
Konfigurasi Server Berbasis Linux dengan Koneksi Internet Multi-Homed

Hendra

STMIK IBBI

Jl. Sei Deli No. 18 Medan, Telp. 061-4567111 Fax. 061-4527548

e-mail: hendra.soewarno@gmail.com

Abstrak

Perusahaan yang menyediakan layanan untuk pertukaran data melalui Internet membutuhkan koneksi Internet cadangan untuk tujuan failover, sehingga gangguan terhadap salah satu koneksi tidak menghentikan layanan yang ada. Server yang memiliki koneksi Internet multi-homed membutuhkan konfigurasi tambahan agar permintaan yang masuk ke suatu interface mendapatkan respon melalui interface yang sama. Tulisan ini membahas tentang konfigurasi server Linux yang memiliki koneksi Internet multi-homed dengan memanfaatkan fasilitas policy routing yang terdapat pada kernel 2.2. Berdasarkan hasil pengujian terhadap shell-script hasil rancangan, masing-masing interface pada server dapat secara aktif melayani permintaan Internet.

Kata kunci: Policy Routing, Source Routing, Multi-homed

Abstract

Companies that provide services to exchange data over the Internet requires a backup Internet connection for failover purposes, so that disruption of one of the connections do not stop that service. Server that has multi-homed Internet connections require additional configuration in order to request that goes to an interface to get the response via the same interface. This paper discusses the configuration of the Linux server that has a multi-homed internet connections using policy routing facilities contained in the kernel 2.2. Based on the test results of the designed shell-scripts, each interface on the server can actively serve Internet request.

Keywords: policy routing, source routing multi-homed

1. Pendahuluan

Perkembangan Internet banyak digunakan oleh perusahaan untuk kepentingan pertukaran data dengan jaringan bisnisnya. Sejalan dengan manfaat yang dirasakan, perusahaan terus mengembangkan berbagai jenis layanan berbasis Internet, sehingga sampai kepada suatu kondisi dimana gangguan pada sistim tersebut dirasakan sebagai gangguan kepada operasi bisnis.

Salah satu faktor penyebab yang sering terjadi pada sistim berbasis Internet adalah gangguan pada koneksi internet. Gangguan yang terjadi dapat disebabkan oleh gangguan koneksi antara server perusahaan dengan *Internet Service Provider* (ISP), maupun gangguan pada internal ISP.

Usaha yang dapat dilakukan perusahaan untuk meningkatkan ketersediaan layanan adalah dengan menggunakan ISP yang menawarkan ketersediaan yang tinggi, disamping itu perusahaan perlu juga mempertimbangkan koneksi internet cadangan dari ISP yang berbeda untuk keperluan *failover*, sehingga jika terjadi gangguan pada salah satu koneksi, maka layanan yang ada dapat segera dialihkan kepada koneksi yang lain melalui failover nama domain yang tersedia di Internet. Pemakaian lebih dari satu ISP untuk tujuan *failover* memperbolehkan suatu perusahaan untuk melanjutkan bisnis tanpa interupsi ketika terjadi gangguan pada provider tertentu[1]

Konfigurasi sistim server yang memiliki koneksi *Internet Multi-homed* adalah tidak sesederhana konfigurasi server dengan satu koneksi internet tunggal, secara default konfigurasi kernel linux adalah menggunakan *destination routing*, sehingga pada satu saat hanya ada satu gateway default yang aktif membalas permintaan dari internet. Agar tujuan penyediaan provider cadangan dapat terwujud, maka pada masing-masing interface harus aktif membalas permintaan yang masuk.

Penelitian ini bertujuan mendapatkan suatu rancangan *shell script* dengan mengaplikasikan fitur policy routing yang tersedia pada kernel Linux 2.2 pada suatu server yang memiliki dua koneksi internet, sehingga masing-masing interface dapat secara aktif melayani permintaan dari internet pada saat yang sama.

Penelitian ini dilakukan pada server Linux dengan distribusi Ubuntu 10.4 LTS yang memiliki satu koneksi internet dedicated, dan satu internet DSL.

