; retry
                               604800 ; expire
                               86400 ; ttl
                               )

;
mylab.com. IN NS PC4.mylab.com.
;
localhost A 127.0.0.1
PC4.mylab.com. A 10.0.1.41
PC3.mylab.com. A 10.0.1.31
PC2.mylab.com. A 10.0.1.21
PC1.mylab.com. A 10.0.1.11
```

Resource Records

db.mylab.com

```
$TTL 86400
mylab.com. IN SOA PC4.mylab.com.
hostmaster@mylab.com. (
    1 ; serial
    28800 ; refresh
    7200 ; retry
    604800 ; expire
    86400 ; ttl
)
;
mylab.com.      IN      NS      PC4.mylab.com.
;
localhost      A        127.0.0.1
PC4.mylab.com. A        10.0.1.41
PC3.mylab.com. A        10.0.1.31
PC2.mylab.com. A        10.0.1.21
PC1.mylab.com. A        10.0.1.11
```

Maksimum umur data cache dalam detik

Record Start of authority (SOA) arti : “Zona authoritative Name server-nya Mylab.com”

- PC4.mylab.com adalah name server
- Email adress PICnya hostmaster@mylab.com

Record Name server (NS).

Address (A) records. Satu entry untuk setiap hostaddress

Blok dalam named.conf

- **options** — List konfigurasi global dan default
- **include** — berisi path file lain yang diperlukan
- **acl** — IP address dalam access control list
- **server** — properties khusus untuk remote servers
- **zone** — informasi khusus untuk zona

- `// generated by named-bootconf.pl`

1. Directory untuk
menempatkan file zone

- `options {`

- `directory "/var/named";`

- `/*`

- `* If there is a firewall between you and nameservers you want`

- `* to talk to, you might need to uncomment the query-source`

- `* directive below. Previous versions of BIND always asked`

