

# **Modul Pra-praktikum IF2230 Jaringan Komputer**

1 - Perkenalan CPT, Frame Switching, LAN, VLAN, dan Subnetting

**Dipersiapkan oleh Sister'21**

**Waktu Mulai :**

Selasa, 24 September 2024, 15:00 WIB

**Waktu Akhir :**

Selasa, 1 Oktober 2024, 13:00 WIB

## I. Daftar Revisi

1. Mengubah waktu mulai dan akhir praktikum
2. Mengubah format nama submisi

## II. Latar Belakang dan Peraturan

Tugas ini ditujukan untuk mempersiapkan peserta untuk praktikum pertama kuliah ini. Dengan menyelesaikan tugas ini, praktikan diharapkan memiliki persiapan dan pengetahuan dasar terhadap materi yang dibutuhkan.

Berikut topik-topik yang menjadi lingkup modul ini:

- Perkenalan Cisco Packet Tracer
- Frame Switching
- LAN dan Subnetting
- VLAN

Kerjakan tugas ini **dengan mengikuti peraturan-peraturan berikut:**

1. **Pra-praktikum ini menjadi syarat untuk praktikum yang akan diadakan terkait modul ini. Tidak mengumpulkan tugas/modul ini akan menyebabkan nilai praktikum 0.**
2. Kumpulkan tugas anda sesuai dengan arahan pengumpulan yang terdapat pada bagian "Deliverables". **Pengumpulan yang tidak sesuai dengan arahan akan mengurangi nilai praktikan.**
3. Praktikan diperbolehkan mengerjakan bersama-sama dengan praktikan lain dan menggunakan referensi & material yang dianggap sesuai ketentuan; namun, praktikan diharapkan memahami segala yang telah dikerjakan pada pengumpulan.
4. Praktikan tidak diperbolehkan untuk menyalin materi referensi atau pekerjaan praktikan lain secara langsung ke dalam pekerjaan praktikan.
5. Tanyakan segala pertanyaan terkait tugas ini pada sheet Q&A.
6. Revisi akan dilakukan secara langsung pada dokumen yang diberikan saat rilis dan akan ditambahkan pada bagian "Daftar Revisi". Bagian-bagian yang direvisi akan diberi warna khusus.

<p><b>Segala bentuk kecurangan akademik (seperti mengganti nama dan mengumpulkan tugas praktikan lain, plagiarisme, dan sebagainya) akan mengakibatkan ketidakkululusan mata kuliah ini (dan sanksi-sanksi lain yang berlaku dalam lingkungan akademik ITB).</b></p>
--

### III. Deliverables

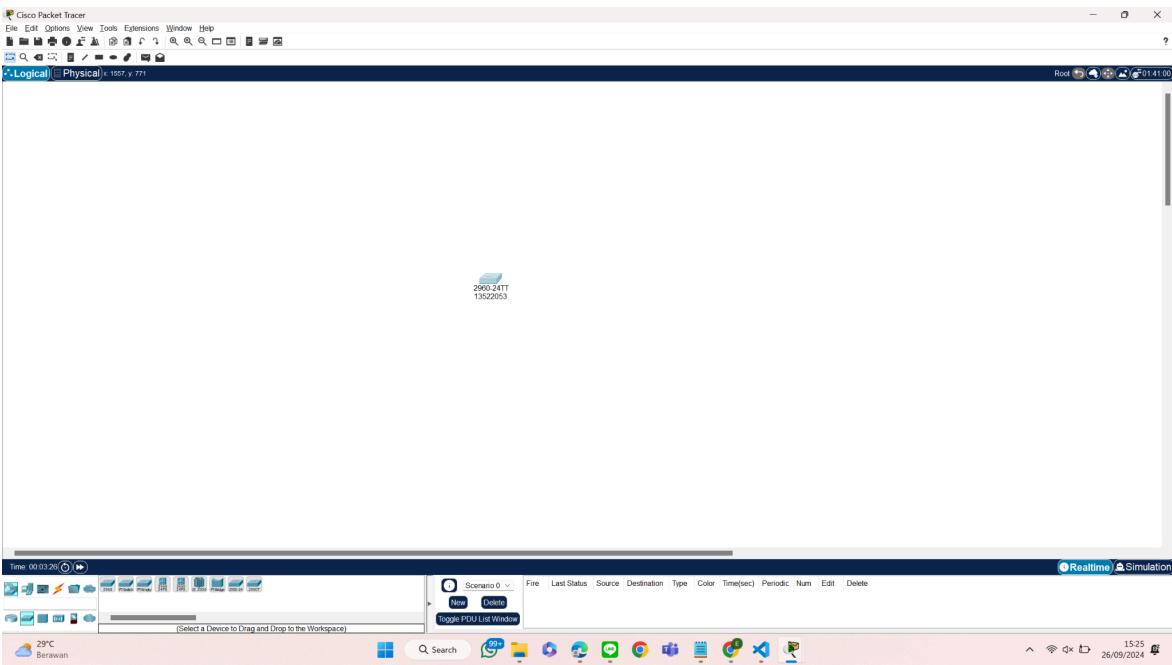
Kumpulkan tugas ini **dengan mengikuti peraturan-peraturan berikut:**