2. Metode Penelitian

Penelitian ini dilakukan dengan mempelajari referensi dari beberapa sumber yang relevan yang terkait dengan metode routing pada protocol internet, dan fitur advanced routing pada Linux, kemudian dilanjutkan dengan mempelajari arsitektur jaringan internet yang ada diperusahaan beserta dengan data koneksi seperti: interface, alamat IP, alamat network, dan alamat gateway untuk masing-masing koneksi internet. Kemudian dilanjutkan dengan perancangan algoritma yang diperlukan yang nantinya akan diimplementasikan dalam bentuk *shell-script*, pengujian terhadap hasil rancangan setting dilakukan dengan menjalankan shell-script dan menampilkan hasil setting pada sisi server. Pengujian pada sisi klien dilakukan dengan melakukan ping langsung kepada alamat IP publik masing interface. Jika ping mendapatkan respon, maka dapat disimpulkan rancangan shell-script adalah efektif.

2.1. Routing pada Internet Protocol

Gateway / router melakukan routing paket data berdasarkan alamat tujuan dari paket. Secara sederhana keputusan routing adalah sebagai berikut:

- Jika host tujuan berada pada jaringan lokal, maka data langsung dikirim ke host tujuan.
- Jika host tujuan berada pada jaringan remote, maka data akan dialihkan ke default gateway/router.

Keputusan routing IP dilakukan melalui pencarian pada suatu tabel sederhana yang dikenal sebagai tabel *routing* atau tabel *forwarding*. Paket akan diarahkan ke tujuan berdasarkan aturan yang ditentukan oleh tabel routing. Tabel routing memetakan berdasarkan alamat tujuan paket data, dan ke gateway/router atau interface yang harus digunakan oleh IP untuk mencapai tujuan.

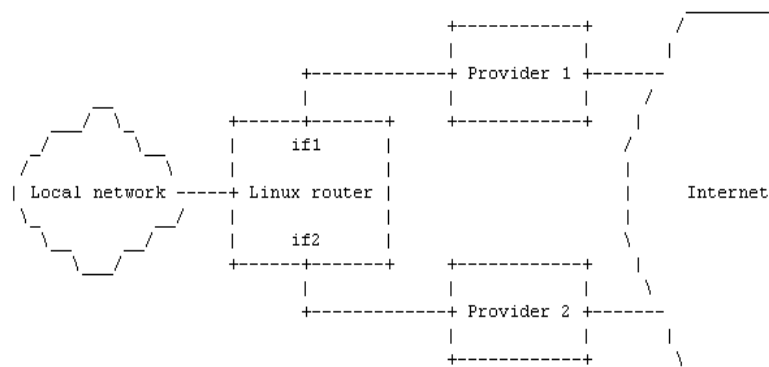
Sesuatu hal yang perlu diperhatikan di dalam suatu tabel *routing* adalah kata *default* pada field tujuan. Isian ini adalah *default route*, dan *gateway* yang disebut dalam isian ini adalah *default gateway*. *Default gateway* digunakan ketika tidak ada daftar route di dalam tabel yang dapat digunakan untuk pembuatan keputusan untuk mencapai alamat tertentu. Semua gateway yang tercantum pada tabel adalah host yang terhubung langsung pada jaringan lokal.

Tabel *routing* tidak menyediakan route dari ujung-ke-ujung. Pada satu route hanya menunjuk kepada satu titik ke gateway berikutnya, yang disebut dengan next hop, yang merupakan perpanjangan jalan menuju ke jaringan tujuan. Suatu host mengantungkan diri pada *gateway* lokal untuk menyampaikan data, dan *gateway* yang satu mengantungkan diri kepada *gateway* lainnya. Ketika suatu datagram bergerak dari satu *gateway* ke *gateway* yang lain, akan diakhirnya setelah mencapai gateway yang secara langsung terkoneksi pada jaringan tujuan dimana host tujuan terkoneksi. IP menggunakan bagian *network* dari alamat IP untuk mengarahkan datagram tersebut antar jaringan, dan alamat lengkap digunakan untuk penyampaian akhir agar mencapai host tujuan.