- `* questions using port 53, but BIND 8.1 uses an unprivileged`

- `* port by default.`

- `*/`

- `// query-source address * port 53;`

- `};`

- `//`

- `// a caching only nameserver config`

- `//`

- `controls {`

- `inet 127.0.0.1 allow { localhost; };`

- `};`

2. Blok untuk mengatur akses

3. Zone untuk root

- zone "." IN {
- type hint;
- file "named.ca";
- };

4. Zone untuk localhost

- zone "localhost" IN {
- type master;
- file "localhost.zone";
- allow-update { none; };
- };

5. Zone untuk reverse address

- zone "0.0.127.in-addr.arpa" IN {
- type master;
- file "named.local";
- allow-update { none; };
- };

Resource Record

- **NS — NAME SERVERS**
 - Menunjukkan nama “name server”.
- **A — THE IP ADDRESS FOR THE NAME**
 - Menunjukkan nomor IP “name server”.
- **PTR — POINTER FOR ADDRESS NAME MAPPING**
 - Digunakan untuk menunjuk name server
- **CNAME — CANONICAL NAME**
 - Menunjukkan nama real host.
- **MX — MAIL EXCHANGE RECORD**
 - Menunjukkan sebagai mail server pada domain tersebut.

Dynamic DNS

- Suatu cara melakukan update DNS server tanpa harus melakukan restart terhadap konfigurasi DNS kita.
- Pada waktu konfigurasi DNS harus ada cara untuk mengupdate, Pada waktu suatu host hidup kita bisa menyediakan address via DHCP, kemudian DHCP meminta DNS untuk merubah record A dan PTR sesuai kebutuhan.
- Kolaborasi antara DNS dan DHCP
- Membutuhkan bind9 dan DHCP3
- Konfigurasi file utama : dhcpd.conf dan named.conf
- Dijelaskan lebih lanjut pada Bagian DHCP Server

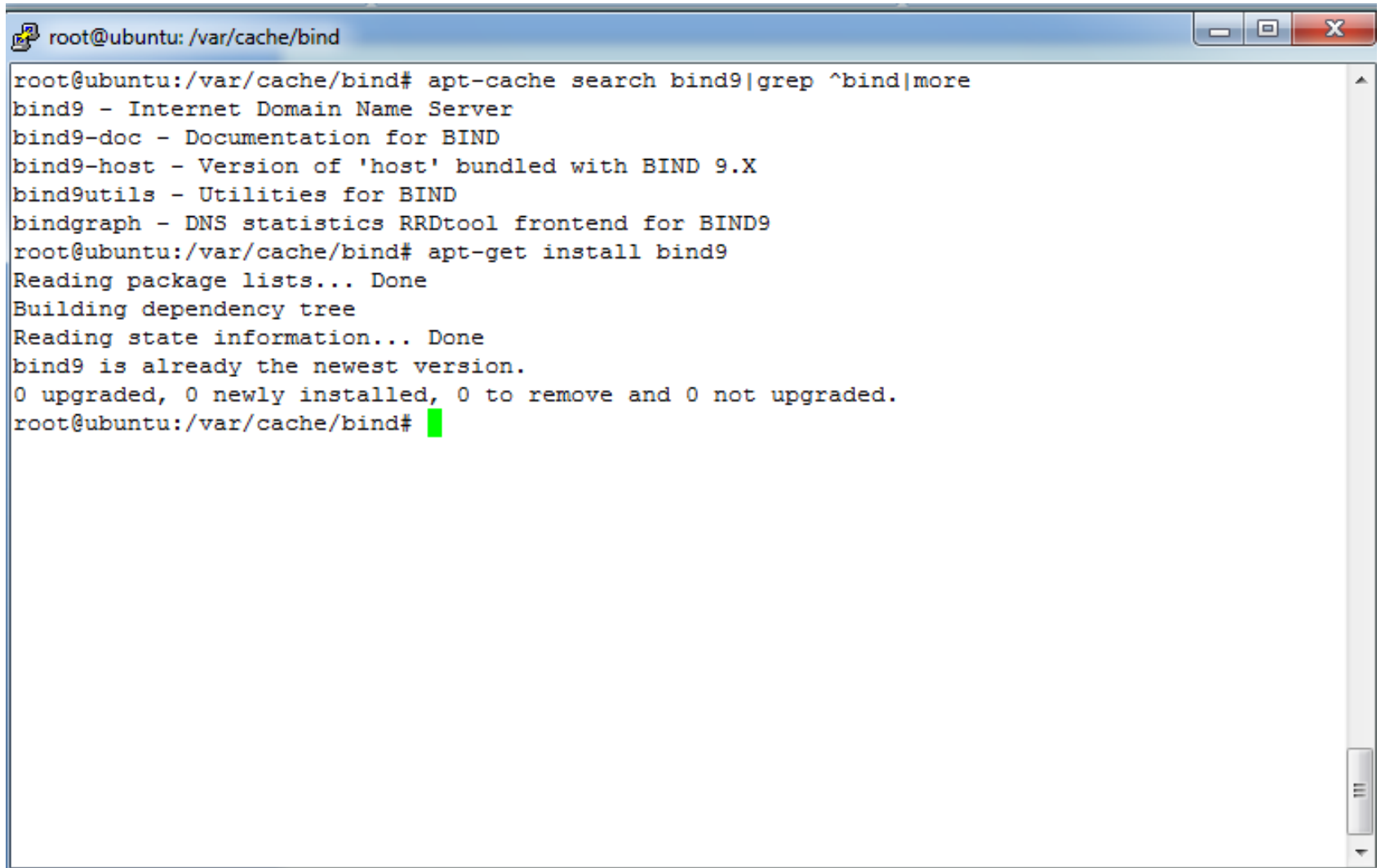
Praktikum DNS

1. Meminta IP otomatis dari server

