ANALISIS PERANCANGAN DAN IMPLEMENTASI *FIREWALL* DAN *TRAFFIC FILTERING* MENGGUNAKAN CISCO ROUTER

Alfin Hikmaturokhman[1,2)], Adnan Purwanto[2)], Rendy Munadi[1)]
[1] Program Pasca Sarjana IT Telkom Bandung Jl. Telekomunikasi, Terusan Buah Batu, Jawa Barat (40257)
[2] Akademi Telkom Sandhy Putra Purwokerto, Jl DI Panjaitan No 128 Purwokerto Jateng .
e-mail : alfin_h21@yahoo.com , mc_pwt@yahoo.com , rnd@ittelkom.ac.id

Abstrak

Cisco Router adalah peralatan utama yang banyak digunakan pada Jaringan Area Luas atau Wide Area Network (WAN). Dengan Cisco Router, informasi dapat diteruskan ke alamat yang berjauhan dan berada di jaringan komputer yang berlainan. Cisco router mempunyai salah satu fungsi yang dapat digunakan sebagai traffic filtering yang apabila diimplementasikan lebih lanjut maka akan menjadi sebuah firewall. Untuk membantu meningkatkan pengamanan suatu jaringan yang ada pada suatu perusahaan/instansi dengan cara yang mudah sehingga jaringan pada suatu perusahaan dapat terlindungi dari ancaman-ancaman yang bersifat merusak, menginfeksi data-data komputer penting di perusahaan tersebut dengan memanfaatkan fungsi dari Cisco Router 1721 series yaitu fungsi Access List. Hasil dari penelitian ini adalah Extended access listyang diterapkan pada Router-router pada jaringan akan membantu menentukan alamat sumber dan tujuan serta protocol dan nomer port yang mengidentifikasikan aplikasi. Dengan menggunakan Aceess List tipe ini akan lebih efisien memperbolehkan user mengakses dan menghentikan pengaksesan host tertentu.
Keywords: Firewall, Router, Access list,Cisco

1. PENDAHULUAN

Di suatu instansi atau perusahaan pastinya banyak sekelompok orang yang menghendaki pengambilan data secara illegal ataupun perusakan jaringan pada perusahaan tertentu. Oleh karena itu dibutuhkan suatu penangkal yang dapat melindungi data ataupun dokumen penting, dikenalah *firewall* dan juga *traffic filtering*. *Firewall* sendiri mengandung pengertian sebagai "pos pemeriksa" yang mengevaluasi trafik-trafik yang keluar dan masuk diantara jaringan *internet* atau privat dengan dunia luar, mengizinkan trafik-trafik tertentu dan memblok yang lainnya.

Tanpa *firewall*, semua komputer berpeluang untuk diakses siapapun dari *internet*. Seseorang yang mengetahui *address* komputer tersebut dapat dengan leluasa mengakses *Telnet* atau menyerang jaringan dengan trafik-trafik yang sifatnya merusak. Dengan adanya *firewall*, keadaannya akan berbeda. seseorang dapat menentukan *rule* keamanan (*security rule*) yang "menuntut" kepatuhan *user* manapun.

Sedangkan *trafic filtering* tidak lain merupakan sebuah teknik untuk mengontrol trafik-trafik yang di-*forward* ke dan dari sebuah jaringan melintasi *router*. Fungsi ini melibatkan perancangan *policy-policy* keamanan. Pada implementasinya *traffic filtering* ini akan di rancang untuk membentuk *environment firewall*.

Dengan adanya implementasi dan perancangan *firewall* dan *traffic filtering* adalah untuk meningkatkan pengamanan suatu jaringan yang ada pada suatu perusahaan/instansi dengan cara yang mudah dengan memanfaatkan fungsi dari *Cisco Router* 1721 *series.*

2. METODE PENELITIAN

Penelitian yang dilakukan dengan cara mensimulasikanya terlebih dahulu menggunakan Packet Tracer kemudian membangun jaringan real yang terdiri dari 3 Router Cisco Router 1721 series beserta server dan workstation.

A. *Firewall*

Firewall atau tembok-api adalah sebuah sistem atau perangkat yang mengizinkan lalu lintas jaringan yang dianggap aman untuk melaluinya dan mencegah lalu lintas jaringan yang tidak aman. Umumnya, sebuah tembok-api diterapkan dalam sebuah mesin terdedikasi, yang berjalan pada pintu gerbang (*gateway*) antara jaringan lokal dan jaringan lainnya. Tembok-api umumnya juga digunakan untuk mengontrol akses terhadap siapa saja yang memiliki akses terhadap jaringan pribadi dari pihak luar .