2.1. Policy Routing

Pada umumnya router dikonfigurasi berdasarkan alamat IP tujuan. Salah satu kemampuan yang ditambahkan pada Kernel Linux 2.2 adalah fitur *multiple routing table*, Kernel Linux 2.2 keatas mendukung sampai kepada 252 tabel *routing* tambahan yang mendukung kemampuan *policy routing* [2], fasilitas ini memperbolehkan pemakai untuk melakukan konfigurasi routing berdasarkan parameter tambahan daripada hanya berdasarkan alamat IP tujuan[3]. Masing-masing tabel *routing* dapat mengandung sejumlah isian aturan *routing lokal* yang diatur oleh *kernel*, dan tabel *routing* utama yang mana sebagian diatur oleh *kernel*, semua tabel *routing* dapat dikendalikan oleh *administrator* atau *software routing* dengan cara mengubah maupun menghapusnya.

Secara *default*, konfigurasi pada Linux adalah menggunakan tabel *routing* utama untuk keputusan routing, atau dengan kata lain hanya terdapat satu *default gateway* yang dapat digunakan. Pada suatu host yang memiliki satu internet tunggal, maka default gateway adalah menunjuk kepada *gateway* dari *provider Internet*. Permasalahan pada suatu *host* memiliki koneksi *Internet multi-homed* adalah bagaimana route jawaban terhadap paket yang masuk melalui interface (dalam hal ini provider) tertentu, harus keluar kembali melalui *interface* yang sama[2] seperti yang ditunjukkan pada Gambar 1.



Gambar 1. Host dengan dua koneksi internet
(Sumber: Bert Hubert, et. al., 2002)

Agar permasalahan tersebut diatas mudah dipahami, akan dibuat beberapa nama simbolik, misalkan \$IF1 adalah untuk interface pertama, dan \$IF2 untuk interface kedua, kemudian \$IP1 adalah alamat IP yang berasosiasi dengan \$IF1 dan \$IP2 adalah alamat IP berasosiasi dengan \$IF2, selanjutnya \$P1 adalah alamat IP untuk gateway Provider1 dan \$P2 adalah alamat IP untuk gateway provider 2, dan akhirnya \$P1_NET adalah network untuk \$P1, dan \$P2_NET adalah IP network untuk \$P2.

Agar masing-masing koneksi Internet dapat aktif membalas koneksi yang masuk, maka perlu dibuat dua tabel routing tambahan yaitu T1 dan T2 yang ditambahkan pada `/etc/iproute2/rt_tables`. Pada masing-masing tabel *routing* dibuat *default route* melalui *gateway* masing-masing *provider* dengan *script* sebagai berikut:

```
ip route add $P1_NET dev $IF1 src $IP1 table T1
ip route add default via $P1 table T1
ip route add $P2_NET dev $IF2 src $IP2 table T2
ip route add default via $P2 table T2
```

Selanjutnya anda membuat tabel *routing* utama, agar melakukan *routing* secara langsung ke jaringan melalui interface yang terkoneksi langsung pada jaringan tersebut seperti *script* yang ditunjukkan berikut:

```
ip route add $P1_NET dev $IF1 src $IP1
ip route add $P2_NET dev $IF2 src $IP2
```

Kemudian, tambahkan default route sesuai dengan keinginan anda:

```
ip route add default via $P1
```

Akhirnya anda membuat aturan *routing* untuk memilih tabel *routing* yang digunakan untuk keputusan *routing* berdasarkan sumber interface agar trafik yang masuk ke suatu *interface* akan mendapatkan jawaban dari *interface* tersebut.

```
ip rule add from $IP1 table T1
ip rule add from $IP2 table T2
```

