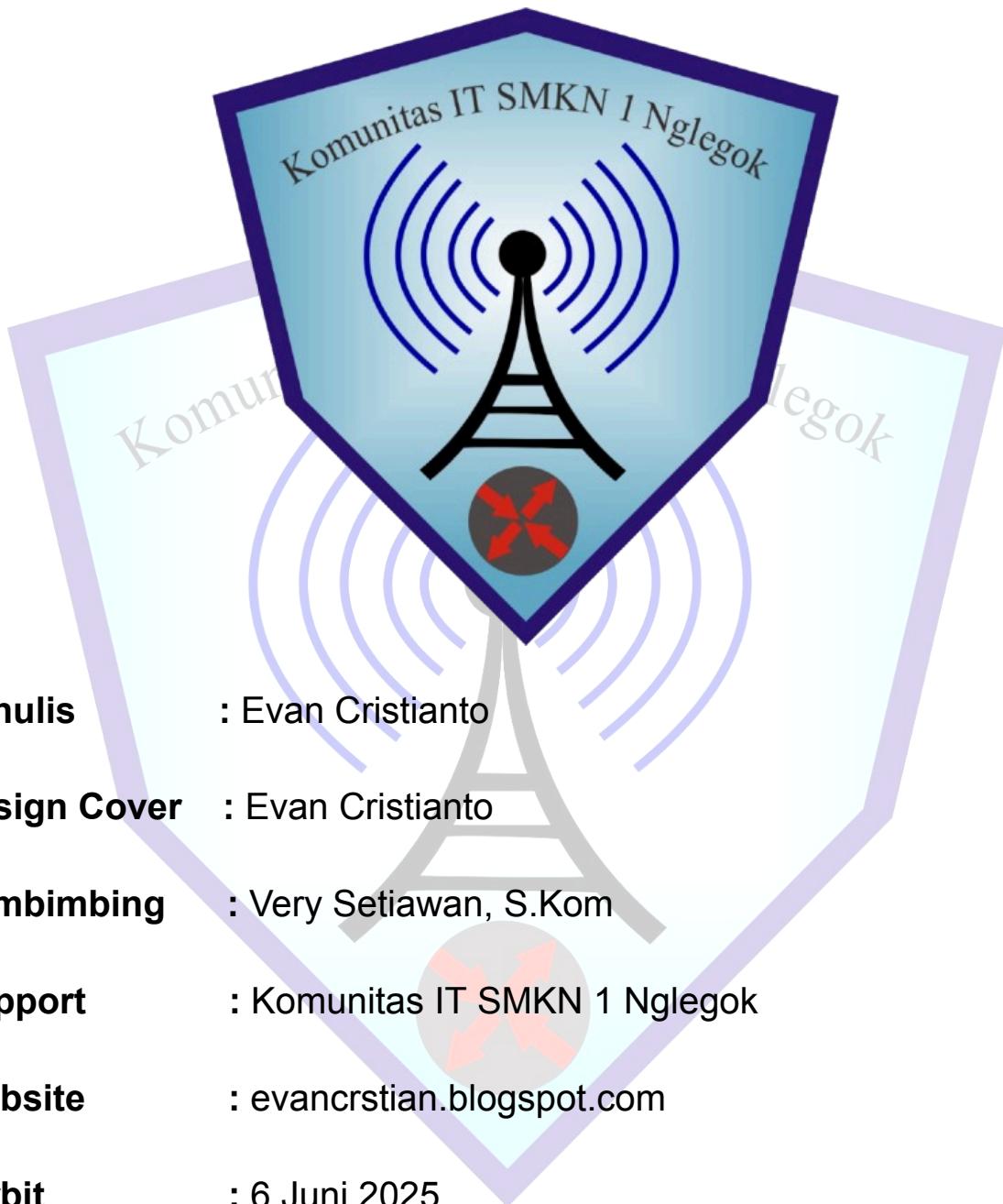




CCNA

FORKITSBOOK





KATA PENGANTAR

Puji syukur atas kehadirat Tuhan yang Maha Esa atas limpah dan Rahmatnya sehingga Buku yang berjudul "For KITS Book: CCNA" telah selesai ditulis dan disusun dan dapat saya selesaikan dengan tepat waktu. Semogabuku ini dan ilmu yang saya dapat bisa bermanfaat bagi kita semua.

Saya mengucapkan terima kasih banyak kepada seluruh pihak yang telah membantu saya dalam membuat buku ini, dan sekali lagi saya sangat berterima kasih khususnya kepada:

1. Tuhan yang Maha Esa
2. Kedua orang tua yang selalu memberikan semangat
3. Pembina Komunitas IT, Bapak Very Setiawan, S.kom
4. Seluruh Alumni, Kakak kelas, dan teman teman Komunitas IT

Saya berharap buku ini dapat berguna bagi semua orang tanpa terkecuali. Karena buku yang saya tulis masih belum dikatakan sempurna. Saya berharap saran serta kritikan dari anda sekalian agar kedepannya menjadi lebih baik. Jadi bagi kalian yang menemukan kesalahan dalam buku saya, mohon saran dan kritikan kalian kirim ke email saya evancristianto629@gmail.com. Terima Kasih.

Hormat Saya

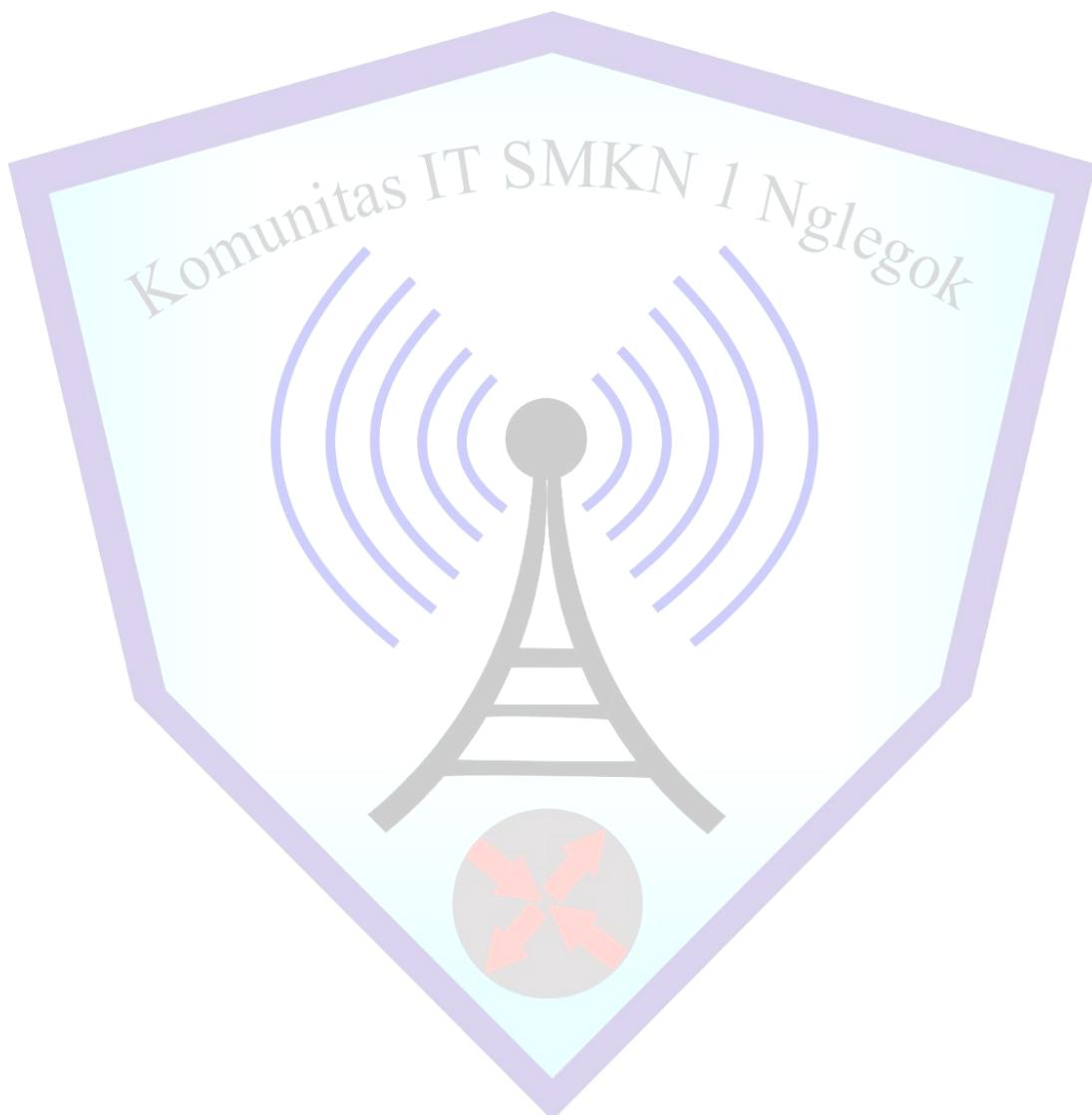


Evan Cristianto

DAFTAR ISI

KATA PENGANTAR.....	3
DAFTAR ISI.....	4
Lab 1 : User mode, Privilege mode, Global configure mode.....	7
USER MODE.....	7
PRIVILAGE MODE.....	7
GLOBAL CONFIGIRE MODE.....	8
Lab 2 : Cisco Enhanced Editing Command.....	9
Lab 3 : Disable IP Domain Lookup.....	10
Lab 4 : Konfigurasi Waktu CISCO.....	11
Lab 5 : Konfigurasi Hostname.....	12
Lab 6 : Melihat Konfigurasi.....	12
Lab 7 : Menyimpan Konfigurasi.....	13
Lab 8 : Menghapus Konfigurasi.....	14
Lab 9 : Konfigurasi Password Cisco.....	15
LAB 10 : Switch Vlan Cisco.....	18
LAB 11 : Vlan Trunking.....	21
LAB 12 : Allwed Trunk.....	23
LAB 13 : Dynamic Trunking Protocol.....	25
LAB 14 Virtual Trunking Protocol.....	27
LAB 15 Trunk Pada MLS.....	29
LAB 16 Security Port Switch.....	30
LAB 17 Spanning Tree Protocol Switch.....	32
LAB 18 Per VLAN Spanning Tree ProtocolP.....	35
LAB 19 Spanning Tree Portfast.....	40
LAB 20 Etherchannel LACP.....	42
LAB 21 Etherchannel PAGP.....	45
LAB 22 Etherchannel Layer 3.....	47
LAB 23 Enable Telnet di Switch.....	48
LAB 24 Enable SSH Switch.....	50
LAB 25 Inter Vlan Routing.....	52
LAB 26 DHCP Server Router.....	54
LAB 27 DHCP Server Pada Switch MLS.....	57
LAB 28 DHCP Relay Cisco.....	58
LAB 29 Routing Pada Switch MLS.....	60
LAB 30 Routing Static Cisco.....	63

LAB 31 Routing Dynamic OSPF.....	65
LAB 32 Routing Dynamic EIGRP.....	67
LAB 33 Routing Dynamic RIPv2.....	70
LAB 34 Redistribute OSPF dan EIGRP.....	74
LAB 35 Redistribute OSPF dan RIP.....	77
LAB 36 Standar Acces List.....	79
Skenario 1.....	79
LAB 37 Standar Acces List.....	83
Skenario 2.....	83
LAB 38 Blokir Telnet.....	85
1. Langkah pertama konfigurasikan IP address sesuai dengan topologi yang telah di buat	
86	
LAB 39 Named AC.....	87
LAB 40 Extended ACL Blokir Ping.....	90
LAB 41 Extended ACL Blokir Http/Https.....	96
LAB 42 Nat Static Cisco.....	101
LAB 43 NAT Dynamic Cisco.....	103
LAB 44 Nat Dynamic Cisco With Exit Interface.....	105
LAB 45 Konfigurasi HSRP.....	107
LAB 46 Konfigurasi VRRP Cisco.....	109
LAB 47 Konfigurasi GLBP Cisco.....	109
LAB 48 WAN HDLC.....	110
LAB 49 PPP Cisco.....	113
LAB 50 PPP PAP.....	115
LAB 51 PPP CHAP.....	118
DAFTAR PUSTAKA.....	121
Biografi Penulis.....	121



Lab 1 : User mode, Privilege mode, Global configure mode

USER MODE

Untuk pertama yaitu user mode. User mode adalah mode yang akan tampil saat pertama kali kita masuk ke dalam cisco. Mode ini ditandai dengan tanda > .

```
Router>?
Exec commands:
<1-99>      Session number to resume
connect       Open a terminal connection
disable       Turn off privileged commands
disconnect    Disconnect an existing network connection
enable        Turn on privileged commands
exit          Exit from the EXEC
logout        Exit from the EXEC
ping          Send echo messages
resume        Resume an active network connection
show          Show running system information
ssh           Open a secure shell client connection
telnet        Open a telnet connection
terminal      Set terminal line parameters
traceroute   Trace route to destination
```

PRIVILAGE MODE

Privilege mode ditandai dengan tanda "#" setelah hostname. Untuk masuk ke mode ini dari user mode, ketikkan perintah enable.Pada mode ini, kita dapat melakukan berbagai konfigurasi yang lebih kompleks. Jika Anda ingin melihat daftar perintah yang tersedia, ketikkan ?.

```

Router>enable
Router#?
Exec commands:
<1-99>      Session number to resume
auto          Exec level Automation
clear         Reset functions
clock          Manage the system clock
configure      Enter configuration mode
connect        Open a terminal connection
copy           Copy from one file to another
debug          Debugging functions (see also 'undebug')
delete         Delete a file
dir            List files on a filesystem
disable        Turn off privileged commands
disconnect     Disconnect an existing network connection
enable         Turn on privileged commands
erase          Erase a filesystem
exit           Exit from the EXEC
logout         Exit from the EXEC
mkdir          Create new directory
more           Display the contents of a file
no             Disable debugging informations
ping           Send echo messages
reload         Halt and perform a cold restart
--More-- |

```

Kita juga bisa melihat versi cisco ataupun konfigurasi, untuk melihatnya kita bisa menggunakan perintah show

```

Router#show version
Cisco IOS Software, 2800 Software (C2800NM-ADVIPSERVICESK9-M), Version 15.1(4)M4,
SOFTWARE (fcl)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2012 by Cisco Systems, Inc.
Compiled Thurs 5-Jan-12 15:41 15:41 by pt_team

ROM: System Bootstrap, Version 15.1(4)M4, RELEASE SOFTWARE (fcl)
cisco2811 uptime is 2 minutes, 15 seconds
System returned to ROM by power-on
System image file is "flash0:c2800nm-advipservicesk9-mz.151-4.M4.bin"
Last reload type: Normal Reload

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wwl/export/crypto/tool/stqrg.html

If you require further assistance please contact us by sending email to
export@cisco.com.
cisco 2811 (MPC860) processor (revision 0x200) with 60416K/5120K bytes of memory
Processor board ID JAD05190MTZ (4292891495)
2 FastEthernet interface(s)
DRAM configuration is 64 bits wide with parity disabled.
255K bytes of non-volatile configuration memory.
249856K bytes of ATA System CompactFlash 0 (Read/Write)

License Info:

```

GLOBAL CONFIGURE MODE

Global Configuration Mode digunakan untuk melakukan konfigurasi lebih lanjut, seperti menambahkan IP address, mengganti hostname, membuat password, dan lain-lain. Pada mode ini, Anda masih dapat menggunakan perintah-perintah dari Privilege Mode, namun perlu menambahkan kata do sebelum perintah tersebut. Contohnya: do ping {ip address}

```
Router(config)#?
Configure commands:
  aaa           Authentication, Authorization and Accounting.
  access-list   Add an access list entry
  banner        Define a login banner
  bba-group     Configure BBA Group
  boot          Modify system boot parameters
  cdp           Global CDP configuration subcommands
  class-map     Configure Class Map
  clock         Configure time-of-day clock
  config-register Define the configuration register
  crypto        Encryption module
  default       Set a command to its defaults
  dial-peer     Dial Map (Peer) configuration commands
  do            To run exec commands in config mode
  dot11         IEEE 802.11 config commands
  enable        Modify enable password parameters
  end           Exit from configure mode
  ephone        define ethernet phone
  ephone-dn    Configure ephone phone lines (Directory Numbers)
  exit          Exit from configure mode
  flow          Global Flow configuration subcommands
  hostname      Set system's network name
--More--
```



Lab 2 : Cisco Enhanced Editing Command

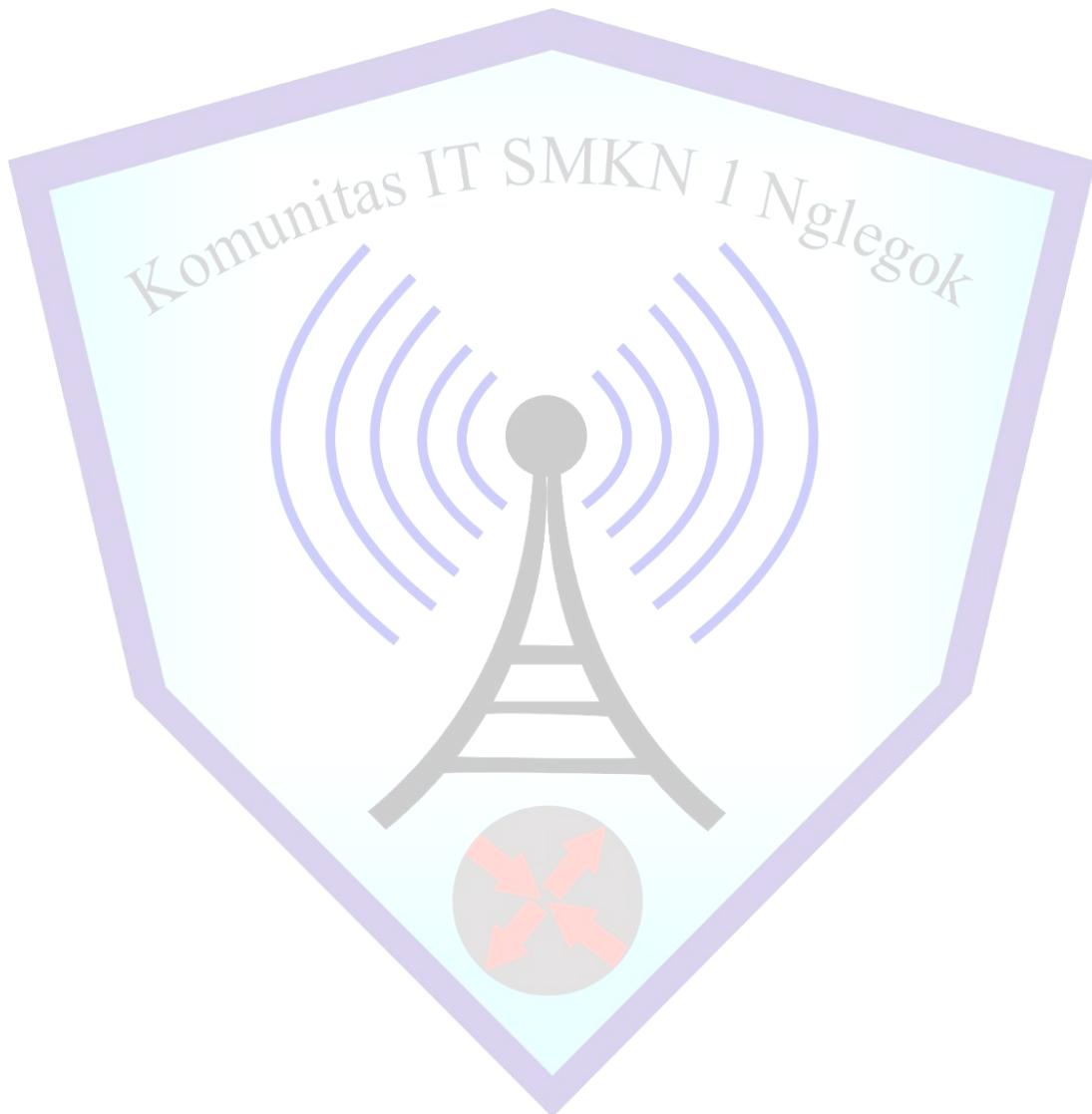
Pada praktik lab kali ini, saya akan menjelaskan tentang Enhanced Editing Command pada perangkat Cisco. Fitur ini merupakan sekumpulan keyboard command yang dapat digunakan untuk mempermudah proses pengeditan konfigurasi di perangkat Cisco.

Ctrl+A	Memindahkan kursor ke awal line
Ctrl+E	Memindahkan kursor ke akhir line
Esc+B	Memindahkan kursor mundur satu kalimat
Ctrl+B atau arah kiri	Memindahkan kursor mundur satu karakter
Ctrl+F atau arah kanan	Memindahkan kursor maju satu karakter
Esc+F	Memindahkan kursor maju satu kalimat
Ctrl+D	Menghapus satu karakter
Backspace	Menghapus satu karakter
Ctrl+R	Menampilkan kembali line
Ctrl+U	Menghapus line
Ctrl+W	Menghapus kalimat
Ctrl+Z	Kembali mundur ke satu mode
Tab	Melanjutkan perintah yang kita ketik
Ctrl+P atau arah atas	Menunjukkan perintah yang pernah kita masukkan (mundur)
Ctrl+N atau arah bawah	Menunjukkan perintah yang pernah kita masukkan (maju)
?	Menunjukkan perintah apa saja yang bisa kita gunakan

Lab 3 : Disable IP Domain Lookup

Ketika kita melakukan kesalahan dalam konfigurasi cisco. Tentunya kita harus menunggu lama untuk menghentikannya, ada cara agar lebih cepat yaitu dengan mengetikkan Ctrl+shift+6

```
Router#conf t
Translating "conf t"...domain server (255.255.255.255) % Name lookup aborted
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#no ip domain lookup
```



Lab 4 : Konfigurasi Waktu CISCO

Pada cisco kita juga mengatur waktu sesuai dengan yang kita inginkan misalnya pada contoh di bawah ini: Untuk keterangannya kita bisa mengetikkan `clock set {jam}:{detik}:{bulan}:{tahun}`

```
| Router#clock set 12:00:00 apr 28 2025  
Router#
```



Lab 5 : Konfigurasi Hostname

Jika kita bosan dengan nama hostname pada cisco kita juga bisa mengubah hostname dengan menggunakan command `hostname {nama hostname yang kalian inginkan}` supaya kita bisa lebih mudah mengenali router saat melakukan konfigurasi

```
Router(config)#hostname Evan-14  
Evan-14(config) #
```

Lab 6 : Melihat Konfigurasi

Jika dalam suatu kondisi kita ingin melakukan troubleshooting tentunya kita harus tau apa konfigurasi yang telah kita buat, oleh karena itu kita dapat melihat menggunakan perintah show running config untuk menunjukan konfigurasi cisco yang telah kita buat

```
Evan-14#show running-config
Building configuration...

Current configuration : 623 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Evan-14
!
!
!
!
!
!
!
ip cef
no ipv6 cef
!
!--More-- |
```

Lab 7 : Menyimpan Konfigurasi

Pada perangkat Cisco, jika kita tidak menyimpan konfigurasi yang telah dibuat, maka seluruh konfigurasi tersebut akan hilang apabila router tiba-tiba mati atau direstart. Oleh karena itu, sangat penting untuk menyimpan konfigurasi setelah melakukan perubahan.

Ada dua cara untuk menyimpan konfigurasi, yaitu:

1. Menggunakan perintah:
copy running-config startup-config
2. Atau cukup dengan perintah singkat:
write

Contohnya dapat dilihat pada gambar di bawah ini:

```
Evan-14#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Evan-14#write
Building configuration...
[OK]
Evan-14#
```

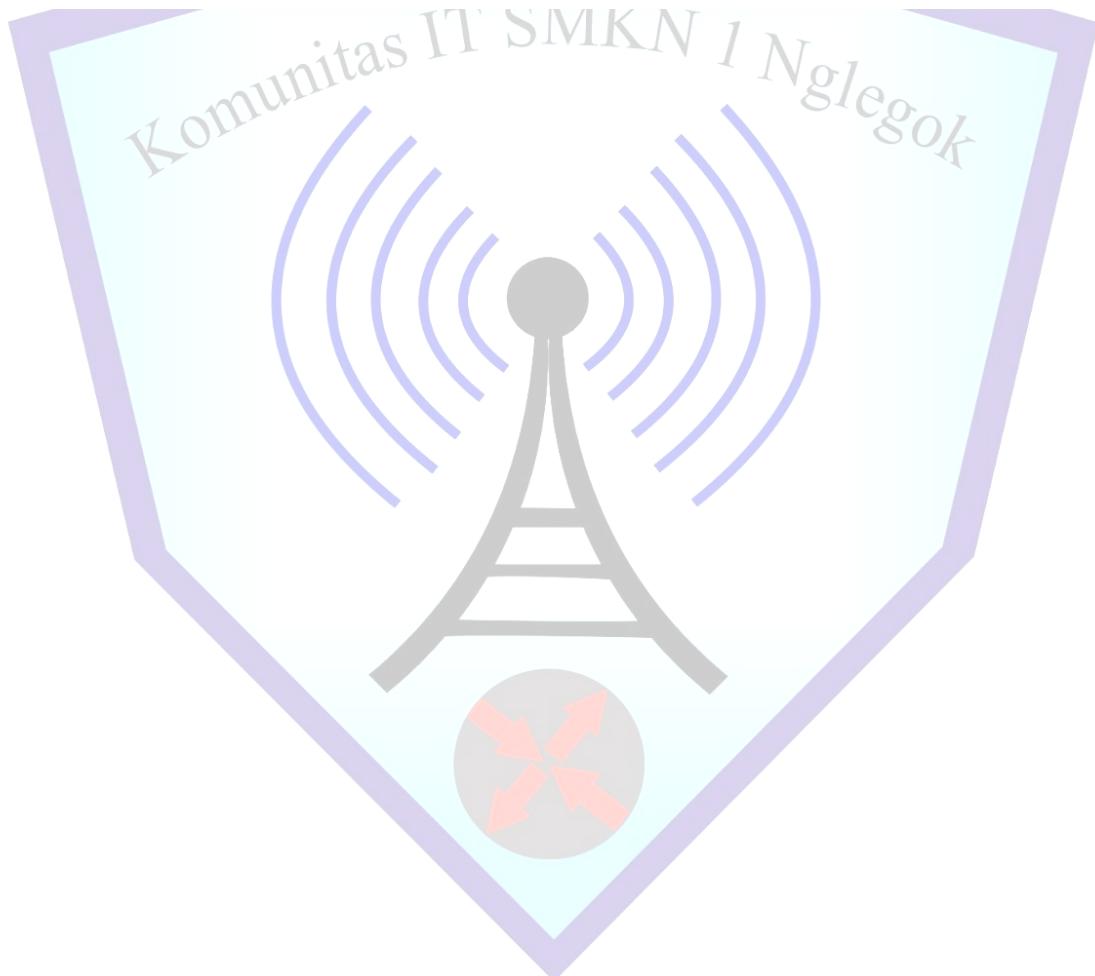
Lab 8 : Menghapus Konfigurasi

Jika kita ingin mengembalikan router ke kondisi awal tanpa konfigurasi apa pun, kita dapat menghapus konfigurasi yang tersimpan dengan perintah:erase startup-configSetelah itu, lakukan restart pada perangkat Cisco dengan perintah:

reloadContoh penggunaan perintah tersebut dapat dilihat pada gambar di bawah ini:

```
Evan-14#erase startup-config
Eraseing the nvram filesystem will remove all configuration files! Continue? [confirm]
[OK]
Erase of nvram: complete
%SYS-7-NV BLOCK INIT: Initialized the geometry of nvram
Evan-14#reload
Proceed with reload? [confirm]
System Bootstrap, Version 12.4(lr) [hqluong lr], RELEASE SOFTWARE (fc1)
Copyright (c) 2005 by cisco Systems, Inc.

Initializing memory for ECC
..
```



Lab 9 : Konfigurasi Password Cisco

Pada praktik lab kali ini, saya akan membahas cara mengganti atau mengatur password pada perangkat Cisco. Pada Cisco, kita dapat memberikan password pada Privilege Mode agar perangkat tidak mudah diakses oleh pihak yang tidak bertanggung jawab. Untuk konfigurasinya, silakan lihat contoh pada gambar di bawah ini:

```
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#enable password password@keren
```

Untuk menambahkan password pada perangkat Cisco, Anda dapat menggunakan perintah berikut: enable password [password yang ingin Anda gunakan]

Jika perintah tersebut sudah dikonfigurasi, maka password telah berhasil diterapkan pada Privilege Mode. Nantinya, setiap kali ada pengguna yang ingin masuk ke Privilege Mode, sistem akan meminta untuk memasukkan password terlebih dahulu. Contoh penerapan perintah ini dapat dilihat pada gambar di bawah ini:

```
Switch>en
Password:
```

Tetapi pembuatan password dengan cara di atas masih belum efektif karena password yang telah kita buat dapat masih bisa dilihat dengan command running-config

```
Switch#show running-config
Building configuration...

Current configuration : 1113 bytes
!
version 12.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Switch
!
enable password password@keren
!
!
!
!
!
!
!
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
--More-- |
```

Pada contoh diatas terlihat bahwa password yang kita konfigurasikan dapat dilihat dengan jelas untuk menghindari itu kita bisa mengatasinya dengan cara mengenkripsi semua password

dengan menggunakan perintah service password-encryption setelah kita konfigurasikan maka semua password yang kita konfigurasikan akan dienkripsi untuk contoh konfigurasi dan pembuktian lihatlah pada gambar di bawah

```
Switch(config)#service password-encryption
Switch(config)#no sh run
^
* Invalid input detected at '^' marker.

Switch(config)#do sh run
Building configuration...
```

Pada gambar di atas terlihat bahwa kita tidak dapat menambahkan password baru karena password yang ingin dibuat sama dengan password yang telah dikonfigurasi sebelumnya menggunakan perintah enable password.Untuk mengatasinya, kita perlu menghapus konfigurasi password yang lama dengan perintah: no enable password Setelah itu, kita dapat menambahkan password baru menggunakan perintah yang lebih aman:enable secret [password yang ingin Anda gunakan] Setelah selesai mengonfigurasi, kita dapat memverifikasi hasilnya dengan melihat konfigurasi yang sedang berjalan menggunakan perintah: show running-config

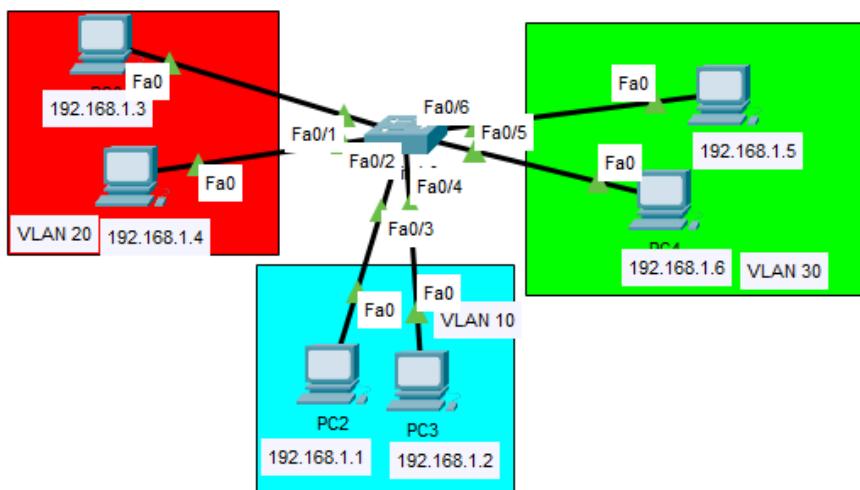
```
Switch(config)#no enable password
Switch(config)#enable secret password@14
Switch(config)#
\\
Switch(config)#do show running-config
Building configuration...

Current configuration : 1126 bytes
!
version 12.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname Switch
!
enable secret 5 $1$mERr$RP.VILZZjhCQISIaUN49m.
!
```

Pada contoh gambar di atas terlihat bahwa password yang kita buat telah dienkripsi

LAB 10 : Switch Vlan Cisco

Pada praktik lab kali ini, saya akan mencoba membuat VLAN pada switch Cisco. VLAN (Virtual Local Area Network) digunakan untuk memisahkan port-port pada switch agar berada dalam jaringan yang berbeda secara logis. Jika dua port berada pada VLAN yang berbeda, maka perangkat (misalnya PC) yang terhubung ke port tersebut tidak akan bisa saling berkomunikasi, meskipun berada dalam jaringan fisik yang sama. Untuk membuktikannya, mari kita lakukan konfigurasi VLAN dengan menggunakan layout seperti yang ditunjukkan pada gambar di bawah ini:



Kalian sebelum melakukan konfigurasi pastikan semua pc dapat ping satu sama lain. Setelah itu masuk ke en/enable untuk mengaktifkan setelah itu kita ketikkan conf t agar kita dapat melakukan konfigurasi

```

Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#hostname SW1
SW1(config)#

```

Setelah itu kita masuk ke dalam konfigurasi switch kita akan menambahkan vlan. Untuk menambahkan vlan kita bisa lihat pada contoh gambar di bawah ini jika kita ingin membuat vlan kita bisa mengetikkan vlan lalu nomor vlan yang ingin kalian buat. Disini sebagai contoh saya

akan membuat vlan 10 dan jika ingin menambahkan nama pada vlan misa menggunakan command name nama yang kalian inginkan

```
SW1(config)#vlan 10
SW1(config-vlan)#name enjang
SW1(config-vlan)#vlan 20
SW1(config-vlan)#name evan
SW1(config-vlan)#vlan 30
SW1(config-vlan)#name gesang
SW1(config-vlan)#ex
```