Parameter proteksi

a. IP *address*
b. *Domain Name*
c. Protokol
d. *Port*

B. Pengenalan Cisco Router

Cisco Router adalah peralatan utama yang banyak digunakan pada Jaringan Area Luas atau *Wide Area Network* (WAN). Dengan *Cisco Router*, informasi dapat diteruskan ke alamat yang berjauhan dan berada di jaringan komputer yang berlainan

Gambar 1 Cisco router 1721

C. *Cisco* IOS

Cisco IOS (*Internetwork Operating System*), yaitu suatu sistem operasi yang berfungsi untuk mengatur dan mengkonfigurasi *Cisco Router*. Seperti sistem operasi *DOS* untuk komputer, *Cisco* IOS menggunakan perintah baris (*command line*) untuk menjalankan suatu perintah.

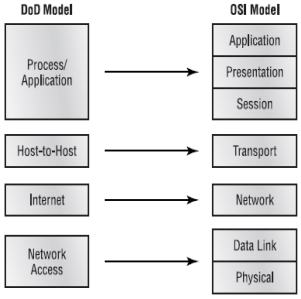
D. Tingkat Akses

1. *User* EXEC *Mode*
   Tingkatan pertama yang dimasuki setelah berhubungan dengan *router,* ditandai oleh **Router> prompt**.
2. *Privileged* EXEC *Mode*
   Dengan mengetikkan perintah *enable* dari *user* EXEC*mode* yang ditandai dengan **Router#prompt**. Pada tingkat *privileged mode* ini konfigurasi-konfigurasi *router* dapat diperiksa dan juga bisa masuk ke *global configuration mode.*
3. *Global Configuration Mode*
   Pada tingkat ini, hampir semua ragam konfigurasi *router* dapat diolah. Cara masuk ke konfigurasi *global* yaitu dengan mengetikkan perintah *configuration terminal* atau *config t* dari **router#prompt**.
4. *Interface Configuration Mode*
   Interface configuration mode adalah suatu *mode* yang digunakan untuk mengkonfigurasikan suatu *interface* tertentu.

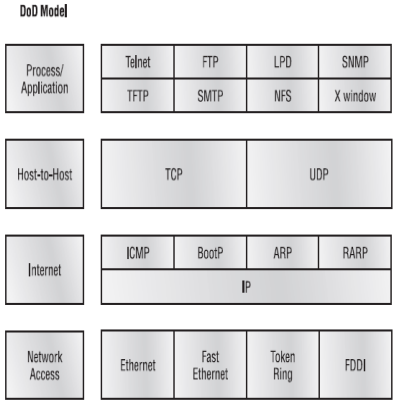
E. TCP/IP dan Model DoD [5]

Pada dasarnya model Dod adalah versi pemadatan model OSI, yang terdiri dari 4 dan bukan tujuh layer, yaitu :

a. Layer Application
b. Layer Host-to-Host
c. Layer Internet
d. Layer Network Access



Gambar 2 Model DoD dan OSI



Gambar 3 *Protocol* TCP/IP

F. Pemberian IP *Address*

Pada LAN umumnya peralatan komputer berada di dalam satu jaringan yang sama. Sedangkan pada WAN, peralatan komputer tersebut berada di dalam jaringan atau *subnet* yang berbeda-beda dan bahkan dengan menggunakan protokol yang berbeda-beda pula. Agar paket-paket data dari jaringan lokal dapat disampaikan ke jaringan lain, perlu menggunakan *router* karena *hub* tidak mampu untuk meneruskan paket-paket ke jaringan yang berlainan atau protokol yang berbeda-beda.

Router meneruskan paket-paket berdasarkan atas alamat-alamat logika (IP *Address*) yang diperolehnya. Sebelum *router* dapat berhubungan satu dengan yang lain dalam jaringan WAN, *interface* dari *router* yang akan dihubungkan tersebut harus diberi IP *Address static*, yang merupakan alamat yang digunakan oleh *router* untuk meneruskan paket-paket. Pada penelitian ini digunakan IP *address* yang digunakan bertipe IPv4.

Tabel 1 Porsi kelas-kelas IPv4 [1]

	8 bit	*8 bit*	*8 bit*	*8 bit*
Kelas A	*Network*	*Host*	*Host*	*Host*
Kelas B	*Network*	*Network*	*Host*	*Host*
Kelas C	*Network*	*Network*	*Network*	*Host*

Subnet mask

Agar perencanaan alamat subnet bekerja, semua mesin jaringan harus tahu bagian mana dari alamat *host* yang akan digunakan sebagai alamat *subnet*. *Subnet mask* adalah sebuah nilai 32-*bit* yang memungkinkan penerima paket IP membedakan bagian *ID* (identifikasi) *network* dari sebuah alamat IP dengan bagian ID *host* dari alamat IP tersebut.