1. Buatlah salinan dari dokumen ini, dan kerjakan tugas-tugas ini pada salinan dokumen praktikan masing-masing.
2. Ikuti arahan dan instruksi yang diberikan pada setiap bagian untuk menyelesaikan pra-praktikum ini. Bagian-bagian yang perlu dikerjakan terdapat pada tabel-tabel dengan *header* kuning dan isi jawaban anda pada bagian dengan label **<Jawab>**.
3. **Semua *screenshot* harus dapat dibaca dengan jelas. Segala *screenshot* yang tidak dapat dibaca dengan jelas dengan akibat apapun tidak akan dinilai.**
4. Simpan tugas anda dengan format berikut:
5. Kumpulan tugas anda melalui form ini. Upload dokumen anda dengan format penamaan: **IF2230\_LabPrep[X]\_<NIM>.pdf**

## IV. Modul Pra-praktikum

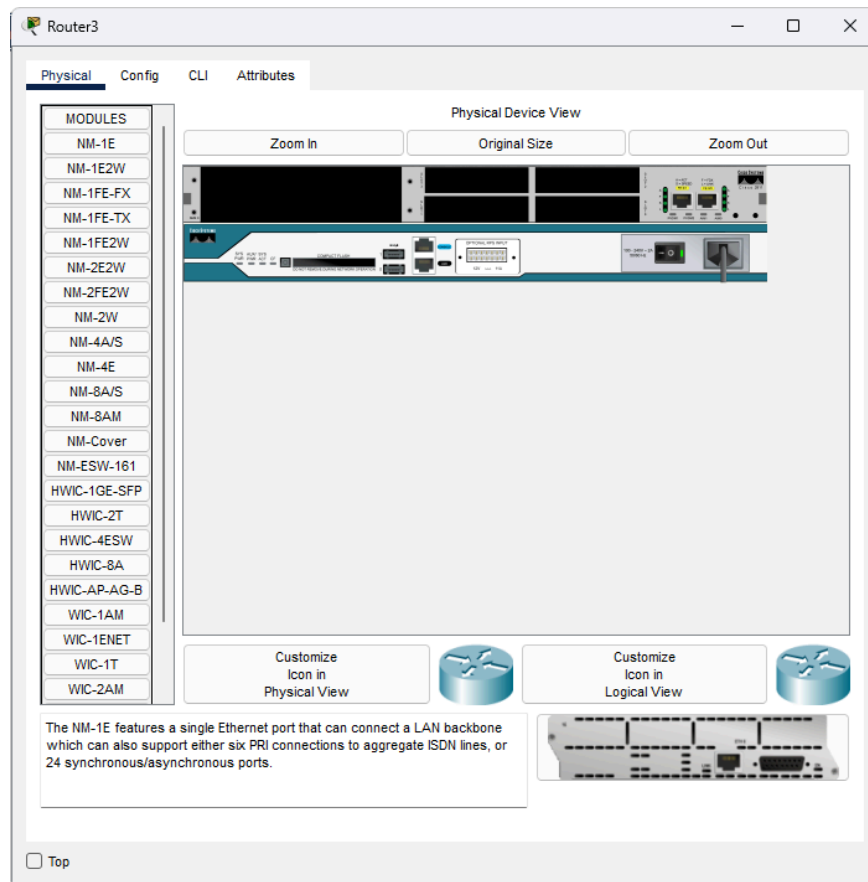
### IV.1. Perkenalan Cisco Packet Tracer