Jika sudah selesai membuat vlan kita bisa mengecek dengan memasukkan perintah dho show vlan

```
-----  
1 default           active   Fa0/7, Fa0/8, Fa0/9, Fa0/10  
                           Fa0/11, Fa0/12, Fa0/13, Fa0/14  
                           Fa0/15, Fa0/16, Fa0/17, Fa0/18  
                           Fa0/19, Fa0/20, Fa0/21, Fa0/22  
                           Fa0/23, Fa0/24, Gig0/1, Gig0/2  
  
10 enjang          active  
20 evan            active  
30 gesang          active  
-----  
1002 raii-default  active  
1003 token-ring-default  active  
1004 fddinet-default  active  
1005 trnet-default  active  
  
VLAN Type SAID      MTU    Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
```

Pada gambar di atas bisa dilihat bahwa default semua port berada pada vlan 1, oleh karena itu kita akan menambahkan/memindah port ke dalam vlan yang telah kita buat. Jika kita ingin memasukkan interface ke dalam vlan yang telah kita buat adalah dengan mengetikkan int range interface yang kalian ingin pindahkan. Selanjutnya jika susdah masuk ke dalam interface kita masukkan command switchport mode access. Lalu ketikkan lagi switchport access vlan no vlan yang kalian ingin masukkan.

```
SW1(config)#int range f0/1,f0/2
SW1(config-if-range)#sw mo acc
SW1(config-if-range)#sw acc vlan 10
SW1(config-if-range)#ex
SW1(config)#int range f0/1,f0/2
SW1(config-if-range)#sw mo acc
SW1(config-if-range)#sw acc vlan 20
SW1(config-if-range)#ex
SW1(config)#int range f0/4,f0/5
SW1(config-if-range)#sw mo acc
SW1(config-if-range)#sw acc vlan 30
SW1(config-if-range)#

```

Jika kita sudah memasukkan setiap interface ke dalam vlan. Kita bisa melakukan pengecheckan kemabli dengan do show vlan. Bisa dilihat bahwa pada vlan yan sebelumnya berada pada vlan 1 setelah kita konfigurasikan maka sudah masuk ke dalam vlan yang kita buat

VLAN Name	Status	Ports
1 default	active	Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gig0/1, Gig0/2
10 enjang	active	Fa0/3, Fa0/4
20 evan	active	Fa0/1, Fa0/2
30 gesang	active	Fa0/5

Untuk melakukan pengecheckan bahwa konfigurasi kita berhasil atau tidak dengan cara ping jika sesama vlan maka hasilnya akan ttl dan jika beda vlan maka hasilnya akan menjadi rto walaupun satu jaringan

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Ping statistics for 192.168.1.1:
  Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Ping statistics for 192.168.1.2:
  Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C
C:\>ping 192.168.1.5

Pinging 192.168.1.5 with 32 bytes of data:

Ping statistics for 192.168.1.5:
  Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
^C
C:\>
```

LAB 11 : Vlan Trunking

Pada lab ini kita akan membuat vlan yang ada pada 2 switch agar bisa saling terkoneksi yaitu dengan menambahkan mode trunk pada setiap portnya

Sebelum kita memulai kita terlebih dahulu membuat vlan dan konfigurasi IP, disini saya tidak akan membahas lagi mengenai vlan karena sudah kita bahas pada lab sebelumnya. Sebagai contoh kalian bisa lihat pada gambar di bawah ini terdapat 3 vlan pada setiap switchnya

VLAN Name	Status	Ports
1 default	active	Fa0/7, Fa0/8, Fa0/9, Fa0/10 Fa0/11, Fa0/12, Fa0/13, Fa0/14 Fa0/15, Fa0/16, Fa0/17, Fa0/18 Fa0/19, Fa0/20, Fa0/21, Fa0/22 Fa0/23, Fa0/24, Gig0/1, Gig0/2
10 enjang	active	Fa0/3, Fa0/4
20 evan	active	Fa0/1, Fa0/2
30 gesang	active	Fa0/5
1 default	active	Fa0/8, Fa0/9, Fa0/10, Fa0/11 Fa0/12, Fa0/13, Fa0/14, Fa0/15 Fa0/16, Fa0/17, Fa0/18, Fa0/19 Fa0/20, Fa0/21, Fa0/22, Fa0/23 Fa0/24, Gig0/1, Gig0/2
10 Evan1	active	Fa0/1, Fa0/2
20 Evan2	active	Fa0/3, Fa0/4
30 Gesang1	active	Fa0/5, Fa0/6

Berikutnya kalian chek terlebih dahulu dan lakukan ping setiap vlannya untuk awal pasti hasilnya akan rto. Selanjutnya kita akan merubah port yang menghubungkan setiap switchnya menjadi mode trunk. Untuk caranya kalian bisa lihat contoh pada gambar di bawah ini

```

SW1(config)#int fa0/6
SW1(config-if)#sw mo tr

SW2(config-if-range)#int fa0/6
SW2(config-if)#sw mo tr

SW2(config-if)#
*LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to down

*LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to up

```

Setelah konfigurasi telah kita lakukan selanjutnya kalian ping lagi maka jika konfigurasi trunk kalian itu benar maka hasilnya akan menjadi ttl jika kita ping sesama vlan

PC3

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.7

Pinging 192.168.1.7 with 32 bytes of data:

Reply from 192.168.1.7: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.8

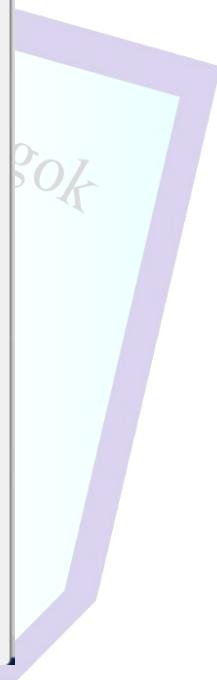
Pinging 192.168.1.8 with 32 bytes of data:

Reply from 192.168.1.8: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

SAMA VLAN 10

 Top

LAB 12 : Allwed Trunk

Pada lab ini kita akan membahas mengenai Allowed Trunk dimana biasanya merubah mode interfaces menjadi mode trunk, maka secara default semua vlan akan masuk ke interface trunk. Sebagai alasan keamanan kita sebagai admin diharuskan untuk membuat konfigurasi supanya hanya vlan tertentu saja yang kita izinkan dapat melewati trunk. Untuk konfigurasinya kita akan menggunakan topologi pada lab sebelumnya. Untuk melihat beberapa saja vlan yang diperbolehkan melewati trunk kita bisa melihat dengan menggunakan perintah do show int trunk seperti pada contoh di bawah ini

```
%LINK-5-CHANGED: Interface FastEthernet0/4, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/5, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/5, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/6, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/6, changed state to up

SW1>
SW1>
SW1>en
SW1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
SW1(config)#do sh int tr
Port      Mode          Encapsulation  Status        Native vlan
Fa0/6    on           802.1q         trunking     1
Fa0/7   desirable    n-802.1q      trunking     1

Port      Vlans allowed on trunk
Fa0/6    1-1005
Fa0/7    1-1005

Port      Vlans allowed and active in management domain
Fa0/6    1,10,20,30
Fa0/7    1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/6    none
Fa0/7    none

SW1(config)#

```

Untuk merubah vlan yang kita akan izinkan untuk melewati trunk kita dapat menggunakan perintah sw tr allowed vlan yang kita izinkan, contoh konfigurasinya seperti pada gambar di bawah ini

```
SW1(config)#int f0/6
SW1(config-if)#sw tr allowed vl 10,20
SW1(config-if)#ex
SW1(config)#

```

Setelah kita konfigurasikan allowed trunk pada kedua switch kita bisa melihatnya pada interface trunk, dan seperti nanti pada vlan allowed on trunk akan berubah dengan vlan yang kita masukkan ke allowed trunk

```

Port      Mode       Encapsulation  Status      Native vlan
Fa0/6    on         802.1q        trunking   1
Fa0/7    desirable  n-802.1q     trunking   1

Port      Vlans allowed on trunk
Fa0/6    1-1005
Fa0/7    1-1005

Port      Vlans allowed and active in management domain
Fa0/6    1,10,20,30
Fa0/7    1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/6    none
Fa0/7    none

SW1(config)#int f0/6
SW1(config-if)#sw tr allowed v1 10,20
SW1(config-if)#ex
SW1(config)#do sh int tr
Port      Mode       Encapsulation  Status      Native vlan
Fa0/6    on         802.1q        trunking   1
Fa0/7    desirable  n-802.1q     trunking   1

Port      Vlans allowed on trunk
Fa0/6    10,20
Fa0/7    1-1005

Port      Vlans allowed and active in management domain
Fa0/6    10,20
Fa0/7    1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/6    10,20
Fa0/7    1,10,20,30

SW1(config)#

```

```

Port      Vlans allowed on trunk
Fa0/6    1-1005
Fa0/7    1-1005

Port      Vlans allowed and active in management domain
Fa0/6    1,10,20,30
Fa0/7    1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/6    1,10,20,30
Fa0/7    1,10,20,30

SW2(config)#sw tr allow vlan 10,20
^
* Invalid input detected at '^' marker.

SW2(config)#int f0/6
SW2(config-if)#sw tr allow vlan 10,20
SW2(config-if)#do sh int tr
Port      Mode          Encapsulation  Status        Native vlan
Fa0/6    on            802.1q         trunking    1
Fa0/7    desirable     n-802.1q      trunking    1

Port      Vlans allowed on trunk
Fa0/6    10,20
Fa0/7    1-1005

Port      Vlans allowed and active in management domain
Fa0/6    10,20
Fa0/7    1,10,20,30

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/6    20
Fa0/7    1,10,20,30

SW2(config-if)#

```

Untuk membuktikan coba ping dari pc di switch1 ke switch2 yang berada pada vlan yang sama

```

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.7

Pinging 192.168.1.7 with 32 bytes of data:

Reply from 192.168.1.7: bytes=32 time=6ms TTL=128
Reply from 192.168.1.7: bytes=32 time<1ms TTL=128
Reply from 192.168.1.7: bytes=32 time=9ms TTL=128
Reply from 192.168.1.7: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.7:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 9ms, Average = 3ms

C:\>ping 192.168.1.6

```

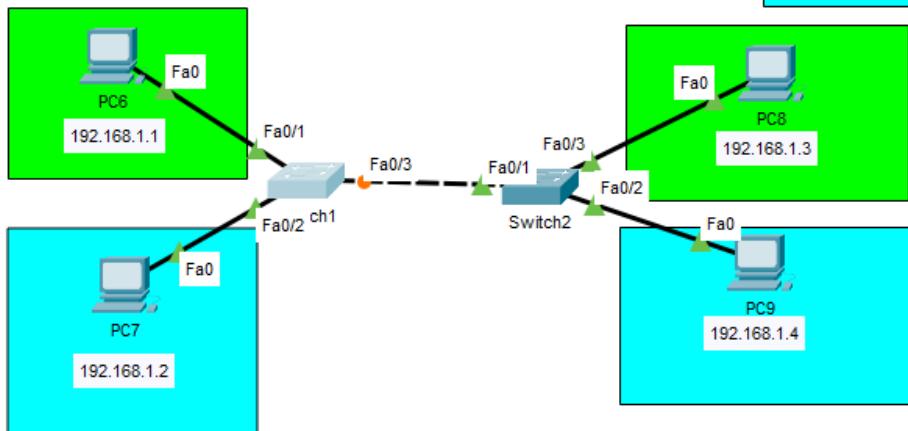
PING PC 1 ke PC 6
SESAMA VLAN 10

LAB 13 : Dynamic Trunking Protocol

Berikutnya kita akan membahas mengenai tentang Dynamic Trunking Protocol adalah protocol yang dapat digunakan untuk membuat interface ke mode trunk secara otomatis. Ada beberapa mode yang dapat digunakan untuk membuat trunk secara dynamic pada switch.

MODE	STATIC TRUNK	STATIC ACCESS	DYNAMIC AUTO	DYNAMIC DESIRABLE
Static Trunk	Trunk	Limited	Trunk	Trunk
Static Access	Limited	Trunk	Access	Access
Dynamic Auto	Trunk	Access	Access	Trunk
Dynamic Desirable	Trunk	Access	Trunk	Trunk

Untuk membuktikan kita akan mencoba konfigurasi beberapa mode, untuk itu kita akan menggunakan topologi seperti gambar di bawah ini



Kita bisa asumsikan bahwa pada layout diatas sudah dikonfigurasi IP, dan juga vlan yang berwarna hijau vlan 10 dan yang berwarna biru vlan 20. Kita akan melanjutkan dengan konfigurasi mode interface yang menghubungkan kedua switch. Untuk konfigurasinya seperti pada contoh di bawah ini.

```
Enter configuration commands, one per line. End with CNTL/Z.  
Switch(config)#int f0/3  
Switch(config-if)#sw mo dynamic auto  
  
Switch(config-if)#  
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
```

```
Switch(config)#int f0/3  
Switch(config-if)#sw mo dyanamic desirable  
^  
% Invalid input detected at '^' marker.  
  
Switch(config-if)#sw mo dynamic desirable  
Switch(config-if)#[legok]
```

Setelah konfigurasi telah kita buat pada kedua interfaces tersebut, mari kita lihat mode yang kita gunakan oleh kedua switch tersebut dengan menggunakan perintah di bawah ini

```
Switch(config)#do sh int f0/3 sw  
Name: Fa0/3  
Switchport: Enabled  
Administrative Mode: dynamic auto  
....  
Switch(config-if)#do sh int f0/3 sw  
Name: Fa0/3  
Switchport: Enabled  
Administrative Mode: dynamic desirable
```

LAB 14 Virtual Trunking Protocol

Pada lab ini kita akan membahas mengenai VTP atau virtual trunking protocol ini adalah virtual yang dapat digunakan untuk melakukan management vlan secara terpusat, intinya dapat menambah, mengedit, menghapus vlan pada satu switch saja intinya ini akan mempermudah perkerjaan dalam mengelola vlan. VTP memiliki 3 mode yaitu

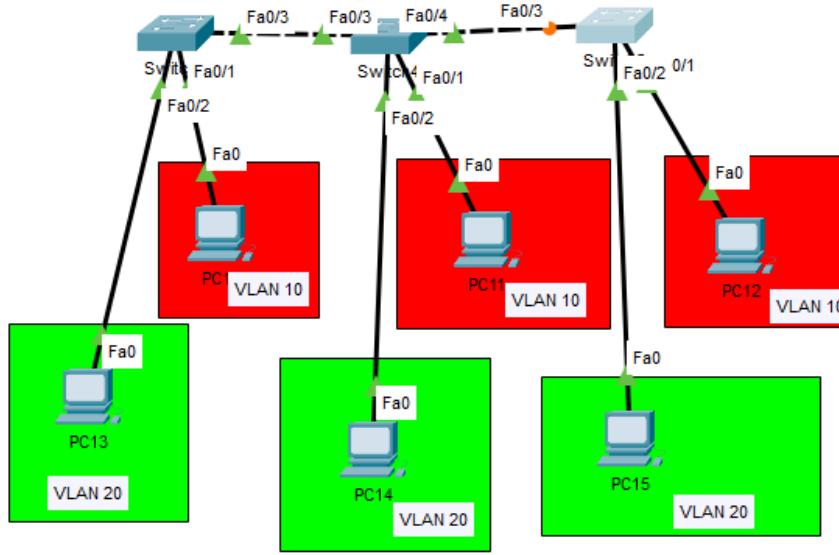
Pada vtp ini terdapat tiga mode yaitu :

► Server pada switch yang menggunakan mode ini akan memiliki hak untuk membuat, mengedit, menghapus vlan. Pada switch yang menggunakan mode ini akan mengirimkan update kepada switch yang lain dengan mode client, sehingga switch yang menggunakan mode client akan mengupdate informasi mengenai vlan yang didapatkan dari switch yang menggunakan mode server.

► Transparent pada mode ini switch akan dapat membuat vlan akan tetapi vlan yang dibuatnya tidak akan dikirim ke switch lain / bersifat lokal. Selain itu switch dengan mode ini tidak akan mendapatkan informasi vlan dari switch yang menggunakan mode server.

► Client pada mode ini switch tidak dapat membuat ataupun menghapus vlan. Switch yang menggunakan mode ini hanya dapat mengupdate informasi vlan yang didapatkan dari switch yang menggunakan mode server.

Untuk membuktikan konsep diatas kita akan mencoba melakukan konfigurasi dengan menggunakan topologi seperti di bawah ini



```

Switch(config-if)#hostname vl 1
^
* Invalid input detected at '^' marker.

Switch(config-if)#hostname swl
swl(config)#vtp mode server
Device mode already VTP SERVER.
swl(config)#vtp domain SW1
Changing VTP domain name from NULL to SW1
swl(config)#vtp pass 1
Setting device VLAN database password to 1
swl(config)#int fa0/3
swl(config-if)#sw mo ac
swl(config-if)#sw %SPANTREE-2-RECV_PVID_ERR: Received 802.1Q BPDU on non trunk
FastEthernet0/3 VLAN1.

%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/3 on VLAN0001. Inconsistent port
type.

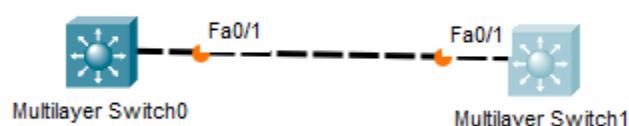
* Incomplete command.
swl(config-if)#sw mo tr

swl(config-if)#
$LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down
$LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
swl(config-if)#

```

LAB 15 Trunk Pada MLS

Pada lab ini kita akan membahas tentang konfigurasi trunk pada MLS. Untuk konfigurasi trunk pada MLS ini akan sedikit berbeda switch pada umumnya. Untuk melihat perbedaannya itu mari kita coba konfigurasi dengan menggunakan topologi di bawah ini



Jika sudah membuat topologi diatas, langsung saja kita coba untuk konfigurasi trunk pada switch MLS

```

Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int f0/1
Switch(config-if)#sw mo tr
Command rejected: An interface whose trunk encapsulation is "Auto" can not be
configured to "trunk" mode.
  
```

Untuk mengatasi masalah tersebut kita hanya perlu menambahkan encapsulation pada trunk di switch MLS, setelah selesai konfigurasikan encapsulation kita bisa melanjutkan dengan mengkonfigurasi trunk

```

Switch(config-if)#sw tr encapsulation dot1q
Switch(config-if)#
  
```

Untuk mengeceknya apakah konfigurasi trunk sudah berhasil maka kita lihat pada interface trunk seperti pada gambar di bawah ini

```

Switch(config-if)#do sh int tr
Port      Mode          Encapsulation  Status        Native vlan
Fa0/1    on            802.1q         trunking     1

Port      Vlans allowed on trunk
Fa0/1    1-1005

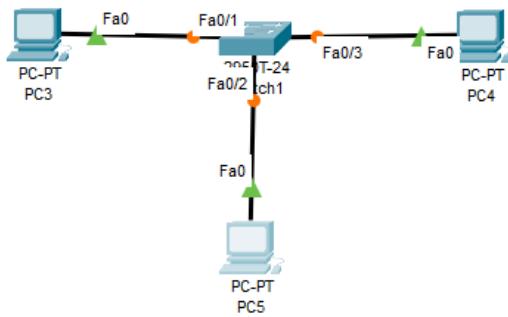
Port      Vlans allowed and active in management domain
Fa0/1    1

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/1    none

Switch(config-if)#
  
```

LAB 16 Security Port Switch

Pada lab ini kita akan membahas tentang konfigurasi port security pada switch, dimana ini akan kita gunakan untuk membatasi perangkat yang diperbolehkan untuk mengakses salah satu port pada switch. Jika kalian sudah memahami mari kita lanjutkan dengan konfigurasinya, sebelumnya kita akan menggunakan topologi seperti pada contoh di bawah ini



Jika sudah kita lanjutkan dengan membuat security portnya, seperti pada gambar di bawah ini

```

Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int f0/1
Switch(config-if)#sw mo acc
Switch(config-if)#sw port-security
Switch(config-if)#sw port-security mac-address sticky
Switch(config-if)#sw port-security violation shutdown
Switch(config-if)#int f0/2
Switch(config-if)#sw mo ac
Switch(config-if)#sw port-security
Switch(config-if)#sw port-security mac-address sticky
Switch(config-if)#sw port-security violation res
Switch(config-if)#sw port-security violation restart
^
% Invalid input detected at '^' marker.

Switch(config-if)#sw port-security violation restrict
Switch(config-if)#int f0/3
Switch(config-if)#sw mo ac
Switch(config-if)#sw port-security
Switch(config-if)#sw port-security mac-address sticky
Switch(config-if)#sw port-security violation pro
Switch(config-if)#sw port-security violation protect
Switch(config-if)#
  
```

Setelah kalian membuat security port kalian coba lakukan ping kesetiap pc supaya switch mencatat mac address setiap pc yang terhubung ke portnya

```
Command Prompt X

Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=6ms TTL=128          PING KE PC 1
Reply from 192.168.1.1: bytes=32 time<1ms TTL=128
Reply from 192.168.1.1: bytes=32 time<1ms TTL=128
Reply from 192.168.1.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 6ms, Average = 1ms

C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128          PING KE PC 2
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time=1ms TTL=128
Reply from 192.168.1.3: bytes=32 time=1ms TTL=128
```

Setelah ping, untuk membuktikan apakah mac address sudah tercatat atau belum kalian bisa coba lihat mac yang sudah tercatat seperti pada contoh ini

```
Switch(config)#do sh mac-address
      Mac Address Table
-----
Vlan   Mac Address        Type      Ports
----  -----
  1    0002.1659.5077    STATIC    Fa0/1
  1    000c.cfe9.c4c6    STATIC    Fa0/3
  1    00e0.b066.a70b    STATIC    Fa0/2
-----
Switch(config)#do sh port-security address
      Secure Mac Address Table
-----
Vlan   Mac Address        Type      Ports      Remaining Age
                                                               (mins)
----  -----  -----
  1    0002.1659.5077  SecureSticky  Fa0/1      -
  1    00e0.b066.a70b  SecureSticky  Fa0/2      -
  1    000c.cfe9.c4c6  SecureSticky  Fa0/3      -
-----
Total Addresses in System (excluding one mac per port) : 0
Max Addresses limit in System (excluding one mac per port) : 1024
```

Jika sudah kalian rubah topologi kalian coba ping dari setiap pc ke pc lain, maka hasil akhirnya akan menjadi time out

```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Request timed out.

Ping statistics for 192.168.1.3:
  Packets: Sent = 2, Received = 0, Lost = 2 (100% loss),

Control-C
^C
C:\>ping 192.168.1.1

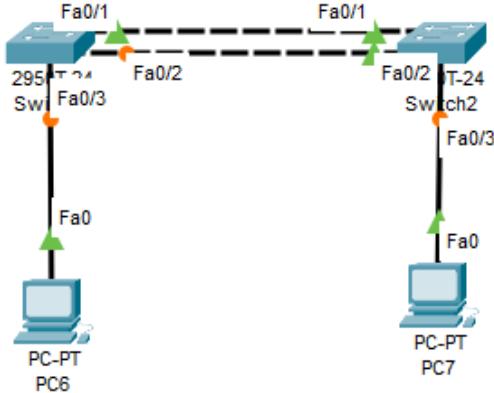
Pinging 192.168.1.1 with 32 bytes of data:

Request timed out.

Ping statistics for 192.168.1.1:
  Packets: Sent = 2, Received = 0, Lost = 2 (100% loss),
```

LAB 17 Spanning Tree Protocol Switch

Pada lab kali ini kita akan membahas tentang spanning tree protocol pada switch, dimana sebuah protocol yang digunakan oleh switch untuk menghindari terjadinya looping pada jaringan. Untuk menghindari hal itu switch telah ditambahkan protocol STP dimana ini akan memblok salah satu port dimana nantinya hanya akan ada salah satu port saja yang akan digunakan sehingga ini akan menghindari dari terjadinya looping.



Untuk melihat status spanning tree pada switch kita dapat menggunakan perintah pada gambar di bawah ini

```

Switch(config)#hostname sw1
sw1(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
              Address     0007.EC0A.9413
              Cost         19
              Port        1 (FastEthernet0/1)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0090.2BAA.D72B
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time   20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/3        Desg FWD 19      128.3    P2p
  Fa0/2        Altn BLK 19      128.2    P2p
  Fa0/1        Root FWD 19      128.1    P2p
  
```

Jika sudah selesai melihat status pada switch 1 kita akan chek juga status port pada switch2 seperti gambar di bawah ini

```

Switch(config)#hostname sw2
sw2(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
              Address     0007.EC0A.9413
              This bridge is the root
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
  Address     0007.EC0A.9413
  Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
  Aging Time  20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/3          Desg FWD 19      128.3    P2p
  Fa0/1          Desg FWD 19      128.1    P2p
  Fa0/2          Desg FWD 19      128.2    P2p

```

Penentuan switch mana yang akan menjadi root bridge dalam jaringan menggunakan Spanning Tree Protocol (STP) sebenarnya mengikuti aturan tertentu. Ketika beberapa switch saling terhubung, mereka secara otomatis akan memilih satu switch sebagai root bridge. Pemilihan ini penting karena root bridge akan menjadi titik pusat referensi untuk menentukan jalur terpendek dalam jaringan dan mencegah terjadinya loop.

Hal pertama yang dilihat oleh switch saat proses pemilihan root bridge adalah nilai priority. Setiap switch memiliki nilai priority, dan semakin kecil angkanya, maka semakin besar kemungkinan switch tersebut terpilih menjadi root bridge. Secara default, nilai priority pada switch adalah 32768. Namun, jika semua switch memiliki nilai priority yang sama karena belum diubah, maka proses pemilihan akan dilanjutkan ke tahap berikutnya.

Apabila nilai priority sama, maka switch akan membandingkan MAC address masing-masing. Dalam hal ini, switch dengan MAC address yang paling kecil atau paling rendah nilainya akan terpilih sebagai root bridge. MAC address merupakan alamat fisik unik yang dimiliki oleh setiap perangkat jaringan, dan nilai yang lebih kecil dianggap lebih “unggul” dalam proses pemilihan ini.

Setelah memahami bahwa root bridge dipilih berdasarkan priority terlebih dahulu, lalu MAC address jika priority-nya sama, sekarang saatnya untuk mencoba memindahkan status root bridge dari switch yang satu ke switch lainnya. Untuk melakukannya, kita bisa mengubah nilai priority dari switch yang ingin kita jadikan root bridge. Misalnya, jika saat ini switch2 menjadi root bridge dan kita ingin agar switch1 yang menjadi root bridge, maka kita perlu menurunkan nilai priority pada switch1 sehingga lebih kecil dari switch2.

Dengan menurunkan nilai priority pada switch1, maka switch tersebut akan lebih diutamakan dalam proses pemilihan root bridge dan akhirnya akan menggantikan switch2 sebagai pusat pengendali jalur dalam jaringan.

```
sw1(config)#spanning-tree vlan 1 priority 10
% Bridge Priority must be in increments of 4096.
% Allowed values are:
 0    4096   8192   12288  16384  20480  24576  28672
 32768 36864 40960 45056 49152 53248 57344 61440
```

Jika dilihat pada gambar diatas maka akan terlibat bahwa kita hanya bisa merubah priority sesuai dengan priority yang telah ditentukan oleh switch cisco. Jika sudah marin kita mencoba untuk merubah prioritynya dengan lebih rendah, setelah itu anda lihat status port seperti pada gambar di bawah ini

```
sw1(config)#spanning-tree vlan 1 priority 28672
sw1(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    28673
  Address    0090.2BAA.D72B
  This bridge is the root
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    28673  (priority 28672 sys-id-ext 1)
  Address    0090.2BAA.D72B
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
  Aging Time 20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/3          Desg FWD 19      128.3    P2p
  Fa0/2          Desg LSN 19      128.2    P2p
  Fa0/1          Desg FWD 19      128.1    P2p
```

Jika sudah kita akan coba lagi merubah supaya switch2 kembali menjadi rot bridge, mari kita lakukan dengan merubah prioritynya lebih kecil daripada priority sw1. Untuk konfigurasinya seperti pada gambar di bawah ini.

```
sw2(config)#spanning-tree vl 1 priority 8192
```

Jika sudah selesai coba kita lihat lagi status port pada switch2 dan bisa dilihat bahwa statusnya telah berubah lagi menjadi root bridge

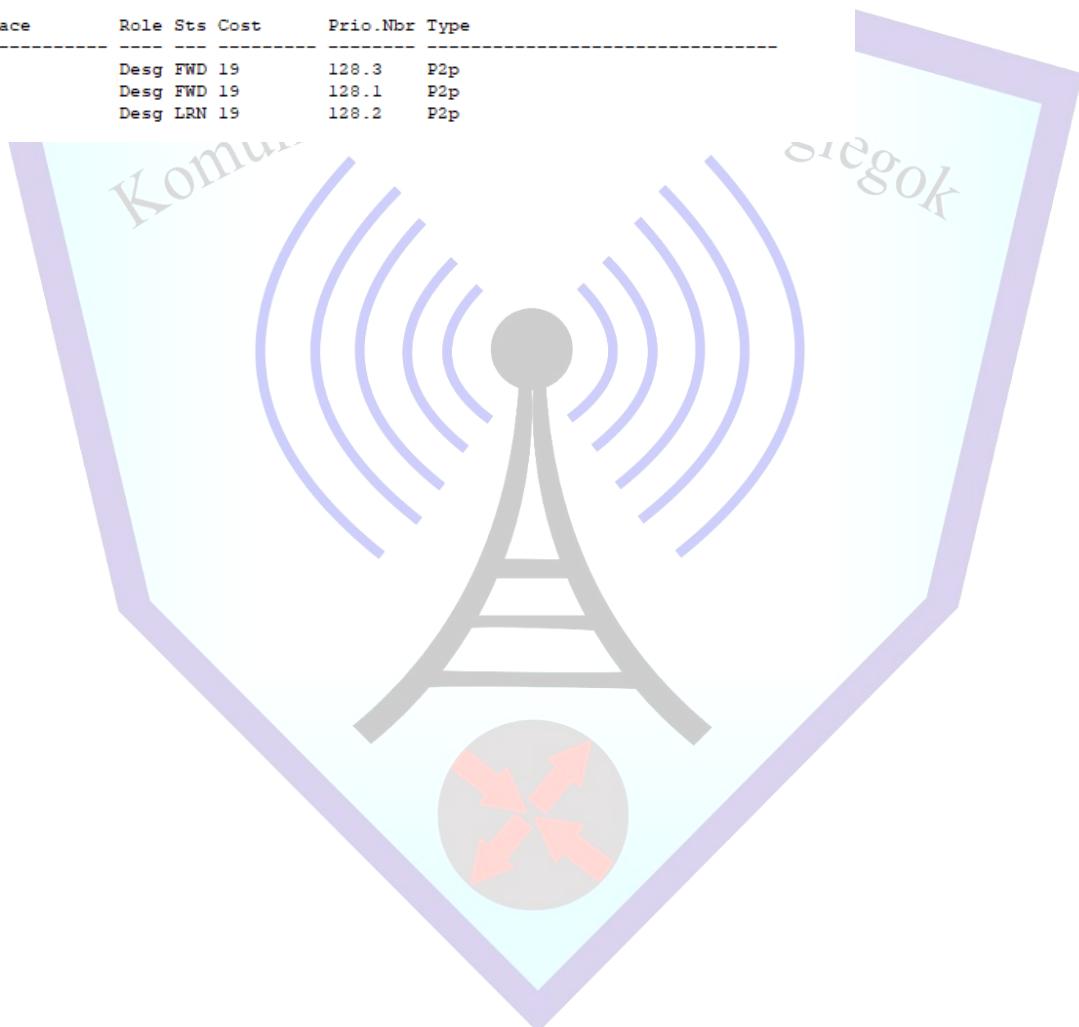
```

sw2(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    8193
  Address    0007.EC0A.9413
  This bridge is the root
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    8193 (priority 8192 sys-id-ext 1)
  Address    0007.EC0A.9413
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
  Aging Time 20

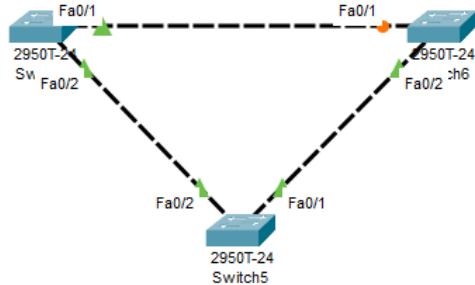
  Interface      Role Sts Cost      Prio.Nbr Type
  -----
  Fa0/3          Desg FWD 19      128.3    P2p
  Fa0/1          Desg FWD 19      128.1    P2p
  Fa0/2          Desg LRN 19     128.2    P2p

```



LAB 18 Per VLAN Spanning Tree ProtocolP

Pada lab ini kita akan membahas mengenai tentang per vlan spanning tree dimana setiap switch cisco akan mengaktifkan fitur ini. Fitur ini memungkinkan setiap vlan memiliki satu switch sebagai root bridge, selain itu setiap vlan juga akan memilih port yang akan di blokir. Untuk membuktikan mari kita coba konfigurasi dengan menggunakan topologi seperti di bawah ini



Pada layout di atas setiap switch memiliki priority yang sama, karena memiliki priority yang sama maka switch akan memilih vlan yang memiliki mac-address yang paling kecil yang akan dipilih sebagai root bridge. Pada gambar di atas terlihat bahwa pada switch 3 memiliki mac address yang paling kecil sehingga akan menjadi root bridge.

