

Membangun Linux Dial-In Server

Linux dapat diakses dari jauh secara langsung lewat modem. Jika akses Internet di salah satu kantor sangat cepat, Linux dapat membagi akses Internet untuk kantor yang lain atau untuk Anda dari rumah. Ini mirip dengan ISP (*Internet Service Provider*) sederhana.



Barangkali di antara kita pernah terpikir bagaimana cara membuat suatu *server* seperti Telkomnet Instan. Kita hanya cukup men-dial suatu nomor telepon, dan kita pun dapat langsung terhubung ke Internet. Tanpa ada biaya pendaftaran atau registrasi lainnya. Server seperti ini sering disebut dengan Dial-In Server. Pada Microsoft Windows NT 4.0, istilah seperti ini disebut *Remote Access Server* (RAS).

Dial-In Server adalah suatu host (mesin) yang dilengkapi dengan modem dan saluran telepon, yang mengizinkan *host* (mesin) lain yang memiliki modem dan saluran telpon untuk menghubungkannya (*Introduction* — <http://www.swcp.com/~jgentry/pers.html>). Dari sisi user atau mesin yang menghubungi server, disebut dengan *dial-out*. Umumnya, dalam melakukan *dial-out*, user menggunakan PPP (*Point to Point Protocol*) agar bisa terhubung ke Internet atau pun ke suatu jaringan lokal perusahaan. Hubungan antara user dan jaringan komputer yang akan diakses dapat dilihat pada Gambar 1.

Salah satu tanggung jawab *network administrator* adalah menyediakan fasilitas agar user dapat dengan mudah melakukan *dial-out*. Untuk lebih jelasnya, mari kita langsung menuju instalasi dan konfigurasi Dial-In Server.

Fasilitas yang dibutuhkan

Umumnya, saat ini Internet menggunakan TCP/IP sebagai 'bahasa' untuk saling berkomunikasi. Untuk itu,

hal-hal yang perlu disiapkan oleh network administrator adalah:

- Nomor telepon yang dapat diakses.
- *Username* dan *Password*.
- Prosedur otentifikasi.
- IP Address dari DNS Server, diperlukan bagi user untuk menerjemahkan nama host (*hostname*) menjadi IP Address dan sebaliknya.

Pada saat artikel ini ditulis, *software* yang digunakan penulis adalah:

- Sistem Operasi GNU/Linux Red Hat 8.0, kernel 2.4.18.
- *mgetty* + *sendfax*-1.0.0.
- *pppd*-2.4.1.

Instalasi dan konfigurasi

➔ Modem

Dalam dunia Linux, dikenal istilah WinModem. Modem jenis ini, umumnya berbentuk modem internal dan dibuat untuk sistem operasi Microsoft Windows. Modem jenis ini, kadang-kadang tidak memerlukan driver karena sebagian driver-nya sudah terdapat pada sistem operasi Microsoft Windows. Di Linux, beberapa jenis modem internal dapat digunakan, tapi sebagian yang lain tidak.

Penulis menggunakan modem US Robotics 56K eksternal pada saat membangun server ini. Alasannya adalah modem eksternal lebih mudah dikenali oleh Linux, meskipun ada beberapa modem internal yang juga cukup mudah. Pada distribusi Red Hat, terdapat fasilitas *kudzu* untuk mendeteksi *hardware* baru, atau fasilitas *hddrake* pada distribusi Mandrake.

Jika ternyata Anda menggunakan modem internal, Anda dapat merujuk pada alamat <http://www.linmodems.org> untuk mendapatkan utiliti *scanmodem* yang dapat membantu untuk mengetahui jenis modem yang digunakan. Alamat lain yang dapat menjadi referensi adalah <http://www.linuxdoc.org/HOWTO/Serial-HOWTO.html> dan <http://www.linuxdoc.org/HOWTO/Modem-HOWTO.html> (Baca juga artikel tentang modem internal di *InfoLINUX* edisi ini atau edisi-edisi sebelumnya—Red).

➔ mgetty+sendfax-1.0.0.tar.gz

Software ini dibutuhkan untuk menangani komunikasi dengan modem. Saat dijalankan, umumnya melalui */etc/inittab*, *getty* akan berjalan di belakang (sebagai *background process*). Modem Anda akan idle selama itu, sampai modem tersebut menerima panggilan. Lalu *getty* akan menentukan jenis panggilan tersebut dan memulai proses komunikasi *modem-to-modem*. Ada beberapa jenis *getty* yang dapat digunakan untuk ini. Penulis menggunakan *software* *mgetty* + *sendfax* yang dapat diambil pada alamat <http://www.leo.org/~doering/mgetty/>.