Cisco Packet Tracer adalah kakas simulasi jaringan komputer yang dikembangkan oleh Cisco. Kakas ini digunakan terutama untuk tujuan pembelajaran dan tidak menyimulasikan jaringan atau perangkat seperti *virtual machine*. Kakas ini khusus menyediakan perangkat-perangkat keras jaringan Cisco.

Tugas 1	
Q	<p>Download Cisco Packet Tracer menggunakan link ini <a href="https://www.netacad.com/resources/get-packet-tracer">Resource Hub: Get Packet Tracer, Virtual Machines, and More (netacad.com)</a></p> <p><i>Log in</i> dengan email anda (akun apa saja, ITB atau pribadi) Dan ikuti instruksi hingga anda dapat mengakses antarmuka Cisco Packet Tracer. Tambahkan perangkat <i>switch</i> 2960 dan ganti <i>display name switch</i> tersebut menjadi NIM anda.</p> <p><b>Screenshoot dan lampirkan hasilnya di bawah!</b></p>
A	 <p>The screenshot shows the Cisco Packet Tracer application window. The main workspace is empty except for a single 2960 switch device (labeled 2960-24TT 13522053) placed in the center. The interface includes a menu bar at the top, a toolbar on the left, and a bottom panel with a 'Scenario' tab and a 'Realtime' simulation button. The Windows taskbar is visible at the bottom of the screen.</p>

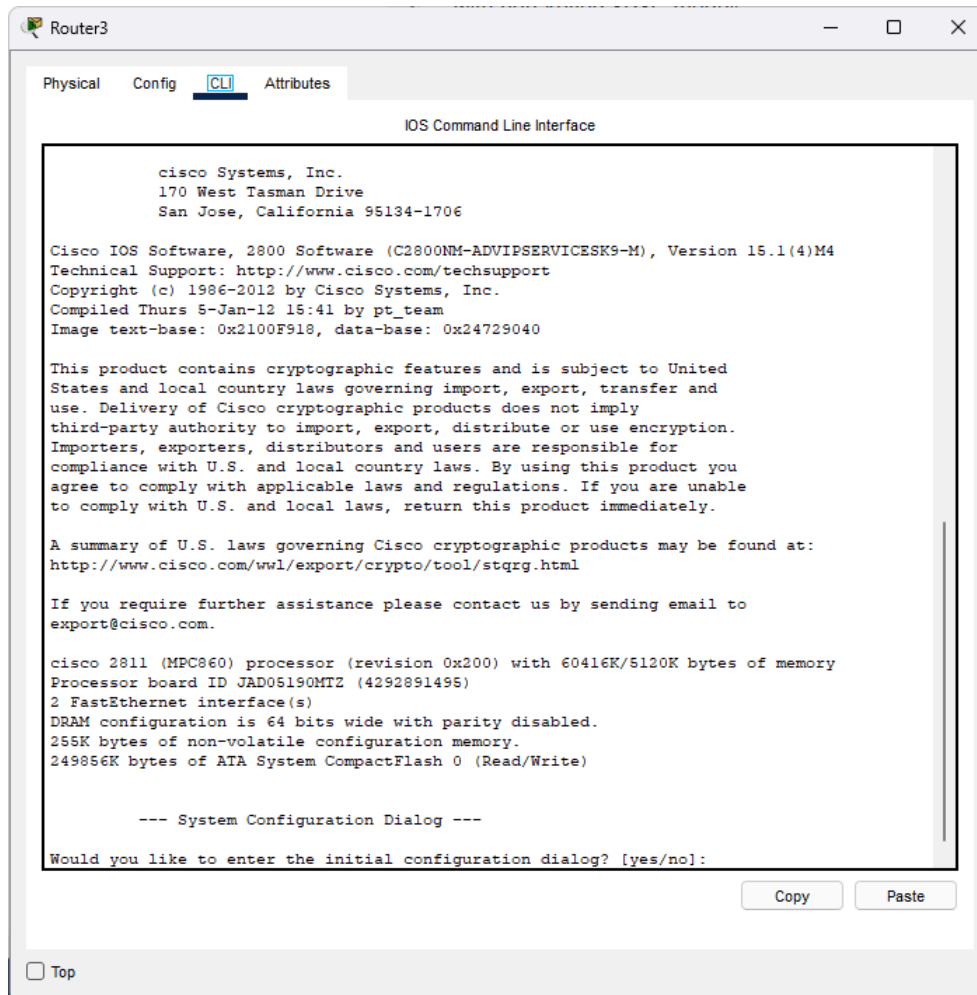
Cisco Packet Tracer menyediakan simulasi perangkat keras pada perangkat-perangkat yang anda tambahkan, yang memungkinkan anda menambah/mengurangi modul-modul pada

