

Membangun Lokal DNS Server Menggunakan BIND

Domain Name Server atau yang disingkat dengan **DNS** adalah sebuah sistem yang menyimpan informasi tentang nama *host* maupun nama domain dalam bentuk data tersebar (*distributed database*) di dalam jaringan komputer. Dengan DNS, suatu alamat IP dapat dialamatkan ke dalam nama *host* dan nama domain yang lebih mudah diingat manusia.

Coba bayangkan, jika Anda harus membuka suatu situs (google misalnya) dengan menghafalkan alamat IP-nya adalah 66.249.89.99 untuk membuka situs itu. Tentu akan lebih mudah bagi kita untuk menghafal dengan nama *www.google.com* daripada 66.249.89.99.

Dahulu sebelum diperkenalkannya DNS sebagai suatu system untuk pengalamatan alamat IP ke dalam suatu nama domain, setiap komputer di jaringan komputer menggunakan file HOST.TXT untuk memetakan alamat IP ke nama domain. Namun setelah pengguna Internet semakin banyak, penggunaan file HOST.TXT ini tidak lagi efektif, karena setiap pengguna harus selalu meng-update file HOST.TXT setiap ada penambahan atau perubahan nama domain baru di dalam jaringan.

Berangkat dari problem itu, maka dibutuhkan sebuah sistem yang bisa mengganti alamat *host* hanya di satu tempat, *host* lain akan mempelajari perubahan tersebut secara dinamis. Dan sistem inilah yang kemudian dikenal dengan nama DNS.

DNS ditemukan oleh Paul Mockapetris pada tahun 1983. Spesifikasi asli muncul di RFC 882 dan 883. Tahun 1987, penerbitan RFC 1034 dan RFC 1035 membuat update terhadap spesifikasi DNS. Hal ini membuat RFC 882 dan RFC 883 tidak berlaku lagi. Beberapa RFC terkini telah memproporsikan beberapa tambahan dari protokol inti DNS.

Pada Artikel ini, penulis akan menjelaskan prinsip kerja dari DNS. Untuk mempermudah, penulis akan menjelaskan cara *setting* DNS Server untuk jaringan lokal.

Instalasi BIND

Sebelum dapat memulai konfigurasi DNS, Anda harus menginstalasikan paket BIND terlebih dahulu. Biasanya paket BIND sudah dipaketkan pada setiap distro Linux. Pada saat penulisan artikel ini, penulis mencoba pada distro Fedora Core 4. Untuk pengguna distro lainnya, Anda tinggal menyesuaikan langkah instalasi BIND sesuai dengan distro yang Anda gunakan.

Untuk menginstalasikan paket BIND di Fedora Core 4, Anda dapat menginstalasi-nya dengan menggunakan *Add/Remove Applications*, lalu klik *Detail* pada bagian DNS Name Server. Cek semua paket yang ada pada group tersebut, *Close*, lalu klik *Update*. Cara lainnya Anda dapat menginstalasikan paket-rpm BIND yang ada pada CD Fedora Core dengan menggunakan perintah RPM.

Jika Anda ingin menginstalasikan dari Source Code, *download* terlebih dahulu paket BIND dari situs <http://www.isc.org>. Versi BIND terakhir yang bisa di-*download* dari situs tersebut adalah BIND 9.3.1.

```
$ wget -c ftp://ftp.isc.org/isc/bind9/9.3.1/bind-9.3.1.tar.gz
```

Ekstrak file source tersebut, dan lakukan kompilasi

```
# tar -xzf bind-9.3.1.tar.gz
# cd bind-9.3.1
# ./configure --prefix=/usr/local/named
# make
# make install
# adduser -d /var/named -s /bin/false named
```

Untuk kebanyakan pengguna Linux, paket BIND yang ada di dalam distro sudah cukup untuk memenuhi kebutuhan. Jadi kalau tidak terlalu dibutuhkan untuk instalasi lewat source, disarankan untuk menggunakan paket BIND yang sudah ada di distro saja.

Setting BIND

File-file yang harus diperhatikan untuk mengkonfigurasi BIND, antara lain:

1. */etc/resolv.conf*: (berisi alamat domain atau alamat IP dari name server).
2. */etc/named.conf* (berisi keterangan letak dan jenis database yang dibutuhkan oleh BIND).
3. */var/named/named.ca*: (berisi informasi data yang berada dalam domain root, yang akan dipergunakan name server jika ada resolver yang akan meminta nama domain diluar nama domain lokal).
4. */var/named/named.local*: (berisi alamat loopback untuk alamat ke diri sendiri dengan alamat 127.0.0.1).