```

SW1(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority 32769
              Address 0001.C741.04E5
              Cost      19
              Port      2 (FastEthernet0/2)
              Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority 32769 (priority 32768 sys-id-ext 1)
              Address 0003.E467.2E6D
              Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----+-----+-----+-----+-----+-----+
  Fa0/2        Root FWD 19      128.2    P2p
  Fa0/1        Altn BLK 19      128.1    P2p

```

```

switch#hostname sw2
SW2(config)#do sh spanning-tree
VLAN001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
              Address     0001.C741.04E5
              Cost         19
              Port        2 (FastEthernet0/2)
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769  (priority 32768 sys-id-ext 1)
              Address     0001.C997.0EEA
              Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time   20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/2          Root FWD 19      128.2    P2p
  Fa0/1          Desg FWD 19     128.1    P2p

```

Setelah kita lanjutkan dengan membuat vlan 10 dan vlan 20 ke setiap switch, dan juga ubah mode portnya menjadi trunk

```

SW1(config)#vlan 10
SW1(config-vlan)#name Evan
SW1(config-vlan)#ex
SW1(config)#vlan 20
SW1(config-vlan)#name 14
SW1(config-vlan)#ex
SW1(config)#int range f0/1-2
SW1(config-if-range)#sw mo tr
SW1(config-if-range)#

```

```

SW2(config)#vlan 10
SW2(config-vlan)#name Evan
SW2(config-vlan)#ex
SW2(config)#vlan 20
SW2(config-vlan)#name 14
SW2(config-vlan)#ex
SW2(config)#int f0/1-2
^
* Invalid input detected at '^' marker.

SW2(config)#int range f0/1-2
SW2(config-if-range)#sw mo tr
SW2(config-if-range)#

```

Setelah selesai membuat vlan coba lihat spanning tree pada salah satu switch. Jika dilihat pada spanning tree pada switch maka vlan yang kita tambahkan tadi akan terlihat.

```
SW1(config)#do sh spanning-tree
sh spanning-treee
^
* Invalid input detected at '^' marker.

SW1(config)#do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
              Address     0001.C741.04E5
              Cost        19
              Port       2 (FastEthernet0/2)
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32769 (priority 32768 sys-id-ext 1)
              Address     0003.E467.2E6D
              Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 20

  Interface   Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/2       Root FWD 19      128.2    P2p
  Fa0/1       Altn BLK 19      128.1    P2p

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    32778
              Address     0001.C997.0EEA
              Cost        19
              Port       1 (FastEthernet0/1)
  Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    32778 (priority 32768 sys-id-ext 10)
              Address     0003.E467.2E6D
              Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
              Aging Time 20
```

Karena pada setiap vlan memiliki port status sendiri oleh karena itu akan mencoba untuk merubah salah satu vlan supaya menjadi bridge, pada kali ini saya akan merubah vlan 10 menjadi root briidge, untuk merubahnya kita tinggal merubah priority

```
SW2 (config)#spanning-tree vlan 10 priority 4096
```

Setelah itu coba lihat pada spanning tree pada switch 2

```

SW2 (config) # spanning-tree vlan 10 priority 4096
SW2 (config) # do sh spanning-tree
VLAN0001
  Spanning tree enabled protocol ieee
  Root ID  Priority    32769
            Address   0001.C741.04E5
            Cost        19
            Port       2 (FastEthernet0/2)
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID Priority    32769 (priority 32768 sys-id-ext 1)
            Address   0001.C997.0EEA
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
            Aging Time 20

  Interface      Role Sts Cost      Prio.Nbr Type
  -----  -----
  Fa0/2          Root FWD 19      128.2    P2p
  Fa0/1          Desg FWD 19     128.1    P2p

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID  Priority    4106
            Address   0001.C997.0EEA
  This bridge is the root
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID Priority    4106 (priority 4096 sys-id-ext 10)

```

Setelah spanning tree pada vlan 10 menjadi root bridge, kita harus merubah vlan semua port yang menghubungkan semua switch menjadi vlan 10 dan supaya switch 2 menjadi root bridge

```

SW1(config)# int range f0/1-2
SW1(config-if-range)# sw mo acc
SW1(config-if-range) # %SPANTREE-2-RECV_PVID_ERR: Received 802.1Q BPDU on non trunk
FastEthernet0/1 VLAN1.

%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/1 on VLAN0001. Inconsistent port
type.

%SPANTREE-2-RECV_PVID_ERR: Received 802.1Q BPDU on non trunk FastEthernet0/2 VLAN1.

%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/2 on VLAN0001. Inconsistent port
type.

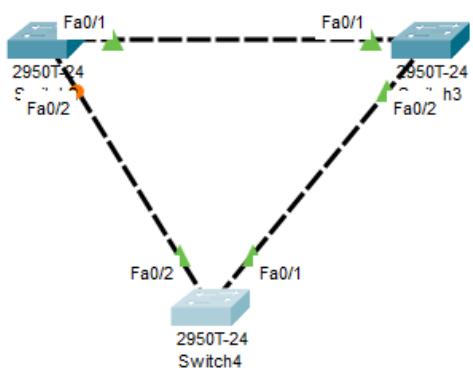
SW1(config-if-range) # sw acc vlan 10
SW2(config) # int range f0/1-2
SW2(config-if-range) # sw mo acc
SW2(config-if-range) # sw %SPANTREE-2-RECV_PVID_ERR: Received 802.1Q BPDU on non trunk
FastEthernet0/2 VLAN1.

%SPANTREE-2-BLOCK_PVID_LOCAL: Blocking FastEthernet0/2 on VLAN0001. Inconsistent port
type.

% Incomplete command.
SW2(config-if-range) # sw acc vlan 10
SW2(config-if-range) #
SW3(config) # int range f0/1-2
SW3(config-if-range) # sw mo acc
SW3(config-if-range) # sw acc vlan 10

```

Jika sudah bisa lihat topologi dan switch2 akan berubah menjadi root bridge, dan port yang tadinya berada pada switch 1 sekarang berpindah ke switch1, jika merubah vlan pada semua switch dengan vlan 1 maka nanti switch akan menjadi root adalah switch 3



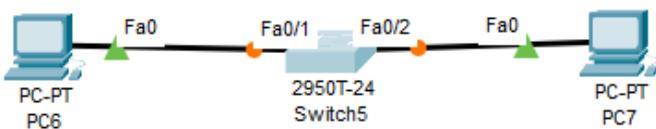
KN1Ngleok

LAB 19 Spanning Tree Portfast

Pada lab kali ini kita akan membahas tentang spanning tree portfast. Secara default port pada switch akan melewati beberapa mode sebelum paket dapat melewati switch.

Blocking (20s)
Listening (15s)
Learning (15s)
Forwarding

Jika dilihat pada tabel diatas terlihat bahwa sebelum paket dapat melewati switch port switch memerlukan waktu sekitar 50s supaya paket dapat melewati switch. Jika belum 50s maka port switch tersebut tidak akan bisa dilewati oleh paket. Hal ini tentunya akan memerlukan waktu yang lama, oleh karena itu pada lab kali ini akan menggunakan fitur spanning tree portfast, dimana ini dapat digunakan untuk mempercepat proses. Akan tetapi spanning tree portfast ini hanya dapat aktif pada interface yang akan terhubung dengan client / server. Untuk memulai konfigurasi mari kita buat topologi sederhana seperti pada gambar di bawah ini



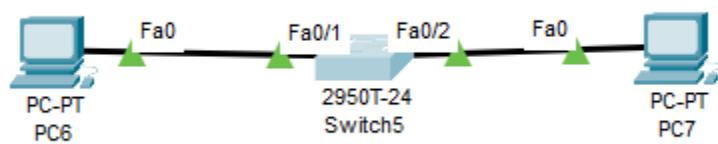
Pada gambar diatas terlihat bahwa saat memasang kabel pada interface switch ke pc maka port yang ada pada switch hanya akan berwarna orange pada saat itu pula kita tidak bisa melakukan pengiriman data. Kali ini kita akan mencoba supaya jika kita memasang interface port switch ke client server maka power tersebut langsung bisa digunakan untuk mengirim paket tanpa harus menunggu lama

```

SW1(config)#int range f0/1-2
SW1(config-if-range)#spanning-tree portfast
*Warning: portfast should only be enabled on ports connected to a single
host. Connecting hubs, concentrators, switches, bridges, etc... to this
interface when portfast is enabled, can cause temporary bridging loops.
Use with CAUTION
  
```

***Portfast has been configured on FastEthernet0/1 but will only have effect when the interface is in a non-trunking mode.**

Setelah selesai melakukan konfigurasi coba anda putuskan kabel yang menghubungkan antara switch dan pc, lalu tambahkan lagi kabel untuk menghubungkan antara switch dengan pc. Jika diperhatikan saat memasang kabel pada switch langsung berubah menjadi hijau tanpa menunggu lama.

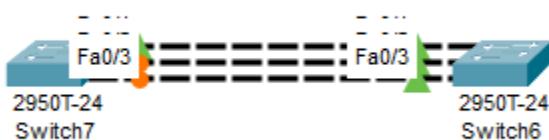


LAB 20 Etherchannel LACP

Pada lab kali ini, saya akan membahas mengenai konfigurasi EtherChannel dengan menggunakan protokol LACP (Link Aggregation Control Protocol). Ketika kita menggunakan dua buah switch yang dihubungkan dengan lebih dari satu link, secara default protokol STP (Spanning Tree Protocol) akan memblokir sebagian link dan hanya mengaktifkan satu link saja untuk mencegah terjadinya loop dalam jaringan. Namun, pada lab ini, saya akan mencoba membuat semua link yang menghubungkan kedua switch tersebut tetap aktif. Untuk mewujudkan hal ini, saya akan menggunakan fitur EtherChannel.

EtherChannel merupakan sebuah teknologi yang memungkinkan penggabungan beberapa link fisik menjadi satu link logis. Dengan demikian, seluruh link akan berfungsi seolah-olah sebagai satu jalur komunikasi, sehingga meningkatkan bandwidth dan efisiensi tanpa menyebabkan konflik dengan STP. Dalam konfigurasi EtherChannel, terdapat beberapa protokol yang dapat digunakan, yaitu LACP, PAgP, dan konfigurasi secara manual (static). LACP (Link Aggregation Control Protocol) adalah protokol standar terbuka (open standard) yang dapat digunakan pada perangkat dari berbagai vendor. PAgP (Port Aggregation Protocol) merupakan protokol proprietary milik Cisco, sehingga hanya bisa digunakan pada perangkat Cisco. Selain itu, konfigurasi EtherChannel juga bisa dilakukan secara manual tanpa menggunakan protokol tertentu. Pada lab ini, saya akan menggunakan protokol LACP karena bersifat open standard dan lebih fleksibel dalam berbagai skenario jaringan.

Seperti yang dilihat pada list di atas bahwa ada tiga protocol yang dapat kita gunakan untuk mengkonfigurasikan etherchannel, akan tetapi pada lab kali ini saya akan membahas tentang konfigurasi etherchannel, dengan menggunakan protocol LACP kita akan menggunakan topologi seperti pada gambar di bawah ini



Jika sudah selesai membuat topologi pada gambar di atas, mari kita lanjutkan dengan mengkonfigurasikan etherchannelnya. Untuk konfigurasinya seperti pada gambar ini

```

Switch(config)#int range f0/1-3
Switch(config-if-range)#sw mo tr
Switch(config-if-range)#channel-group 1 mode active
Switch(config-if-range)#
Creating a port-channel interface Port-channel 1

```

Setelah selesai kita melanjutkan dengan membuat konfigurasi di switch 2

```

SW2(config)#int range f0/1-3
SW2(config-if-range)#sw mo tr
SW2(config-if-range)#channel-group 1 mode passive
^
% Invalid input detected at '^' marker.

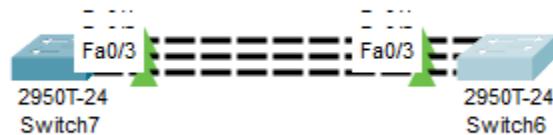
SW2(config-if-range)#channel-group 1 mode passive
SW2(config-if-range)#
Creating a port-channel interface Port-channel 1

```

Jika dilihat ada perbedaan mode yang digunakan, sw1 menggunakan mode active sedangkan switch2 menggunakan mode passive ini bertujuan supaya etherchannel dapat aktif. Untuk itu lihatlah pada tabel di bawah ini

Mode	Active	Passive
Active	Yes	Yes
Passive	Yes	No

Setelah melihatnya maka nanti semua port akan menyala seperti pada gambar di bawah ini



Jika kalian ingin melihat status dari group 1 anda dapat menggunakan perintah seperti di bawah ini

```
Switch(config)#do sh etherchannel summary
Flags: D - down      P - in port-channel
      I - stand-alone s - suspended
      H - Hot-standby (LACP only)
      R - Layer3      S - Layer2
      U - in use       f - failed to allocate aggregator
      u - unsuitable for bundling
      w - waiting to be aggregated
      d - default port
```

```
Number of channel-groups in use: 1
Number of aggregators: 1

Group Port-channel Protocol Ports
----+-----+-----+
1    Pol(SU)        LACP   Fa0/1(P) Fa0/2(P) Fa0/3(P)
```

```
SW2(config)#do sh etherchannel summary
Flags: D - down      P - in port-channel
      I - stand-alone s - suspended
      H - Hot-standby (LACP only)
      R - Layer3      S - Layer2
      U - in use       f - failed to allocate aggregator
      u - unsuitable for bundling
      w - waiting to be aggregated
      d - default port
```

```
Number of channel-groups in use: 1
Number of aggregators: 1

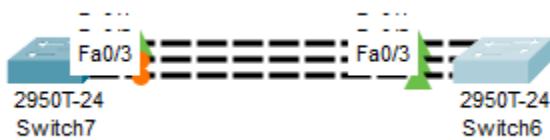
Group Port-channel Protocol Ports
----+-----+-----+
1    Pol(SU)        LACP   Fa0/1(P) Fa0/2(P) Fa0/3(P)
```

Jika sebelumnya kita menggunakan 2 mode yang berbeda sekarang mari kita coba menggunakan duo mode yang sama yaitu mode passive untuk konfigurasi seperti pada gambar di bawah ini

```
Switch(config-if-range)#channel-group 3 mo passive
Switch(config-if-range)#
SW2(config)#int range ru/1-3
SW2(config-if-range)#channel-group 3 mo passive
SW2(config-if-range)#

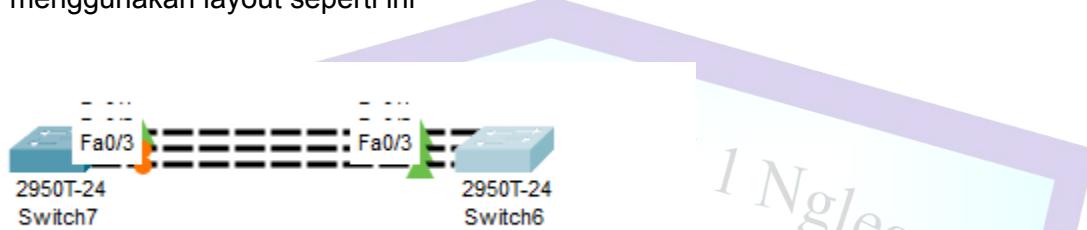
```

Jika dilihat pada gambar di bawah ini maka akan terlihat bahwa etherchannel tidak berjalan ini dikarenakan kita tadi menggunakan mode passive pada kedua switch



LAB 21 Etherchannel PAgP

Pada lab kali ini kita akan membahas tentang etherchannel PAgP. Etherchannel PAgP atau port Aggregation ini berbeda dari etherchannel LACP yang bisa digunakan pada semua vendor etherchannel PAgP ini hanya di gunakan oleh perangkat cisco aja. Untuk konfigurasinya kita akan menggunakan layout seperti ini



Konfigurasi etherchannel PAgP terdapat perbedaan mode, jika pada Lacp kita menggunakan mode active dan mode passive pada Pagb ini kita menggunakan konfigurasi seperti pada gambar di bawah ini

```
SW2 (config-if-range)#channel-group 1 mode desirable
SW2 (config-if-range)#
-
SW2 (config-if-range)#channel-group 1 mode auto
SW2 (config-if-range)#

```

Setelah selesai memasukkan interface pada channel group 1 kita harus masuk ke interface channel group 1 dan ubah modenya menjadi trunk

```
SW1 (config-if-range)#int p1
SW1 (config-if)#sw mo tr
SW1 (config-if)#

```

```
SW2 (config)#int p0.1
SW2 (config-if)#sw mo tr
SW2 (config-if)#

```

Pada contoh di atas terlihat bahwa kita menggunakan perintan int p1 digunakan tuntuk masuk ke interface port channel 1. Jika mode port sudah di ganti menjadi mode trunk, kita bisa lihat apakah konfigurasi etherchannel sudah selesai dengan perintah show ip etherchannel summary

```
SW1#show int fa0/1
SW1(config-if)#do sh etherchannel sum
Flags: D - down P - in port-channel
I - stand-alone S - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
```

```
Number of channel-groups in use: 1
Number of aggregators: 1
```

Group	Port-channel	Protocol	Ports

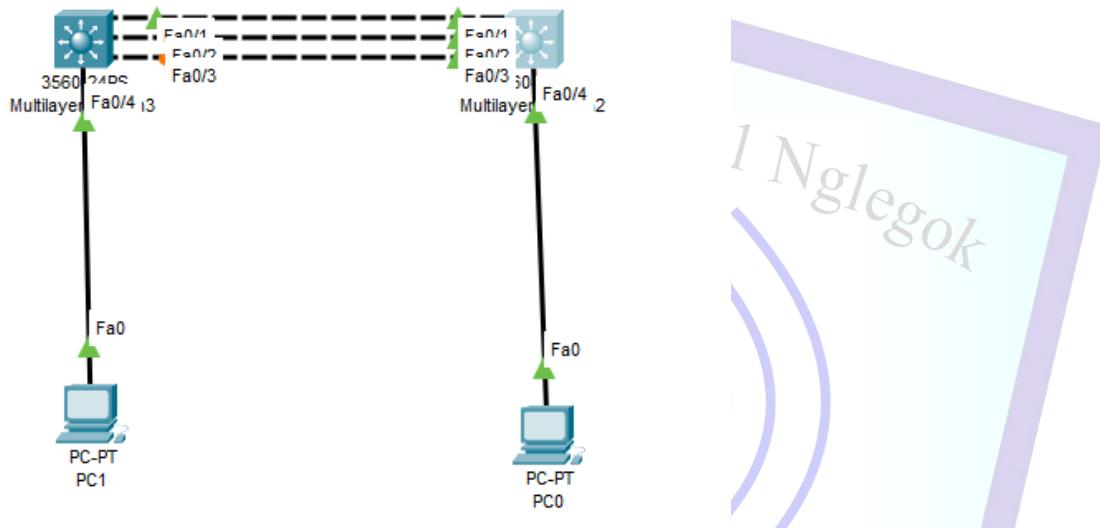
```
SW2#show int fa0/1
SW2(config-if)#do sh etherchannel sum
Flags: D - down P - in port-channel
I - stand-alone S - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
```

```
Number of channel-groups in use: 1
Number of aggregators: 1
```

Group	Port-channel	Protocol	Ports
1	Pol(SD)	PAgP	Fa0/1(D) Fa0/2(D) Fa0/3(D)

LAB 22 Etherchannel Layer 3

Lab ini bertujuan untuk menghubungkan dua switch Layer 3 (Multilayer Switch) menggunakan EtherChannel di Layer 3. EtherChannel adalah teknologi Cisco yang menggabungkan beberapa link fisik menjadi satu link logis untuk meningkatkan bandwidth dan redundansi.



Pada masing-masing switch, port Fa0/4 dikonfigurasi sebagai Layer 3 (dengan perintah no switchport) dan diberi alamat IP. Perintah no switchport mengubah interface dari Layer 2 menjadi Layer 3. Setelah itu, alamat IP dikonfigurasi agar bisa berkomunikasi dengan PC di masing-masing jaringan.

```

MLS1(config)#int f0/4
MLS1(config-if)#no sw
MLS1(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
MLS1(config-if)#ip add 192.168.1.1 255.255.255.0
MLS1(config-if)#
MLS2(config)#int f0/4
MLS2(config-if)#no sw
MLS2(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up
MLS2(config-if)#ip add 192.168.2.1 255.255.255.0
    
```

Kesalahan ini muncul karena ada ketidaksesuaian antara mode port dan port-channel. Untuk EtherChannel Layer 3, semua port yang tergabung harus diubah ke mode Layer 3 terlebih dahulu sebelum dimasukkan ke dalam channel-group.

```
MLS1(config-if-range)#no sw
Command rejected (Port-channel): Either port is L2 and port-channel is L3, or vice-versa
Command rejected (Port-channel): Either port is L2 and port-channel is L3, or vice-versa
Command rejected (Port-channel): Either port is L2 and port-channel is L3, or vice-versa
MLS1(config-if-range)#

```

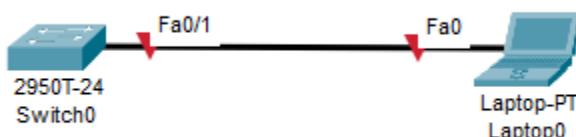
Perintah ini membuat EtherChannel secara manual (mode on), dan memasukkan port yang dipilih ke dalam channel-group 1.

```
MLS1(config-if-range)#channel-group 1 mo on  
MLS1(config-if-range)#+
```



LAB 23 Enable Telnet di Switch

Pada lab ini kita akan membahas mengenai cara untuk mengaktifkan telnet pada switch cisco. Pada switch cisco kita bisa menggunakan telnet untuk remote access ke switch. Kita akan menggunakan topologi seperti di bawah ini



Seperti yang kita tahu bahwa untuk remote menggunakan telnet kita perlu mengkonfigurasikan ip pada switch terlebih dahulu.

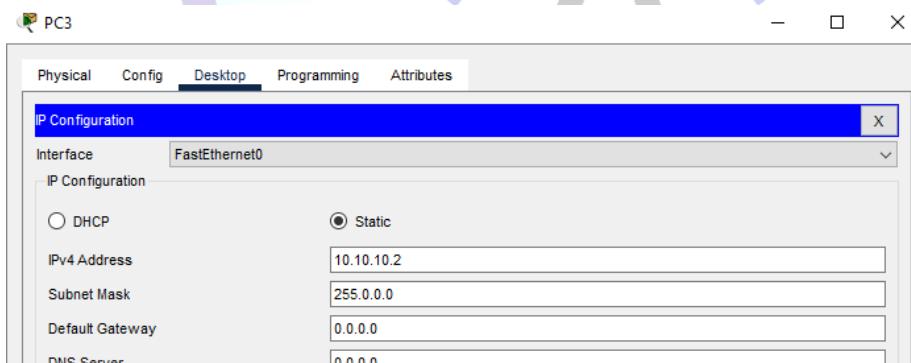
```

Switch(config-if)#int vlan 1
Switch(config-if)#no sh

Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up

Switch(config-if)#ip add 10.10.10.1 255.255.255.0
  
```



Setelah kalian selesai konfigurasi IP kita akan mencoba ping dari pc ke IP konfigurasi pada vlan 1.

```
C:\>ping 10.10.10.1

Pinging 10.10.10.1 with 32 bytes of data:

Reply from 10.10.10.1: bytes=32 time<1ms TTL=255

Ping statistics for 10.10.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Jika sudah kalian ping, selanjutnya kita akan membuat user dan mengaktifkan telnet. Pada gambar di bawah konfigurasi vty 0 3 itu berarti jumlah user yang dapat login menggunakan telnet

```
Switch(config)#username admin password 19
Switch(config)#line vty 0 3
Switch(config-line)#login local
Switch(config-line)#exit
```

Jika kalian sudah selesai membuat konfigurasi kita bisa mencoba dengan login pada cmd pc, bisa dilihat konfigurasi yang telah kita buat sudah berhasil

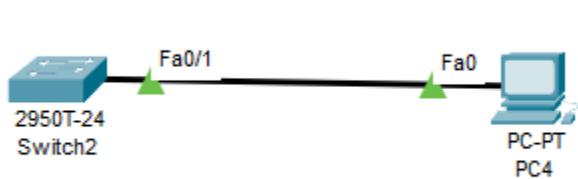
```
C:\>telnet 10.10.10.1
Trying 10.10.10.1 ...Open

User Access Verification

Username: admin
Password:
```

LAB 24 Enable SSH Switch

Pada lab ini kita akan membahas tentang cara memgaktifkan SSH pada switch cisco. Untuk mengaktifkan ssh pada cisco konfigurasinya hampir sama dengan telnet, hanya saja sedikit tambahan konfigurasi. Kita akan menggunakan layout seperti pada gambar di bawah ini



Setelah selesai membuat topologi, kita akan melanjutkan dengan menambahkan ip pada vlan 1, ini diperlukan karena kita tidak bisa menambahkan ip pada interface switch

```

Switch(config)#int vlan 1
Switch(config-if)#no dh
^
% Invalid input detected at '^' marker.

Switch(config-if)#no sh

Switch(config-if)#
%LINK-S-CHANGED: Interface Vlan1, changed state to up

%LINEPROTO-S-UPDOWN: Line protocol on Interface Vlan1, changed state to up

Switch(config-if)#ip add 192.168.1.1 255.255.255.0
Switch(config-if)#ex

```

Jika sudah selesai menambahkan ip pada vlan 1 kita di haruskan menambahkan nama domain pada switch

```

Switch(config)#ip domain-name kits.net
Switch(config)#crypto key generate rsa
^ Please define a hostname other than Switch.

```

Terlihat bahwa konfigurasi tidak berhasil di karenakan hostname pada switch masih default oleh karena itu kita harus menggantinya seperti ini

```

Switch(config)#hostname SW1
SW1(config)#crypto key generate rsa
The name for the keys will be: SW1.kits.net
Choose the size of the key modulus in the range of 360 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [512]:
^ Generating 512 bit RSA keys, keys will be non-exportable...[OK]

```

Sudah selesai kita lanjutkan dengan mengaktifkan ssh pada switch untuk konfigurasinya kalian bisa lihat seperti pada gambar di bawah

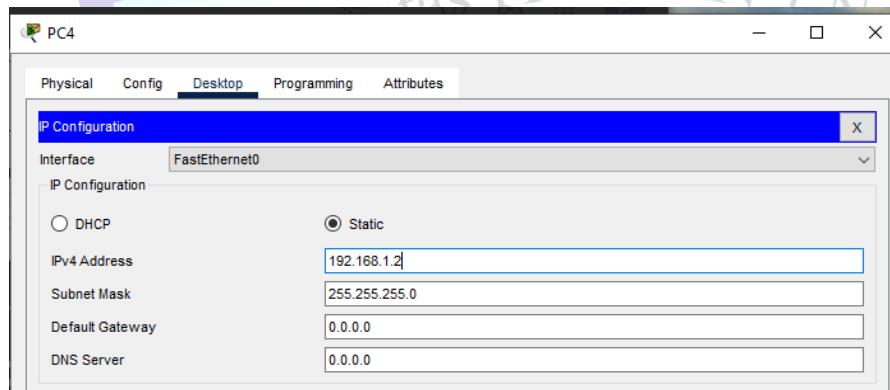
```

SW1(config)#line vty 0 4
*Mar 1 0:3:6.577: RSA key size needs to be at least 768 bits for ssh version 2
*Mar 1 0:3:6.579: %SSH-5-ENABLED: SSH 1.5 has been enabled
SW1(config-line)#transport input ssh
SW1(config-line)#login local
SW1(config-line)#

```

SW1(config-line)#login local
 SW1(config-line)#enable secret 123
 SW1(config)#username ssh password admin
 SW1(config)#

Kita tambahkan juga ip pada pc



Jika konfigurasi pada switch sudah selesai kita akan mencoba untuk akses ssh sebelum kita coba, lakukan ping terlebih dahulu

```

C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

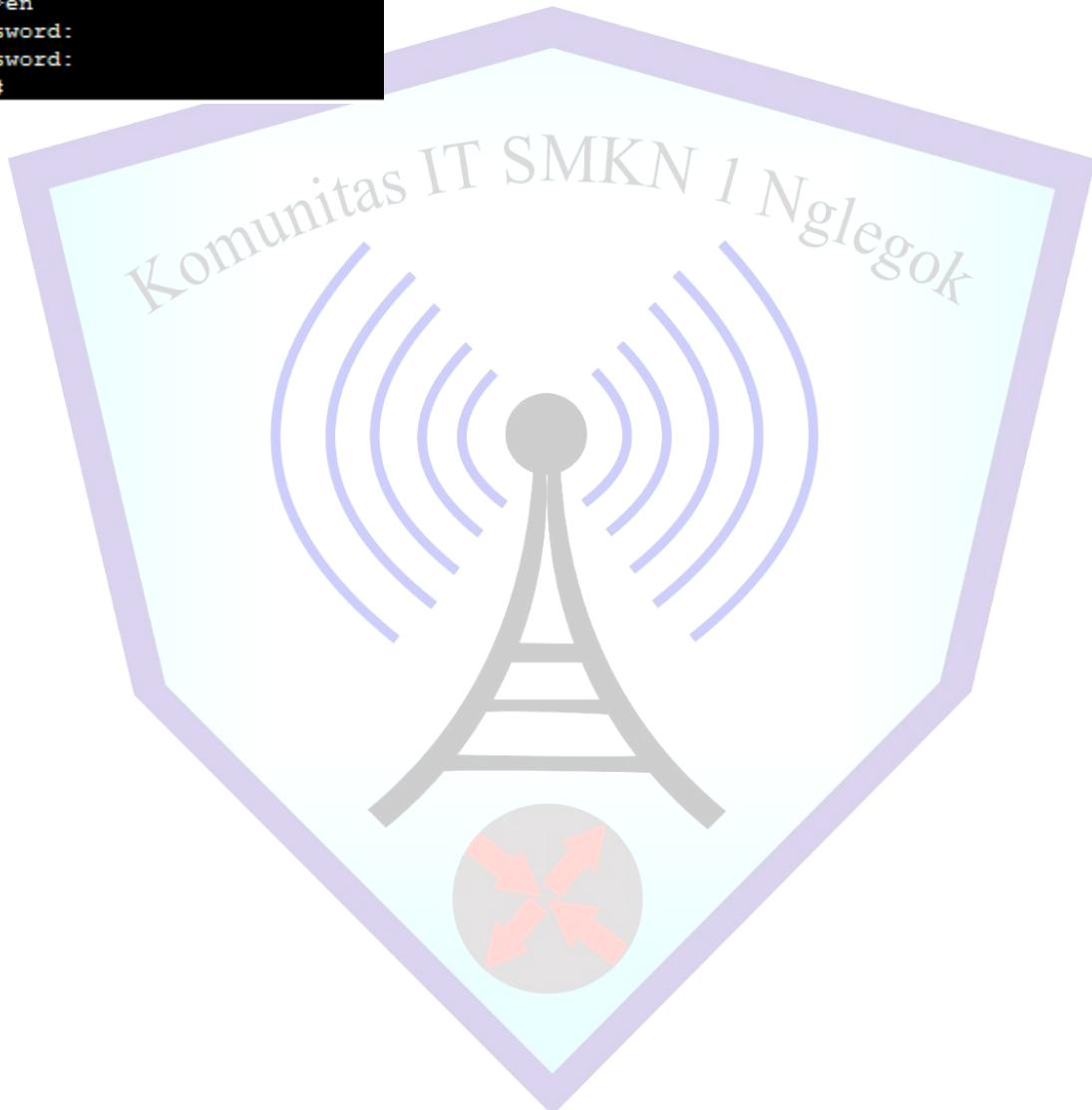
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

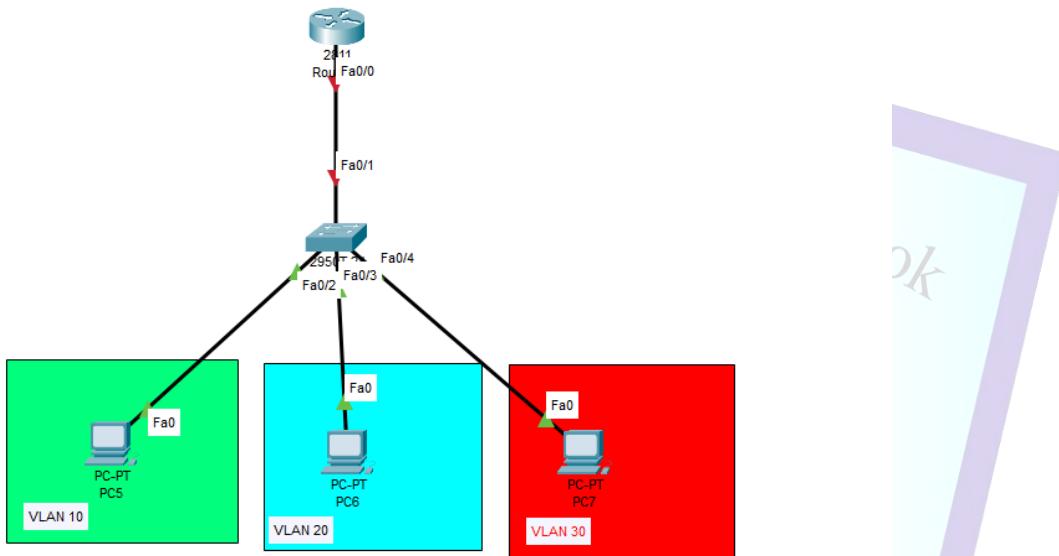
Jika sudah ping kita lanjutkan dengan akses ssh ke switch

```
C:\>ssh -l ssh 192.168.1.1  
Password:  
  
SW1>en  
Password:  
Password:  
SW1#
```



LAB 25 Inter Vlan Routing

Pada lab ini kita akan mencoba intervlan, kita akan mencoba menghubungkan vlan berbeda dengan router, tujuannya yaitu agar beda vlan dapat terhubung satu sama lain. Untuk konfigurasinya kita akan menggunakan layout jaringan seperti di bawah ini



Sebelum itu kalian kofigurasikan ip terlebih dahulu sesuai gambar di atas dan jangan lupa kalian tambahkan gateway pada setiap pc. Jika sudah kita lanjutkan kongigurasi vlan pada switch, nah jika kalian sudah selesai membuat vlan jangan sampai lupa untuk mengubah port yang mengarah ke router menjadi mode trunk.