```
root@ubuntu: /var/cache/bind
root@ubuntu:/var/cache/bind# dhclient
There is already a pid file /var/run/dhclient.pid with pid 8508
killed old client process, removed PID file
Internet Systems Consortium DHCP Client V3.1.1
Copyright 2004-2008 Internet Systems Consortium.
All rights reserved.
For info, please visit http://www.isc.org/sw/dhcp/

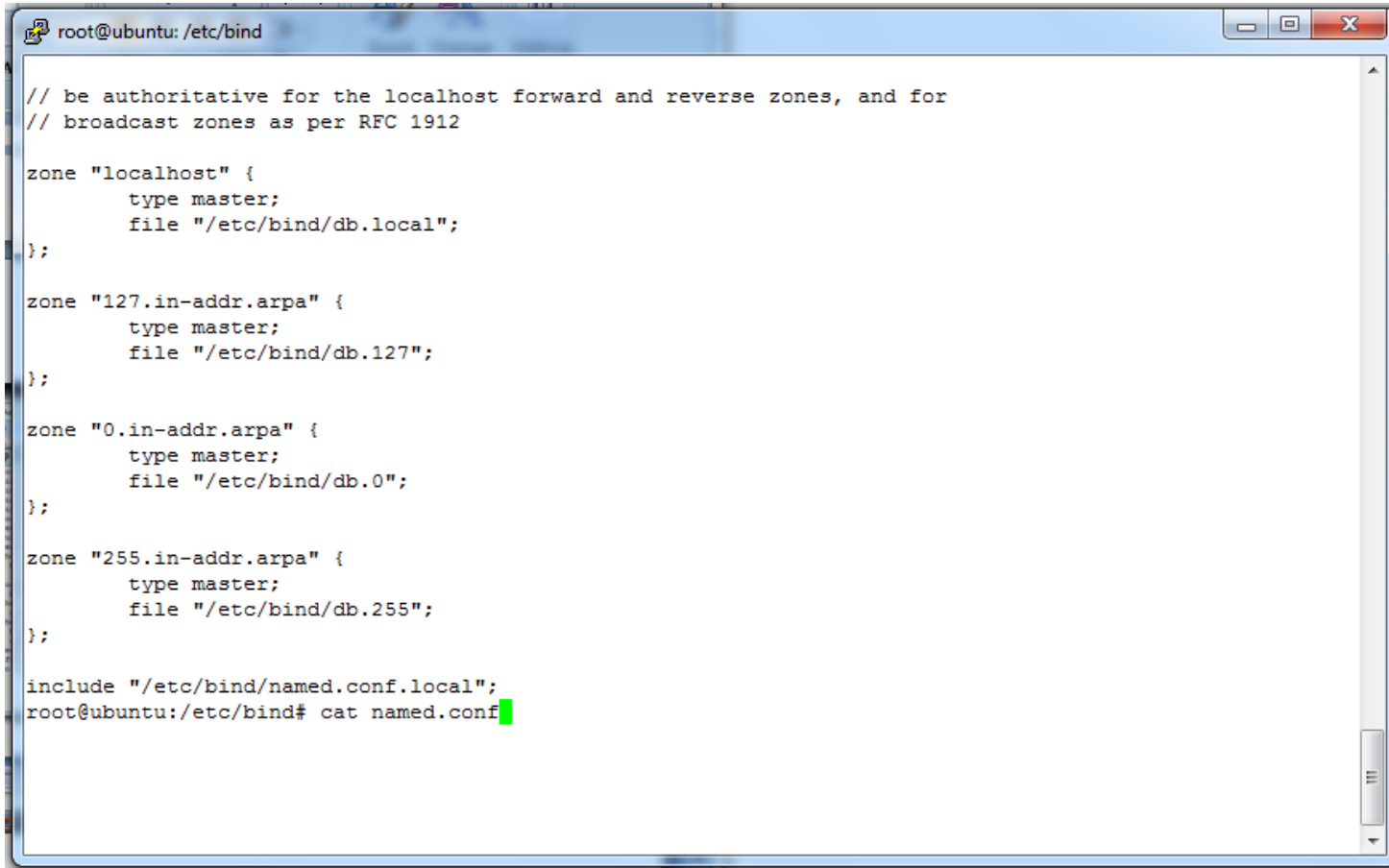
Listening on LPF/pan0/1a:ad:94:c2:62:a5
Sending on LPF/pan0/1a:ad:94:c2:62:a5
Listening on LPF/eth0/00:0c:29:4e:0b:db
Sending on LPF/eth0/00:0c:29:4e:0b:db
Sending on Socket/fallback
DHCPCREQUEST of 10.252.12.244 on eth0 to 255.255.255.255 port 67
DHCPACK of 10.252.12.244 from 10.252.12.1
 * Reloading /etc/samba/smb.conf smbd only
  ...done.
bound to 10.252.12.244 -- renewal in 1598 seconds.
root@ubuntu:/var/cache/bind#
```

2. Mencari dan menginstall paket DNS : bind9



```
root@ubuntu: /var/cache/bind
root@ubuntu:/var/cache/bind# apt-cache search bind9|grep ^bind|more
bind9 - Internet Domain Name Server
bind9-doc - Documentation for BIND
bind9-host - Version of 'host' bundled with BIND 9.X
bind9utils - Utilities for BIND
bindgraph - DNS statistics RRDtool frontend for BIND9
root@ubuntu:/var/cache/bind# 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:/var/cache/bind#
```

3. Membaca file named.conf

A terminal window titled 'root@ubuntu: /etc/bind' showing the contents of the named.conf file. The window has a blue title bar and standard Ubuntu window controls (minimize, maximize, close). The text is as follows:

```
root@ubuntu: /etc/bind
// be authoritative for the localhost forward and reverse zones, and for
// broadcast zones as per RFC 1912

zone "localhost" {
    type master;
    file "/etc/bind/db.local";
};

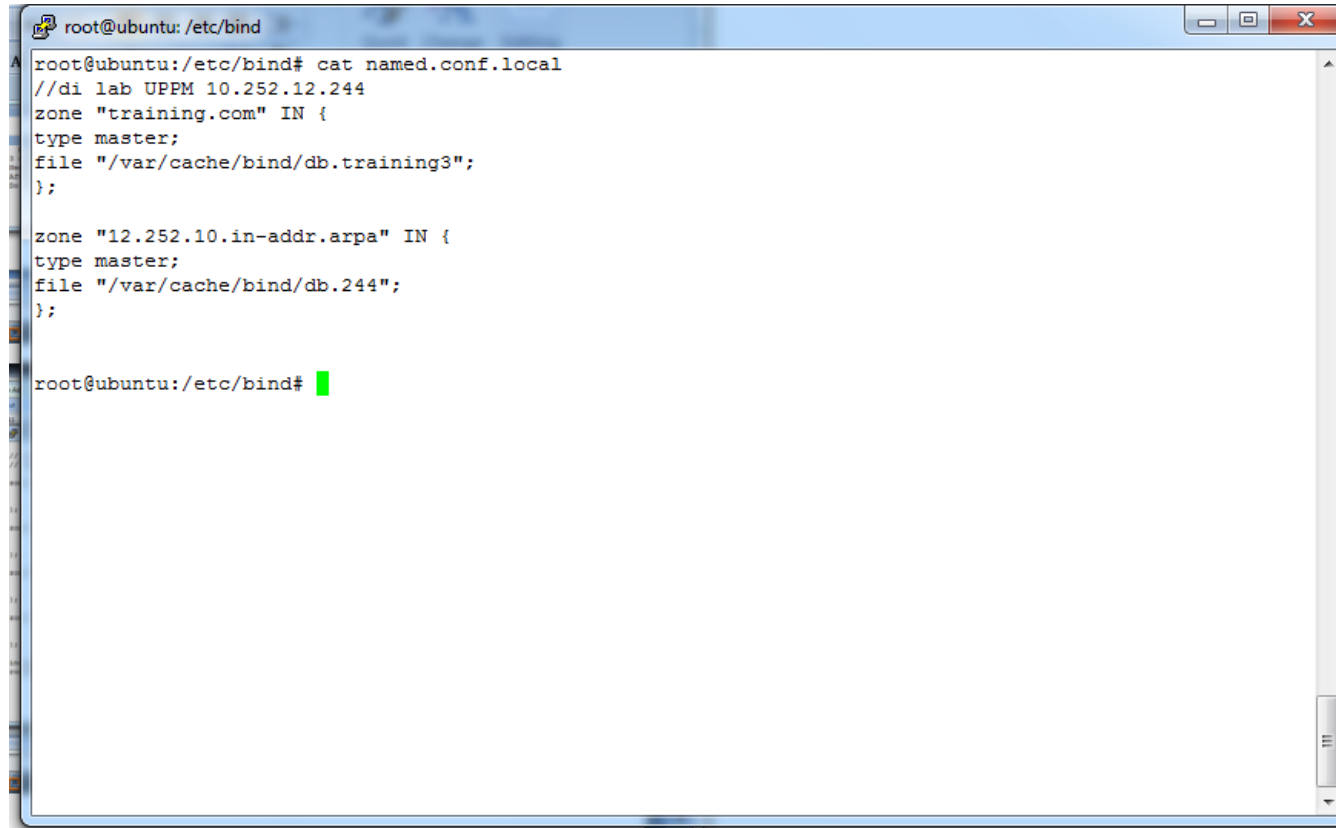
zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};

zone "0.in-addr.arpa" {
    type master;
    file "/etc/bind/db.0";
};

zone "255.in-addr.arpa" {
    type master;
    file "/etc/bind/db.255";
};