Persiapan setting domain baru

Dalam studi kasus kali ini, kita akan membuat domain baru untuk DNS Server dan mail server, ftp server, http server, dengan data sebagai berikut:

Nama Domain	: example.com
Nama DNS Server	: ns.example.com
IP DNS Server	: 192.168.2.1
Nama Mail Server	: mail.example.com
Nama HTTP Server	: www.example.com
Nama FTP Server	: ftp.example.com

```

root@server1:~# rpm -qa | grep bind
bind-9.3.1-4
bind-utils-9.3.1-4
bind-libs-9.3.1-4
ypbind-1.17.2-5
bind-chroot-9.3.1-4
root@server1 ~#

```

Mendaftar paket-paket instalasi BIND.

```

root@server1:/var/named/chroot/var/named#
File Edit View Terminal Tabs Help
[root@server1 ~]# chmod 755 /var/named/
[root@server1 ~]# chmod 755 /var/named/chroot/
[root@server1 ~]# chmod 755 /var/named/chroot/var/
[root@server1 ~]# chmod 755 /var/named/chroot/var/named/
[root@server1 ~]# chmod 755 /var/named/chroot/var/run/
[root@server1 ~]# chmod 755 /var/named/chroot/var/run/named/
[root@server1 ~]# cd /var/named/chroot/var/named/
[root@server1 named]# ln -s ../../ chroot
[root@server1 named]# chkconfig --levels 235 named on
[root@server1 named]# /etc/init.d/named restart
Stopping named: [FAILED]
Starting named: [ OK ]
[root@server1 named]#

```

Langkah-langkah konfigurasi BIND-Chroot.

Karena kita menggunakan IP Address yang sama untuk DNS Server dan (Mail Server, FTP Server, HTTP Server), maka kita akan menggunakan record CNAME (nama alias) dengan mendefinisikan bahwa ns.example.com mempunyai nama alias (mail.example.com, ftp.example.com, dan www.example.com).

Persiapan setting domain baru

Agar komputer dapat mengenali nama domain example.com dengan IP Address 192.168.2.1, maka kita definisikan di file /etc/resolv.conf sebagai berikut:

```

search mailku.com
nameserver 192.168.2.1

```

Arti dari baris diatas bahwa resolver akan melakukan pencarian domain example.com dengan Name Server dengan IP Address 192.168.2.1.

Menjalankan BIND-Chroot

Untuk alasan keamanan, penulis juga akan menjelaskan langkah-langkah menjalankan BIND-chroot. Pertama-tama, periksa apakah paket bind-chroot sudah terinstalasi dengan baik di sistem Anda.

```

# rpm -qa | grep bind-chroot
bind-chroot-9.3.1-4

```

Selanjutnya ketikkan perintah-perintah berikut ini:

```

# /etc/init.d/named stop
# chmod 755 /var/named/
# chmod 775 /var/named/chroot/
# chmod 775 /var/named/chroot/var/
# chmod 755 /var/named/chroot/var/named/
# chmod 775 /var/named/chroot/var/run/

```

```

# chmod 777 /var/named/chroot/
# cd /var/named/chroot/var/
# ln -s ../../ chroot
# chkconfig --levels 235 named on
# /etc/init.d/named restart

```

Mengedit file named.conf

Sekarang kita akan mendefinisikan bahwa DNS Server ini akan menjadi “authoritative name server” untuk domain example.com. Selain itu kita juga akan mendefinisikan reverse DNS Zone example.com. Untuk melakukan hal tersebut, Anda dapat mengedit file /etc/named.conf seperti di bawah ini:

```

## Edit Ini Di Komputer
## 192.168.2.1 - Master DNS Server
## named.conf
# Let only the local machine control the server:
controls {
    inet 127.0.0.1 allow { localhost; } keys { rndckey; };
};

# Include other things into the configuration file, things that # automatic configuration tools that modify this file might make # a mess of: (Though, if you don't use such tools, you can do # all the configuration within this file.)

include "/etc/named.custom";

# Cryptographic key to allow certain things: (Refer to a