```

Switch>en
Switch#conf
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#int f0/1
Switch(config-if)#sw mo tr
Switch(config-if)#int f0/2
Switch(config-if)#sw mo acc
Switch(config-if)#sw ac vlan 10
% Access VLAN does not exist. Creating vlan 10
Switch(config-if)#sw acs vlan 10
^
% Invalid input detected at '^' marker.

Switch(config-if)#sw acc vlan 10
Switch(config-if)#int f0/3
Switch(config-if)#sw mo ac
Switch(config-if)#sw acc vlan 20
% Access VLAN does not exist. Creating vlan 20
Switch(config-if)#sw acc vlan 20
Switch(config-if)#int f0/4
Switch(config-if)#sw mo ac
Switch(config-if)#sw acc vlan 30
% Access VLAN does not exist. Creating vlan 30
Switch(config-if)#sw acc vlan 30
Switch(config-if)#

```

Untuk konfigurasi pada router hal pertama yang dilakukan adalah menyalakan interface ke arah switch, ada 2 cara untuk menyalakannya yaitu pada menu router bisa juga menggunakan command, disini saya menggunakan command

```

Router(config)#int f0/0.10
Router(config-subif)#no sh
Router(config-subif)#no shutdown

```

Setelah port dinyalakan kita akan melanjutkan membuat port untuk vlan dengan tetap menggunakan port asli, setelah selesai lanjutkan dengan konfigurasi ip pada interface tersebut untuk ipnya kalian bisa menggunakan gateway

```

Router(config-subif)#encapsulation dot1q 10
Router(config-subif)#ip add 192.168.1.1 255.255.255.0

```

Jika sudah selesai lanjutkan dengan menambahkan interface pada vlan yang lainnya

```

Router(config-subif)#encapsulation dot1q 20
Router(config-subif)#ip add 192.168.10.1
% Incomplete command.
Router(config-subif)#ip address 192.168.10.1 255.255.255.0

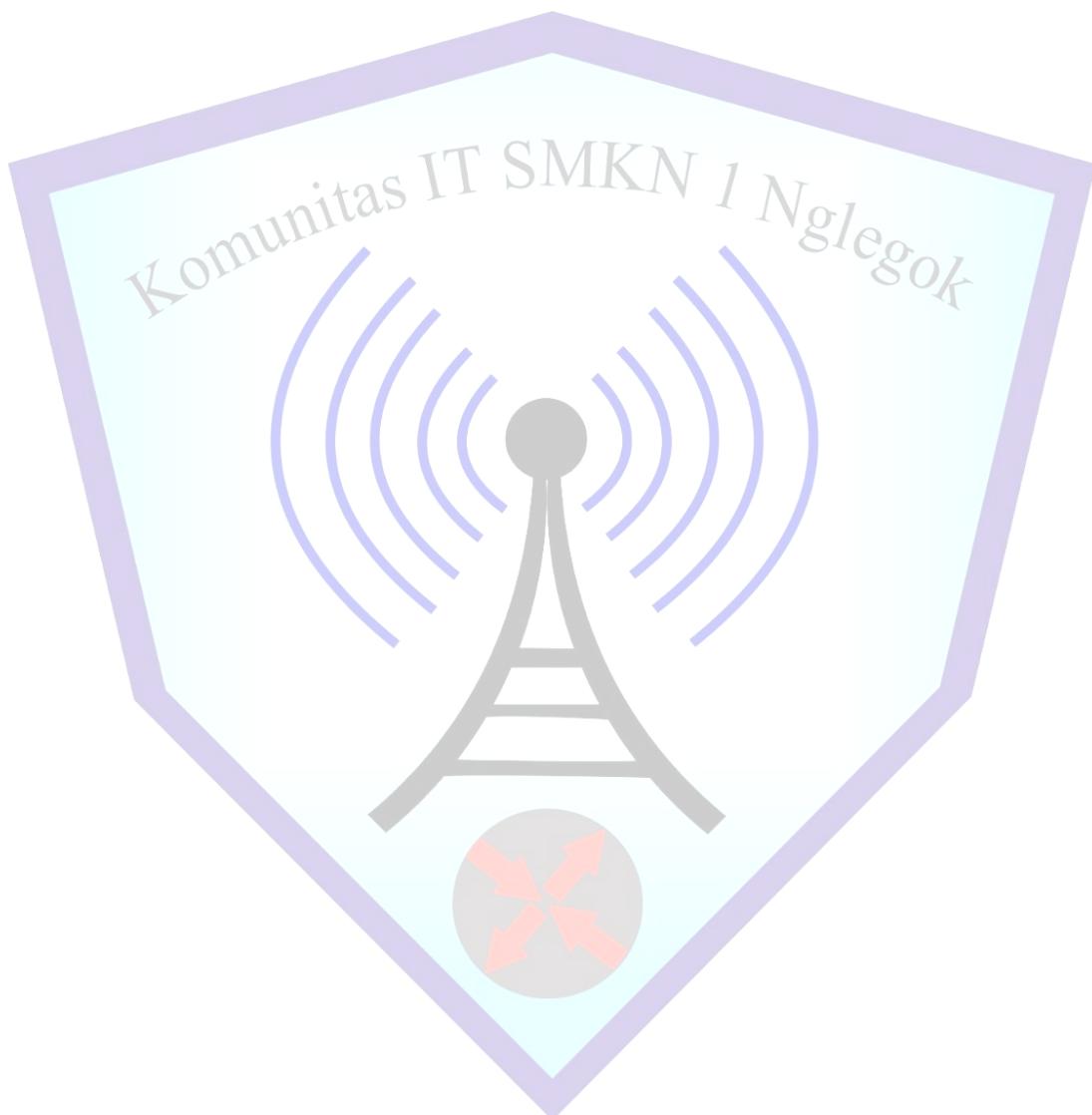
Router(config-subif)#int f0/0.30
Router(config-subif)#encapsulation dot1q 30
Router(config-subif)#ip add 192.168.20.1 255.255.255.0

```

Jika konfigurasi telah selesai kalian coba ping sesama pc tetapi yang beda vlan disini jika hasilnya TTL atau successful maka sudah berhasil

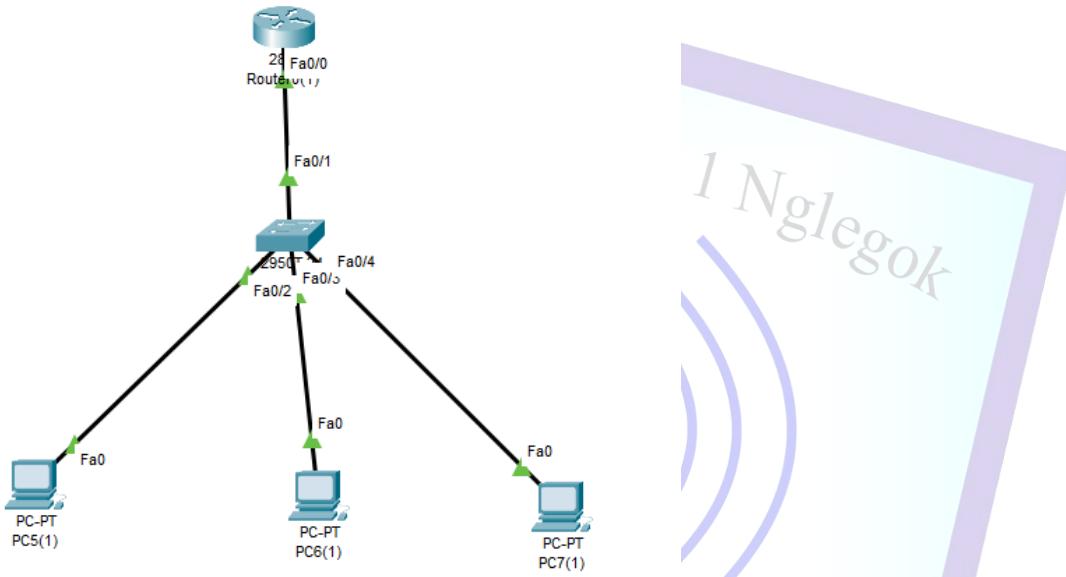
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic
	Successful	PC5	PC6	ICMP		0.000	N
	Successful	PC7	PC5	ICMP		0.000	N

HASIL PING PC 1 KE PC 2 dan 3



LAB 26 DHCP Server Router

Sebelumnya kita perlu mengkonfigurasikan ip address kepada setiap pc secara static. Pada lab ini kita akan mencoba untuk membuat supaya setiap pc bisa mendapatkan ip dynamic yang didapatkan dari router



Setelah itu kita lakukan selanjutnya adalah menambahkan ip address ke interface router yang mnegarah ke switch

```

Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int f0/0
Router(config-if)#no shut
Router(config-if)#ip address 192.168.1.1 255.255.255.0
% 192.168.1.0 overlaps with FastEthernet0/0.10
Router(config-if)#ip dhcp pool
^
% Invalid input detected at '^' marker.

Router(config-if)#ip dhcp pool coba
Router(dhcp-config)#network 192.168.1.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.1.1
Router(dhcp-config)#dn-server 8.8.8.8
^
% Invalid input detected at '^' marker.

Router(dhcp-config)#dns-server 8.8.8.8
Router(dhcp-config)#

```

Keterangan Konfigurasi di atas

ip address {ip address} {subnet mask}

Perintah ini digunakan untuk menetapkan alamat IP statis pada sebuah interface jaringan (seperti FastEthernet atau GigabitEthernet) pada perangkat Cisco.

- Contoh: ip address 192.168.10.1 255.255.255.0
Artinya interface akan dikonfigurasi dengan alamat IP 192.168.10.1 dan subnet mask 255.255.255.0.
Langkah ini penting agar perangkat dapat terhubung ke jaringan dan bertindak sebagai gateway atau pengelola DHCP.

ip dhcp pool {nama pool}

Perintah ini digunakan untuk membuat sebuah pool DHCP yang berfungsi sebagai sumber alamat IP dinamis bagi klien jaringan.

- Contoh: ip dhcp pool LAN-Pool
Setelah perintah ini dijalankan, konfigurasi DHCP lanjutan dilakukan dalam mode konfigurasi DHCP pool.

network {network address} {subnet mask}

Digunakan untuk menentukan alamat jaringan yang akan digunakan untuk mendistribusikan IP kepada klien DHCP.

- Contoh: network 192.168.10.0 255.255.255.0
Ini berarti DHCP server akan memberikan IP dari jaringan 192.168.10.0/24 kepada perangkat klien.

default-router {ip gateway}

Perintah ini menetapkan alamat default gateway yang akan diberikan kepada klien DHCP.

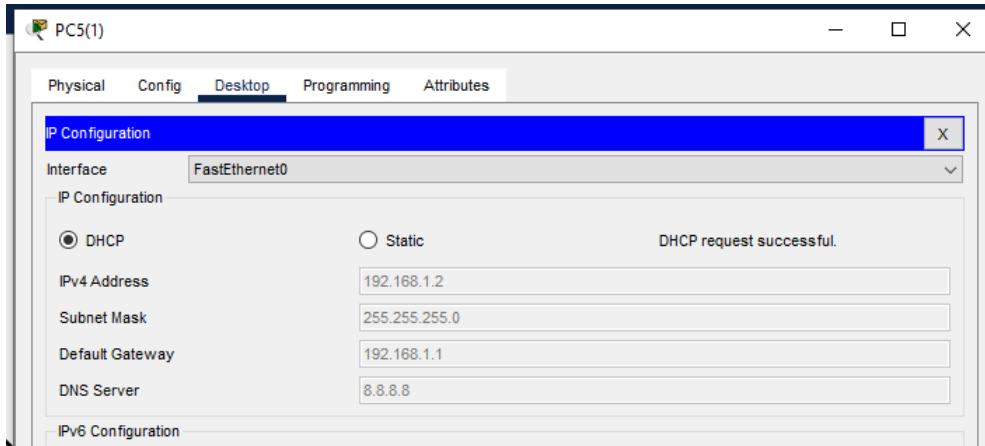
- Contoh: default-router 192.168.10.1
Dengan ini, setiap klien yang mendapat IP dari DHCP server juga akan diarahkan untuk menggunakan 192.168.10.1 sebagai jalur keluar ke jaringan lain (misalnya internet).

dns-server {ip dns}

Perintah ini menetapkan alamat server DNS (Domain Name System) untuk klien DHCP. DNS berfungsi untuk menerjemahkan nama domain (seperti google.com) menjadi alamat IP.

- Contoh: dns-server 8.8.8.8
Dalam contoh ini, klien DHCP akan menggunakan DNS publik milik Google.

Jika semua konfigurasi telah selesai kita lanjutkan dengan mengaktifkan dhcp pada pc, caranya kalian klik pc lalu lanjutkan ke menu dekstop IP dan kalian ubah menjadi DHCP

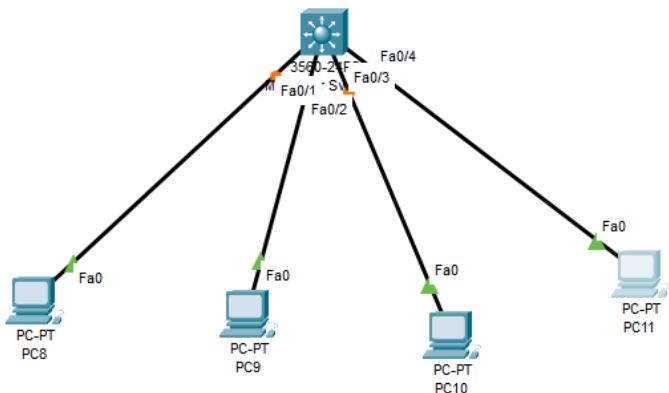


Kon... 80K



LAB 27 DHCP Server Pada Switch MLS

Pada lab ini kita akan membahas tentang dhcp MLS. Pada switch MLS kita bisa menambahkan DHCP seperti pada router. Untuk konfigurasinya sama dengan konfigurasi dhcp pada router



legok

Untuk konfigurasi kalian bisa lihat pada contoh gambar di bawah ini

```

Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#ip dhcp pool evan
Switch(dhcp-config)#network 192.168.10.0 255.255.255.0
Switch(dhcp-config)#default-route 192.168.10.1
Switch(dhcp-config)#dns-server 8.8.8.8
Switch(dhcp-config)#ex
Switch(config)#ip dhcp pool keran
Switch(dhcp-config)#network 192.168.20.0 255.255.255.0
Switch(dhcp-config)#default-route 192.168.20.1 255.255.255.0
^
* Invalid input detected at '^' marker.

Switch(dhcp-config)#default route 192.168.20.1 255.255.255.0
^
* Invalid input detected at '^' marker.

Switch(dhcp-config)#default route 192.168.20.1
^
* Invalid input detected at '^' marker.

Switch(dhcp-config)#default-route 192.168.20.1
Switch(dhcp-config)#dn-server 8.8.8.8
^
* Invalid input detected at '^' marker.

Switch(dhcp-config)#dns-server 8.8.8.8

```

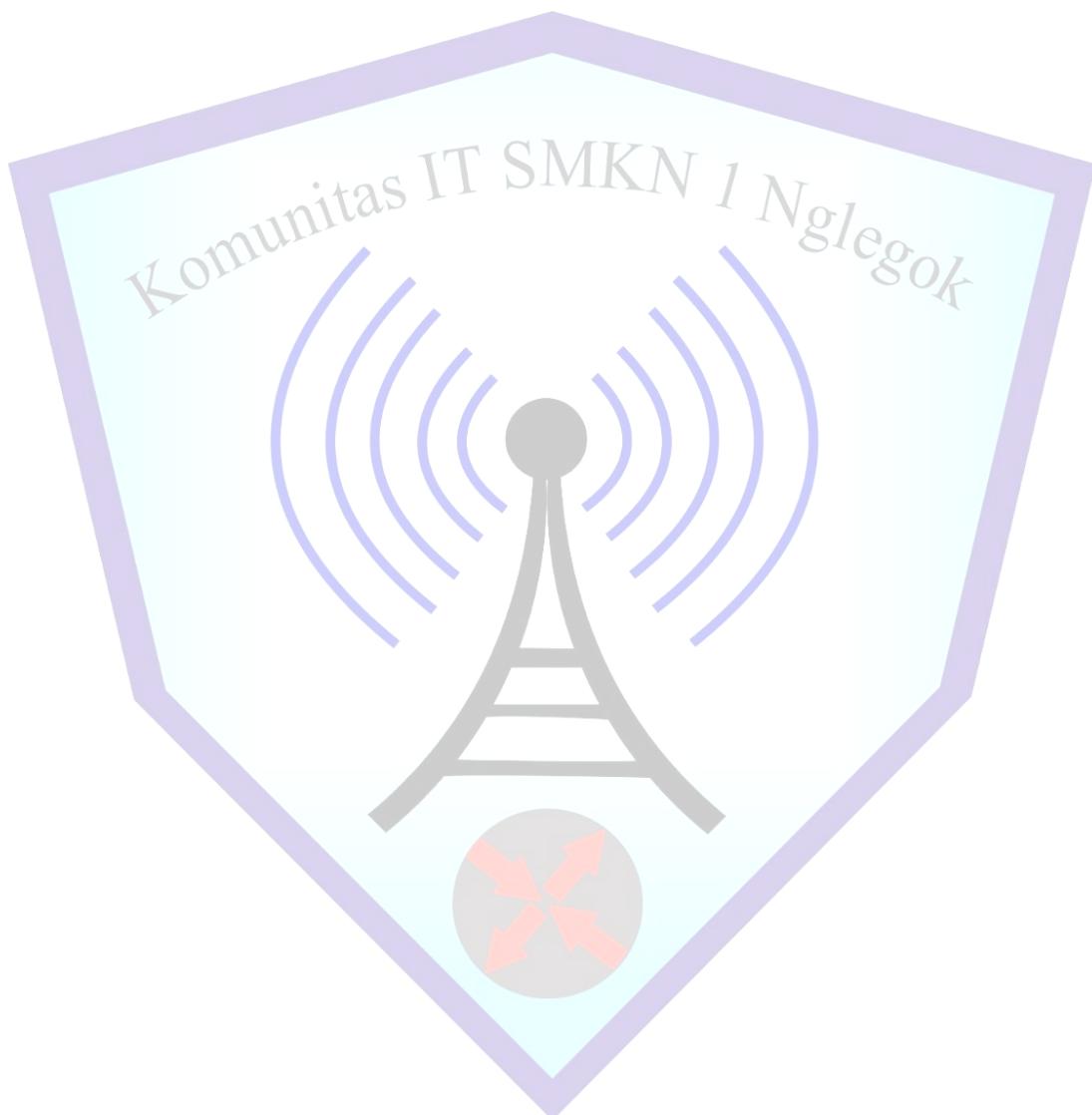
Jika sudah selesai menambahkan DHCP kita perlu menambahkan ip ke interface VLan supaya interface tersebut dapat berfungsi sebagai dhcp server. Untuk ip dikonfigurasikan akan menggunakan ip gateway yang kita isikan saat membuat dhcp

```

Switch(config)#int vlan 10
Switch(config-if)#ip add 192.168.10.1 255.255.255.0
Switch(config-if)#int vlan 20
Switch(config-if)#ip add 192.168.20.1 255.255.255.0
^

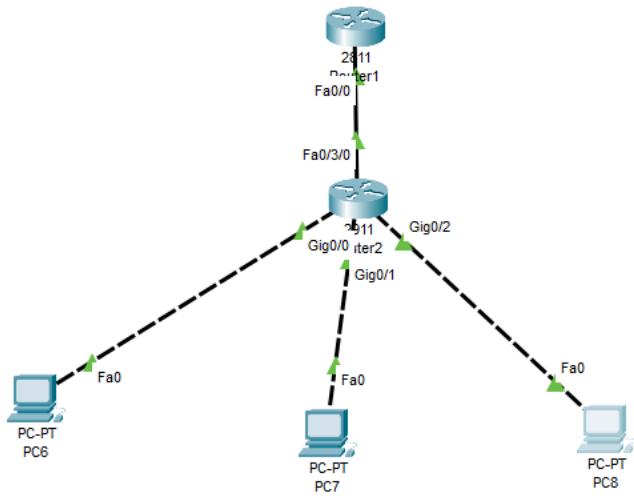
```

Jika kalian sudah selesai membuatnya kalian chek apakah sudah berhasil seperti pada lab sebelumnya



LAB 28 DHCP Relay Cisco

Lab ini kita akan membahas mengenai dhcp relay. Dhcp relay adalah satu parameter yang digunakan untuk meneruskan packet dari DHCP server menuju ke dhcp client. Agar kita memahami dhcp relay mari kita akan mencoba dhcp relay



Pertama kita akan mengonfigurasikan IP address, contohnya seperti pada gambar di bawah ini

```
server(config)#int f0/0
server(config-if)#ip add 10.10.10.1 255.255.255.0
server(config-if)#no sh
```

```

client(config)#int g0/0
client(config-if)#ip add 192.168.1.1 255.255.255.0
client(config-if)#no sh

client(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

client(config-if)#int g0/1
client(config-if)#ip add 192.168.2.1 255.255.255.0
client(config-if)#no sh

client(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

client(config-if)#int g0/2
client(config-if)#ip add 192.168.3.1 255.255.255.0
client(config-if)#no sh

client(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/2, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/2, changed state to up

```

Jika sudah selesai konfigurasikan DHCP server pada router server yang akan kita berikan ke pada client. Untuk konfigurasinya seperti pada gambar di bawah ini

```

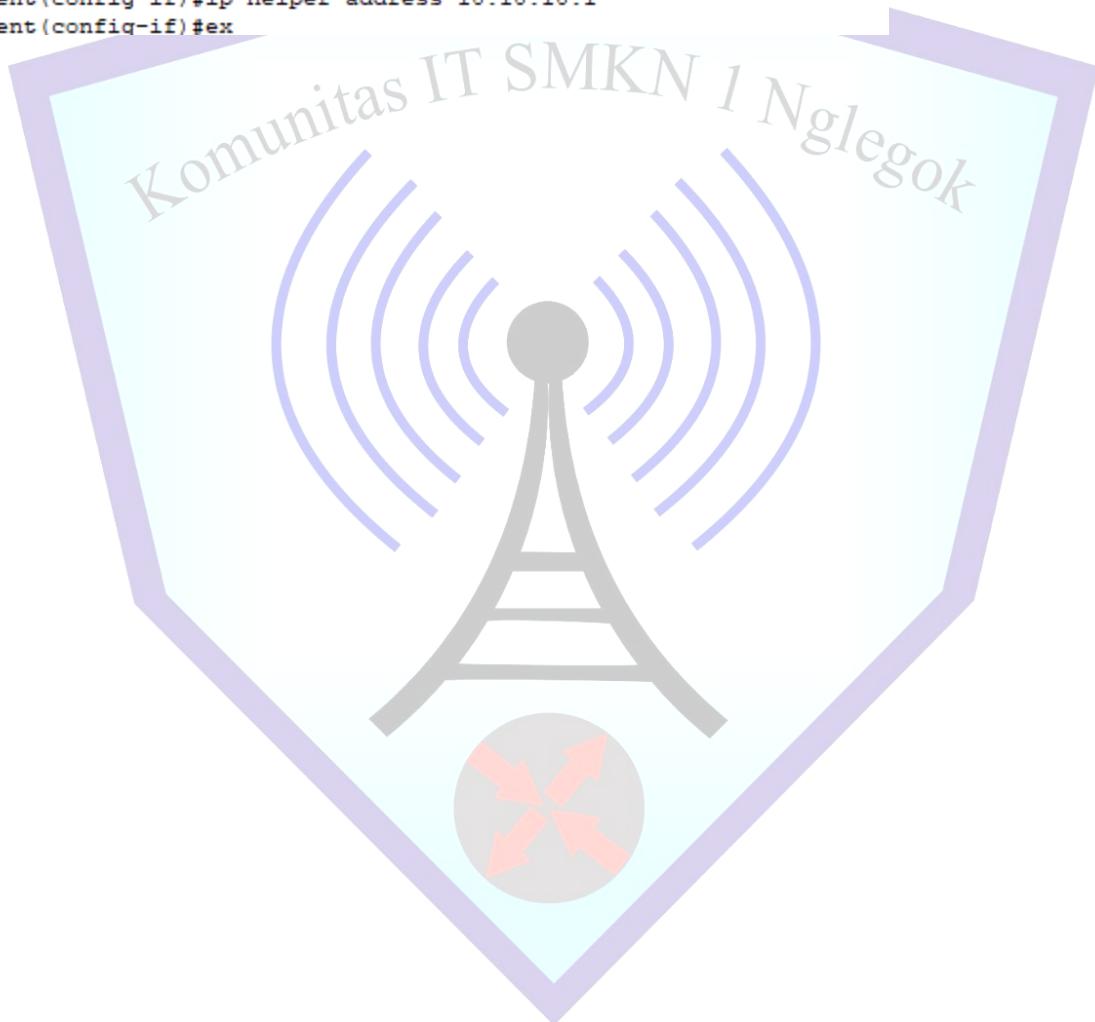
server(config)#ip dhcp pool L1
server(dhcp-config)#net 192.168.1.0 255.255.255.0
server(dhcp-config)#def 192.168.1.1
server(dhcp-config)#dns 10.10.10.1
server(dhcp-config)#ex
server(config)#ip dhcp pool L2
server(dhcp-config)#net 192.168.2.0 255.255.255.0
server(dhcp-config)#def 192.168.2.1
server(dhcp-config)#dns 10.10.10.1
server(dhcp-config)#ex
server(config)#ip dhcp pool L3
server(dhcp-config)#net 192.168.3.0 255.255.255.0
server(dhcp-config)#def 192.168.3.1
server(dhcp-config)#dns 10.10.10.1
server(dhcp-config)#ex

```

Setelah dhcp server dibuat mari kita lanjutkan dengan mengkonfigurasikan dhcp relay pada client, konfigurasinya seperti pada gambar di bawah ini

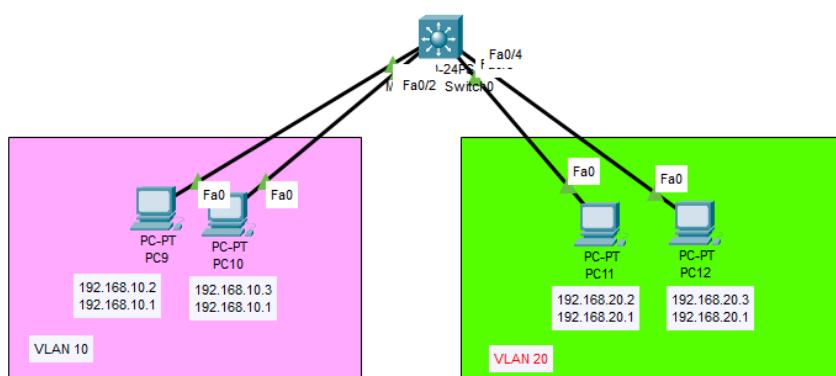
```
client(config)#int g0/0
client(config-if)#ip helper-address 10.10.10.1
client(config-if)#int g0/1
^
% Invalid input detected at '^' marker.

client(config-if)#int g0/1
client(config-if)#ip helper-address 10.10.10.1
client(config-if)#int g0/2
client(config-if)#ip helper-address 10.10.10.1
client(config-if)#exit
```



LAB 29 Routing Pada Switch MLS

Pada praktikum kali ini, kita akan mempelajari cara membuat dan mengonfigurasi Switch Virtual Interface (SVI). SVI adalah antarmuka virtual yang digunakan pada switch untuk mengizinkan komunikasi antar VLAN tanpa memerlukan perangkat router tambahan. Dengan kata lain, SVI memungkinkan sebuah switch untuk melakukan inter-VLAN routing, yaitu proses komunikasi data antar VLAN yang berbeda dalam jaringan. Biasanya, routing antar VLAN dilakukan dengan menggunakan perangkat router, namun dengan adanya SVI, fungsi ini bisa dijalankan langsung oleh switch asalkan switch tersebut mendukung fitur routing atau disebut juga sebagai Multi-Layer Switch (MLS), yang merupakan switch dengan kemampuan layer 3. Dalam konfigurasi SVI, kita akan memberikan alamat IP pada masing-masing VLAN di switch. Alamat IP ini akan berperan sebagai default gateway bagi perangkat-perangkat (host) yang tergabung dalam VLAN tersebut. Jadi, setiap perangkat dalam VLAN akan mengarahkan lalu lintas keluar VLAN-nya ke alamat IP VLAN yang dikonfigurasi di switch. Secara fungsional, konfigurasi SVI ini hampir sama seperti konfigurasi routing pada umumnya, hanya saja dilakukan langsung pada switch layer 3. Hal ini sangat berguna dalam jaringan skala menengah hingga besar karena lebih efisien dan hemat biaya dibandingkan menggunakan router terpisah. Untuk mempermudah pemahaman dan implementasi, pada praktikum ini kita akan menggunakan sebuah topologi jaringan tertentu yang telah dirancang sedemikian rupa. Topologi ini akan mencakup beberapa VLAN yang terhubung ke switch layer 3, dengan masing-masing VLAN memiliki SVI-nya sendiri. Melalui topologi ini, peserta praktikum akan melakukan konfigurasi langsung dan mengamati bagaimana komunikasi antar VLAN dapat berlangsung melalui SVI. Dengan mempelajari dan mempraktikkan konfigurasi SVI, diharapkan peserta mampu memahami konsep dasar inter-VLAN routing menggunakan switch layer 3 serta dapat menerapkannya dalam skenario jaringan nyata.



Setelah itu kita akan melanjutkan dengan menambahkan vlan

```
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#name 1
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name 2
Switch(config-vlan)#ex
Switch(config)#int range f0/1-2
Switch(config-if-range)#sw mo acc
Switch(config-if-range)#sw acc vlan 10
Switch(config-if-range)#int range f0/3-4
Switch(config-if-range)#sw mo acc
Switch(config-if-range)#sw acc vlan 20
Switch(config-if-range)#[
```

Setelah selesai membuat vlan kita lanjutkan dengan interfacenya, dengan cara masuk ke interface vlan yang sudah dibuat sebelumnya. Setelah itu kita konfigurasi ip terhadap interface tersebut, sebagai konfigurasinya kita akan menggunakan ip gateway yang ada pada vlan tersebut

```
Switch(config)#int vlan 10
Switch(config-if)#
*LINK-5-CHANGED: Interface Vlan10, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan10, changed state to up

Switch(config-if)#ip address 192.168.10.1 255.255.255.0
Switch(config-if)#int vlan 20
Switch(config-if)#
*LINK-5-CHANGED: Interface Vlan20, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan20, changed state to up

Switch(config-if)#ip add 192.168.20.2 255.255.255.0
Switch(config-if)#[
```

Jika sudah kita lanjutkan dengan menguji ping pc dengan vlan yang berbeda

```
C:\>ping 192.168.10.1

Pinging 192.168.10.1 with 32 bytes of data:

Reply from 192.168.10.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 25ms, Average = 6ms

C:\>ping 192.168.20.1

Pinging 192.168.20.1 with 32 bytes of data:

Request timed out.
Request timed out.