include "/etc/bind/named.conf.local";
root@ubuntu:/etc/bind# cat named.conf
```


4. Membuat file named.conf.local

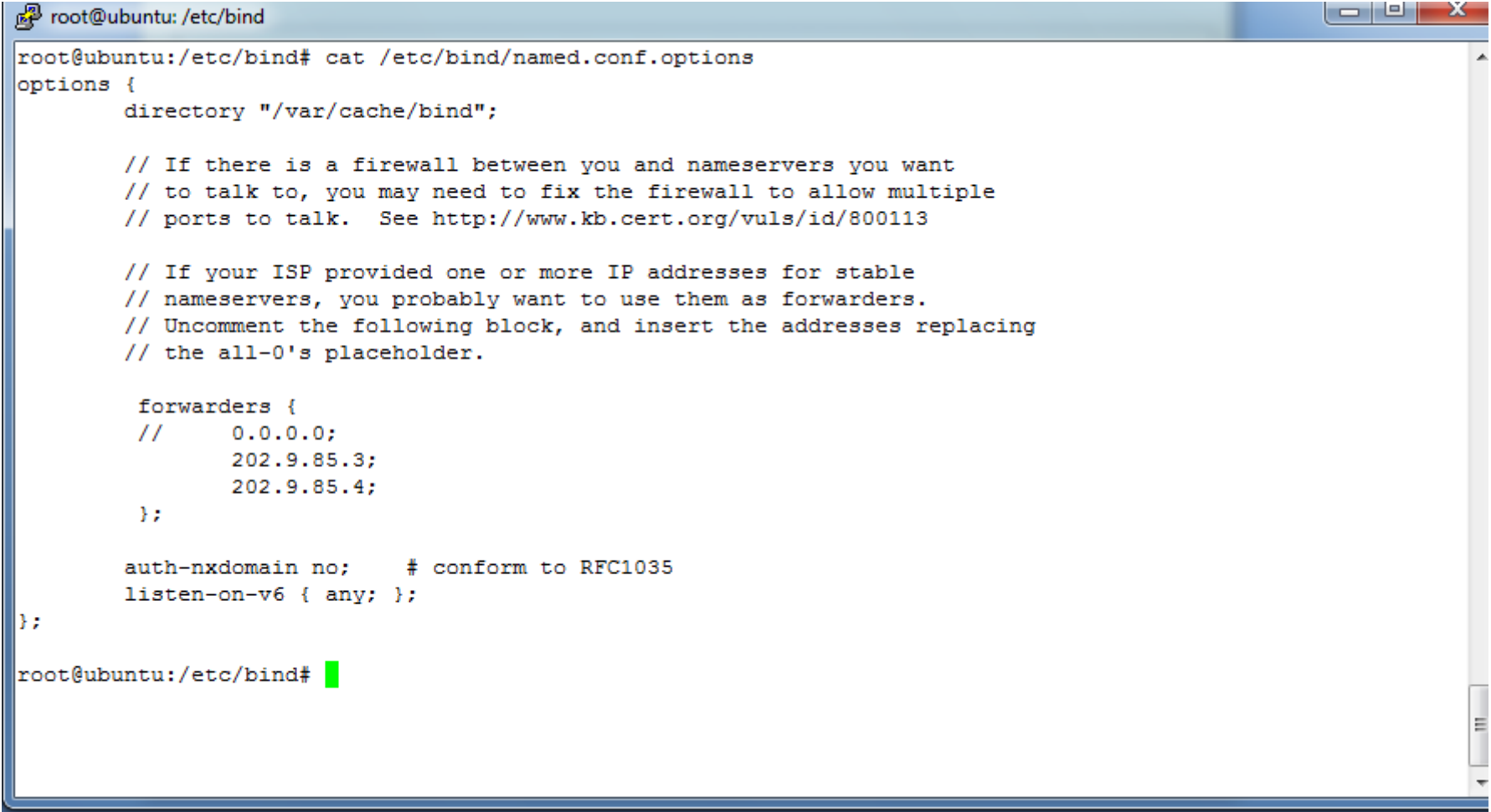
A terminal window titled 'root@ubuntu: /etc/bind' showing the output of the 'cat named.conf.local' command. The output lists two DNS zones: 'training.com' and '12.252.10.in-addr.arpa', both configured as master zones with specific database files.