Tabel 2 Pembagian *network* dan *host address* berdasarkan *subnet mask default* [1]

Kelas	*Format*	*Default Subnet*
A	*Net.Node.Node.Node*	255.0.0.0
B	*Net.Net.Node.Node*	255.255.0.0
C	*Net.Net.Net.Node*	255.255.255.0

G. *Protocol routing* [5]

Supaya suatu paket dapat mencapai tujuannya, diperlukan suatu peralatan untuk mengatur paket-paket tersebut agar mencapai tujuannya dengan jalan yang tersingkat. Untuk itu digunakan *router* yang fungsi utamanya adalah untuk menentukan jalur dan meneruskan paket-paket dari suatu jaringan ke jaringan lain. Agar *router* dapat mengetahui bagaimana meneruskan paket-paket ke alamat yang dituju dengan menggunakan jalur yang baik, *router* menggunakan peta atau tabel *routing*

***Routing Information Protocol* (RIP)**

RIP (*Routing Information Protocol*) adalah *routing protocol* yang termasuk jenis *distance* vektor. RIP menggunakan jumlah lompatan (*hop count*) sebagai *metric* dengan 15 *hop* maksimum. Jadi *hop-count* yang ke-16 tidak dapat tercapai dan *router* akan memberikan pesan *error* ? *destination is unreachable*? (tujuan tidak tercapai).

H. *Access list* (ACL) [5]

Cisco Router mengunakan metode yang disebut "*packet filter*" untuk mengatur akses lalulintas data melewati *router*. Paket-paket data yang datang ke *router* difilter (disaring) untuk menentukan paket data mana yang akan ditolak dan paket data mana yang akan diteruskan ke suatu alamat jaringan (*network address*) atau ke suatu alamat komputer (*host address*) tertentu. Metode paket filter yang dipakai oleh *Cisco Router* menggunakan daftar akses yang berfungsi sebagai berikut [5]:

a. Setiap paket data yang diterima oleh *router* dicocokkan dengan isi daftar akses yang diterapkan pada *router interface* baris per baris
b. Bila ditemukan suatu baris yang cocok, maka paket data tersebut diteruskan atau ditolak berdasarkan perintah dari baris tersebut
c. Jika tidak ada baris yang cocok, perlu diketahui bahwa semua daftar *access list* jika dibuat, secara otomatis akan diakhiri dengan perintah ?*implicit deny*? yang berarti jika ijin tidak disebutkan secara khusus dalam daftar akses maka paket akan ditolak.

1. Daftar Akses IP *Extended* [5]

Daftar Akses IP *Extended* (*Extended* IP *Access list*) lebih rumit dan memiliki lebih banyak parameter yang dapat diatur antara lain: alamat pengirim (source *address*), alamat penerima (destination *address*), *port number*, dan protokol seperti dibawah ini:

Router(config)#access-list <nomer daftar akses IP extended> <permit/deny> <protocol> <source *address*> <wildcard mask> <destination *address*> <wildcard mask> <operator> <information *port*>

Lalu diterapkan pada *interface* yang digunakan, perintahnya adalah

Router(config)#<interface yang digunakan>

Router (config)#IP access-group <nomer daftar akses IP extended> <in/out>

3. HASIL DAN PEMBAHASAN

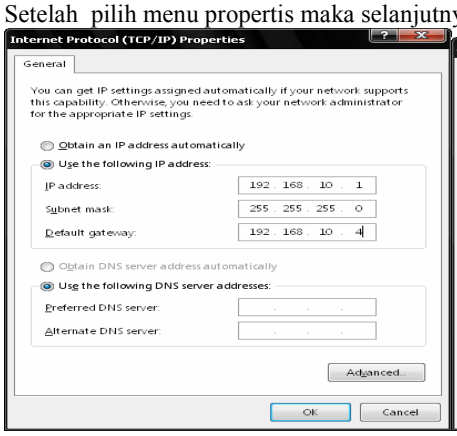
A. Implementasi *firewall* dan *traffic filtering*