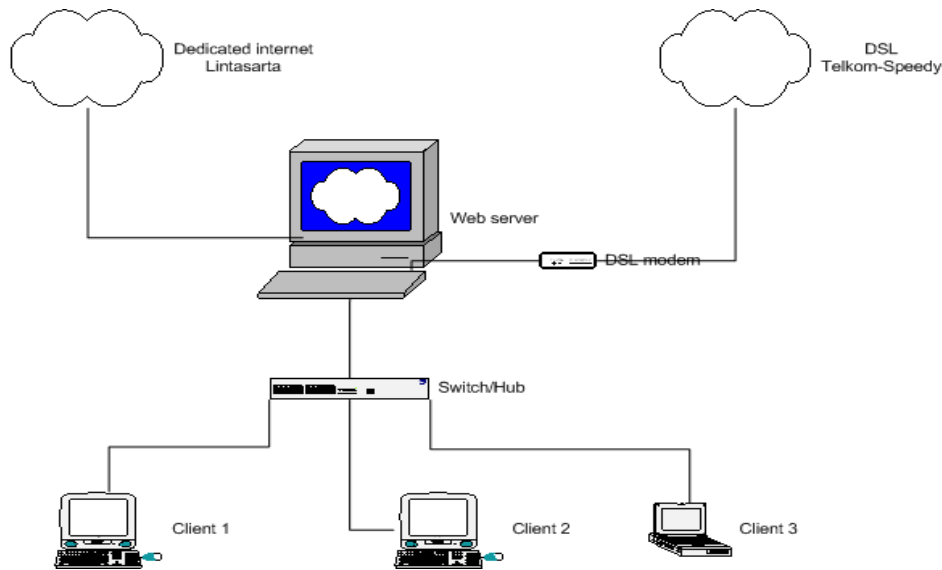
Secara ringkas dapat dikatakan bahwa anda membutuhkan dua routing tabel tambahan diluar dari tabel *routing* utama. Kemudian membuat suatu *route* ke *gateway* dan membuat suatu *default route* melalui *gateway* tersebut, dan membuat aturan pemakaian masing-masing tabel

3. Pembahasan dan Hasil

3.1. Arsitektur Jaringan

Adapun konfigurasi jaringan yang ada di perusahaan, yaitu terdiri dari satu Web server yang memiliki dua koneksi internet yaitu koneksi *dedicated* yang menggunakan *provider* LintasArta, dan satu

lagi koneksi internet yang menggunakan provider Telkom speedy yang mendapat alokasi alamat IP tetap sebagaimana yang ditunjukkan pada Gambar 2.



Gambar 2. Arsitektur jaringan di perusahaan

Berdasarkan arsitektur tersebut diatas, maka dapat disimpulkan bahwa Web server memiliki konfigurasi koneksi *mutli-homed*, dimana terdiri dari *interface* jaringan internet Lintasarta (eth0), interface jaringan internet telkom Speedy (eth1 --> ppp0), dan interface ke LAN (eth2). Pada masing-masing interface memiliki pengalamatan seperti yang ditunjukkan pada Tabel 1.

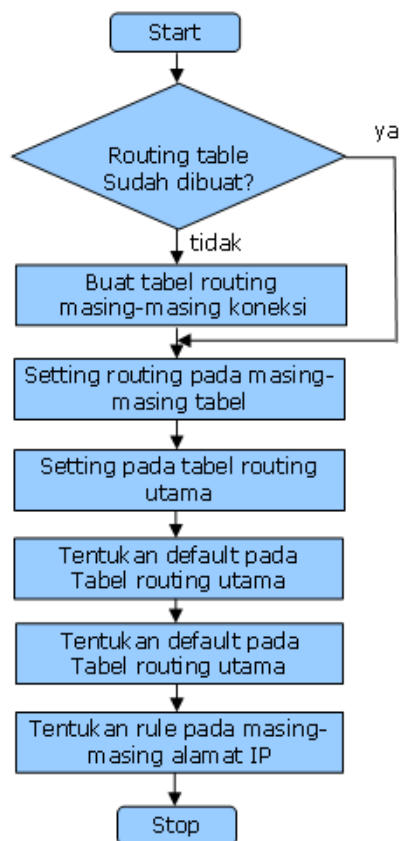
Tabel 1. Setting pada jaringan perusahaan Interface

Iface	Pengalamatan	Keterangan
eth0	IP Address: 182.23.35.xxx Mask: 255.255.255.248 Network Address: 182.23.35.xxx Default GW: 182.23.35.xxx	Koneksi ke ISP LintasArta
ppp0	IP Address: 125.165.41.xxx Mask: 255.255.255.255 P-t-P: 125.165.40.1 Default GW: 125.165.40.1	Koneksi ke ISP Telkom Speedy (eth1-->ppp0)
eth2	IP Address:192.168.0.1 Netmask: 255.255.255.0	Koneksi ke LAN

Secara default sistim operasi Linux hanya dapat berlaku satu *default gateway*. Sehingga menyebabkan permintaan yang masuk dari eth0 (182.23.35.xxx) maupun ppp0 (125.165.41.xxx) hanya akan keluar melalui *interface* yang berlaku sebagai *default gateway*. Kondisi ini menyebabkan hanya satu koneksi internet yang efektif melayani permintaan dari internet dengan kata lain koneksi yang masuk dari interface yang berlaku sebagai default gateway, maka akan dibalas oleh interface yang sama, sedangkan koneksi yang masuk dari interface provider yang kedua, maka akan dibalas oleh *interface* yang berlaku sebagai *default gateway*.