```
root@ubuntu: /etc/bind# cat named.conf.local
//di lab UPPM 10.252.12.244
zone "training.com" IN {
type master;
file "/var/cache/bind/db.training3";
};

zone "12.252.10.in-addr.arpa" IN {
type master;
file "/var/cache/bind/db.244";
};

root@ubuntu: /etc/bind#
```

5. Membuat file named.conf.local

A terminal window titled 'root@ubuntu: /etc/bind' showing the output of the command 'cat /etc/bind/named.conf.options'. The output displays the configuration for the named.conf.options file, including options for directory, forwarders, and listen-on-v6.

```
root@ubuntu: /etc/bind# cat /etc/bind/named.conf.options
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 {
        //      0.0.0.0;
        202.9.85.3;
        202.9.85.4;
    };

    auth-nxdomain no;          # conform to RFC1035
    listen-on-v6 { any; };
};

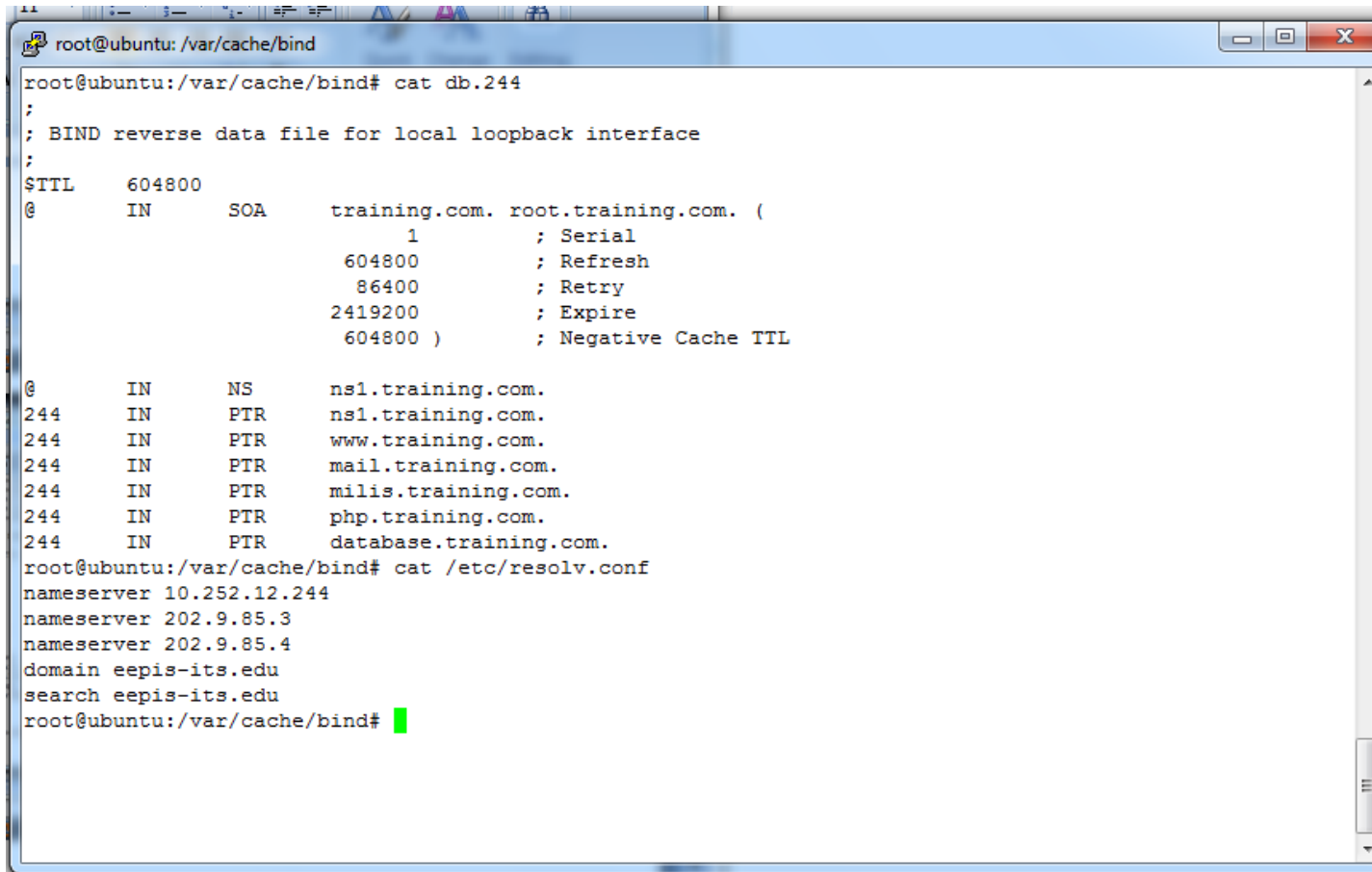
root@ubuntu: /etc/bind# █
```

6. Membuat file db.training3