Ping statistics for 192.168.20.1:
    Packets: Sent = 3, Received = 0, Lost = 3 (100% loss),

Control-C
```

Bisa dilihat jika hasilnya masih time out. Ini terjadi karena kita belum mengkonfigurasikan routing pada switch maka pc yang berbeda vlan tidak akan bisa saling berkomunikasi. Kita tambahkan konfigurasi routing seperti pada gambar di bawah ini

```
Switch(config)#ip routing
Switch(config)#

```

Setelah selesai konfigurasi routing pada switch coba ping dari pc ke pc yang berbeda vlan

```
C:\>ping 192.168.20.2

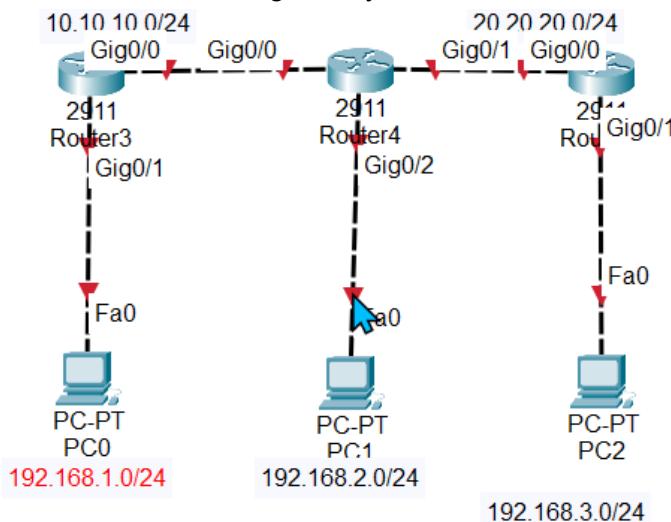
Pinging 192.168.20.2 with 32 bytes of data:

Reply from 192.168.20.2: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.20.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

LAB 30 Routing Static Cisco

Pada lab ini kita akan membahas mengenai routing static pada cisco dimana ini digunakan untuk menghubungkan beberapa network yang berbeda. Untuk pemahaman yang lebih lanjut kita akan mencoba konfigurasinya



Sebelum kita konfigurasi routing kita konfigurasikan terlebih dahulu ip address

```

Router(config)#int g0/0
Router(config-if)#ip add 10.10.10.1 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip add 10.10.10.2 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

Router(config-if)#int g0/2
Router(config-if)#ip add 192.168.2.1 255.255.255.0
Router(config-if)#no sh
  
```

```

Router(config)#int g0/0
Router(config-if)#ip add 20.20.20.2 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip add 192.168.3.1 255.255.255.0
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

```

Disini kita akan menambahkan routing table pada router supaya dapat mengenali network yang dituju. Untuk melihat apakah router sudah memiliki routing table apa belum bisa menggunakan perintah seperti pada contoh di bawah ini

```

R1(config)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
      C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
      L        10.10.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
      C        192.168.1.0/24 is directly connected, GigabitEthernet0/1
      L        192.168.1.1/32 is directly connected, GigabitEthernet0/1

```

Jika sudah terlihat bahwa tidak ada routing table mengenai network 192.168.2.0 karena itu kita harus menambahkan network tersebut ke dalam routing table, untuk menambahkannya kita menggunakan cara seperti pada contoh di bawah ini

```

R1(config)#ip route 192.168.2.0 255.255.255.0 10.10.10.2
R1(config)#ip route 192.168.3.0 255.255.255.0 10.10.10.2

```

Kita akan menambahkan juga routing table ke semua router agar network pengirim dapat kembali ke pengirim sehingga dapat terhubung satu sama lain, untuk konfigurasinya kalian bisa lihat gambar di bawah ini

```

R2(config)#ip route 192.168.3.0 255.255.255.0 20.20.20.2
R2(config)#ip route 192.168.1.0 255.255.255.0 10.10.10.1

R3(config)#ip route 192.168.1.0 255.255.255.0 20.20.20.1
R3(config)#ip route 192.168.2.0 255.255.255.0 20.20.20.1

```

Jika sudah kalian lihat routing table pada semua router sesuai dengan gambar di bawah ini

```
R1(config)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
L        10.10.10.1/32 is directly connected, GigabitEthernet0/0
  192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/1
L        192.168.1.1/32 is directly connected, GigabitEthernet0/1
S        192.168.2.0/24 [1/0] via 10.10.10.2
S        192.168.3.0/24 [1/0] via 10.10.10.2
```

```
L - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route

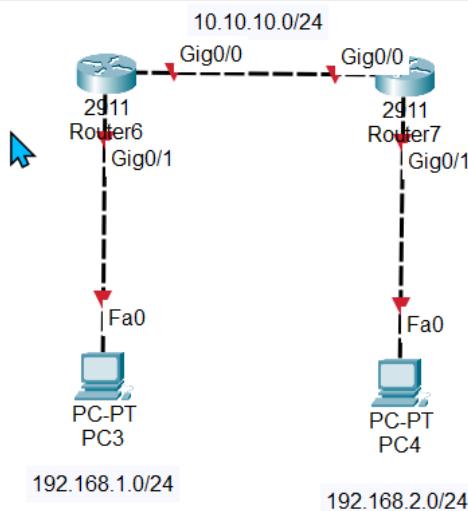
Gateway of last resort is not set

  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
L        10.10.10.2/32 is directly connected, GigabitEthernet0/0
  20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.20.20.0/24 is directly connected, GigabitEthernet0/1
L        20.20.20.1/32 is directly connected, GigabitEthernet0/1
S        192.168.1.0/24 [1/0] via 10.10.10.1
  192.168.2.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.2.0/24 is directly connected, GigabitEthernet0/2
L        192.168.2.1/32 is directly connected, GigabitEthernet0/2
S        192.168.3.0/24 [1/0] via 20.20.20.2
```

Jika semua sudah kalian lihat, kalian coba lakukan ping jika hasilnya ttl maka konfigurasi yang telah kalian buat berhasil

LAB 31 Routing Dynamic OSPF

Pada lab ini kita akan melanjutkan dari lab sebelumnya dimana sebelumnya kita menggunakan routing static, dynamic OSPF. Untuk lebih lanjutnya kita akan mencoba untuk konfigurasi dengan menggunakan topologi sederhana seperti pada gambar di bawah ini



Sebelum kita melakukan konfigurasi lanjutan kita akan mengkonfigurasi ip terlebih dahulu seperti pada gambar di bawah ini

```

R1(config)#int g0/0
R1(config-if)#ip add 10.10.10.1 255.255.255.0
R1(config-if)#no sh

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

R1(config-if)#int g0/1
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

%
% Invalid input detected at '^' marker.

R2#conf ft
^
% Invalid input detected at '^' marker.

R2#conf t
Enter configuration commands, one per line. End with CNTL/Z.
R2(config)#int g0/0
R2(config-if)#ip add 10.10.10.2 255.255.255.0
R2(config-if)#no sh
R2(config-if)#int g0/1
R2(config-if)#ip add 192.168.2.1 255.255.255.0
R2(config-if)#

```

Jika sudah kita lanjutkan dengan konfigurasi OSPF

```
R1(config)#router ospf 1
R1(config-router)#net 10.10.10.0 0.0.0.255 area 0
R1(config-router)#net 192.168.2.0 0.0.0.255 area 0
R1(config-router)#[I]
```

Pada gambar di atas terlihat bahwa untuk netmask OSPF menggunakan wildcard mask dimana merupakan kebalikan dari subnetmask

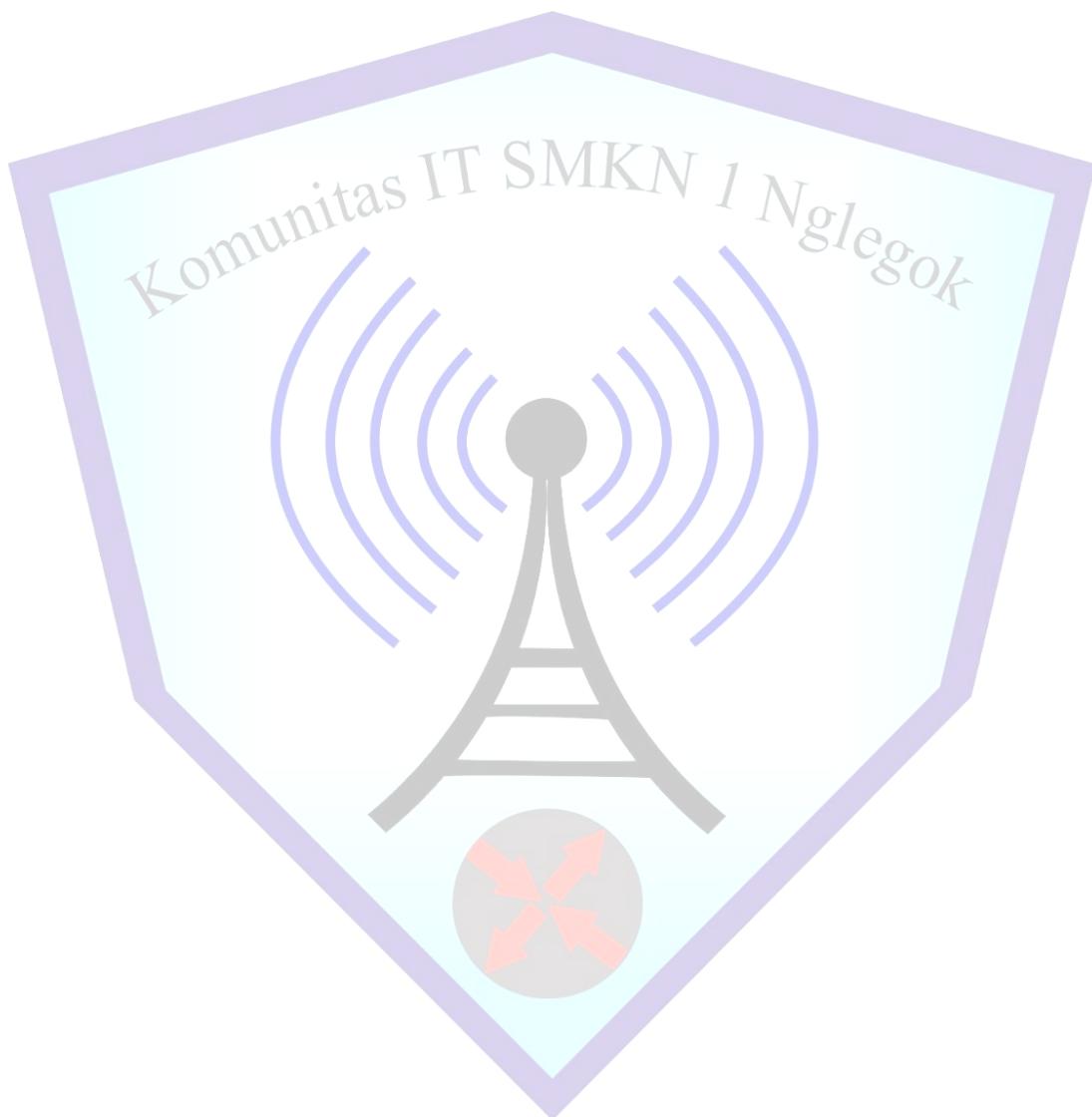
```
R2(config)#router ospf 1
R2(config-router)#net 10.10.10.0 0.0.0.255 area 0
R2(config-router)#net 10.10.10.0 0.0.0.255 area 0
00:11:51: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.1.1 on GigabitEthernet0/0 from
R2(config-router)#
R2(config-router)#net 192.168.2.0 0.0.0.255 area 0
R2(config-router)#[I]
```

Selesai coba kalian lihat pada routing table, maka nanti akan secara otomatis seperti pada gambar di bawah ini, kalian lihat pada setiap router

```
R1(config-router)#do sh ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      p - periodic downloaded static route

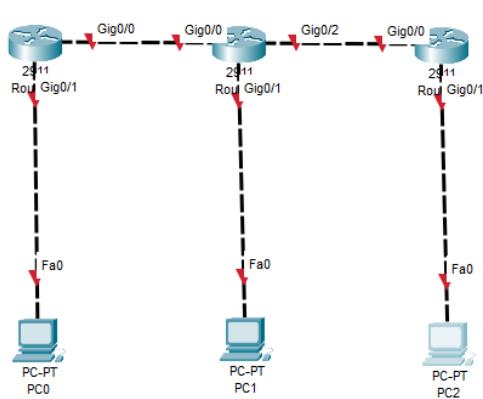
Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
L        10.10.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/1
L        192.168.1.1/32 is directly connected, GigabitEthernet0/1
O        192.168.2.0/24 [110/2] via 10.10.10.2, 00:00:56, GigabitEthernet0/0
R1(config-router)#[I]
```



LAB 32 Routing Dynamic EIGRP

Pada lab ini kita akan membahas mengenai routing dynamic EIGRP, dimana juga termasuk dalam interior gateway protocol, sama seperti ospf dan rip. Untuk lebih lanjut mari kita coba konfigurasi dengan menggunakan topologi jaringan seperti pada gambar di bawah ini



Sama seperti sebelumnya langkah pertama yang akan kita lakukan adalah mengkonfigurasikan ip pada setiap interface router

```
R1(config)#int g0/0
R1(config-if)#ip add 10.10.10.1 255.255.255.0
R1(config-if)#no sh

R1(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
int g0/1
R1(config-if)#int g0/1
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh

router(config)#int g0/0
router(config-if)#ip add 10.10.10.2 255.255.255.0
router(config-if)#no sh

router(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
int g0/1
router(config-if)#int g0/1
router(config-if)#ip add 192.168.20.1 255.255.255.0
router(config-if)#no sh

router(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
int g0/2
router(config-if)#int g0/2
router(config-if)#ip add 20.20.20.1 255.255.255.0
router(config-if)#no sh
```

```
R3(config-if)#int g0/0
R3(config-if)#ip add 20.20.20.2 255.255.255.0
R3(config-if)#no sh
R3(config-if)#int g0/1
R3(config-if)#ip add 192.168.168.3.1 255.255.255.0
^
* Invalid input detected at '^' marker.

R3(config-if)#ip add 192.168.3.1 255.255.255.0
R3(config-if)#no sh

R3(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
```

Setelah selesai mari kita lihat tabel routing pada semua router

```
R1(config-if)#do sh ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
L        10.10.10.1/32 is directly connected, GigabitEthernet0/0
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, GigabitEthernet0/1
L        192.168.1.1/32 is directly connected, GigabitEthernet0/1

Router(config-if)#do sh ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

      10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/24 is directly connected, GigabitEthernet0/0
L        10.10.10.2/32 is directly connected, GigabitEthernet0/0
      20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.20.20.0/24 is directly connected, GigabitEthernet0/2
L        20.20.20.1/32 is directly connected, GigabitEthernet0/2
      192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.20.0/24 is directly connected, GigabitEthernet0/1
L        192.168.20.1/32 is directly connected, GigabitEthernet0/1
```

```
R3(config-if)#do sh ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.20.0.0/24 is directly connected, GigabitEthernet0/0
L        20.20.2.0/32 is directly connected, GigabitEthernet0/0
  192.168.3.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.3.0/24 is directly connected, GigabitEthernet0/1
L        192.168.3.1/32 is directly connected, GigabitEthernet0/1
```

Pada semua router terlihat tidak semua network ada pada routing table, untuk itu kita akan menambahkannya akan tetapi kita tidak akan menambahkan routing table dengan cara static melainkan akan menggunakan routing dynamic EIGRP

```
R1(config)#router eigrp 1
R1(config-router)#no auto sum
R1(config-router)#net 192.168.10.0
R1(config-router)#net 10.10.10.0
```

Disini kita akan menambahkan wildcards mask pada semua router

```
Router(config)#router eigrp 1
Router(config-router)#no auto sum
Router(config-router)#net 192.168.20.0
Router(config-router)#net 10.10.10.0
Router(config-router)#
%DUAL-5-NBRCHANGE: IP-EIGRP 1: Neighbor 10.10.10.1 (GigabitEthernet0/0) is up: new
adjacency
```

```
Router(config-router)#net 20.20.20.0
```

```
R3(config)#router eigrp 1
R3(config-router)#no auto sum
R3(config-router)#192.168.30.0
^
% Invalid input detected at '^' marker.

R3(config-router)#net 192.168.30.0
R3(config-router)#net 30.30.30.0
--
```

LAB 33 Routing Dynamic RIPv2

Pada lab ini kita akan membahas mengenai konfigurasi routing dynamic RIPv2, RIPv2 merupakan pengembangan dari RIPv1, untuk melihat perbedaannya lihatlah penjelasan berikut:

Tipe Routing:

- RIPv1: Classful (tidak menyertakan subnet mask)
- RIPv2: Classless (menyertakan subnet mask)

Dukungan VLSM (Variable Length Subnet Mask)

- RIPv1: Tidak mendukung
- RIPv2: Mendukung

Metode Pengiriman Update:

- RIPv1: Broadcast ke alamat 255.255.255.255
- RIPv2: Multicast ke alamat 224.0.0.9

Autentikasi:

- RIPv1: Tidak ada
- RIPv2: Mendukung autentikasi (plaintext dan MD5)

Kompatibilitas Jaringan:

- RIPv1: Terbatas pada jaringan sederhana
- RIPv2: Bisa digunakan di jaringan kompleks

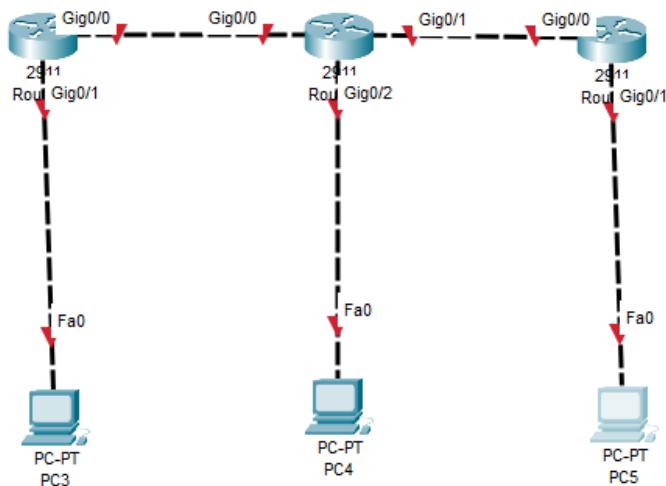
Isi Update Routing:

- RIPv1: Tidak menyertakan informasi subnet mask
- RIPv2: Menyertakan subnet mask

Efisiensi Jaringan:

- RIPv1: Kurang efisien karena broadcast
- RIPv2: Lebih efisien karena multicast

Setelah kalian memahami perbedaan kedua versi tersebut, kita lanjutkan untuk melakukan konfigurasi RIPv2



Sebelum kita memulai konfigurasi pastikan kita setting ip address sesuai dengan pada layout diatas. Untuk contoh konfigurasinya ip address seperti contoh gambar di bawah ini

```
R1(config)#int g0/0
R1(config-if)#ip add 10.10.10.1 255.255.255.252
R1(config-if)#no sh

R1(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

R1(config-if)#int g0/1
R1(config-if)#ip add 192.168.10.1 255.255.255.0
R1(config-if)#no sh

R2(config)#int g0/0
R2(config-if)#ip ad 10.10.10.2 255.255.255.252
R2(config-if)#no sh

R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

R2(config-if)#int g0/1
R2(config-if)#ip ad 20.20.20.1 255.255.255.252
R2(config-if)#no sh

R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

R2(config-if)#int g0/2
R2(config-if)#ip ad 192.168.20.1 255.255.255.0
R2(config-if)#no sh
```

```
R3(config)#int g0/0
R3(config-if)#ip ad 20.20.20.2 255.255.255.252
R3(config-if)#no sh

R3(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

R3(config-if)#int g0/1
R3(config-if)#ip ad 192.168.30.1 255.255.255.0
R3(config-if)#no sh
```

Jika semua ip kalian konfigurasikan kita lanjutkan dengan membuat konfigurasi RIP pada router, untuk contoh konfigurasi kalian bisa lihat pada contoh di bawah ini

```
R1(config)#route rip
R1(config-router)#version 2
R1(config-router)#no auto summary
R1(config-router)#net 10.10.10.0
R1(config-router)#net 192.168.10.0
R1(config-router)#end
```

Pada gambar di atas terlihat jika saya menambahkan perintah version2 ini digunakan untuk merubah versi dari RIP, setelah itu ada perintah no auto summary yang digunakan supaya routing tida supary, network yang ditambahkan ini adalah network yang terhubung langsung pada router

```
R2(config)#router rip
R2(config-router)#version 2
R2(config-router)#no auto summary
R2(config-router)#net 10.10.10.0
R2(config-router)#net 20.20.20.0
R2(config-router)#net 192.168.20.0
R2(config-router)#end

no config-rr#ex
R3(config)#router rip
R3(config-router)#version 2
R3(config-router)#no auto summary
R3(config-router)#net 20.20.20.0
R3(config-router)#net 192.168.30.0
R3(config-router)#end
no
```

Jika semua sudah selesai kita akan melihat apakah rip sudah berjalan atau belum, untuk melihatnya kita lihat pada table routing apakah network sudah ditambahkan atau belum. Untuk melihat tabel routing bisa menggunakan perintah show ip route

```
R2#show ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/30 is directly connected, GigabitEthernet0/0
L        10.10.10.2/32 is directly connected, GigabitEthernet0/0
  20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.20.20.0/30 is directly connected, GigabitEthernet0/1
L        20.20.20.1/32 is directly connected, GigabitEthernet0/1
R  192.168.10.0/24 [120/1] via 10.10.10.1, 00:00:22, GigabitEthernet0/0
    192.168.20.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.20.0/24 is directly connected, GigabitEthernet0/2
L        192.168.20.1/32 is directly connected, GigabitEthernet0/2
R  192.168.30.0/24 [120/1] via 20.20.20.2, 00:00:07, GigabitEthernet0/1

R1#show ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        10.10.10.0/30 is directly connected, GigabitEthernet0/0
L        10.10.10.1/32 is directly connected, GigabitEthernet0/0
  20.0.0.0/8 is variably subnetted, 1 subnets
R  20.20.20.0/30 [120/1] via 10.10.10.2, 00:00:12, GigabitEthernet0/0
    192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.10.0/24 is directly connected, GigabitEthernet0/1
L        192.168.10.1/32 is directly connected, GigabitEthernet0/1
R  192.168.20.0/24 [120/1] via 10.10.10.2, 00:00:12, GigabitEthernet0/0
R  192.168.30.0/24 [120/2] via 10.10.10.2, 00:00:12, GigabitEthernet0/0

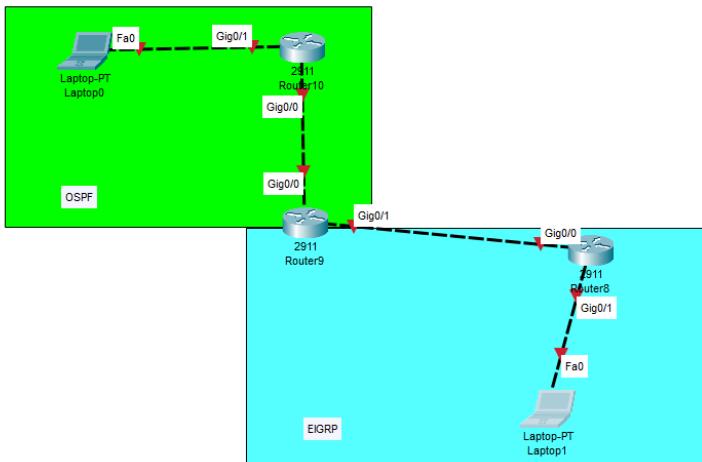
R3#show ip ro
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
      i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
      * - candidate default, U - per-user static route, o - ODR
      P - periodic downloaded static route

Gateway of last resort is not set

  10.0.0.0/30 is subnetted, 1 subnets
R  10.10.10.0/30 [120/1] via 20.20.20.1, 00:00:20, GigabitEthernet0/0
  20.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        20.20.20.0/30 is directly connected, GigabitEthernet0/0
L        20.20.20.2/32 is directly connected, GigabitEthernet0/0
E  192.168.10.0/24 [120/2] via 20.20.20.1, 00:00:20, GigabitEthernet0/0
    192.168.20.0/24 [120/1] via 20.20.20.1, 00:00:20, GigabitEthernet0/0
    192.168.30.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.30.0/24 is directly connected, GigabitEthernet0/1
L        192.168.30.1/32 is directly connected, GigabitEthernet0/1
```

LAB 34 Redistribute OSPF dan EIGRP

Pada lab ini kita akan membahas tentang redistribute OSPF dan EIGRP, sebelum melakukan konfigurasinya kita akan mengenal terlebih dahulu apa itu redistribute. Redistribute adalah protocol yang ada pada routing yang akan digunakan untuk mengirim informasi routing pada protocol routing yang berbeda. Agar routing table yang dibuat oleh kedua routing protocol dapat dire distributkan harus mengkonfigurasu redistribuse pada kedua routing protocol, supaya kedua routing dapat saling bertukar informasi routing yang mereka buat



Untuk konfigurasi pertama kita akan konfigurasikan terlebih dahulu ip address pada router dan pc sesuai dengan layout diatas untuk contoh konfigurasinya kalian bisa lihat seperti pada gambar di bawah ini

```

Core(config)#int g0/0
Core(config-if)#ip add 10.10.10.1 255.255.255.252
Core(config-if)#no sh

Core(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

Core(config-if)#int g0/1
Core(config-if)#ip add 11.11.11.1 255.255.255.252
Core(config-if)#
Core(config-if)#no sh

R1(config)#
R1(config)#int g0/0
R1(config-if)#ip add 10.10.10.2 255.255.255.252
R1(config-if)#no sh
R1(config-if)#int g0/1
R1(config-if)#ip add 192.168.1.1 255.255.255.0
R1(config-if)#no sh

```

```
R2(config)#int g0/0
R2(config-if)#ip add 11.11.11.2 255.255.255.252
R2(config-if)#no sh

R2(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

R2(config-if)#int g0/1
R2(config-if)#ip add 192.168.2.1 255.255.255.0
R2(config-if)#no sh
```

Setelah konfigurasi selesai, kita lanjutkan dengan konfigurasi routing static pada ketiga router. Untuk konfigurasinya seperti pada gambar di bawah ini. Disclaimer jangan lupakan bahwa untuk R1 menggunakan OSPF dan untuk R2 adalah router EIGRP

Untuk Core konfigurasikan OSPD dan EIGRP, untuk konfigurasinya kalian bisa lihat seperti pada contoh di bawah ini

```
Core(config)#router eigrp 1
Core(config-router)#net 10.0.0.0
Core(config-router)#net 11.0.0.0
Core(config-router)#
%DUAL-5-NBRCHANGE: IP-EIGRP 1: Neighbor 11.11.11.2 (GigabitEthernet0/1) is up: new
adjacency

Core(config-router)#no auto sum
Core(config-router)#ex
Core(config)#router ospf 1
Core(config-router)#net 10.10.10.0 0.0.0.3 area 0
Core(config-router)#net 11.11.11.
00:12:10: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.1.1 on GigabitEthernet0/0 from LOADING
to FULL,
Core(config-router)#net 11.11.11.0 0.0.0.3 area 0
Core(config-router)#ex
Core(config)#do write
```

Jika kalian sudah melakukan semua konfigurasi selanjutnya kalian coba lakukan ping

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

```

Ping dari laptop 1 ke laptop 2 masih time out ini dikarenakan table routing yang dimiliki EIGRP tidak dimiliki oleh OSPF begitu pula sebaliknya. Untuk menyebarkan table routing tersebut kita harus menggunakan redistribute, redistribute ini harus kita konfigurasikan pada router yang menghubungkan kedua protocol routing tersebut, yaitu core, untuk konfigurasinya kalian bisa lihat seperti pada contoh di bawah ini

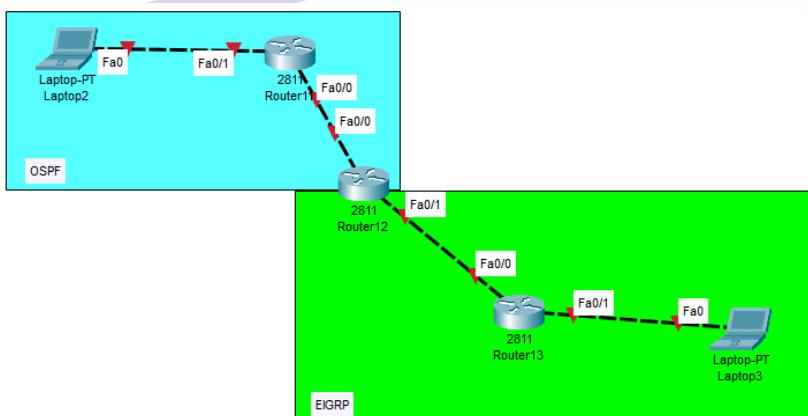
```
Core(config)#router eigrp 1
Core(config-router)#redistribute ospf 1 metric ?
<1-4294967295> Bandwidth metric in Kbits per second
Core(config-router)#redistribute ospf 1 metric
* Incomplete command.
Core(config-router)#router ospf 1
Core(config-router)#redistribute eigrp ?
<1-65535> Autonomous system number
Core(config-router)#redistribute eigrp 1 ?
metric      Metric for redistributed routes
metric-type OSPF/IS-IS exterior metric type for redistributed routes
subnets    Consider subnets for redistribution into OSPF
tag        Set tag for routes redistributed into OSPF
<cr>
Core(config-router)#redistribute eigrp 1 subnets
```

Jika sudah selesai kalian coba lakukan ping kembali. Terlihat pada contoh dibawah ini bahwa hasilnya TTL artinya konfigurasi redistribute yang telah kita buat sudah berhasil

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	Laptop0	Laptop1	ICMP		0.000	N	0	(edit)	

LAB 35 Redistribute OSPF dan RIP

Pada lab ini merupakan kelanjutan dari lab sebelumnya, jika sebelumnya kita membuat redistribute OSPF dan EIGRP pada lab ini kita akan membuat redistribute OSPF dan RIP, untuk membuatnya kita akan menggunakan layout seperti pada gambar di bawah ini



Untuk konfigurasi pertama kita konfigurasikan terlebih dahulu IP address seperti pada gambar di bawah ini

```

Core(config)#int f0/0
Core(config-if)#ip ad 10.10.10.1 255.255.255.252
Core(config-if)#no sh

Core(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

Core(config-if)#int f0/1
Core(config-if)#ip add 20.20.20.1 255.255.255.252
Core(config-if)#no sh

-----
R1(config)#int f0/0
R1(config-if)#ip ad 10.10.10.2 255.255.255.252
R1(config-if)#no sh

R1(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

R1(config-if)# int f0/1
R1(config-if)#ip ad 192.168.1.1 255.255.255.0
R1(config-if)#no sh

```

```
R2(config)#int f0/0
R2(config-if)#ip ad 20.20.20.2 255.255.255.252
R2(config-if)#no sh

R2(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

R2(config-if)#int f0/1
R2(config-if)#ip ad 192.168.2.1 255.255.255.0
R2(config-if)#no sh
```

Setelah konfigurasinya selesai kita akan melanjutkan dengan mengkonfigurasikan routing pada ketiga router, untuk router core akan menjadi jembatan kedua routing protocol oleh karena itu kita harus mengkonfigurasikan OSPF dan RIP pada router core, untuk contoh konfiugrasinya seperti pada gambar di bawah ini

```
Core(config)#router ospf 1
Core(config-router)#net 10.10.10.0 0.0.0.3 area 0
Core(config-router)#net 20.20.20.0 0.0.0.3 area 0
Core(config-router)#ex
Core(config)#router rip
Core(config-router)#net 10.10.10.0
Core(config-router)#net 20.20.20.0
Core(config-router)#no auto sum
```

Jika sudah kita lanjutkan dengan konfigurasi OSPF pada R1 seperti pada gambar di bawah ini

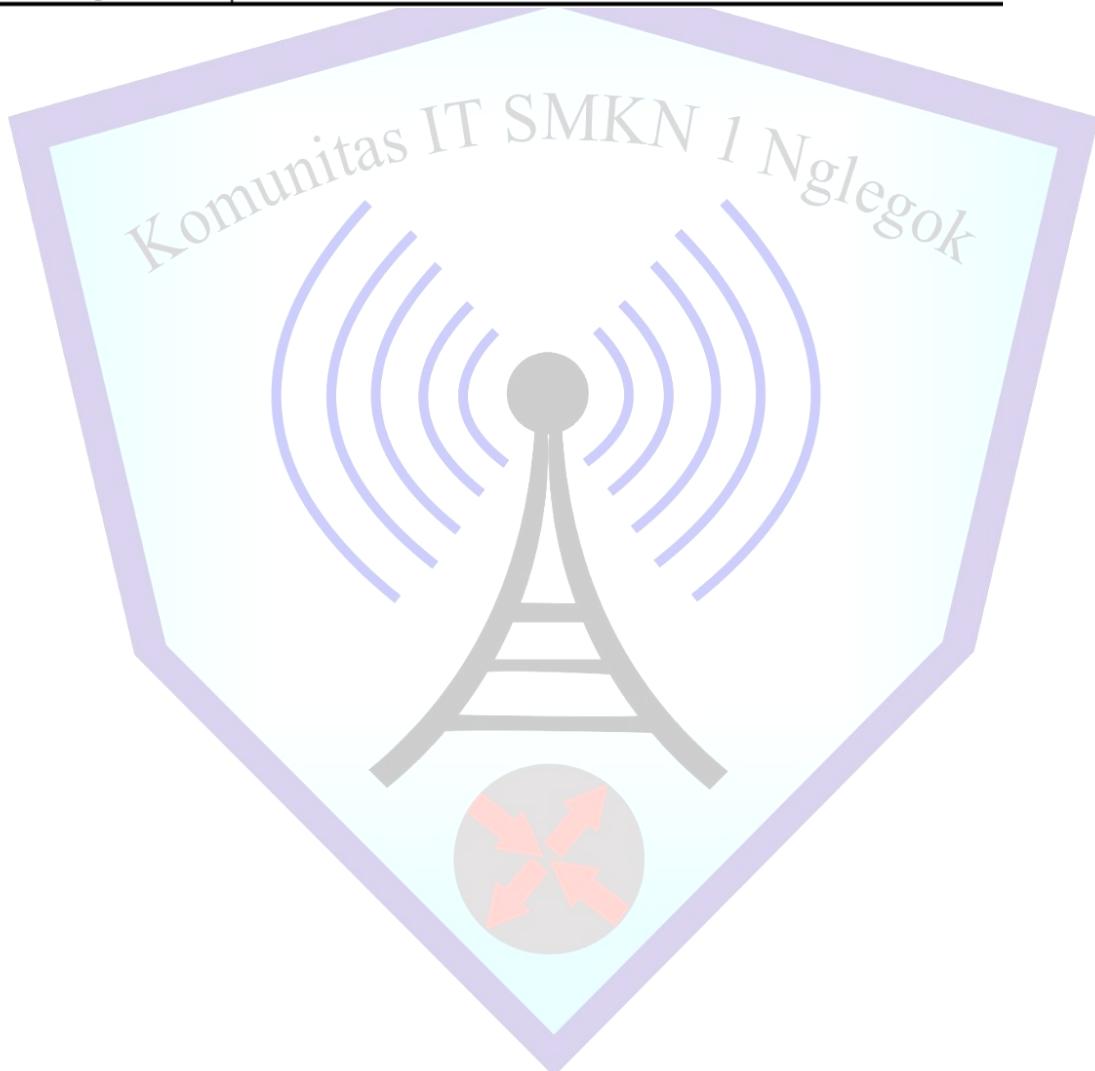
```
R1(config)#router ospf 1
R1(config-router)#net 10.10.10.0 0.0.0.3 area 0
R1(config-router)#net 192.168.1.0 0.0.0
00:06:36: %OSPF-5-ADJCHG: Process 1, Nbr 20.20.20.1 on FastEthernet0/0 from LOADING to
FULL, Loading Done

^
* Invalid input detected at '^' marker.