Setelah mendapatkan file *mgetty* + *sendfax*-1.0.0.tar.gz, maka langkah-langkah instalasinya adalah sebagai berikut:

1. Salin file tersebut ke dalam direktori */usr/local/src*.
2. Ekstrak file tersebut dengan perintah:
`tar zxvf mgetty+sendfax-1.0.0.tar.gz`
3. Masuk dalam direktori *mgetty* + *sendfax*-1.0.0
4. Salin file *policy.h-dist* menjadi *policy.h*. Anda dapat mengubah hal-hal yang menurut Anda perlu di sini. Tetapi, konfigurasi default ini, berjalan baik pada komputer penulis.
5. Kemudian edit file *Makefile* untuk menentukan lokasi instalasi, *system defines* dan *libraries*. Konfigurasi *default* ini juga berjalan baik pada komputer penulis.
6. Kemudian jalankan perintah *make* diikuti dengan *make install* untuk melakukan instalasi.

Untuk menjalankan mgetty, edit file `/etc/inittab`. Pada komputer penulis, modem disambungkan di `/dev/ttyS0` (COM1), sehingga pada file `/etc/inittab`, penulis menambahkan baris ini:

```
S0:2345:respawn:/sbin/mgetty ttyS0 -D /dev/ttyS0
```

Option `-D` memberitahukan mgetty agar hanya menerima data, bukan fax. Setelah menambahkan baris tadi, jalankan perintah `init -q` untuk memaksa `initd` membaca file `/etc/inittab`. Hal ini akan membuat mgetty dijalankan.

Kemudian perintah berikut akan membuat semua user dapat menjalankan `pppd`:

```
# chmod u + s /usr/sbin/pppd
```

Agar pengguna Windows tidak mendapatkan layar login setelah koneksi dengan `ppp` berhasil, maka kita dapat menggunakan fasilitas AutoPPP agar mgetty dapat menjalankan `ppp` pada saat memulai koneksi. Untuk itu, sebelum mengompilasi mgetty, pada file `Makefile`, pada baris 110 tambahkan kode berikut:

```
CFLAGS = -O2 -Wall -pipe -DAUTO_PPP
```

Kemudian simpan perubahan itu, dan kompilasilah dengan menggunakan perintah `make` dan `make install`. Sebagai catatan, pada Red Hat 8.0 penulis, pada saat mengompilasi mgetty, secara otomatis dukungan AutoPPP telah diikutsertakan.

Setelah itu, pada file `/etc/mgetty + sendfax/login.config`, tambahkan baris berikut:

```
/AutoPPP/- - /usr/sbin/pppd file /etc/ppp/options
```

Jika sudah ada, maka aktifkan fungsi tersebut dengan menghapus tanda pagar (`#`) pada awal baris tersebut. Sekarang kita dapat melanjutkan pada bagian konfigurasi `pppd`.

➔ Pppd

`pppd` umumnya secara otomatis sudah ikut terinstalasi pada saat menginstalasi Linux atau minimal tersedia pada distribusi Linux yang Anda gunakan. Untuk versi terbaru dapat diambil pada

alamat <ftp://cs.anu.edu.au/pub/software/ppp/ppp-2.4.1.tar.gz>

Untuk melihat apakah pada Linux Anda sudah terinstalasi `pppd`, gunakan perintah berikut:

```
# whereis pppd
```

Jika sudah terinstalasi, perintah di atas akan menunjukkan lokasi software `pppd` (umumnya `/usr/sbin/pppd`) dan Anda dapat melewati langkah berikut ini untuk langsung menuju pada konfigurasi `pppd`. Jika belum, maka instalasi dapat dilakukan melalui CD Linux yang Anda memiliki. Jika Anda menggunakan distribusi Red Hat, maka untuk melakukan instalasinya adalah sebagai berikut:

```
# mount /dev/cdrom /mnt/cdrom
```

```
# rpm -ivh /mnt/cdrom/RedHat/RPMS/pppd-2.4.1
```

Untuk versi *source*, maka langkah-langkahnya adalah sebagai berikut:

1. Salin file `pppd-2.4.1.tar.gz` ke dalam direktori `/usr/local/src`
2. Ekstrak file tersebut:

```
# tar xzf pppd-2.4.1.tar.gz
```

3. Masuklah dalam direktori `pppd-2.4.1`

```
# cd pppd-2.4.1
```

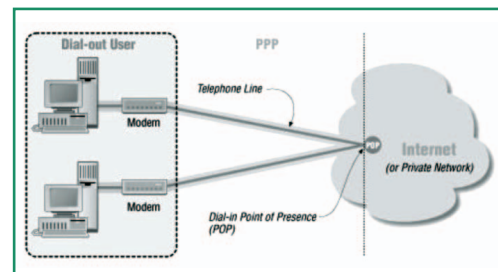
4. Memulai proses instalasi (ada baiknya untuk membaca file `README` terlebih dahulu yang berada dalam direktori tersebut).

```
# configure --prefix=/usr/local/ppp
```

```
# make
```

```
# make install
```

Selain itu, kernel Linux Anda juga harus mendukung PPP. Umumnya, pada saat instalasi Linux, dukungan terhadap PPP sudah disertakan. Jika ternyata kernel yang digunakan belum mendukung PPP, maka Anda perlu



▲ Gambar 1. Dial-out PPP

mengompilasi ulang kernel tersebut. Untuk langkah-langkah mengompilasi kernel, silahkan merujuk pada alamat <http://www.linuxdoc.org/HOWTO/Kernel-HOWTO.html>.

➔ Konfigurasi

`pppd` membutuhkan informasi mengenai antarmuka serial-nya, antara lain *PPP Link Control Protocol* (LCP), *IP Control Protocol* (IPCP), dan otentifikasi. Umumnya modem menggunakan interface serial RS-232 (DOS atau MS Windows mengenali serial port ini sebagai COM). Linux mengenali device interface serial RS-232 ini seperti pada Tabel 1.

File-file konfigurasi PPP terletak pada direktori `/etc/ppp`. Tadi kita telah mengonfigurasi agar mgetty membaca file `/etc/ppp/options`. Isi file tersebut pada komputer penulis pertama-tama hanyalah berisi baris:

```
lock
```

Kemudian penulis menambahkan baris lain sehingga menjadi seperti berikut:

```
-detach
debug
asyncmap 0
modem
crtcts
lock
require-pap
refuse-chap
login
```

Tabel 1. Device Interface Serial Linux pada PC

Linux	DOS	IRQ	I/O Address
Device File	Equivalent		Base
<code>/dev/ttyS0</code>	COM1	4	0x3f8
<code>/dev/ttyS1</code>	COM2	3	0x2f8
<code>/dev/ttyS2</code>	COM3	4	0x3e8
<code>/dev/ttyS3</code>	COM4	3	0x2e8

```
proxyarp
netmask 255.0.0.0
ms-dns 10.1.1.5
```

Keterangan dari masing-masing opsi di atas adalah sebagai berikut:

-detach	berfungsi agar pppd tidak membuat background process baru
debug	berfungsi agar mencatat transaksi pada file <code>/var/log/messages</code>
asynmap 0	mengizinkan agar pppd dapat bekerja melalui koneksi telnet/rlogin
modem	memberitahu pppd untuk menggunakan modem control lines
crtstcts	memberitahu pppd untuk menggunakan hardware flow control untuk mengendalikan aliran data ke serial port
lock	berfungsi agar tidak ada proses lain selain pppd yang dapat menggunakan device ini
require-paprefuse-chap	gunakan mode otentifikasi PAP (Password Authentication Protocol) dan menolak mode otentifikasi CHAP (Challenge Handshake Authentication Protocol)
login	gunakan <code>/etc/passwd</code> sehingga tidak perlu menggunakan file <code>/etc/ppp/pap-secrets</code>
proxyarp	tambahkan pada tabel ARP (Address Resolution Protocol) IP Address dari client dan Ethernet Address
netmask	memberikan nilai netmask pada NIC
ms-dns	lokasi dari DNS Server

Jika opsi login digunakan, maka file `/etc/ppp/pap-secrets` tidak lagi dibutuhkan. Anda dapat menghapus file tersebut atau mengisinya dengan baris:

```
★ ★ "" ★
```