***1. Setting* IP *address* komputer.**

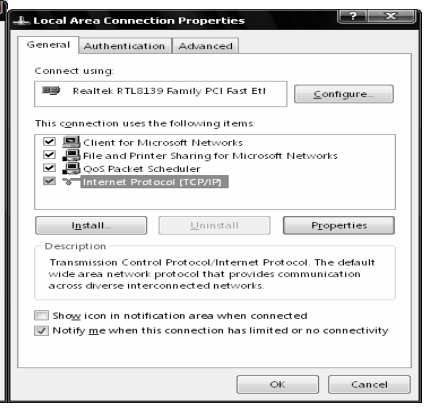
Agar komputer dapat diakses dari komputer lain maupun dari *Router* maka perlu diberi IP *Address*. IP *address* disini berfungsi sebagai alamat dari suatu *device* baik itu alamat sumber maupun digunakan alamat tujuan. Konfigurasi IP *address* pada komputer langkah-langkahnya adalah sebagai berikut :

a. Pilih *Setting* > *network conection* > *Properties*. Maka akan keluar tampilan seperti gambar 4. Pilih *Internet Protocol* (TCP/IP) lalu *Properties* untuk masuk ke pengaturan selanjutnya.

b. Setelah pilih menu propertis maka selanjutnya akan tampil *Internet Protocol* (TCP/IP) *Properties*.



Gambar 5 *Internet protocol* (TCP/IP) *properties*.



Gambar 4 LAN *properties*

2. Setting Router

Mengkonfigurasi *Router*

Router tidak mempunyai layar monitor untuk berinteraksi dengan *network administrator*, oleh karena itu, kita membutuhkan sebuah *PC* untuk men-*setup* sebuah *Router*. *PC* tersebut harus disambungkan ke *Router* tersebut dengan salah satu dari cara berikut:

1. Melalui *Console port*
2. Melalui *Auxilary Port*
3. Melalui *Telnet*

Pada perancangan penelitian ini hanya menggunakan *Console port* dalam mengkonfigurasi *Router*.

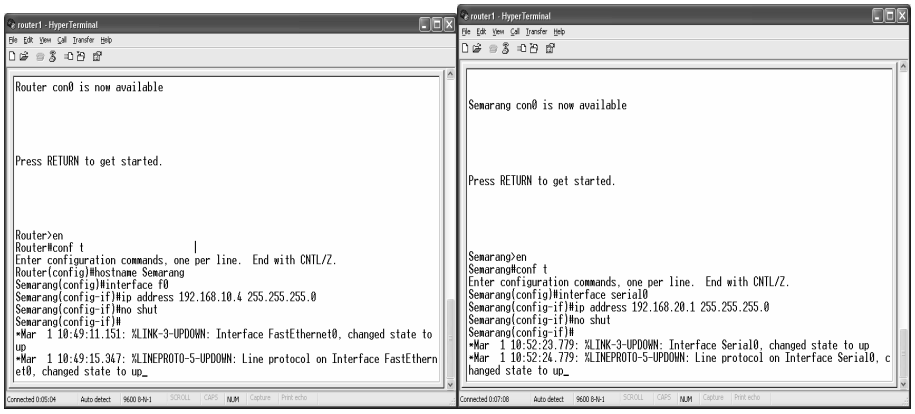
Men-konfigurasi *Router* melalui *port Console*

Console port adalah sebuah *port* pada *Router* yang disediakan untuk menghubungkan *Router* tersebut pada "dunia luar". Sebuah kabel *Roll Over* dibutuhkan untuk menghubungkan *Serial Interface*pada *PC* dan *Console port* pada *Router* tersebut. Setelah *Router* terhubung dengan *PC*, *Router* dapat dikonfigurasi dengan menjalankan aplikasi *HyperTerminal* dari *PC*.

a. Konfigurasi hostname dan *Interface fastethernet* 0.

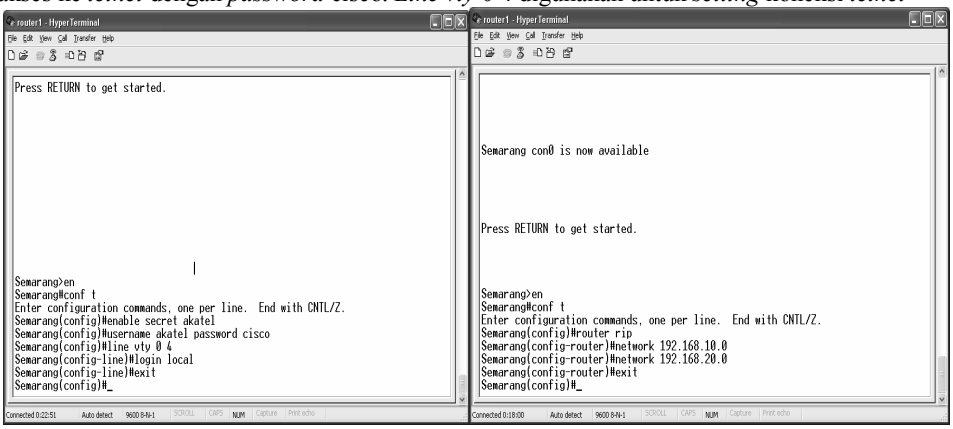
Untuk *Setting hostname* dan *Interface* masuk ke *mode previledge*. langkah-langkahnya adalah sebagai berikut :