```

```

file
# with the information, here, and in other configuration files,
# rather than copy the information in that file into each
# configuration file.)

include "/etc/rndc.key";

# Maksud dari baris dibawah ini adalah, bahwa kita mendefinisikan
# DNS Server kita sebagai Primary Name Server untuk domain
# example.com dan file example.com.zone adalah nama file yang
# merupakan zone-zone file dari domain example.com

zone "example.com" {
    type master;
    file "example.com.zone";
    allow-update { key "rndckey"; };
    allow-transfer { 192.168.2/8; };
    notify yes;
};

# Baris dibawah ini adalah berisi Reverse DNS zone. Ini diperlukan
# agar DNS Server Anda dapat menerjemahkan dari nomor IP Address
# ke nama host pemilik IP Address dalam jaringan tersebut

zone "2.168.192.in-addr.arpa" {
    type master;

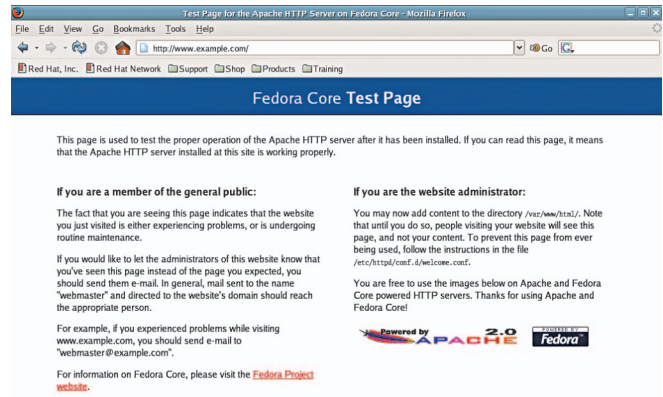
```

```

root@server1:~# host 192.168.2.1
1.2.168.192.in-addr.arpa domain name pointer ns.example.com.
root@server1:~# host ns.example.com
ns.example.com has address 192.168.2.1
root@server1:~# host -t ns example.com
example.com name server ns.example.com.
root@server1:~# host -t mx example.com
example.com mail is handled by 10 mail.example.com.example.com.
root@server1:~# host -t cname mail.example.com
mail.example.com is an alias for ns.example.com.
root@server1:~# host -t cname www.example.com
www.example.com is an alias for ns.example.com.
root@server1:~# host -t cname ftp.example.com
ftp.example.com is an alias for ns.example.com.
root@server1:~#

```

Testing DNS Server Menggunakan Perintah host



Test DNS Server untuk HTTP Server.

```

file "2.168.192.in-
addr.arpa.zone";
allow-update { key
"rndckey"; };
allow-transfer {
192.168.2/8; };
notify yes;
};

```

Setting zone-file example.com

Sekarang kita akan mengonfigurasi Primary Name Server untuk domain example.com pada host ns.example.com dengan IP Address 192.168.2.1. Berdasarkan pada file /etc/named.conf, kita akan membuat file /var/named/chroot/var/named/example.com.zone sebagai berikut:

```

$TTL 86400
@      IN      SOA
ns.example.com. root.example.
com. (
                                50 ;
serial
                                28800 ;
refresh (8 hours)
                                7200 ;
retry (2 hours)
                                604800 ;
retire (1 week)
                                86400 ;
ttl (1 day)
)

      IN      NS
ns.example.com.

      IN      MX      10
mail.example.com.

ns      IN      A
192.168.2.1

mail    IN      CNAME    ns

```

```

www      IN      CNAME    ns
ftp      IN      CNAME    ns

```

Maksud dari baris di atas adalah sebagai berikut:

- **TTL (Time To Live):** mendefinisikan waktu lamanya data berada dalam data-base.
- **SOA (Start Of Authority):** mendefinisikan hostname yang merupakan awal dari suatu zone.
- **ns.example.com:** merupakan hostname yang memegang tanggung jawab terhadap domain example.com.
- **root.example.com:** merupakan alamat e-mail administrator yang memegang tanggung jawab terhadap domain example.com.
- **50;serial:** merupakan nomor serial dari zone file yang akan bertambah jika ada perubahan data.
- **28800;refresh:** merupakan selang waktu yang diperlukan secondary name server untuk memeriksa perubahan pada Primary Name Server.
- **7200;retry:** merupakan selang waktu secondary name server untuk mengulang pengecekan pada primary name server.
- **604800;retire:** merupakan selang waktu zone file dipertahankan bila secondary name server tidak dapat melakukan pengecekan ke primary name server.
- **86400;ttl:** merupakan nilai default TTL untuk semua resource record pada zone-file.
- **IN NS ns.example.com:** mendefinisikan bahwa hostname ns.example.com yang memegang tanggung jawab terhadap domain example.com.
- **IN MX 10 mail.example.com:** mendefinisikan bahwa hostname mail.example.com sebagai Mail Server pada domain example.com.

com sebagai Mail Server pada domain example.com.

- **ns IN A 192.168.2.1:** mendefinisikan bahwa hostname ns.example.com mempunyai IP Address 192.168.2.1.
- **mail IN CNAME ns:** mendefinisikan bahwa hostname ns.example.com mempunyai nama alias mail.example.com.
- **www IN CNAME ns:** mendefinisikan bahwa hostname ns.example.com mempunyai nama alias www.example.com.
- **ftp IN CNAME ns:** mendefinisikan bahwa hostname ns.example.com mempunyai nama alias ftp.example.com.