3.2. Algoritma

Agar kedua koneksi yang tersedia dapat secara aktif melayani permintaan dari Internet, maka perlu ditambah dua tabel *routing* yang nantinya akan digunakan menjadi aturan *routing* bagi masing-masing *interface* (dalam hal ini alamat IP untuk masing-masing *interface*). Adapun algoritma yang digunakan pada proses konfigurasi ditunjukkan pada Gambar 3.



Gambar 3. Langkah-langkah setting

Pada awalnya akan dilakukan pemeriksaan terlebih dahulu apakah tabel *routing* yang dibutuhkan untuk masing-masing koneksi Internet telah tersedia, jika belum maka akan dibuat masing-masing tabel *routing*. Selanjutnya akan dibuat *setting routing* pada masing-masing tabel *routing*, selanjutnya dilakukan setting routing untuk tabel routing utama, dan *setting default gateway* untuk tabel routing utama (hal ini diperlukan untuk kebutuhan koneksi dari server ke internet), selanjutnya dibuat aturan pemakaian masing-masing tabel routing untuk masing-masing interface yang terkoneksi ke internet melalui alamat IP masing-masing

3.2. Shell Script

Berdasarkan algoritma yang telah dirancang sebelumnya, maka dapat dibuat *shell script* yang nantinya dapat aktifkan setiap kali proses *startup server* sebagai berikut:

```

rc.local
#!/bin/sh

#interface
IF1=eth1
IF2=ppp0

#ips
IP1=182.23.35.114
IP2=125.165.41.65

#gateways
P1=182.23.35.113
P2=125.165.40.1

#ip network

```

```

P1_NET=182.23.35.112
P2_NET=125.165.40.1

sudah=$(cat /etc/iproute2/rt_tables | grep 'LINTAS')

#tabel routing hanya perlu dibuat satu kali
if [ "$sudah" = "" ]
then
  #buat dua routing table untuk masing2 interface
  echo 1 LINTAS >> /etc/iproute2/rt_tables
  echo 2 TELKOM >> /etc/iproute2/rt_tables
fi

#create routing tables
ip route add $P1_NET dev $IF1 src $IP1 table LINTAS
ip route add default via $P1 table LINTAS
ip route add $P2_NET dev $IF2 src $IP2 table TELKOM
ip route add default via $P2 table TELKOM

#main routing table
ip route add $P1_NET dev $IF1 src $IP1
ip route add $P2_NET dev $IF2 src $IP2

#default route preference
ip route add default via $P1

#routing rules untuk masing-masing IP koneksi internet
ip rule add from $IP1 table LINTAS
ip rule add from $IP2 table TELKOM

```

Agar script ini dapat dieksekusi setiap saat pada *booting server*, maka dapat diletakkan pada file `rc.local`.

3.3. Pengujian

3.3.1. Pengujian pada Sisi Server

Pengujian terhadap efektifitas shell-script hasil rancangan dilakukan dengan proses boot ulang server, dan dilakukan pemeriksaan hasil pembuatan table local dengan menampilkan isi file `rt_tables` dengan perintah `cat` sebagaimana yang ditunjukkan sebagai berikut:

```

server@idnet:~$ cat /etc/iproute2/rt_tables
#
# reserved values
#
255  local
254  main
253  default
0    unspec
#
# local
#
#1   inr.ruhep
1   LINTAS
2   TELKOM

```

Berdasar hasil tampilan tersebut diatas menunjukkan bahwa terdapat dua table routing local dengan nama LINTAS dan TELKOM, kemudian ditampilkan aturan *route* untuk masing-masing tabel *routing* dengan perintah `ip route show` sebagai berikut:

```

server@idnet:~$ ip route show table LINTAS
182.23.35.112 dev eth1 scope link src 182.23.35.114