R1(config-router)#net 192.168.1.0 0.0.0.255 area 0
...
R2(config)#router rip
R2(config-router)#net 20.20.20.0
R2(config-router)#net 192.168.2.0
R2(config-router)#no auto sum
```

Jika sudah konfigurasikan OSPF dan RIP pada ketiga router, selanjutnya konfigurasikan redistribute, untuk contohmya seperti pada contoh di bawah in

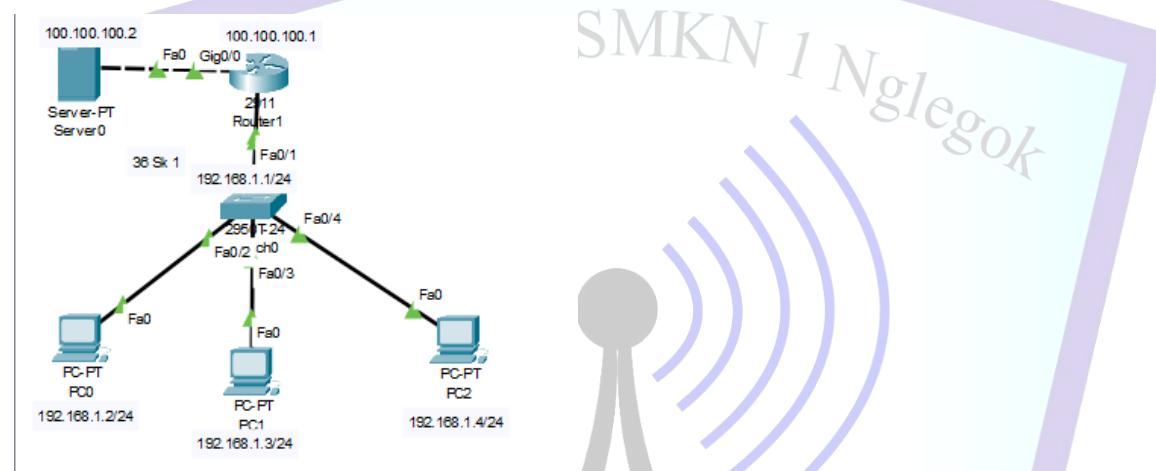
```
* cisco@core# Core(config-router)#redistribute rip ?  
metric      Metric for redistributed routes  
metric-type OSPF/IS-IS exterior metric type for redistributed routes  
subnets    Consider subnets for redistribution into OSPF  
tag        Set tag for routes redistributed into OSPF  
<CL>  
Core(config-router)#redistribute rip subnets  
Core(config-router)#[
```



LAB 36 Standar Acces List

Pada skenario ini, saya akan membuat sebuah topologi jaringan yang terdiri dari tiga client. Dari ketiga client tersebut, hanya satu client yang akan diizinkan untuk berkomunikasi dengan router, sementara dua client lainnya akan diblokir aksesnya. Untuk mendukung skenario ini, saya akan menggunakan topologi jaringan sebagai berikut.

Skenario 1



1. Pertama tama kalian konfigurasikan IP address terlebih dahulu seperti topologi

```
Router(config)#int gig0/0
Router(config-if)#ip add 100.100.100.1 255.255.255.252
Router(config-if)#no sh
Router(config-if)#int gig0/1
Router(config-if)#ip add 192.168.1.1. 255.255.255.0
^
* Invalid input detected at '^' marker.

Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
```

2. Kita akan mencoba memasukkan command “access-list”, jika kita tahu akan command kita bisa mengetahui dengan menambahkan tanda tanya pada bagian akhir command.

Bisa kita lihat dibawah bahwa

- a. Angka 1 sampai 00 merupakan standart access list
- b. Dan angka 100-199 merupakan extended access list

```
Router(config)#access-list ?
<1-99>    IP standard access list
<100-199>  IP extended access list
```

3. Saya akan mencoba melanjutkan command access-list dengan menambahkan angka 1 dan jika kia ingin mengetahui isinya bisa dengan menambahkan ? bisa kita lihat disini yaitu:

- a. Deny = berarti tidak mengizinkan
- b. Permit = berarti mengizinkan
- c. Remark = berguna untuk memberikan komentar di ACL

```
Router(config)#access-list 1 ?
deny   Specify packets to reject
permit  Specify packets to forward
remark Access list entry comment
```

4. Kita akan melanjutkan kembali dengan menambahkan permit, karena saya hanya mengizinkan 1 client yang dapat berkomunikasi melalui router. Bisa dilihat ada 3 kemungkinan yaitu;

- a. A.B.C.D = digunakan untuk suatu rentang ip
- b. Any = berlaku untuk semuanya
- c. Host = Berlaku hanya 1 IP

```
Router(config)#access-list 1 permit ?
A.B.C.D  Address to match
any      Any source host
host    A single host address
```

5. Sebagai contoh saya akan mengizinkan 192.168.1.2 saja yang dapat berkomunikasi dengan router

```
Router(config)#access-list 1 permit host 192.168.1.2
```

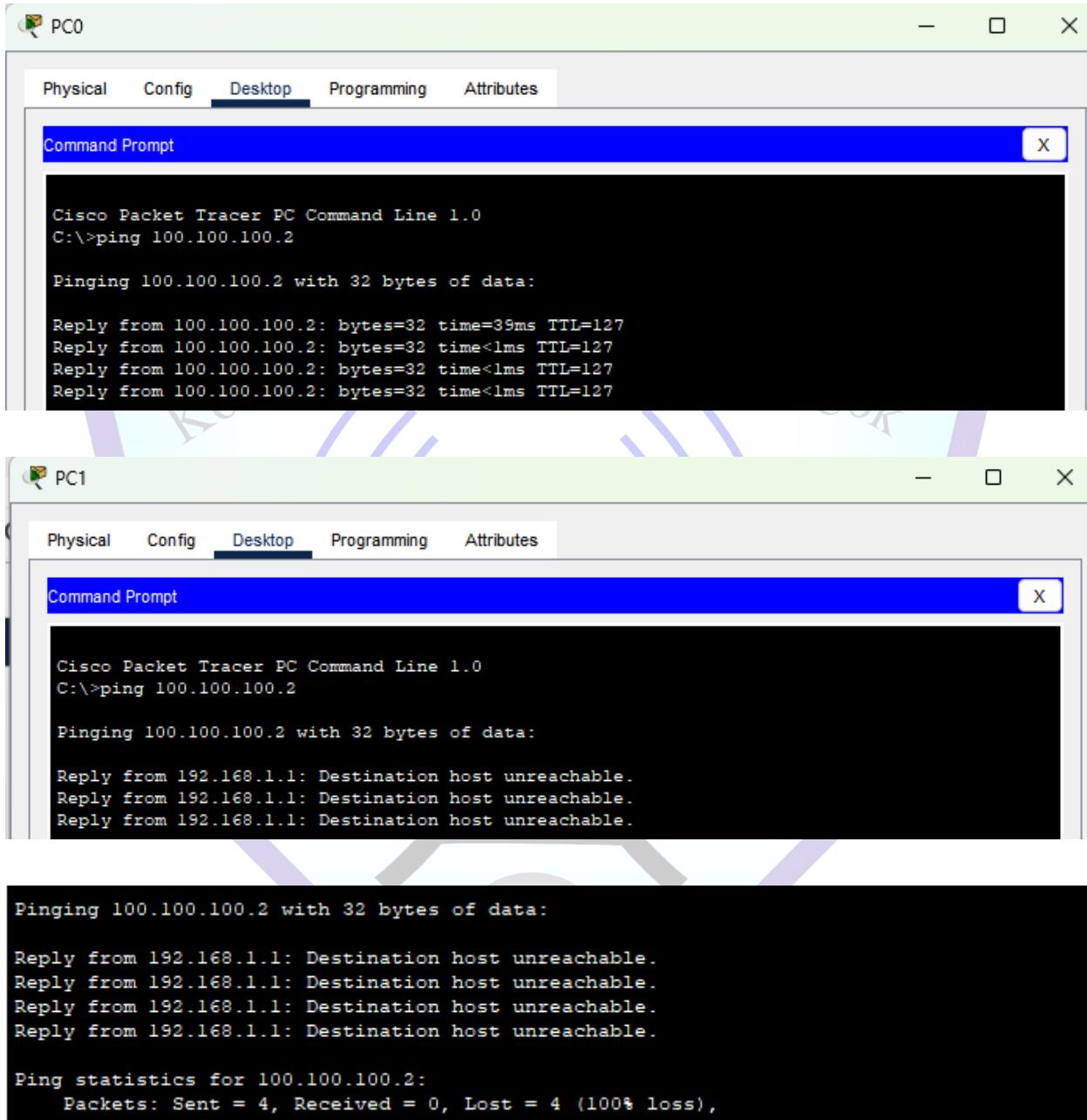
6. Disini kita akan masuk ke interface gig0/1

```
Router(config)#int gig0/1
```

7. Disini saya akan mengetikkan access-group 1, saya akan mengetikkan 1 karena kita akan membuat ACL di angka 1

```
Router(config-if)#ip access-group 1 ?
  in  inbound packets
  out outbound packets
Router(config-if)#ip acces-1
Router(config-if)#ip acces-lis
Router(config-if)#ip access
Router(config-if)#ip access-group 1 in
```

8. Untuk melakukan pengujian kalian coba ping setiap pc ke IP server, jika pada ip yang telah kalian permit tadi ttl maka kofigurasi kalain sudah berhasil



PC0

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 100.100.100.2

Pinging 100.100.100.2 with 32 bytes of data:

Reply from 100.100.100.2: bytes=32 time=39ms TTL=127
Reply from 100.100.100.2: bytes=32 time<1ms TTL=127
Reply from 100.100.100.2: bytes=32 time<1ms TTL=127
Reply from 100.100.100.2: bytes=32 time<1ms TTL=127
```

PC1

Physical Config Desktop Programming Attributes

Command Prompt

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 100.100.100.2

Pinging 100.100.100.2 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
```

```
Pinging 100.100.100.2 with 32 bytes of data:

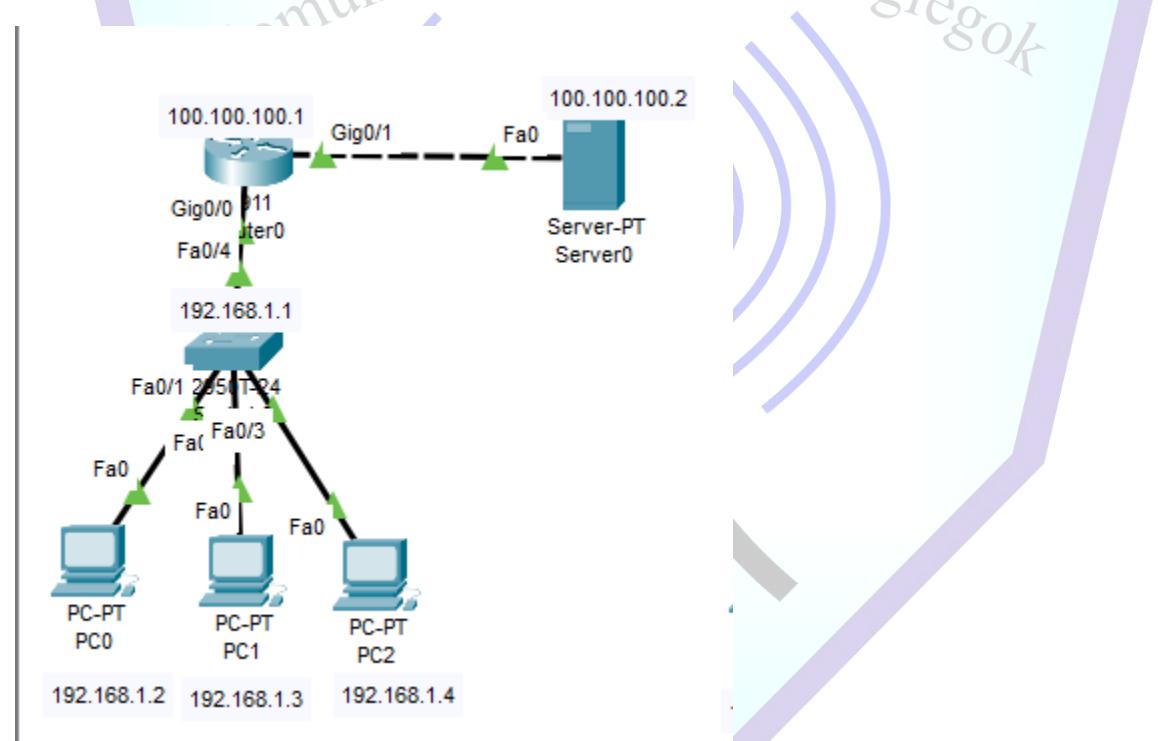
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 100.100.100.2:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

LAB 37 Standar Acces List

Pada skenario ini, saya akan membuat sebuah topologi jaringan yang terdiri dari tiga client. Dari ketiga client tersebut, hanya satu client yang akan diblokir aksesnya sehingga tidak dapat berkomunikasi dengan router, sedangkan dua client lainnya tetap dapat berkomunikasi secara normal. Untuk menerapkan skenario ini, saya akan menggunakan topologi jaringan seperti berikut.

Skenario 2



1. Pertama kita konfigurasikan terlebih dahulu sesuai dengan konfigurasi

```

outer(config-if)#int gig0/1
outer(config-if)#ip add 100.100.100.1 255.255.255.0
^
Invalid input detected at '^' marker.

outer(config-if)#ip add 100.100.100.1 255.255.255.0
outer(config-if)#no sh

outer(config-if)#int gig0/0
outer(config-if)#ip add 192.168.1.1 255.255.255.0
outer(config-if)#no sh
  
```

-
2. Disini saya akan mencoba membuat access-list 1 dimana akan menolak dengan pilihan deny lalu langsung masukkan saja ip saja dimana itu akan terhitung sebagai network address. Setelah itu kita akan permit pada any tetap di access-list 1 supaya client yang lain tidak terhitung sebagai implisit deny

```
outer(config)#access-list 1 deny 192.168.1.3  
outer(config)#access-list 1 permit any
```

3. Setelah itu kita akan mengaktifkan ACI pada interface gig0/1

```
outer(config)#int g0/0  
outer(config-if)#ip access-group 1 in  
outer(config-if)#
```

4. Kita akan melakukan pengetesan dengan melakukan ping dari pc ke server.

```
C:\>ping 100.100.100.1  
  
Pinging 100.100.100.1 with 32 bytes of data:  
  
Reply from 100.100.100.1: bytes=32 time<1ms TTL=255  
  
Ping statistics for 100.100.100.1:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 100.100.100.1  
  
Pinging 100.100.100.1 with 32 bytes of data:  
  
Reply from 192.168.1.1: Destination host unreachable.  
  
Ping statistics for 100.100.100.1:  
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

```
CISCO Packet Tracer 10 Command Line 1.0
C:\>ping 100.100.100.1

Pinging 100.100.100.1 with 32 bytes of data:

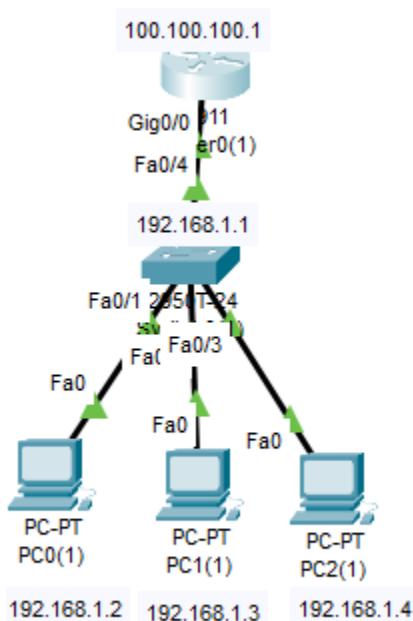
Reply from 100.100.100.1: bytes=32 time<1ms TTL=255

Ping statistics for 100.100.100.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```



LAB 38 Blokir Telnet

Dalam skenario ini, saya akan membuat sebuah topologi jaringan yang terdiri dari tiga client. Dari ketiga client tersebut, hanya satu client yang akan diblokir agar tidak dapat melakukan akses Telnet ke router, sementara dua client lainnya tetap dapat mengakses Telnet secara normal. Untuk mendukung konfigurasi ini, saya akan menggunakan topologi jaringan seperti berikut.



IT SMKN 1 Nglegok

- Langkah pertama konfigurasikan IP address sesuai dengan topologi yang telah di buat

```

Router(config)#int gig0/0
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
    
```

- Disini saya akan menolak akses yaitu deny pada Ip 192.168.1.2 dan sisanya akan saya izinkan dengan command access-list 1 permit any

```

Router(config)#access-list 1 deny 192.168.1.2
Router(config)#access-list 1 permit any
    
```

- Selanjutnya masukkan command line vty 0 4 dan jalankan juga access-class 1 in, kita akan mengatur password dengan memasukkan command password

```
Router(config)#line vty 0 4
Router(config-line)#access-class 1 in
Router(config-line)#password evan
Router(config-line)#login
Router(config-line)#exit
```

4. Jika sudah kita akan melakuka tes dengan telnet pada setiap pc, hasilnya pada ip yang kalian deny maka tidak akan bisa tersambung, sebaliknya pada ip lainnya akan dimintai password dan berhasil

```
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...
% Connection refused by remote host
```

```
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open
```

```
User Access Verification

Password:
Router>
```

```
C:\>telnet 192.168.1.1
Trying 192.168.1.1 ...Open

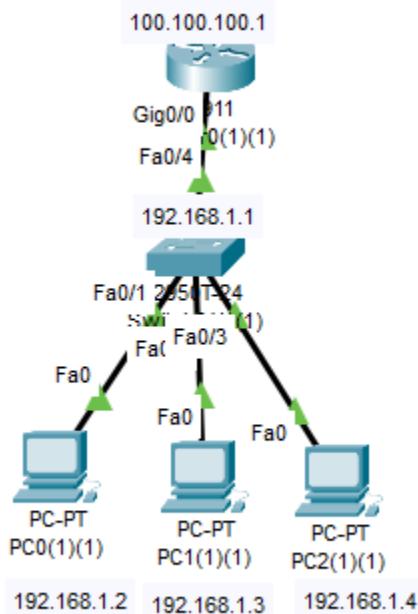
User Access Verification

Password:
Router>
```



LAB 39 Named AC

Named ACL adalah fitur yang memungkinkan kita untuk membuat Access Control List (ACL) dengan menggunakan nama sebagai pengenal, bukan hanya nomor. Hal ini sangat berguna ketika kita memiliki banyak ACL, karena memudahkan dalam membedakan dan mengelola setiap ACL berdasarkan nama yang spesifik. Pada skenario ini, saya akan menggunakan topologi seperti di bawah untuk menerapkan Named ACL.



1. Pertama tama kita akan konfigurasikan ip address terlebih dahulu sesuai dengan topologi

```
Router(config)#int g0/0
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
```

2. Saya akan menggunakan command "ip access-list standart. Setelah itu kita akan mendapat menggunakan tanda tanya untuk mengetahui apa yang bisa dilakukan

```
Router(config)#ip access-list standard evan
Router(config-std-nacl)#
<1-2147483647> Sequence Number
default      Set a command to its defaults
deny        Specify packets to reject
exit        Exit from access-list configuration mode
no          Negate a command or set its defaults
permit      Specify packets to forward
remark     Access list entry comment
```

3. Saya akan mengizinkan 1 IP dari client

```
Router(config-std-nacl)#permit ?
  A.B.C.D  Address to match
    any      Any source host
    host     A single host address
Router(config-std-nacl)#permit permit host 192.168.1.3
                           ^
% Invalid input detected at '^' marker.

Router(config-std-nacl)#permit host 192.168.1.3
Router(config-std-nacl)#exit
```

4. Aktifkan named ACL yang baru kita buat tadi

```
Router(config-if)#ip acc
% Incomplete command.
Router(config-if)#ip
Router(config-if)#ip acc
Router(config-if)#ip access-group evan in
```

5. Lakukan pengujian dengan cara ping dari masing masing client ke router.

Hasilnya akan ada 1 pc yang dapat berkomunikasi sesuai dengan named ACL

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=5ms TTL=255
Reply from 192.168.1.1: bytes=32 time=3ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
  Minimum = 0ms, Maximum = 5ms, Average = 2ms
```

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

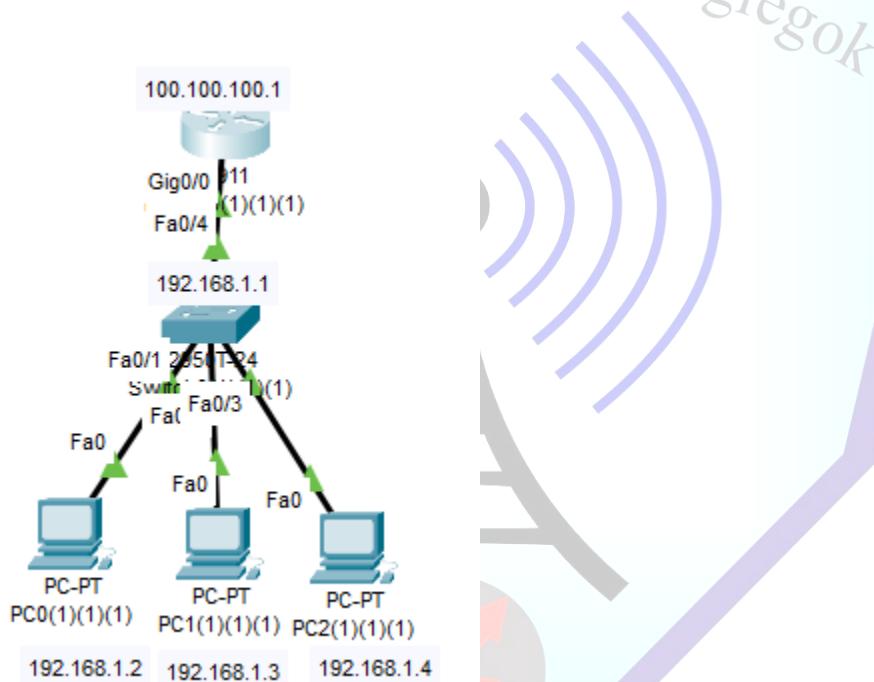
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\>
```



LAB 40 Extended ACL Blokir Ping

Extended ACL adalah jenis Access Control List yang digunakan untuk melakukan penyaringan lalu lintas jaringan secara lebih spesifik, berdasarkan protokol, alamat IP sumber dan tujuan, serta port. Dalam skenario ini, saya akan membuat Extended ACL untuk memblokir akses ping (ICMP) dari salah satu client. Konfigurasinya dilakukan dengan menggunakan perintah deny untuk protokol ICMP, diikuti oleh alamat IP client yang ingin diblokir. Setelah itu, saya menambahkan perintah permit ip any any agar lalu lintas lainnya tetap diizinkan, dan tidak terblokir oleh aturan implicit deny di akhir ACL. Topologi yang akan digunakan dalam skenario ini dapat dilihat pada gambar di bawah.



1. Seperti biasa kita akan atur ip terlebih dahulu seperti pada topologi

```
Router(config)#int g0/0
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
```

2. Disini kita akan mencoba ping ke router sebelum melakukan exented ACL blokir ping, maka semua client pasti akan bisa berkomunikasi dengan router

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=4ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=2ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 4ms, Average = 1ms
```

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=3ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 0ms
```

3. Disini kita akan membuat 100 ACL dimana seperti yang saya jelaskan pada LAB 36 dimana angka 100-199 digunakan untuk extended ACL, setelahnya saya akan memilih deny untuk memblokir salah satu dari client supaya tidak bisa ping. Setelah itu saya pilih icmp yaitu protokol dan saya aka memilih tujuan dengan IP 192.168.1.3 lalu tambahkan any dibelakangnya

```
Router(config)#access-list 100 deny ?
  ahp    Authentication Header Protocol
  eigrp   Cisco's EIGRP routing protocol
  esp     Encapsulation Security Payload
  gre     Cisco's GRE tunneling
  icmp   Internet Control Message Protocol
  ip      Any Internet Protocol
  ospf   OSPF routing protocol
  tcp     Transmission Control Protocol
  udp     User Datagram Protocol
Router(config)#access-list 100 deny icmp host 192.168.1.3 any
Router(config)#access-list 100 permit ip any any
```

4. Setelah itu aktifkan konfigurasi di port gig0/0

```
Router(config)#int g0/0
Router(config-if)#ip access-group 100 in
```

5. Saya akan melakukan pengujian ping, pada ip yang telah kita blokir tadi maka tidak akan berhasil

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=4ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time=2ms TTL=255

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms
```

```
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<lms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=3ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 3ms, Average = 0ms

C:\>
C:\>ping 192.168.1.1

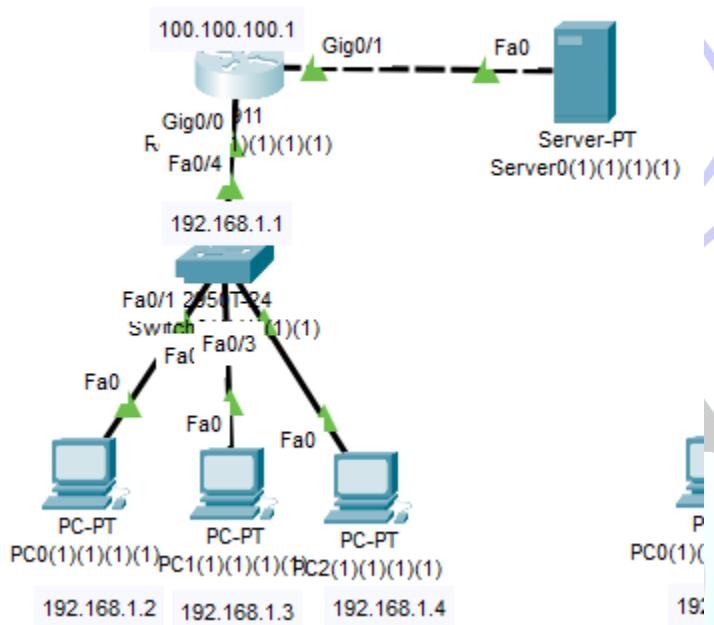
Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

LAB 41 Extended ACL Blokir Http/Https

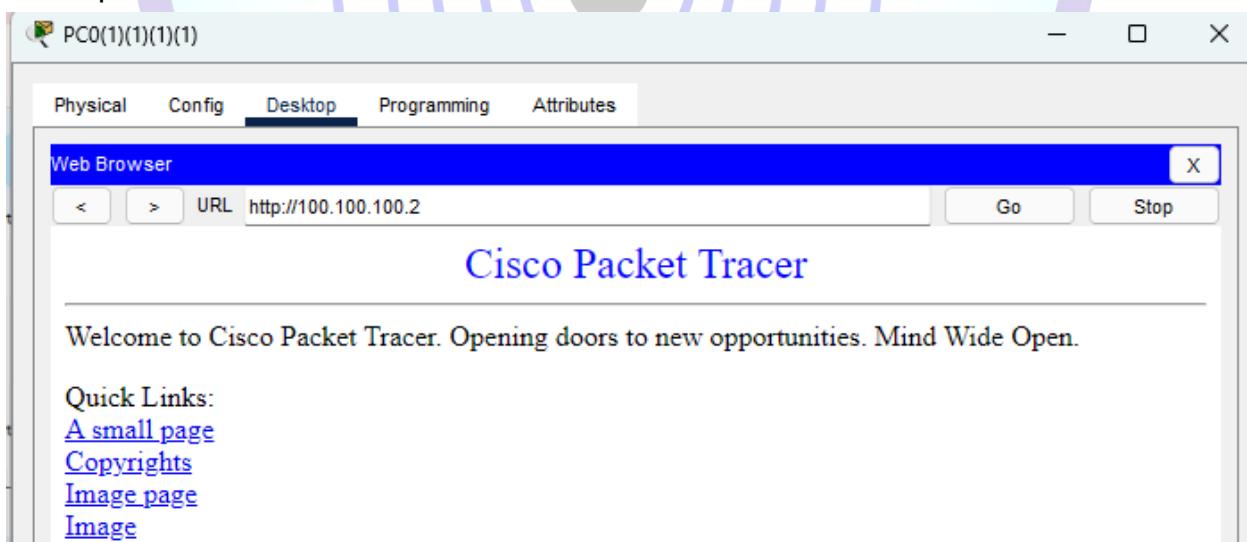
Extended ACL adalah fitur yang digunakan untuk memfilter lalu lintas jaringan secara lebih rinci, berdasarkan protokol, alamat IP sumber dan tujuan, serta nomor port. Dalam skenario ini, saya akan membuat Extended ACL untuk memblokir akses HTTP dan HTTPS dari salah satu client. Konfigurasinya dilakukan dengan menggunakan perintah deny untuk protokol TCP, dengan port 80 (HTTP) dan port 443 (HTTPS), serta mencantumkan alamat IP client yang ingin diblokir. Setelah itu, saya menambahkan perintah permit ip any any agar lalu lintas lainnya tetap diizinkan, dan tidak diblokir oleh aturan implicit deny di akhir ACL. Topologi jaringan yang digunakan dalam skenario ini dapat dilihat pada gambar di bawah.