1. masuk ke *mode previledge* dengan cara ketik "*enable*", setelah itu ketik *hostname* untuk memberi nama dari *Router* yang digunakan dan untuk *interface*nya seperti pada gambar 6. konfigurasi *interface fastethernet* 0 dengan IP *address* 192.168.10.4. Perintah "*no shut*" digunakan untuk mengaktifkan *interface fastethernet* tersebut.



Gambar 6 Konfigurasi *Interface Fastetherne* **Gambar 7** Konfigurasi *interface serial* 0

2. Konfigurasi *Interface serial* 0 dengan IP address 192.168.20.1. perintah "*no shut"* digunakan untuk mengaktifkan *Interface.*
3. Konfigurasi *password* dan koneksi *Telnet* dengan *password* akatel. *Username* digunakan untuk akses ke *telnet* dengan *password* cisco. *Line vty* 0 4 digunakan untuk *setting* koneksi *telnet*

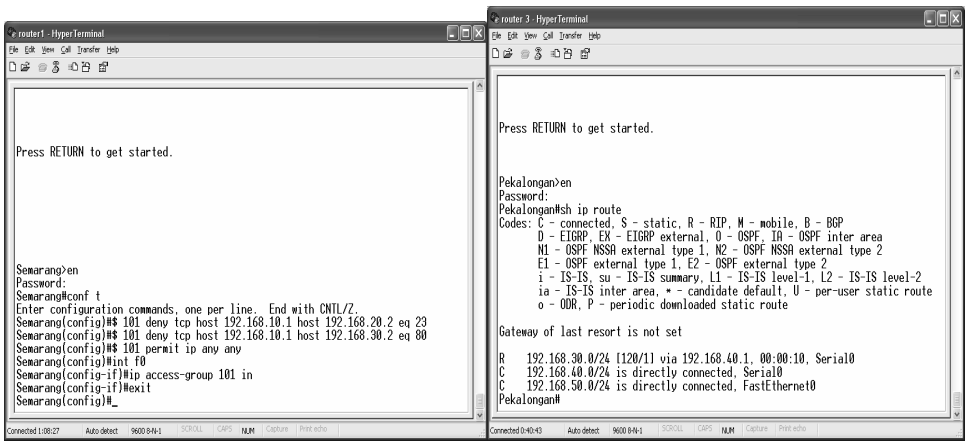


Gambar 8 Konfigurasi *password* dan koneksi *Telnet* **Gambar 9** Konfigurasi RIP

4. Konfigurasi *Routing protocol*. *Routing Protocol* yang digunakan adalah *dynamic Routing* dengan jenis RIP (*Routing Information Protocols*). Untuk langkah-langkahnya adalah seperti pada gambar 9. Router rip adalah perintah yang digunakan untuk konfigurasi RIP. Selanjutnya network yang terhubung dari router ke jaringannya adalah 192.168.10.0 dan 192.168.20.
5. *Setting Access-List*

Access-List yang digunakan adalah *Extended Access-List*. Langkah-langkah konfigurasinya adalah seperti pada gambar 10. Perintah pada gambar 10 menunjukkan bahwa *Access list* yang digunakan adalah *extended access list* dapat dilihat dari nomer rangenya yaitu 101 bertujuan menolak *host* 192.168.10.1 untuk mengakses *telnet* (23) ke *host* 192.168.20.2.

1. Perintah selanjutnya dengan nomer *range* 101 bertujuan menolak *host* 192.168.10.1 untuk mengakses http (80) ke *host* 192.168.30.2.
2. *Permit IP any any* menunjukkan pengijinan akses ke *telnet* dan http selain host 192.168.10.1
3. *Access list* tersebut diterapkan pada *Interface* F0



Gambar 10 Konfigurasi *Extended* ACL

Gambar 11 Menampilkan *routing* ip yang digunakan

B. Analisis cara kerja sistem perancangan

Setelah mengkonfigurasi seluruh *device* yang digunakan sekarang saatnya menganalisis cara kerja dari sistem perancangan inplementasi *firewall* dan *Traffic Filtering* Adapun IP *address* dari masing-masing *device* ditunjukkan pada tabel 3.