Akan tetapi jika kita menggunakan opsi login, berarti semua user yang terdaftar pada `/etc/passwd` akan dapat melakukan akses untuk men-dial, kecuali jika Anda melarangnya pada file `/etc/ppp/pap-secrets`. Kadang-kadang, untuk alasan keamanan, kita perlu untuk membatasi siapa saja yang memiliki akses dan yang tidak. Pada sistem penulis secara *default account* yang

terbentuk pada saat instalasi Red Hat 8.0 antara lain adalah root, bin, daemon, adm, lp, dan sync. Untuk melarang mereka agar tidak dapat melakukan dial-in, maka pada file `/etc/ppp/pap-secrets` cukup ditambahkan baris:

#user	server	secrets	address
★	★	" "	★
root	★	-	★
bin	★	-	★
daemon	★	-	★
adm	★	-	★
lp	★	-	★
sync	★	-	-

Setelah itu, agar PPP dapat bekerja, tentunya komputer client harus memiliki IP Address. Umumnya client tidak memiliki IP Address sendiri, sehingga PPP harus memberikan IP Address bagi client. Untuk melakukan hal itu, buatlah sebuah file baru lagi dengan nama `/etc/ppp/options.ttySO` (jika pada server, modem disambungkan pada `/dev/ttySO`). Berikut adalah contoh file `/etc/ppp/options.ttySO` milik penulis:

```
# Format :
# w.x.y.z : a.b.c.d
#
# Dengan
# w.x.y.z = IP Address dari Server Dial-In (RAS) ini.
# a.b.c.d = IP Address yang akan diberikan kepada client,
# jika dia berhasil login
#
10.1.1.5:10.1.1.50
```

Sampai di sini, Anda sudah selesai membuat sebuah Linux Dial-In Server. Selanjutnya mari kita lakukan pengujian untuk melihat apakah server yang kita bangun dapat bekerja dengan baik.

Pengujian

Kita dapat melakukan pengujian dengan cara menjalankan pppd:

```
$ /usr/sbin/pppd
```

jika berhasil, saat Anda menjalankan:

```
$ /sbin/ifconfig -a
```

akan muncul tampilan sebagai berikut:

```
ppp0 Link encap:Point-to-Point Protocol
```

```
POINTOPOINT NOARP MULTICAST
MTU:1500 Metric:1
RX packets:0 errors:0 dropped:0
overruns:0 frame:0
TX packets:0 errors:0 dropped:0
overruns:0 carrier:0
collisions:0 txqueuelen:3
RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

kemudian pada file `/var/log/messages` akan terdapat log seperti ini:

```
Mar 25 16:48:15 intranet pppd[4929]: pppd
2.4.1 started by LOGIN, uid 0
Mar 25 16:48:15 intranet pppd[4929]:
Using interface ppp0
Mar 25 16:48:15 intranet pppd[4929]:
Connect: ppp0 <-> /dev/ttySO
Mar 25 16:48:15 intranet /etc/hotplug/
net.agent: assuming ppp0 is already up
Mar 25 16:48:19 intranet
ppp(pam_unix)[4929]: session opened for
user pungi by LOGIN(uid=0)
Mar 25 16:48:19 intranet pppd[4929]: user
pungi logged in
Mar 25 16:48:23 intranet pppd[4929]:
found interface eth0 for proxy arp
Mar 25 16:48:23 intranet pppd[4929]: local
IP address 10.1.1.5
Mar 25 16:48:23 intranet pppd[4929]:
remote IP address 10.1.1.50
```

Setelah pppd melaporkan IP Address Local dan Remote, hal ini menunjukkan bahwa koneksi PPP telah siap digunakan. Perlu dipastikan bahwa IP client (*remote*) harus satu subnet dengan IP Server (*local*).

Salah satu alasan dari pembangunan Dial-In Server antara lain adalah untuk memanfaatkan *resource* yang tersedia. Jika pembangunan Dial-In Server ini diterapkan pada perkantoran, maka karyawan kantor tersebut dapat terhubung dengan jaringan kantor tersebut dengan cara melakukan dial-up ke kantor itu. Seandainya jaringan kantor tersebut terhubung dengan Internet, maka karyawan kantor tersebut dapat ikut menikmati layanan Internet yang tersedia di kantor. Dengan cara ini pula, System Administrator dapat melakukan administrasi sistem, secara *remote*.¹

Pungi (punki@softhome.net)