```
root@ubuntu: /var/cache/bind
root@ubuntu:/etc/bind# cd /var/cache/bind/
root@ubuntu:/var/cache/bind# ls
db.179 db.231 db.57 db.72 db.training1 db.training3
db.183 db.244 db.69 db.training db.training2
root@ubuntu:/var/cache/bind# cat db.training3
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA     training.com. root.training.com. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS      ns1.training.com.
@         IN      MX      10      mail.training.com.
;
www       IN      A        10.252.12.244
mail      IN      A        10.252.12.244
milis     IN      A        10.252.12.244
php       IN      A        10.252.12.244
database  IN      A        10.252.12.244
;

root@ubuntu:/var/cache/bind# █
```

7. Membuat file db.244 dan edit file resolv.conf



```
root@ubuntu: /var/cache/bind
root@ubuntu: /var/cache/bind# cat db.244
;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@        IN      SOA      training.com. root.training.com. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL

@        IN      NS       ns1.training.com.
244     IN      PTR      ns1.training.com.
244     IN      PTR      www.training.com.
244     IN      PTR      mail.training.com.
244     IN      PTR      milis.training.com.
244     IN      PTR      php.training.com.
244     IN      PTR      database.training.com.
root@ubuntu: /var/cache/bind# cat /etc/resolv.conf
nameserver 10.252.12.244
nameserver 202.9.85.3
nameserver 202.9.85.4
domain eepis-its.edu
search eepis-its.edu
root@ubuntu: /var/cache/bind#
```

8. Testing DNS

```
root@ubuntu: /var/cache/bind
root@ubuntu:/var/cache/bind# nslookup 10.252.12.244
Server:          10.252.12.244
Address:         10.252.12.244#53

244.12.252.10.in-addr.arpa    name = mail.training.com.
244.12.252.10.in-addr.arpa    name = milis.training.com.
244.12.252.10.in-addr.arpa    name = database.training.com.
244.12.252.10.in-addr.arpa    name = ns1.training.com.
244.12.252.10.in-addr.arpa    name = php.training.com.
244.12.252.10.in-addr.arpa    name = www.training.com.

root@ubuntu:/var/cache/bind# nslookup www.training.com
Server:          10.252.12.244
Address:         10.252.12.244#53

Name:   www.training.com
Address: 10.252.12.244

root@ubuntu:/var/cache/bind# nslookup mail.training.com
Server:          10.252.12.244
Address:         10.252.12.244#53

Name:   mail.training.com
Address: 10.252.12.244

root@ubuntu:/var/cache/bind# █
```

9. Testing DNS

```
root@ubuntu: /var/cache/bind
root@ubuntu: /var/cache/bind# dig 10.252.12.244

; <<>> DiG 9.5.1-P2 <<>> 10.252.12.244
;; global options: printcmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 27360
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 14

;; QUESTION SECTION:
;10.252.12.244.                IN      A

;; ANSWER SECTION:
10.252.12.244.                0      IN      A      67.215.65.132

;; AUTHORITY SECTION:
.                289251 IN      NS      a.root-servers.net.
.                289251 IN      NS      k.root-servers.net.
.                289251 IN      NS      m.root-servers.net.
.                289251 IN      NS      d.root-servers.net.
.                289251 IN      NS      e.root-servers.net.
.                289251 IN      NS      g.root-servers.net.
.                289251 IN      NS      i.root-servers.net.
.                289251 IN      NS      j.root-servers.net.
.                289251 IN      NS      l.root-servers.net.
.                289251 IN      NS      f.root-servers.net.
.                289251 IN      NS      b.root-servers.net.
.                289251 IN      NS      h.root-servers.net.
.                289251 IN      NS      c.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net.          48650  IN      A      198.41.0.4
```

Latihan Soal

1. Sebutkan fungsi dari DNS !
2. Jelaskan dengan kata-kata sendiri cara kerja DNS !
3. Jelaskan dengan kata-kata sendiri bagaimana tahapan menginstall DNS dan beri keterangan !