Tabel 3 IP *address* masing-masing *device*

.	Interface			Subnet Mask
Device	FastEthernet 0	Serial 0	Serial 1	
PC 1	192.168.10.1			255.255.255.0
PC 5	192.168.50.1			255.255.255.0
PC Server	192.168.30.2			255.255.255.0
Router Semarang	192.168.10.4	192.168.20.1		255.255.225.0
Router Jakarta	192.168.30.4	192.168.40.1	192.168.20.2	255.255.255.0
Router Pekalongan	192.168.50.4	192.168.40.2		255.255.255.0

Setelah IP *address* tersebut diterapkan pada semua *device* tidak semua *device* dapat terhubung hanya *device* yang mempunyai IP *address* dalam jaringan yang sejenis saja yang dapat terhubung. Oleh karena itu, digunakanlah salah satu fungsi dari IOS *Router* yang berguna menghubungkan jaringan yang berbeda jaringan yaitu RIP (*Routing Information Protocols)*. Cara kerja dari RIP itu sendiri adalah sebagai berikut :

1. RIP merupakan sebuah *Routing Protocol* jenis *distance-vector*. *Protocol distance vector* menemukan jalur terbaik ke sebuah *remote* dengan menilai jarak. *Route* dengan hop yang paling sedikit menujukan *network* yang dituju akan menjadi *Route* terbaik.
2. RIP secara *default* memilki sebuah nilai jumlah hop maksimum yang diijinkan yaitu 15, yang berarti nilai 16 dianggap tidak terjangkau.
3. RIP v1 menggunakan hanya *classful Routing*, yang berarti semua alat di *network* harus menggunakan *subnet mask* yang sama.
4. RIP tidak bekerja berdasarkan kecepatan, melainkan berdasarkan jumalh hop minimum.

Untuk lebih memahami RIP perhatikan contoh tabel *Routing* dibawah ini (diambil dari *Router* Pekalongan) :
Ketikkan perintah sh IP *Route* pada *mode previledge*, maka akan tampilannya seperti gambar 11.

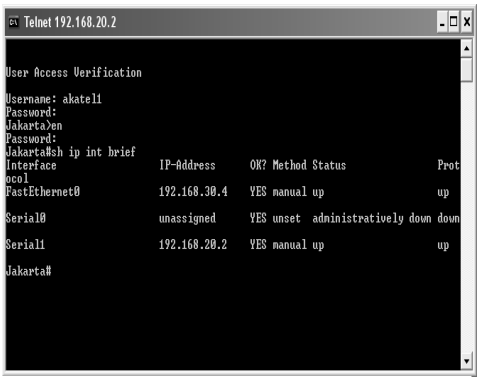
Pada gambar 11 dapat dianalisa bahwa pada *Routing table* yang telah dibentuk pada *Router* Pekalongan memiliki entri menggunakan kode R dan C. Kode C berarti *network* terhubung secara langsung (*direcly connection)*. Kode R berarti bahwa *network* menambahkan secara dinamis menggunakan *Routing Protocol* RIP. Angka [120/1] adalah *administrative distance* dari *Route* (120) bersama dengan jumlah hop ke *network* tersebut (1 hop).

Administrative distance digunakan untuk mengukur apa yang disebut *trustworthiness* (tingkat kepercayaan) dari informasi *Routing*. *Router* yang memiliki AD terendah maka itu yang akan dimasukkan di *Routing tabel*.

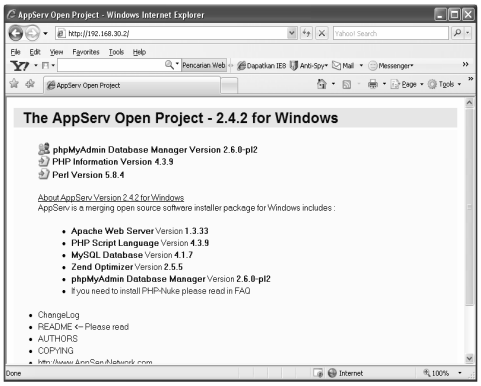
Setelah *Router* satu dengan yang lain dapat berkomunikasi dengan baik maka selanjutnya adalah melakukan pembatasan akses dari sebuah paket data. Pembatasan akses ini dikenal dengan istilah *Traffic Filtering* yang apabila diimplementasikan lebih lanjut maka akan menjadi sebuah *firewall.* Sesuai dengan skenario yang telah dibuat yaitu

a. Menolak *host* PC 1, menolak *host* PC 5, untuk mengakses *Telnet* pada *Router* Jakarta
b. Menolak *host* PC 1, menolak *host* PC 5, untuk mengakses *http* pada *Server* Jakarta
c. Mengijinkan yang lainnya untuk mengakses pada *Telnet* dan *http* pada *Router* dan *Server* Jakarta.