Setting reverse DNS 2.168.192.in-addr.arpa.zone

Selanjutnya kita akan mengonfigurasi Primary Reverse DNS pada host ns.example.com dengan IP Address 192.168.2.1 untuk jaringan 192.168.2.0 yang telah didefinisikan dalam file /etc/named.conf dengan membuat zone-file /var/named/chroot/var/named/2.168.192.in-addr.arpa.zone sebagai berikut:

```

$TTL 86400
@      IN      SOA
ns.example.com. hostmaster.
example.com. (
                                50 ;
serial
                                28800 ;
refresh
                                7200 ;
retry
                                604800 ;
expire
                                86400 ;
ttl
)

```


	IN	NS
	ns.example.com.	

1	IN	PTR
	ns.example.com.	

Maksud dari baris di atas adalah sebagai berikut:

- Penjelasan yang lain, sama dengan penjelasan pada bagian example.com.zone.
- 1 IN PTR ns.example.com: mendefinisikan bahwa hostname ns.example.com mempunyai IP Address 192.168.2.1.

Testing DNS Server

Sekarang kita akan mencoba menjalankan BIND dengan menjalankan perintah:

```
# /etc/init.d/named restart
```

Bila BIND telah berjalan dengan baik, Anda dapat melihat port 53 BIND telah berjalan dengan mengetikkan perintah nmap.

```
# nmap localhost
```

Kemudian kita dapat melakukan beberapa query ke DNS Server dengan menggunakan perintah:

```
# host 192.168.2.1
1.2.168.192.in-addr.arpa domain
name pointer ns.example.com.
# host ns.example.com
ns.example.com has address
192.168.2.1
# host -t ns example.com
example.com name server
ns.example.com.
# host -t mx example.com
example.com mail is handled by
10 mail.example.com.example.com.
# host -t cname mail.example.com
mail.example.com is an alias for
ns.example.com.
# host -t cname www.example.com
www.example.com is an alias for
ns.example.com.
# host -t cname ftp.example.com
ftp.example.com is an alias for
ns.example.com.
```

Jika semua query test berhasil, maka Anda telah berhasil membuat DNS Server dengan domain example.com dengan name server ns.example.com dan routing e-mail domain example.com ke mail server mail.example.com, routing www domain example.com ke www.example.com dan routing ftp domain example.com ke ftp.example.com.

Selanjutnya Anda dapat mengetes masing-masing routing dengan menjalankan service Web server Apache misalnya di browser Anda. Ketikkan www.example.com, jika keluar tampilan default webserver di komputer Anda.

Test juga pada mail server yang Anda miliki, dengan memasukkan alamat mail servernya menjadi mail.example.com. Jika berhasil send dan receive mail, berarti Anda telah berhasil mengimplementasikan DNS Server dengan baik.

Semoga dari penjelasan cara membuat DNS Server Lokal ini, dapat membuat Anda paham bagaimana konsep domain di Internet itu bekerja. **Supriyanto** (supriyanto@infolinux.co.id)

Professional 100% Linux Training & Solution

Ingin Menguasai Linux Secara LENGKAP ?!

Paket A-Z Linux

- Linux Concept & Fundamental
- Linux System Administration
- Linux Internet + Intranet Server
- Linux Security

56 hours (14 day @ 4 hours)

Only : Rp.4.850.000,-

Special Offer Crash Programme !

PATIN (Paket Intensif)

- Linux Concept and Fundamental
- Linux System Administration
- Linux Internet + Intranet Server

42 hours (6 days@ 7 hour)

Only : Rp.3.750.000,-

PAKIS (Paket Ekonomis)

- Linux Concept and Fundamental
- Linux System Administration
- Linux Internet + Intranet Server.

44 hours (11 days@ 4 hour)

Only : Rp.3.650.000,-

Ketik: Info PATIN atau Info PAKIS kirim SMS ke 0856 7771030 SMS Server powered by eSMSis (www.eSMSis.com)

SMS Server & Gateway

eSMSis Ver. 1.5

- Web based, Internet Ready
- Broadcast, GroupCast, MemberCast
- Scheduled SMS, Product Informations
- Auto Response, Remote SMS, Alert, etc

visit: www.eSMSis.com

MySMSPass

Start Making Money from your Website!

- SMS Authentication System for Web Content
- Short Number by Telco Operators

Demo Website : www.InfoLINUX.web.id/sections



Linuxindo

PUSAT : Wisma Bisnis Indonesia Suite #415 - JAKARTA

BANDUNG: (022) 7234192 - CIREBON: (0231) 200418 - SOLO: (0271) 662318

PERINGATAN ! Linux bisa membuat Anda kecanduan, menambah PD dan belum ada obatnya. Tidak Setiap Paket Promosi tersedia di Cabang.

(021) 5362390

www.Linuxindo.com