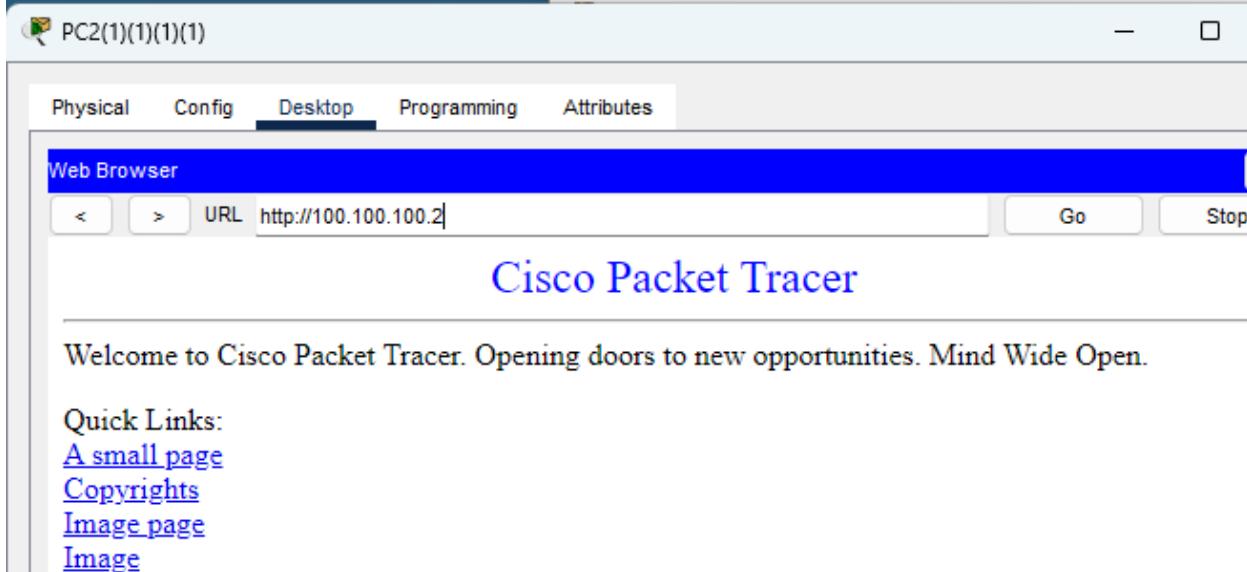
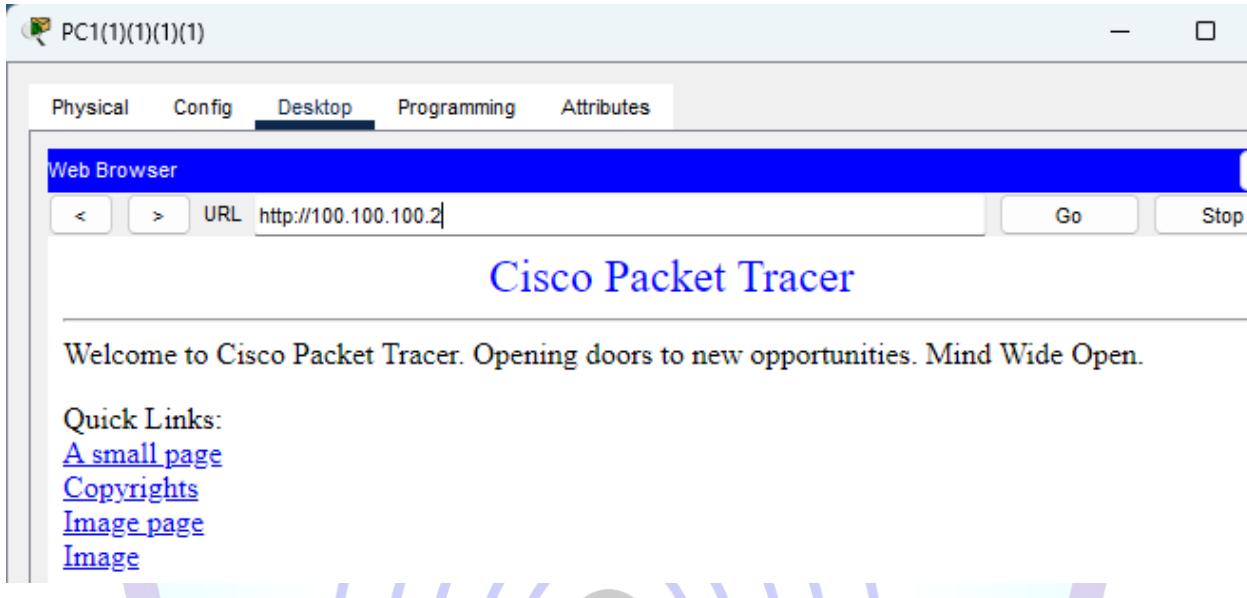


1. Pertama tama seperti biasa kita konfigurasikan terlebih dahulu IP sesuai dengan topologi yang telah dibuat

```
Router>en
Router#conf
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#int g0/1
Router(config-if)#ip add 100.100.100.1 255.255.255.0
Router(config-if)#no sh
Router(config-if)#int g0/0
Router(config-if)#ip add 192.168.1.1
% Incomplete command.
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
Router#
```

2. Sebelum kita melakukan konfigurasi kita akan mencoba akses web dari client dan pasti semua berhasil





3. Disini kita lakukan access-list pada 100 untuk extended ACL dan kita deny dengan protokol tcp dan tujuannya host dengan ip 192.168.1.2. Disini akan lakukan untuk tcp http dan https, lalu kita lakukan permit dengan protokol IP menuju ke any, supaya yang lain tidak menjadi implisit

```
Router(config)#access-list 100 deny tcp host 192.168.1.2 any eq 80
Router(config)#access-list 100 deny tcp host 192.168.1.2 any eq 443
Router(config)#
```

4. Aktifkan Exented aclnya di port gig0/0

```
Router(config)#int g0/0
Router(config-if)#ip access-group 100 in
Router(config-if)#ex
```

5. Kita coba akses browser dari semua client maka client dengan IP 192.168.1.2 tidak akan bisa mengakses webnya



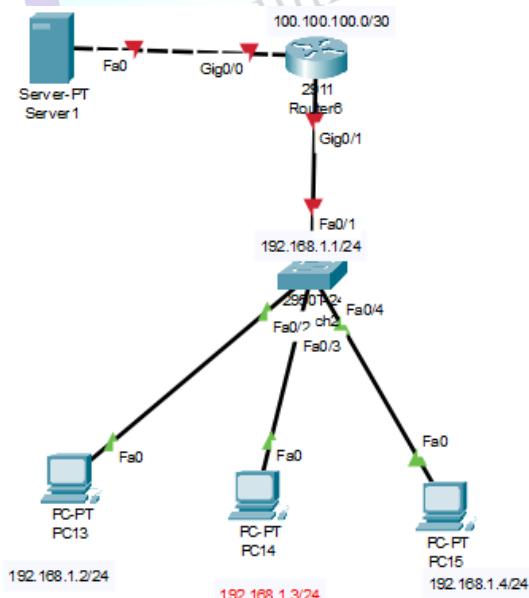


LAB 42 Nat Static Cisco

Pada lab ini kita akan menjelaskan mengenai NAT static pada cisco. NAT sendiri memiliki tiga jenis yaitu:

1. NAT static
2. NAT dynamic
3. NAT PAT

Pada lab ini kita hanya berfokus kepada NAT static, NAT ini termasuk ke pada jenis one to one NAT, dimana ini hanya bisa digunakan untuk mentranslasikan satu IP private menjadi IP public. Untuk pemahaman lebih lanjut mari kita coba untuk melakukan konfigurasi dengan menggunakan layout seperti pada contoh di bawah ini



Sebelum memulai konfigurasi kita pastikan terlebih dahulu bahwa semua ip sudah kita konfigurasikan, dan jangan lupa pada internet tidak perlu ditambahkan gateway

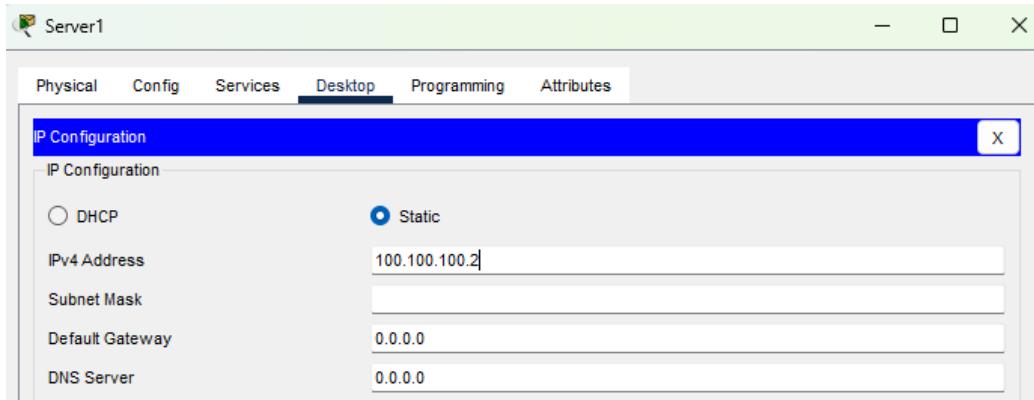
```

Router(config)#int g0/0
Router(config-if)#ip ad 100.100.100.1 255.255.255.252
Router(config-if)#no sh

Router(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip ad 192.168.1.1 255.255.255.0
Router(config-if)#no sh
    
```



Jika semua IP telah kita konfigurasikan seperti pada layout, pastikan semua pc tidak dapat terkoneksi dengan internet

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit
	Failed	PC13	Server1	ICMP		0.000	N	0	(edit)
	Failed	PC14	Server1	ICMP		0.000	N	1	(edit)
	Failed	PC15	Server1	ICMP		0.000	N	2	(edit)

Jika semua pc sudah dipastikan tidak bisa ping ke internet kita lanjutkan dengan membuat NAT, untuk konfigursi NAT kita bisa lihat pada contoh gambar di bawah ini

```
Router(config)#ip nat inside source static ?  
A.B.C.D Inside local IP address  
tcp Transmission Control Protocol  
udp User Datagram Protocol  
Router(config)#ip nat inside source static 192.168.1.2 ?  
A.B.C.D Inside global IP address  
Router(config)#ip nat inside source static 192.168.1.2 100.100.100.1  
Router(config)#ip nat inside source static 192.168.1.3 100.100.100.1  
Router(config)#ip nat inside source static 192.168.1.4 100.100.100.1
```

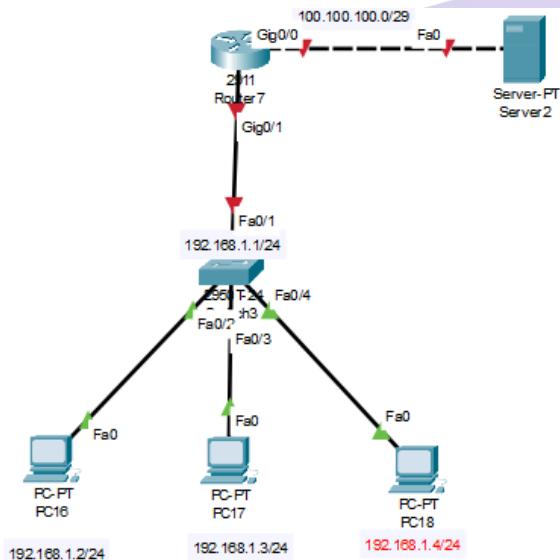
Pada gambar di atas terlihat bahwa setelah kita menambahkan ip nat inside source static kita harus menambahkan ip yang private yang ingin kita translasi. Setelah ip private kita juga harus menambahkan ip public yang ingin kita arahkkan untuk translasi. Jika konfigurasi pada NAT sudah selesai kita lanjutkan dengan menambahkan nat yang telah kita buat ke interface router

```
Router(config)#int g0/0
Router(config-if)#ip nat outside
Router(config-if)#int g0/1
Router(config-if)#ip nat inside
```

Jika semua konfigurasi telah dilakukan kalian uji coba dengan cara ping ke setiap pc ke internet

LAB 43 NAT Dynamic Cisco

Pada lab ini kita akan membahas mengenai konfigurasi NAT dynamic pada cisco. NAT dynamic termasuk pada tipe many to many NAT dimana NAT ini dapat kita gunakan untuk mentranslasikan IP Private dengan jumlah yang banyak menjadi lebih dari satu IP public. Supaya kalian lebih paham kita akan mencobanya



Sebelumnya kita pastikan terlebih dahulu bahwa semua IP address sudah dikonfigurasikan seperti pada layout diatas

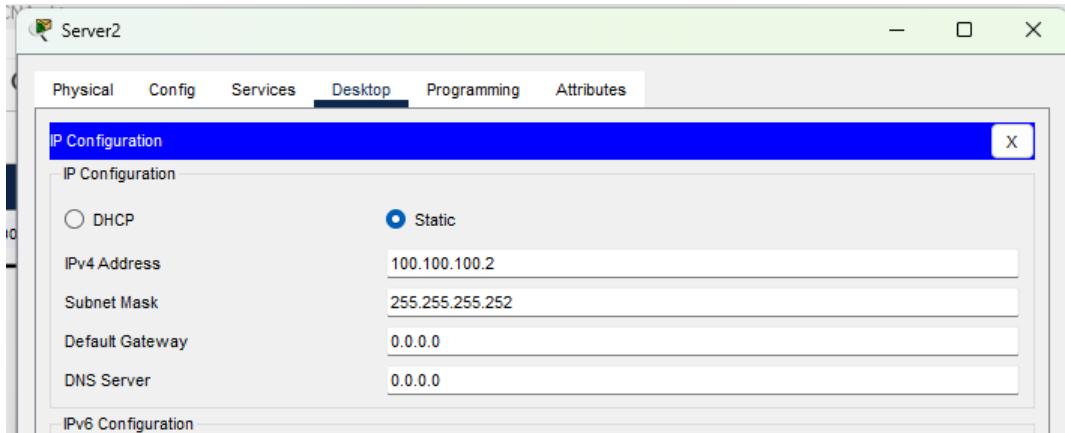
```

Router(config)#int g0/0
Router(config-if)#ip ad 100.100.100.1 255.255.255.252
Router(config-if)#no sh

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip add 192.168.1.1 255.255.255.0
Router(config-if)#no sh
    
```



Jika sudah kita lanjutkan dengan konfigurasinya. Kita perlu konfigurasi Access list terlebih dahulu dan juga kita perlu membuat pool NAT untuk IP Public. Konfigurasinya kalian bisa lihat pada contoh di bawah ini

```
Router(config)#access-list 1 permit any
Router(config)#ip nat pool pool1 100.100.100.1 100.100.100.6 netmask 255.255..255.248
% Invalid input detected at '^' marker.

Router(config)#ip nat pool pool1 100.100.100.1 100.100.100.6 netmask 255.255.255.248
```

Pada gambar di atas saat kita membuat pool untuk IP public yang akan kita gunakan kita bisa menambahkan lebih dari satu IP public. Jika access list dan pool NAT sudah dibuat kita lanjutkan dengan membuat NAT

```
Router(config)#ip nat inside source list 1 ?
  interface  Specify interface for global address
  pool      Name pool of global addresses
Router(config)#ip nat inside source list 1 pool ?
  WORD     Name pool of global addresses
Router(config)#ip nat inside source list 1 pool pool1
```

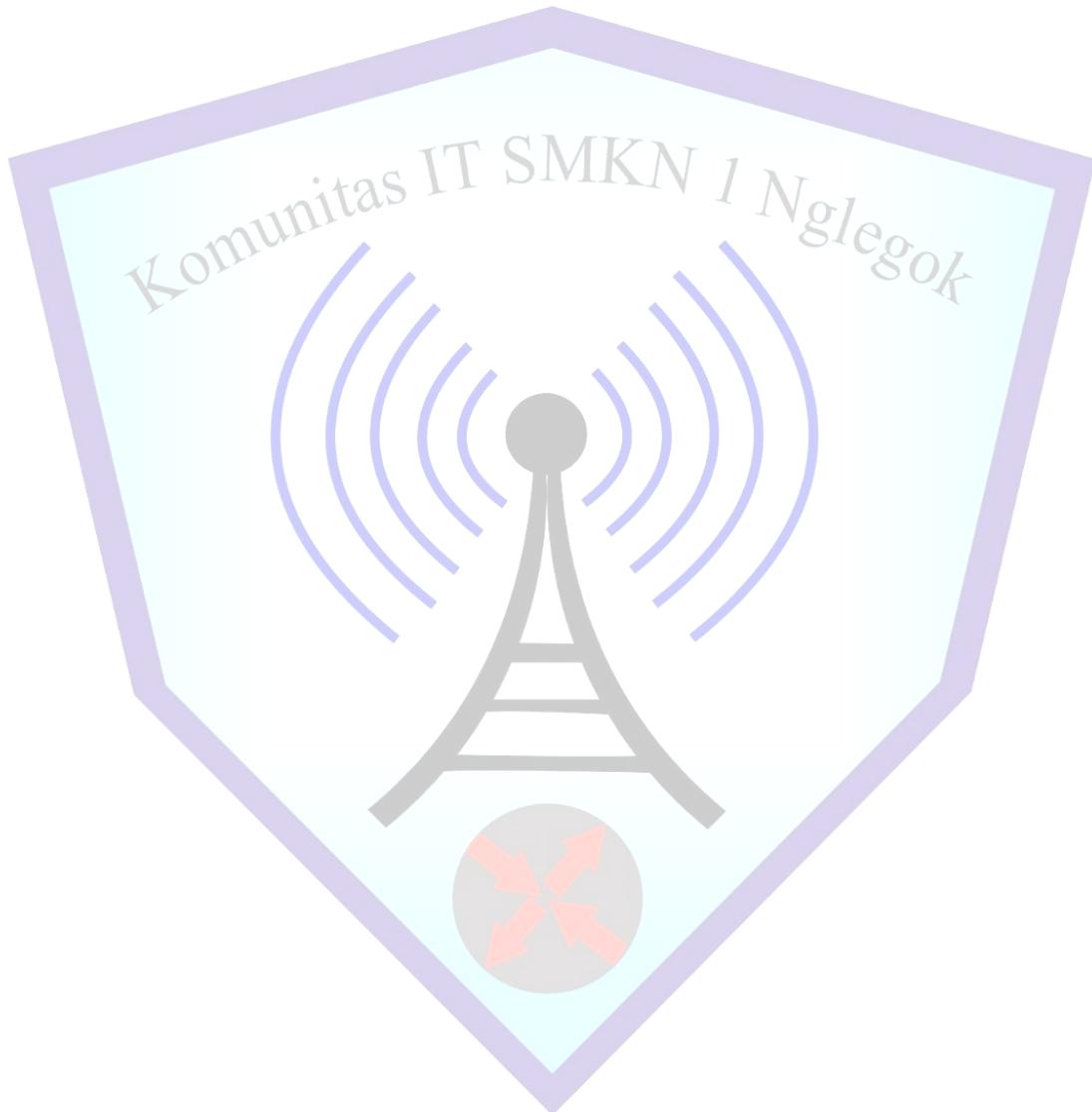
Pada gambar diatas terlihat bahwa setelah perintah ip nat inside source list 1 terdapat dua konfigurasi lanjutan yaitu interface dan pool, pada lab ini kita akan ppilih pool. Jika nat sudah ada kita selanjutnya akan memasukkan nat tersebut ke interface router

```
Router(config)#int g0/0
Router(config-if)#ip nat outside
Router(config-if)#int g0/1
Router(config-if)#ip nat inside
Router(config-if)#

```

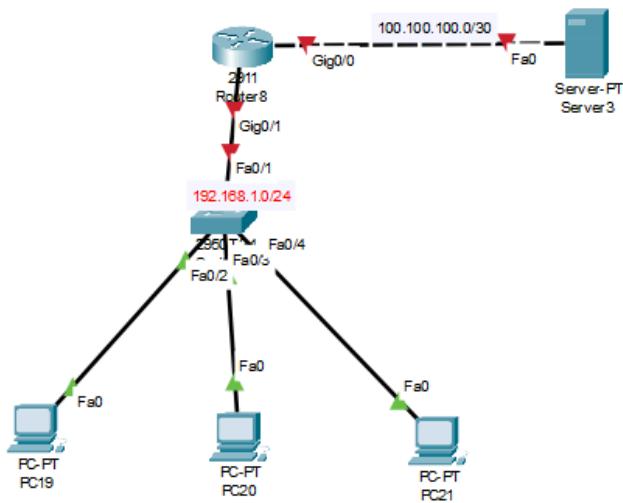
Sebagai pembuktian apakah konfigurasi yang telah kita buat berhasil atau tidak kita lakukan dengan ping

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit
	Successful	PC16	Server2	ICMP	Dark Green	0.000	N	0	(edit)
	Failed	PC17	Server2	ICMP	Red	0.000	N	1	(edit)
	Successful	PC18	Server2	ICMP	Magenta	0.000	N	2	(edit)



LAB 44 Nat Dynamic Cisco With Exit Interface

Pada lab ini kita akan membahas mengenai NAT, Network Address Translating digunakan untuk menerjemahkan alamat IP address pada jaringan local menjadi alamat IP address jaringan public ataupun sebaliknya. Pada lab ini kita akan menggunakan dynamic nat dengan menggunakan exit interface. Langsung saja kita akan mencoba konfigurasinya



Sebelum kita memulai konfigurasinya pastikan bahwa semua IP sudah dikonfigurasikan seperti pada gambar di atas. Pada layout diatas diasumsikan bahwa server tersebut adalah server yang ada di internet untuk membuat server seperti pada server yang ada di internet kita hanya perlu memberikan ip address pada server tanpa menambahkan gateway

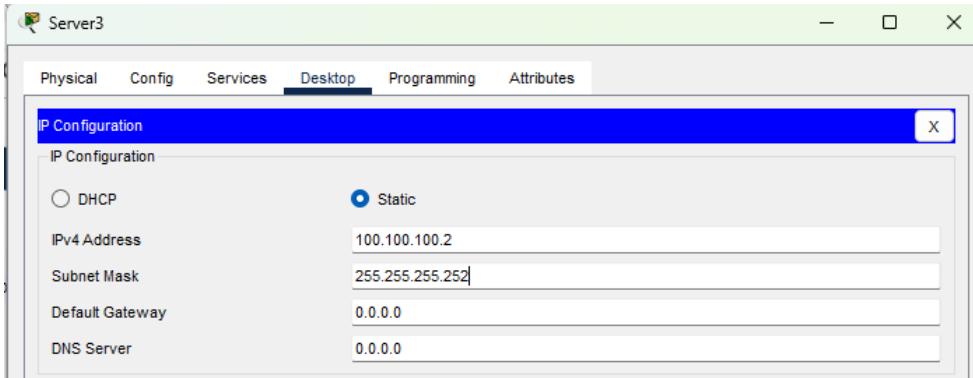
```

Router(config)#int g0/0
Router(config-if)#ip add 100.100.100.1 255.255.255.252
Router(config-if)#no sh

Router(config-if)#
*LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

*LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#int g0/1
Router(config-if)#ip ad 192.168.1.1 255.255.255.0
Router(config-if)#no sh
  
```



Selanjutnya kita lanjutkan dengan membuat NAT pada router. Untuk contoh konfigurasinya kalian bisa lihat seperti pada gambar di bawah ini

```

Router(config)#access-list 1 permit any
Router(config)#in nat inside source ?
  list   Specify access list describing local addresses
  static Specify static local->global mapping
Router(config)#ip nat inside source list ?
  <1-199>  Access list number for local addresses
  WORD      Access list name for local addresses
Router(config)#ip nat inside source list 1 ?
  interface  Specify interface for global address
  pool       Name pool of global addresses
Router(config)#ip nat inside source list 1 interface g0/0
  
```

Sebelum kita menambahkan NAT, langkah pertama yang perlu dilakukan adalah membuat Access List. Access List ini berfungsi untuk menentukan alamat IP mana yang akan dikenakan NAT. Setelah membuat Access List, baru kita bisa melanjutkan untuk mengonfigurasi NAT. Jika kita menggunakan Access List dengan nomor, misalnya nomor 1, kita akan menuliskan perintah untuk mengizinkan alamat IP tertentu agar dapat dikenakan NAT. Jika menggunakan Access List yang bernama, kita cukup menggunakan nama yang sudah kita buat. Setelah Access List selesai dibuat, kita bisa lanjut dengan konfigurasi NAT dinamis. Pada bagian ini, kita akan menambahkan perintah untuk NAT menggunakan Access List yang telah dibuat, serta menentukan interface yang terhubung ke internet. Jangan lupa juga untuk menetapkan interface yang terhubung ke jaringan lokal (inside) dan yang terhubung ke internet (outside). Jika sudah, kita bisa memverifikasi bahwa NAT berfungsi dengan baik menggunakan perintah tertentu. Dengan langkah-langkah tersebut, kita akan berhasil mengonfigurasi NAT dinamis dan memungkinkan alamat IP lokal bisa mengakses internet menggunakan satu alamat IP publik.

Setelah selesai membuat NAT kita diharuskan untuk memasukkan NAT yang sudah kita buat pada interface router. Untuk contohnya kalian bisa lihat seperti pada gambar di bawah ini

```
Router(config)#int g0/0
Router(config-if)#ip nat outside
Router(config-if)#int g0/1
Router(config-if)#int g0/1
Router(config-if)#ip nat inside
```

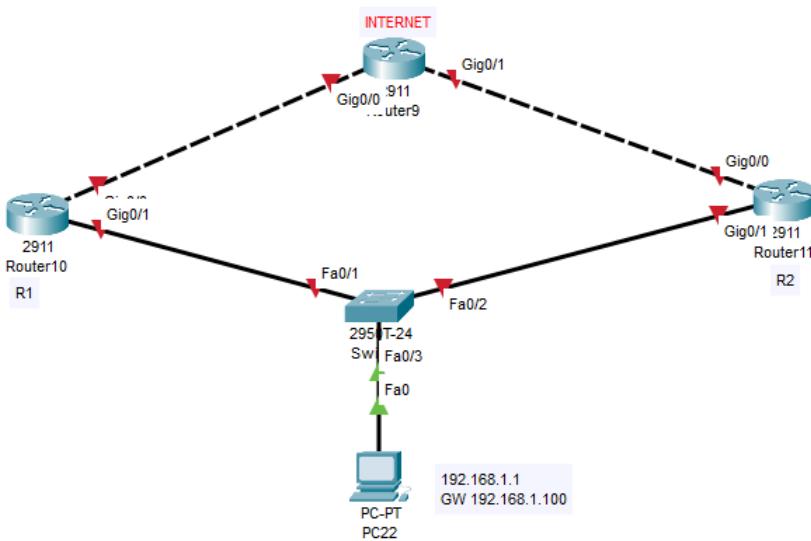
Setelah kalian selesai membuat NAT mari kita uji cobakan ping dari beberapa PC ke server. Jika konfigurasi NAT anda berhasil maka hasilnya pasti semua pc akan dapat ping menuju server internet

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit
	Successful	PC21	Server3	ICMP		0.000	N	0	(edit)
	Successful	PC20	Server3	ICMP		0.000	N	1	(edit)



LAB 45 Konfigurasi HSRP

HSRP adalah salah satu dari protocol FHRP. HSRP juga merupakan Cisco proprietary atau protocol bawaan cisco yang hanya bisa berjalan apabila router pada network juga menggunakan cisco. Pada HSRP hello packet dikirim menggunakan protocol UDP dengan port 1985 dengan multicast ip 224.0.0.2 untuk versi 1 untuk versi 2 menggunakan ip 224.0.0.102. Untuk memahami lebih lanjut mari kita coba untuk konfigurasi HSRP, untuk itu saya akan menggunakan topologi sederhana sederhana seperti pada contoh di bawah ini



Sebelum masuk ke HSRP mari kita asumsikan bahwa kita akan membuat supaya client nanti bisa akses ke internet menggunakan jalur via 192.168.1.1 sebagai jalur utama yang berarti R1 akan menjadi Active Router lalu untuk jalur cadangan akan menggunakan jalur via 192.168.1.2 yang berarti R2 R2 akan menjadi Standby Router. Setelah itu pastikan bahwa semua IP sudah dikonfigurasikan seperti pada gambar diatas, dan untuk ip loopback pastikan anda konfigurasikan pada router yang berperan sebagai internet.

Setelah IP sudah dikonfigurasikan pada semua router, lanjutkan dengan mengkonfigurasi protokol routing seperti EIGRP (atau protokol routing lain yang diinginkan).

- Untuk konfigurasi EIGRP pada lab kali ini, saya tidak akan membahas detailnya.
- Jika Anda ingin referensi cara konfigurasi EIGRP, silakan lihat pada Lab 31.
- Jangan lupa, untuk router yang terhubung ke Internet, masukkan juga network loopback ke dalam konfigurasi EIGRP agar loopback dapat ter-routing dengan benar.

Setelah konfigurasi EIGRP selesai, kita akan melanjutkan ke konfigurasi HSRP (Hot Standby Router Protocol) pada router R1 dan R2.

Contoh konfigurasi HSRP dapat dilihat pada gambar di bawah ini

```
|-----+-----+
R1(config-if)#standby 1 ip 192.168.1.100
% Warning: address is not within a subnet on this interface
R1(config-if)#standby preempt
|-----+-----|
R2(config)#int g0/1
R2(config-if)#standby ip 192.168.1.100
% Warning: address is not within a subnet on this interface
R2(config-if)#standby 1 ip 192.168.1.100
% Address 192.168.1.100 in group 0
R2(config-if)#standby 1 ip 192.168.1.100
% Address 192.168.1.100 in group 0
R2(config-if)#standby 1 preempt
R2(config-if)#standby 1 priority 10
```

Jika sudah selesai mengkonfigurasikan HSRP lanjutkan dengan mengkonfigurasikan HSRP lanjutkan dengan konfigurasi passive interface dengan menggunakan perintah seperti pada contoh di bawah ini

```
|-----+-----+
R1(config)#router eigrp 1
R1(config-router)#passive-interface g0/1
R1(config-router)#exit
|-----+-----|
R2(config)#router eigrp 1
R2(config-router)#passive-interface g0/1
R2(config-router)#exit
```

Setelah selesai mengkonfigurasikan passive interface mari kita coba lihat HSRP yang sudah kita buat, dengan menggunakan perintah seperti pada gambar di bawah ini

```
R1#show standby br
      P indicates configured to preempt.
      |
Interface  Grp  Pri P State     Active          Standby        Vir
Gig0/1      1    100  Init       unknown        unknown        192
R1#show standby fastethernet 0/1
^
* Invalid input detected at '^' marker.

R1#show standby fastethernet 0/1
^
* Invalid input detected at '^' marker.

R1#
R1#R1#show standby interface FastEthernet0/1
^
* Invalid input detected at '^' marker.

R1#show standby fastethernet g0/1
^
* Invalid input detected at '^' marker.

R1#show standby g0/1
GigabitEthernet0/1 - Group 1
  State is Init (interface down)
  Virtual IP address is 192.168.1.100
  Active virtual MAC address is unknown
    Local virtual MAC address is 0000.0C07.AC01 (vl default)
  Hello time 3 sec, hold time 10 sec
    Next hello sent in 1.180 secs
  Preemption disabled
  Active router is unknown
  Standby router is unknown
  Priority 100 (default 100)
  Group name is hsrp-Gig0/1-1 (default)

R1#
```

```
R2#show standby g0/1
GigabitEthernet0/1 - Group 0
  State is Active
    6 state changes, last state change 00:23:00
    Virtual IP address is 192.168.1.100
    Active virtual MAC address is 0000.0C07.AC00
      Local virtual MAC address is 0000.0C07.AC00 (vl default)
    Hello time 3 sec, hold time 10 sec
      Next hello sent in 0.128 secs
    Preemption disabled
    Active router is local
    Standby router is unknown
    Priority 100 (default 100)
  Group name is hsrp-Gig0/1-0 (default)
```

Keterangan :

1. show standby brief perintah ini digunakan untuk memperlihatkan interface HSRP
2. show standby fastethernet 0/1 perintah ini digunakan untuk melihat status HSRP yang ada pada interface 0/1.
3. State is ini menunjukkan status dari interface HSRP apakah statusnya Active ataupun Standby
4. 3 state changes ini menunjukkan bahwa pada interface HSRP ini telah terjadi perubahan konfigurasi sebanyak 3 kali
5. last state change {lama waktu} ini menunjukkan kapan terakhir kali konfigurasi pada interface HSRP dirubah
6. Virtual IP address is {ip address} ini menunjukkan IP virtual pada interface HSRP
7. Preemption {enable/disable} ini menunjukkan status preemption
8. Active roter is {ip address} ini menunjukkan ip pada interface HSRP
9. Priority 10 ini menunjukkan prioritas pada interface HSRP
10. Group name {nama} ini menunjukkan nama group yang ada pada interface HSRP

LAB 46 Konfigurasi VRRP Cisco

Virtual Router Redundancy Protocol (VRRP) adalah sebuah protokol yang digunakan untuk menciptakan redundansi pada gateway default dalam sebuah jaringan IP. Tujuan utama VRRP adalah untuk menghindari terputusnya koneksi jaringan jika router utama mengalami gangguan atau kegagalan. Dengan VRRP, beberapa router dapat bekerja sama dalam sebuah grup, di mana salah satu router akan berperan sebagai router utama (Master), sementara router lainnya menjadi cadangan (Backup). Apabila router utama tidak lagi dapat berfungsi, maka secara otomatis salah satu router cadangan akan mengambil alih peran sebagai gateway default, tanpa mengganggu koneksi jaringan.

Cara kerja VRRP melibatkan penggunaan sebuah alamat IP virtual yang dibagikan di antara semua router dalam grup VRRP tersebut. Alamat IP virtual ini digunakan oleh perangkat klien sebagai gateway default mereka. Router yang aktif sebagai Master akan menangani seluruh lalu lintas jaringan yang ditujukan ke IP virtual tersebut. Untuk menentukan siapa yang menjadi Master, digunakan sistem prioritas. Router dengan nilai prioritas tertinggi akan menjadi Master. Jika dua router memiliki prioritas yang sama, maka router dengan alamat IP tertinggi akan dipilih. Selain itu, terdapat fitur yang disebut preempt, yang memungkinkan router dengan prioritas lebih tinggi untuk merebut kembali posisi Master jika sebelumnya sempat turun dan kemudian aktif kembali.

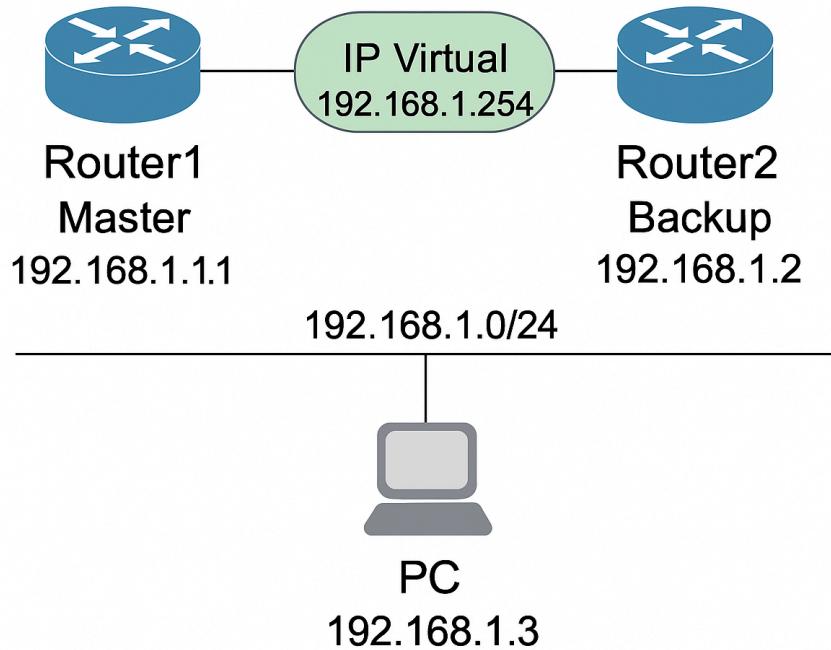
Konfigurasi VRRP pada perangkat Cisco dapat dilakukan pada antarmuka jaringan yang terhubung ke jaringan internal. Sebagai contoh, jika terdapat dua router yaitu Router1 dan Router2, maka masing-masing dikonfigurasikan dengan alamat IP fisik yang berbeda, namun berbagi satu IP virtual yang akan digunakan sebagai gateway oleh klien. Pada Router1, yang akan dijadikan Master, diberikan nilai prioritas lebih tinggi, misalnya 120. Sedangkan pada Router2, yang akan menjadi Backup, diberi nilai prioritas standar seperti 100. Konfigurasi dilakukan dengan perintah-perintah seperti `vrrp [group-number] ip [virtual-ip-address]`, diikuti dengan pengaturan prioritas dan aktivasi fitur preempt. Dengan konfigurasi ini, Router1 akan berfungsi sebagai Master dan menangani lalu lintas, sedangkan Router2 akan siaga. Jika Router1 gagal, Router2 secara otomatis akan mengantikannya tanpa memerlukan perubahan konfigurasi di sisi klien.