Untuk *Traffic Filtering* digunakan salah satu fitur IOS *Router* yaitu *Access-List*. *Access-List* yang digunakan disini adalah ACL jenis *Extended* ACL.

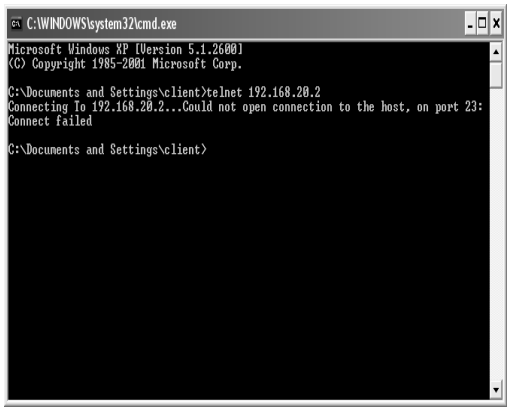


Gambar 12 Akses *telnet* ke *router* jakarta

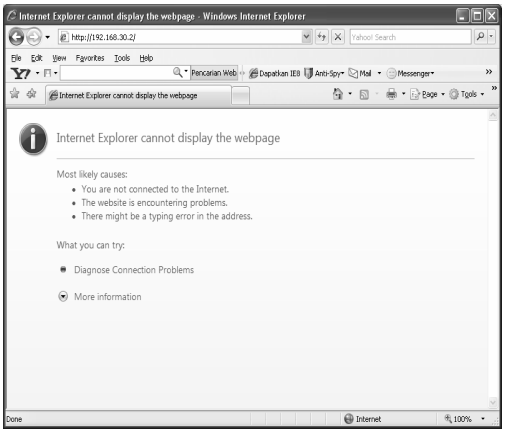


Gambar 13 Akses http ke computer server

Gambar 12 dan 13 menunjukkan bahwa *Telnet* ke *Router* Jakarta dari *Router* semarang dan akses *http* ke komputer *Server* berhasil. Untuk dapat menjalin koneksi *Telnet* sebelumnya harus di *Setting* terlebih dahulu *line vty* , serta isi semua IP *address* pada masing-masing *device*. Sebelum dipasang atau diterapkan *Extended* ACL maka koneksi akan berjalan dengan baik. Apabila sebuah *Router* dapat mengakses *Telnet* maka *Router* tersebut dapat merusak serta mematikan sistem yang ada pada sebuah *Router* dimana *Router* tersebut berperan sebagai *gateway* di suatu jaringan. oleh karena itulah sebaiknya diterapkan *Extended* ACL. Setelah diterapkannya *access list* maka hasilnya akan seperti pada gambar 14 dan 15.



Gambar 15 koneksi gagal ke server jakarta



Gambar 14 Koneksi gagal ke telnet router jakarta

Gambar 14 menunjukan bahwa setelah diterapkan *Extended* ACL maka koneksi ke *Telnet* gagal. *Extended* ACL memeriksa *destination, source, protocol*. Tidak seperti standart ACL yang hanya dapat memeriksa paket-paket berdasarkan IP *address* sumber. Seperti pada gambar 10 cara kerja dari suatu *Extended* ACL adalah sebagai berikut :

Pada operasi normal saat sebuah paket melintasi *Router*, maka *Router* akan mencari *rute* yang tepat untuk mencapai tujuan dan menetapkan *Interface* mana yang harus digunakan untuk keluarnya paket dari *Router*. Saat menggunakan *access-list*, sebelum paket dapat memasuki atau keluar dari *interface router*, disana telah terdapat filter-filter yang diberlakukan pada *interface* tersebut yang akan menguji atau memeriksa paket.