```

```
default via 182.23.35.113 dev eth0
```

```
server@idnet:~$ ip route show table TELKOM
125.165.40.1 dev ppp0 scope link src 125.165.41.65
default via 125.165.40.1 dev ppp0
```

Berdasarkan hasil tampilan tersebut diatas menunjukkan bahwa route pada masing-masing tabel routing telah berhasil dikonfigurasi, dimana untuk eth0 akan menggunakan default melalui 182.32.345.113, sedangkan untuk ppp0 akan menggunakan *default* melalui 125.165.40.1, dan akhirnya ditampilkan *rule routing* pada *host* tersebut sebagai berikut:

```
server@idnet:~$ ip rule show
32764: from 125.165.41.65 lookup TELKOM
32765: from 182.23.35.114 lookup LINTAS
32766: from all lookup main
32767: from all lookup default
```

Berdasarkan hasil tampilan tersebut diatas menunjukkan bahwa aturan pemakaian tabel routing adalah, jika koneksi masuk dari alamat IP 125.165.41.65 (ppp0) maka keputusan *routing* akan dicari ke tabel routing TELKOM, kemudian jika koneksi masuk dari alamat IP 182.23.35.114 (eth0) maka keputusan routing akan dicari ke tabel routing LINTAS, kemudian semua koneksi lainnya akan menggunakan tabel *routing* utama dan *default*.

3.3.2. Pengujian pada Sisi Klien

Untuk melakukan pengujian terhadap hasil setting, maka akan dilakukan ping dari sisi internet menggunakan koneksi internet eksternal. Proses ping dilakukan terhadap masing-masing alamat IP untuk masing-masing interface sebagai berikut:

```
server@idnet:~$ ping 182.23.35.114
PING 182.23.35.114 (182.23.35.114) 56(84) bytes of data.
64 bytes from 182.23.35.114: icmp_seq=1 ttl=49 time=922 ms
64 bytes from 182.23.35.114: icmp_seq=2 ttl=49 time=245 ms
64 bytes from 182.23.35.114: icmp_seq=4 ttl=49 time=1103 ms
```

```
server@idnet:~$ ping 125.165.41.65
PING 125.165.41.65 (125.165.41.65) 56(84) bytes of data.
64 bytes from 125.165.41.65: icmp_seq=1 ttl=62 time=514 ms
64 bytes from 125.165.41.65: icmp_seq=2 ttl=62 time=605 ms
64 bytes from 125.165.41.65: icmp_seq=3 ttl=62 time=514 ms
```

Berdasarkan hasil ping tersebut diatas, maka terlihat bahwa hasil konfigurasi dapat efektif sehingga permintaan yang masuk ke masing-masing interface akan mendapat jawaban dari masing-masing *interface*.

4. Kesimpulan

Server Linux yang memiliki koneksi internet *multi-homed* untuk tujuan failover layanan membutuhkan masing-masing interface koneksi internet dapat aktif menjawab permintaan yang masuk melalui interface masing-masing. Untuk tujuan tersebut dapat menggunakan fitur policy routing yang tersedia sejak kernel Linux 2.2. dengan menambahkan tabel routing lokal untuk masing-masing interface yang dilanjutkan dengan pembuatan aturan routing yang berlaku bagi masing-masing interface.

Shell-script hasil rancangan dapat digunakan untuk konfigurasi suatu server Linux yang memiliki dua koneksi ke internet sehingga masing-masing koneksi dapat aktif melayani permintaan koneksi dari internet pada saat yang sama.

5. Saran

Agar penyediaan hasil setting dapat efektif untuk tujuan failover, maka perusahaan perlu berlangganan fasilitas failover nama domain yang tersedia secara komersil.

Daftar Pustaka

- [1] Marc Goodman, SMEs Use Multiple ISPs for Greater Internet Reliability, Disaster Recovery Journal, Spring 2010 Volume 23 Number 2, 2010.
 - [2] Bert Hubert et. al., Linux Advanced Routing & Traffic Control How To Ver 1.0, <http://www.opencontent.org/openpub/>, 2002.
 - [3] Craig Hunt, TCP/IP Network Administration Thrid Edition, O'Reilly Media, Inc., United States of America. 2002.
 - [4] Martin A. Brown, Guide to IP Layer Network Administration with Linux, <http://linux-ip.net/html/index.html>, March 2007.
-