Setelah konfigurasi selesai, pengujian dapat dilakukan dengan mematikan interface atau mematikan Router1 secara langsung, dan mengamati apakah Router2 secara otomatis mengambil alih fungsi gateway. Jika konfigurasi berhasil, perpindahan ini akan terjadi secara otomatis dan lalu lintas jaringan tidak akan terputus. VRRP juga dapat dimonitor menggunakan perintah `show vrrp` pada perangkat Cisco untuk melihat status saat ini dari tiap router dalam grup VRRP. Karena VRRP merupakan protokol standar terbuka, ia dapat digunakan secara lintas vendor, tidak terbatas hanya pada perangkat Cisco saja.

Secara keseluruhan, penggunaan VRRP dalam jaringan memberikan solusi yang efektif untuk memastikan ketersediaan gateway secara terus-menerus. Ini sangat penting terutama pada lingkungan jaringan yang kritis seperti perusahaan, kampus, dan pusat data. Dengan

mengimplementasikan VRRP, administrator jaringan dapat meningkatkan keandalan jaringan dan menjamin layanan tetap berjalan meskipun terjadi kegagalan perangkat.

Agar lebih memahami kita langsung saja praktik, disini saya akan praktik di gns3 dan menggunakan topologi seperti di bawah ini



Setelah kalian membuat topologinya, lanjutkan dengan membuat konfigurasinya, kalian bisa contoh seperti pada di bawah ini untuk penjelasannya sebagai berikut

- Mengaktifkan mode konfigurasi.
- Mengatur IP address 192.168.1.1/24 pada interface GigabitEthernet0/0 di RouterA.
- no shutdown mengaktifkan interface (defaultnya mati).

-
- write memory menyimpan konfigurasi agar tidak hilang setelah reboot.

```
RouterA> enable
RouterA# configure terminal
RouterA(config)# interface GigabitEthernet0/0
RouterA(config-if)# ip address 192.168.1.1 255.255.255.0
RouterA(config-if)# no shutdown
RouterA(config-if)# exit
RouterA(config)# end
RouterA# write memory
```

Penjelasan Konfigurasi. Sama seperti RouterA, hanya berbeda IP address: RouterB dikonfigurasi dengan 192.168.1.2/24. Ini memungkinkan RouterA dan RouterB berada dalam satu jaringan (192.168.1.0/24).

```
RouterB> enable
RouterB# configure terminal
RouterB(config)# interface GigabitEthernet0/0
RouterB(config-if)# ip address 192.168.1.2 255.255.255.0
RouterB(config-if)# no shutdown
RouterB(config-if)# exit
RouterB(config)# end
RouterB# write memory
```

- Mengaktifkan VRRP di interface GigabitEthernet0/0.
- vrrp 1 ip 192.168.1.254 → Mengatur Virtual IP Address (VIP) yang digunakan oleh VRRP (yang akan digunakan sebagai default gateway oleh host di jaringan).
- priority 110 → RouterA diberikan prioritas 110 (default VRRP priority adalah 100, artinya RouterA akan jadi master jika lebih tinggi dari RouterB).
- preempt → Memungkinkan RouterA untuk mengambil alih peran master VRRP jika sebelumnya turun dan kembali online.
- write memory → Menyimpan konfigurasi.

```
RouterA(config)# interface GigabitEthernet0/0
RouterA(config-if)# vrrp 1 ip 192.168.1.254
RouterA(config-if)# vrrp 1 priority 110
RouterA(config-if)# vrrp 1 preempt
RouterA(config-if)# end
RouterA# write memory
```

konfigurasi dasar pada RouterA, di mana interface GigabitEthernet0/0 diaktifkan dan diberikan alamat IP 192.168.1.1 dengan subnet mask 255.255.255.0. Perintah no shutdown digunakan untuk mengaktifkan interface tersebut, lalu konfigurasi disimpan dengan write memory. Konfigurasi ini bertujuan agar RouterA siap digunakan dalam jaringan dan nantinya bisa dikonfigurasi untuk mendukung VRRP.

```
RouterB(config)# interface GigabitEthernet0/0
RouterB(config-if)# vrrp 1 ip 192.168.1.254
RouterB(config-if)# vrrp 1 priority 100
RouterB(config-if)# vrrp 1 preempt
RouterB(config-if)# end
RouterB# write memory
```

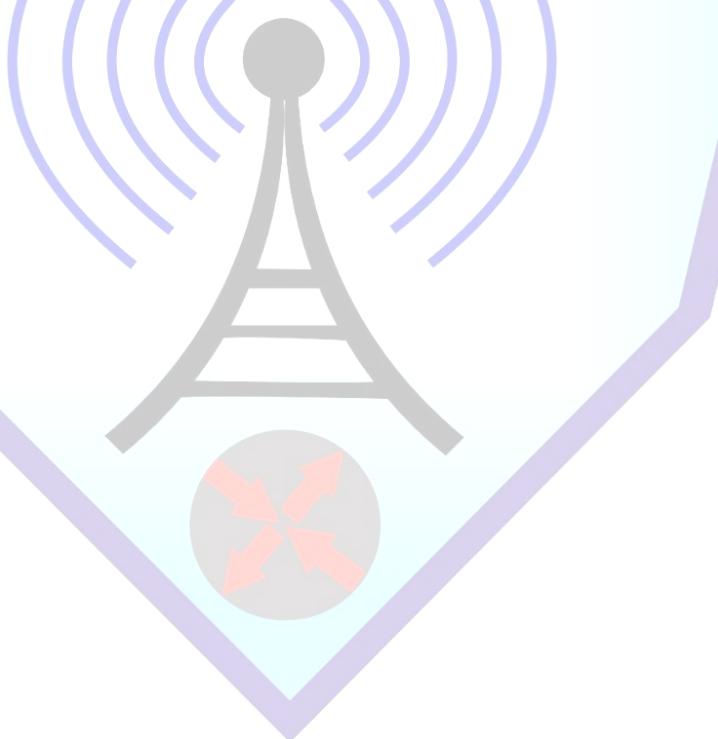
menampilkan konfigurasi serupa namun dilakukan pada RouterB. Interface GigabitEthernet0/0 diaktifkan dan diberikan alamat IP 192.168.1.2 dengan subnet mask yang sama, yaitu 255.255.255.0. Sama seperti RouterA, interface diaktifkan dengan perintah `no shutdown`, dan konfigurasi disimpan. Perbedaan utama dengan gambar pertama adalah IP address-nya, yang menandakan bahwa RouterB akan menjadi router kedua dalam jaringan yang sama dan siap untuk konfigurasi VRRP sebagai cadangan RouterA.

- **IP Address:** 192.168.1.100
- **Subnet Mask:** 255.255.255.0
- **Gateway Default:** 192.168.1.254 (Virtual IP VRRP)

LAB 47 Konfigurasi GLBP Cisco

Gateway Load Balancing Protocol (GLBP) memungkinkan beberapa router berbagi beban sebagai gateway default dengan menggunakan satu IP virtual yang sama. Hal ini memungkinkan redundansi dan load balancing, di mana lalu lintas jaringan dibagi secara merata antara router-router yang tergabung dalam grup GLBP. Router dengan prioritas lebih tinggi akan bertindak sebagai Active Virtual Gateway (AVG), yang mengelola pembagian trafik ke router lainnya. Jika router AVG gagal, router lain dalam grup dapat mengambil alih peran tersebut secara otomatis. Konfigurasi preempt memungkinkan router dengan prioritas lebih tinggi untuk kembali menjadi AVG jika sudah aktif kembali.

Dengan GLBP, keandalan dan efisiensi penggunaan router meningkat karena semua router dapat aktif, bukan hanya satu yang berperan sebagai gateway.



Komunitas IT SMKN 1 Nglelok

LAB 48 WAN HDLC

Pada lab ini kita akan membahas tentang WAN HDLC. WAN teknologi yang difunakan untuk mengubungkan router yang jauh, dan tidak bisa dijangkau oleh kabel Fast Ethernet. WAN menggunakan kabel serial dan sifatnya adalah point to point.

Sedangkan HDLC adalah protokol WAN yang bekerja pada layer 2. Pada cisco HDLC merupakan cisco prorprietary, yang berarti HDLC pada router cisco hanya dapat bekerja pada perangkat cisco aja. Agar lebih paham langsung saja kita praktikan menggunakan topologi sederhana seperti pada gambar di bawah ini



Jika sudah membuat topologi diatas mari kita lihat apakah protokol HDLC aktif secara default pada interface serial, untuk itu kita dapat mencobanya dengan melihat status pada interface serial

```
R1#show int se0/0/0
Serial0/0/0 is administratively down, line protocol is down (disabled)
  Hardware is HD64570
    MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
      reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation HDLC, loopback not set, keepalive set (10 sec)
    Last input never, output never, output hang never
    Last clearing of "show interface" counters never
    Input queue: 0/75/0 (size/max/drops); Total output drops: 0
    Queueing strategy: weighted fair
    Output queue: 0/1000/64/0 (size/max total/threshold/drops)
      Conversations 0/0/256 (active/max active/max total)
      Reserved Conversations 0/0 (allocated/max allocated)
      Available Bandwidth 1158 kilobits/sec
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
      0 packets input, 0 bytes, 0 no buffer
      Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
      0 packets output, 0 bytes, 0 underruns
      0 output errors, 0 collisions, 1 interface resets
      0 output buffer failures, 0 output buffers swapped out
      0 carrier transitions
    DCD=down DSR=down DTR=down RTS=down CTS=down
```

Pada kabel serial ini sebenarnya terdapat dua jenis yaitu serial DTE dan serial DCE, serial DCE ini digunakan oleh ISP, sebagai penentu clock rate. Clock rate harus diatur supaya dapat berjalan, untuk melihatnya kalian bisa lihat gambar di bawah ini

```
R1#show controllers se0/0/0
Interface Serial0/0/0
Hardware is PowerQUICC MPC860
DCE V.35, CLOCK rate 20000000
idb at 0x81081AC4, driver data structure at 0x81084AC0
SCC Registers:
```

```
R2#show controllers se0/0/0
Interface Serial0/0/0
Hardware is PowerQUICC MPC860
DTE V.35 TX and RX clocks detected
idb at 0x81081AC4, driver data structure at 0x81084AC0
SCC Registers:
```

Pada gambar diatas terlihat bahwa pada R2 DTE tidak ada clock ratenya ini dikarenakan DTE hanya akan menyesuaikan clock ratenya dengan DCE. Untuk merubah clockrate kita bisa menggunakan pertintah seperti pada contoh di bawah ini

```
R1(config)#int se0/0/0
R1(config-if)#clock rate ?
Speed (bits per second)
1200
2400
4800
9600
19200
38400
56000
64000
72000
125000
128000
148000
250000
500000
800000
1000000
1300000
2000000
4000000
<300-4000000> Choose clockrate from list above
R1(config-if)#clock rate 125000
```

Gambar diatas ada beberapa pilihan untuk konfigurasi clock rate, pada gambar diatas clock rate menggunakan satuan bit. Untuk konfigurasi lengkapnya seperti pada contoh di bawah ini

```
R1(config-if)#do sh controllers se0/0/0
Interface Serial0/0/0
Hardware is PowerQUICC MPC860
DCE V.35, clock rate 125000
idb at 0x81081AC4, driver data structure at 0x81084AC0
SCC Registers:
```

Setelah selesai mari kita konfigurasikan ip address pada kedua router, seperti pada gambar di bawah ini

```
R1(config-if)#int se0/0/0
R1(config-if)#ip ad 11.11.11.2 255.255.255.0
R1(config-if)#no sh

R2(config)#int s0/0/0
R2(config-if)#ip ad 11.11.11.1 255.255.255.0
R2(config-if)#no sh
```

Setelah itu mari kita uji coba ping seperti ini

```
R1(config-if)#do ping 11.11.11.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.11.11.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5) round-trip min/avg/max = 4/5/9 ms
```

Komunitas IT SMKN 1 Nglelok



LAB 49 PPP Cisco

Pada lab ini kita akan membahas tentang PPP pada cisco. PPP merupakan salah satu WAN protocol yang bekerja pada layer 2 dengan cara mengencapsulasi frame menjadi paket PPP. Untuk konfigurasinya kita akan menggunakan topologi seperti pada gambar di bawah ini



Sebelum kita mulai konfigurasi utama terlebih dahulu kita konfigurasikan IP address dulu, untuk konfigurasinya seperti pada gambar di bawah ini

```
R1(config)#int s0/0/0
R1(config-if)#ip ad 11.11.11.2 255.255.255.0
R1(config-if)#no sh

R2(config)#int s0/0/0
R2(config-if)#ip ad 11.11.11.1 255.255.255.0
R2(config-if)#no sh
```

Setelah konfigurasi IP selesai kita akan melanjutkan dengan konfigurasi PPP. Untuk konfigurasinya kita harus mengubah encapsulasi pada interface serial terlebih dahulu karena defaultnya pada cisco interface serial diencapsulasi dengan HDLC, oleh karena itu kita harus merubahnya terlebih dahulu. Untuk merubah encapsulasi pada interface serial bisa menggunakan perintah seperti pada contoh di bawah ini

```
R1(config-if)#int s0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#

```

Setelah selesai mengkonfigurasi encapsulasi interface serial akan down, ini bisa terjadi karena perbedaan encapsulasi pada kedua router, oleh karena itu kita harus merubah encapsulasi pada router R2

```
R2(config-if)#int s0/0/0
R2(config-if)#encapsulation ppp
R2(config-if)#

```

Untuk memastikan apakah encapsulasi sudah selesai atau belum bisa dilihat dengan menambahkan command seperti pada contoh di bawah ini

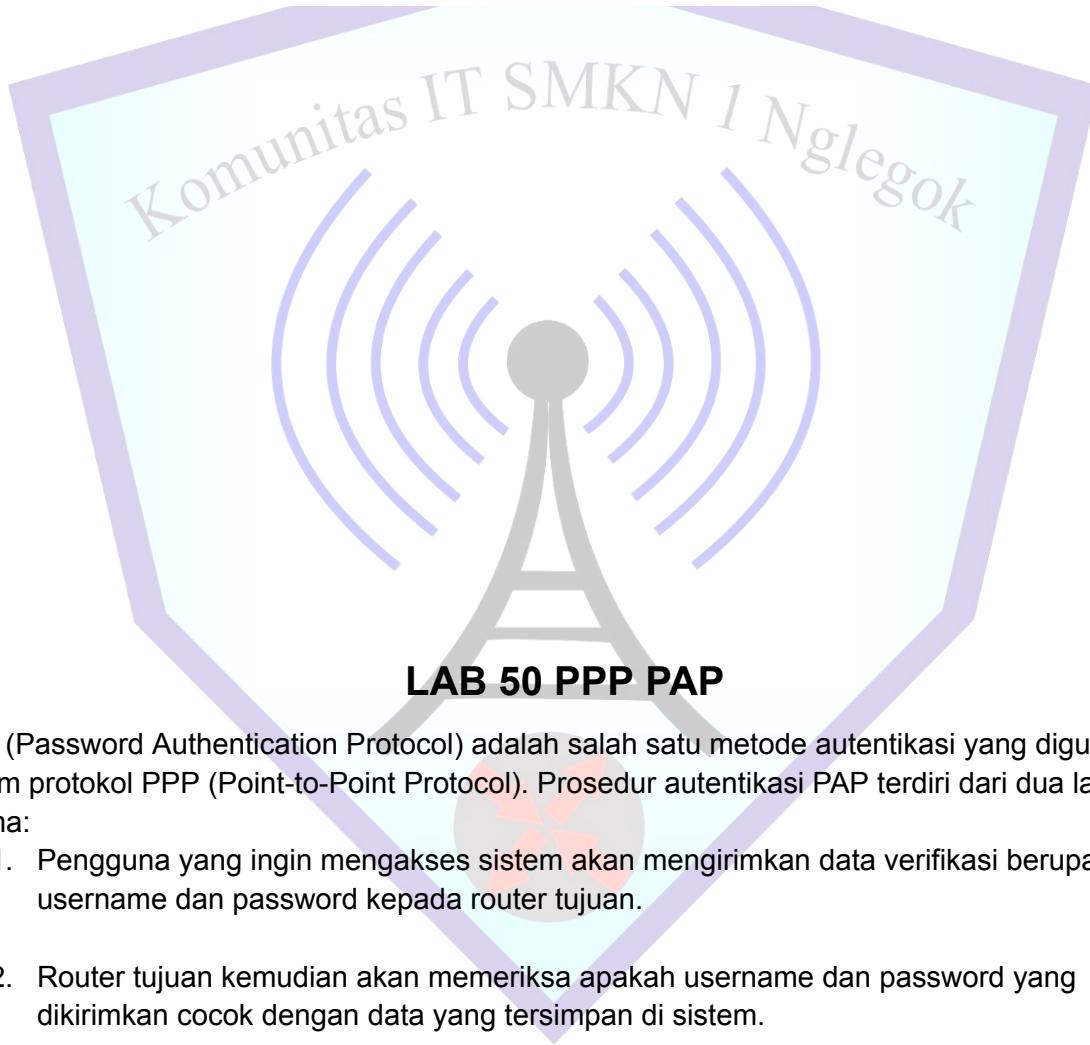
```
R1#show int s0/0/0
Serial0/0/0 is up, line protocol is up (connected)
  Hardware is HD64570
  Internet address is 11.11.11.2/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  LCP Open
  Open: IPCP, CDPCP

```

```
R2#show int s0/0/0
Serial0/0/0 is up, line protocol is up (connected)
  Hardware is HD64570
  Internet address is 11.11.11.1/24
  MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  LCP Open
  Open: IPCP, CDPCP
```

Jika kedua router sudah di pastikan menggunakan encapsulasi PPP mari kita coba ping antar router

```
R1#ping 11.11.11.2  
  
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 11.11.11.2, timeout is 2 seconds:  
!!!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 5/5/8 ms
```



PAP (Password Authentication Protocol) adalah salah satu metode autentikasi yang digunakan dalam protokol PPP (Point-to-Point Protocol). Prosedur autentikasi PAP terdiri dari dua langkah utama:

1. Pengguna yang ingin mengakses sistem akan mengirimkan data verifikasi berupa username dan password kepada router tujuan.
2. Router tujuan kemudian akan memeriksa apakah username dan password yang dikirimkan cocok dengan data yang tersimpan di sistem.
 - Jika sesuai, maka koneksi akan diterima.
 - Jika tidak cocok, maka koneksi akan ditolak.

Namun, perlu diketahui bahwa router Cisco mendukung dua metode autentikasi utama dalam PPP, yaitu:

1. PAP (Password Authentication Protocol)

- Metode ini menggunakan pengiriman username dan password dalam bentuk teks biasa (plaintext), sehingga dari sisi keamanan, PAP dianggap kurang aman karena dapat disadap (sniffed) oleh pihak tidak bertanggung jawab.

2. CHAP (Challenge Handshake Authentication Protocol)

- CHAP merupakan metode autentikasi yang lebih aman dibandingkan PAP. Dalam CHAP, proses autentikasi dilakukan melalui mekanisme tantangan dan respons (challenge-response).
- Router penghubung akan mengirimkan sebuah challenge (tantangan) ke klien. Klien kemudian merespons tantangan tersebut dengan hash dari username, password, dan nilai challenge yang dikirim sebelumnya.
- Karena password tidak pernah dikirim secara langsung melalui jaringan, CHAP lebih aman dibandingkan PAP.

Pada lab ini kita akan membahas tentang authentikasi PAP, untuk authentikasi CHAP akan kita bahas pada lab berikutnya. Untuk konfigurasi PAP kita akan menggunakan topologi seperti pada gambar di bawah ini



Sebelum kita mulai konfigurasi PAP pastikan bahwa ip sudah kalian setting seperti pada contoh diatas

```
R1(config)#int s0/0/0
R1(config-if)#ip ad 11.11.11.1 255.255.255.0
R1(config-if)#no sh

R2(config)#int s0/0/0
R2(config-if)#ip ad 11.11.11.2 255.255.255.0
R2(config-if)#no sh
```

Jika ip address sudah setting lanjutkan dengan konfigurasi PPP pada kedua router, contohnya seperti pada contoh di bawah ini

```
R1(config-if)#int s0/0/0
R1(config-if)#encapsulation ppp
R1(config-if)#
R2(config-if)#int s0/0/0
R2(config-if)#encapsulation ppp
R2(config-if)#

```

Pada dua gambar diatas terlihat bahwa kita mengkonfigurasikan encapsulation PPP ini digunakan untuk merubah encapsulasi pada interface s0/0/0

Setelah selesai megkonfigurasi PPP, kita lanjutkan dengan mengkonfigurasi username dan password untuk authentikasi PAP. Untuk contoh konfigurasinya seperti pada contoh di bawah ini

```
R1(config)#username R1 pass 123
R2(config)#
R2(config)#username R2 pass 456

```

Jika sudah kalian buat username beserta passwordnya, selanjutnya ita konfigurasi PAP pada kedua router. Untuk konfigurasi PAP seperti pada contoh di bawah ini

```
R1(config)#int s0/0/0
R1(config-if)#ppp authentication pap
R1(config-if)#
*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
R1(config-if)#ppp pap sent-username R2 password 456
R1(config-if)#
R2(config)#int s0/0/0
R2(config-if)#ppp authentication pap
R2(config-if)#
*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down
R2(config-if)#ppp pap sent-username R1 password 123

```

Untuk mengetahui apakah PPP PaP sudah berjalan atau tidak bisa mengeceknya dengan menggunakan pertintah seperti pada contoh di bawah ini

```
R1#debug ppp negotiation
PPP protocol negotiation debugging is on
R1#
```

Setelah kalian mengetik perintah , coba kita disable enable interface pada R2 seperti pada contoh di bawah ini

```
Enter configuration commar
R1(config)#int s0/0/0
R1(config-if)#no sh
R1(config-if)#sh
```

Setelah itu lihat pada R1, jika authentikasi PPP PAP berhasil maka nanti akan muncul notifikasi seperti pada contoh di bawah ini:

```
R1(config-if)#
*LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

Serial0/0/0 PPP: Using default call direction
Serial0/0/0 PPP: Treating connection as a dedicated line
Serial0/0/0 PPP: Phase is ESTABLISHING, Active Open

Serial0/0/0 LCP: State is Open
Serial0/0/0 PPP: Phase is AUTHENTICATING
Serial0/0/0 LCP: State is Open
Serial0/0/0 PPP: Phase is AUTHENTICATING
Serial0/0/0 Using hostname from interface PAP
Serial0/0/0 Using password from interface PAP
Serial0/0/0 PAP: O AUTH-REQ id 17 len 15
Serial0/0/0 PAP: I AUTH-REQ id 17 len 15
Serial0/0/0 PAP: Authenticating peer
Serial0/0/0 PAP: Phase is FORWARDING, Attempting Forward
Serial0/0/0 PAP: Phase is FORWARDING, Attempting Forward
Serial0/0/0 PPP: Phase is FORWARDING, Attempting Forward
Serial0/0/0 Phase is ESTABLISHING, Finish LCP
Serial0/0/0 Phase is UP
```

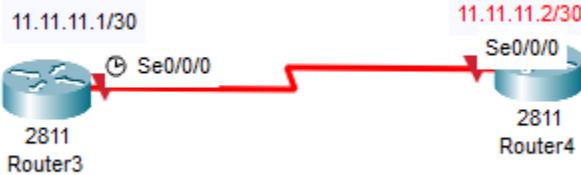
Selanjutnya mari kita coba untuk melakukan ping dari R1 menuju R2 seperti pada contoh di bawah ini

```
R1(config)#do ping 11.11.11.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.11.11.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
```

LAB 51 PPP CHAP

Pada lab sebelumnya kita telah mencoba untuk menggunakan PAP untuk authentikasi, pada lab ini kita akan menggunakan CHAP. CHAP ini adalah protocol authentikasi three-way handshaking, yang memberikan keamanan lebih tinggi dari pada PAP, pada protocol ini password akan disimpan dan akan mengenkripsi password untuk keperluan authentikasi. Untuk contoh konfigurasi kalian bisa gunakan topologi seperti pada contoh di bawah ini



Sebelum kita memulai konfigurasi kita akan mengonfigurasikan IP terlebih dahulu seperti tertera pada contoh diatas

```

R1(config)#int s0/0/0
R1(config-if)#ip add 11.11.11.1 255.255.255.252
R1(config-if)#no sh
Router1(config)#
R2(config)#int s0/0/0
R2(config-if)#ip add 11.11.11.2 255.255.255.252
R2(config-if)#no sh
    
```

Setelah selesai kita lanjutkan dengan konfigurasi PPP CHAP, untuk konfigurasinya seperti pada contoh di bawah ini

```

R1(config-if)#encapsulation ppp
R1(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down

R1(config-if)#ppp authentication chap
R1(config-if)#ex

R2(config-if)#encapsulation ppp
R2(config-if)#ppp authentication chap
R2(config-if)#ex
    
```

Setelah kalian mengaktifkan PPP CHAP,kita harus mengkonfigurasikan username dan password, untuk contoh konfigurasi username kita harus menggunakan hostname router lawan untuk passwod kedua router harus sama. **CATATAN hostname menggunakan nama lawannya**

```

R1(config)#username R2 password rvc
R1(config)#
    
```

```

R2(config)#username R1 password rvc
R2(config)#
    
```

Untuk melihat PPP CHAP berjalan dengan baik kita bisa menggunakan perintah debug ppp negotiation untuk melihat authentikasinya. Untuk contoh kalian bisa lihat seperti pada contoh

```

R1#debug ppp negotiation
PPP protocol negotiation debugging is on
    
```

Jika pada R1 kalian sudah mengaktifkan debug PPP, kalian coba disable enable pada interface R2 seperti pada contoh di bawah ini

```
R2#config)#int s0/0/0
R2(config-if)#sh

R2(config-if)#
*LINK-5-CHANGED: Interface Serial0/0/0, changed state to administratively down

*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to down

R2(config-if)#no sh

R2(config-if)#
*LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

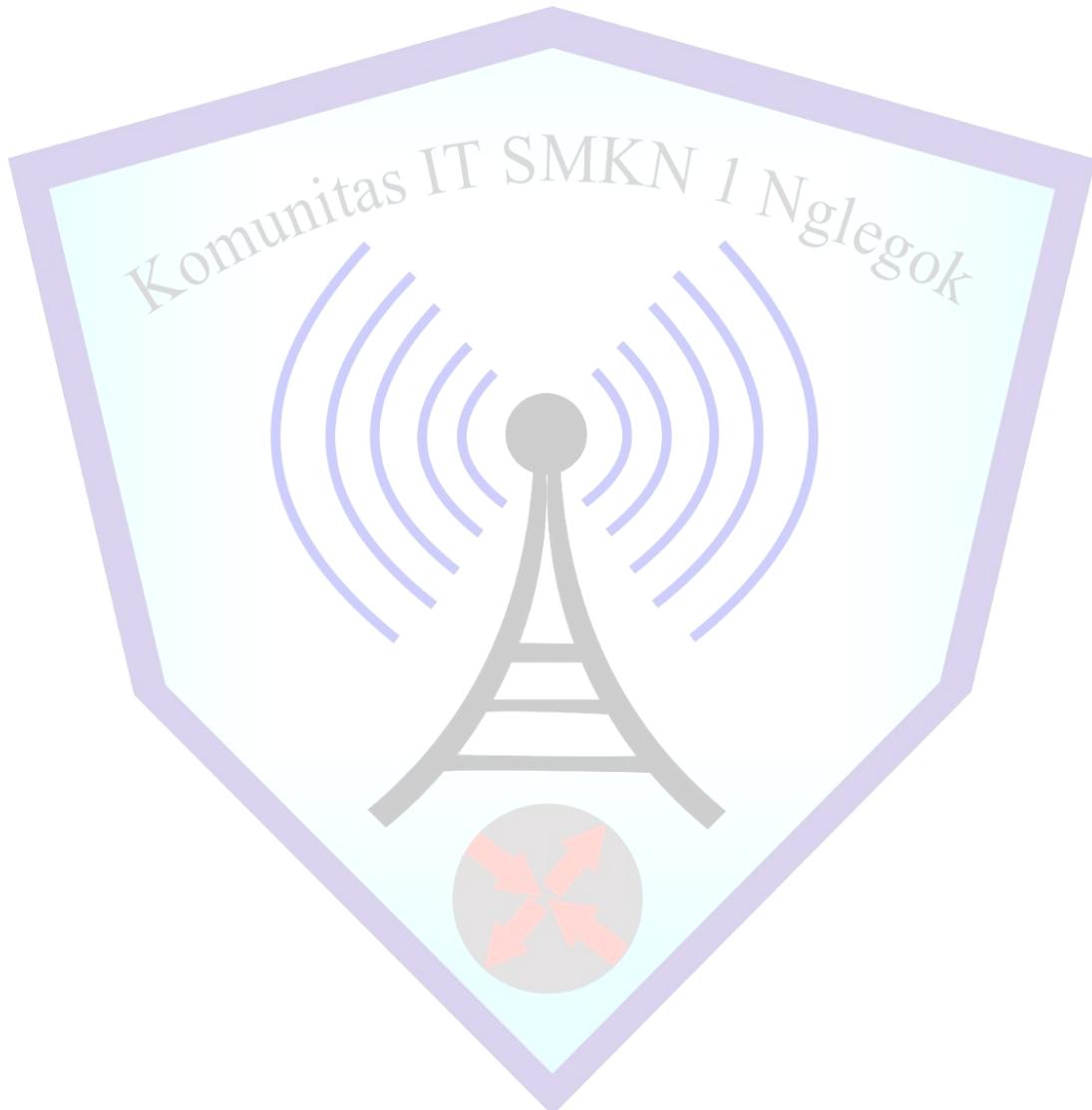
*LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up
```

Jika authentifikasi berhasil maka pada R1 akan muncul notifikasi seperti pada contoh gambar di bawah ini

```
Serial0/0/0 LCP: State is Open
Serial0/0/0 PPP: Phase is AUTHENTICATING
Serial0/0/0 IPCP: O CONFREQ [Closed] id 1 len 10
Serial0/0/0 IPCP: I CONFREQ [Closed] id 1 len 10
Serial0/0/0 IPCP: O CONFACK [Closed] id 1 len 10
Serial0/0/0 IPCP: I CONFACK [Closed] id 1 len 10
Serial0/0/0 IPCP: O CONFREQ [Closed] id 1 len 10
Serial0/0/0 IPCP: I CONFREQ [REQsent] id 1 len 10
Serial0/0/0 IPCP: O CONFACK [REQsent] id 1 len 10
Serial0/0/0 IPCP: I CONFACK [REQsent] id 1 len 10
Serial0/0/0 PPP: Phase is FORWARDING, Attempting Forward
Serial0/0/0 Phase is ESTABLISHING, Finish LCP
Serial0/0/0 Phase is UP
```

Selain itu kalian juga bisa mengetahui apakah authentifikasi berhasil atau tidak dengan menggunakan ping. Jika ping berhasil maka authentifikasi berhasil jika tidak berhasil/ rto berarti sebaliknya authentifikasi kalian gagal

```
R1(config)#do ping 11.11.11.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 11.11.11.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/9/15 ms
```



DAFTAR PUSTAKA

CCNA - M Syahrur Rojab

Komunitas IT SMKN 1 Nglelok



Biografi Penulis



Perkenalkan nama saya Evan Cristianto, saya biasanya dipanggil evan. Saya lahir di Blitar 25 Oktober 2008. Saya membuat buku ini saat berusia 16 tahun. Saya mempunyai orang tua yang terus memberi semangat serta dukungan sehingga dapat menyelesaikan buku dengan tepat waktu. Ayah saya bernama Edi Susanto dan Ibu saya bernama Desiati.

Sebelum saya bersekolah di SMKN 1 Nglegok saya terlebih dahulu sekolah di SMPN 2 Gandusari dan melanjutkan di SMKN 1 Nglegok Jurusan Teknik Komputer dan jaringan. Alasan saya memilih TKJ karena saya dari kecil sudah menyukai computer. Karena keinginan saya untuk menjadi IT yang handal saya bergabung dengan Komunitas IT SMKN 1 Nglegok agar saya dapat berkembang tidak hanya skil tapi seluruhnya. Semoga buku yang saya buat ini serta semua ilmu dapat berguna, serta bermanfaat bagi semua orang tanpa terkecuali. Terima Kasih.