Sebuah *access-list* terdiri dari daftar *rule* atau *statement* yang secara berurut menguji paket-paket yang keluar masuk. *Rule-rule* ini menguji berbagai informasi spesifik dalam sebuah paket seperti IP *address source*, IP *address destination*, *protocol*. Paket yang masuk diuji terlebih dahulu mengikuti *rule-rule* yang ditetapkan hingga kondisi tertentu terpenuhi. Jika tidak ada yang terpenuhi pada *rule* pertama maka paket diserahkan ke baris kedua. Jika tidak ada kondisi yang sesuai, maka terdapat konsekuensi "*deny all*".
Adapun penjelasan dari listing konfigurasi adalah sebagai berikut :

1. Pilih sebuah nomer untuk membuat *extended* ACL. Nomer *extended* ACL berada dalam range 100-199, untuk penelitian ini digunakan nomer 101
2. Gunakan *statement deny*

Semarang(config)#Access-List 101 deny

3. Karena akan menolak *Telnet* dan *http* maka harus memilih TCP sebagai *Protocol layer transport*. Karena *http* dan *Telnet* berada pada *Protocol* TCP.

Semarang(config)#Access-List 101 deny TCP

4. Tambahkan alamat IP sumber yang ingin disaring, kemudian tambahkan alamat host IP tujuan

Semarang(config)#Access-List 101 deny TCP host 192.168.10.1 host 192.168.20.2

5. Tambahkan perintah *eq Telnet* untuk menyaring host 192.168.10.1 melakukan *Telnet* ke 192.168.20.2.

Semarang(config)#Access-List 101 deny TCP host 192.168.10.1 host 192.168.20.2 eq Telnet

6. Tambahkan perintah *eq http* (80) untuk menyaring *host* 192.168.10.1 melakukan akses *http* ke 192.168.30.2.

Semarang(config)#Access-List 101 deny TCP host 192.168.10.1 host 192.168.30.2 eq http

7. Sangat penting untuk menambahkan baris ini selanjutnya untuk membuat *statement permit*.

Semarang(config)#Access-List 101 permit IP any any

8. *Statement permit* tersebut harus diterapkan karena jika hanya menambahkan *statement deny,* semua akan ditolak.
9. Terapkan *Access-List* ke *FastEthernet* 0 pada *Router* Semarang untuk menghentikan lalu lintas *Telnet* dan akses ke *http* pada saat sampai pada *Interface* yang pertama.

Semarang(config)#int f0
Semarang(config-if)#IP access-group 101 in
Semarang(config-if)#^Z

4. KESIMPULAN

1. Rangkaian sistem yang dibangun dari simulasi menggunakan *packet tracer* 5.0 dan kemudian diterapkan pada cisco router 1721 berfungsi untuk mengijinkan paket data tertentu maupun menolak paket data tertentu juga.
2. Sistem penolakan maupun pengijinan suatu paket data menggunakan salah satu fitur dari OSI router yaitu Access-List. Dalam hal ini access-list berperan sebagai traffic filtering yang apabila diimplementasikan lebih lanjut akan menjadi sebuah *firewall.*
3. *Access-list* yang digunakan bertipe *Extended access list* dimana *extended access-list* akan membantu menentukan alamat sumber dan tujuan serta *protocol* dan nomer *port* yang mengidentifikasikan aplikasi. Dengan menggunakan tipe ini akan lebih efisien memperbolehkan *user* mengakses dan menghentikan pengaksesan *host* tertentu.

5. DAFTAR PUSTAKA

[1]Arifin, Zaenal, 2003. "Langkah mudah mengkonfigurasi Router Cisco", Andi OFFSET, Yogyakarta
[2]*Gateway*.http://id.wikipedia.org/wiki/Gateway.htm (diakses tanggal 18 Maret 2009 pukul 10.39)
[3]Hangga, Fraedi, , 2008"Laporan Praktek Kerja Lapangan II", Akademi Teknik Telekomunikasi Sandhy Putra, Purwokerto.
[4]*IPv4*.http://id.wikipedia.org/wiki/Alamat_IP_versi_4.htm (diakses tanggal 18 Maret 2009 pukul 10.38)
[5]Lammlee, Todd, 2005. "CCNA Cisco Certified *Network* Associate Study Guide", PT Elex Komputindo, Jakarta,
[6]*Protokol*.http://id.wikipedia.org/wiki/Protokol_internet.htm (diakses tanggal 14 Maret 2009 pukul 17.52)
[7]Purbo, W.Onno dan Tony Wiharjito, 2000 ,"Buku Pintar Internet Keamanan Jaringan Internet", PT Elex Media Komputindo, Jakarta,.
[8]Rafiudin, Rahmat, 2006, "Membangun Firewall dan Traffic filtering berbasis CISCO,", Andi OFFSET, Yogyakarta,