perangkat tersebut. Antarmuka ini dapat diakses dengan *double click* pada perangkat, dan memilih bagian '*Physical*' pada menu bagian atas.



*Gambar 1. Antarmuka Konfigurasi Perangkat Keras pada Cisco ISR2811*

*Router* dan *switch* dikonfigurasi pada lokasinya secara langsung melalui *serial port* yang diakses menggunakan *rollover cable*, umumnya menggunakan perangkat lunak seperti [PuTTY](#) atau [SecureCRT](#). Namun, Cisco Packet Tracer menyederhanakan proses ini dengan menyediakan CLI (Command Line Interface) pada antarmuka yang sama dengan antarmuka perangkat keras dengan memilih '*CLI*' pada menu bagian atas.



Gambar 2. CLI pada Cisco ISR2811

Apa itu *rollover cable*? *rollover cable* adalah salah satu konfigurasi [RJ45](#) dengan jenis kabel [UTP \(Unshielded Twisted Pair\)](#). Kabel-kabel dihubungkan dengan urutan *rolling*: 1-to-8, 2-to-7, 3-to-6, dan seterusnya hingga 8-to-1. Terdapat konfigurasi kabel lain seperti *Straight-through* dan *Crossover*. Naumn, kabel-kabel ini sudah jarang digunakan sejak munculnya [Auto MDI-X](#).

Terdapat beberapa mode konfigurasi pada CLI Perangkat-perangkat Cisco. Saat pertama kali terhubung, pengguna mengakses dengan mode *User EXEC*. Mode ini ditandai dengan simbol `'>'` sebagai pemisah antara *hostname* (nama perangkat) dan terminal aktif. Untuk menampilkan petunjuk-petunjuk terkait *command* CLI, pengguna dapat mengetikkan `'?'` untuk menampilkan daftar *command* yang ada atau menunjukkan *completion* berdasarkan *state* dari CLI.

```

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
Router>|

```

*Gambar 3. Command pada Mode User EXEC Cisco ISR2811*

Pada daftar *command* di atas, hanya sedikit konfigurasi perangkat yang dapat dilakukan pada mode ini. Untuk melakukan konfigurasi pada perangkat, pengguna pertama harus memasuki mode *privileged EXEC* dengan *command* 'enable'. Mode *privileged EXEC* ditandai dengan simbol '#' sebagai pemisah antara *hostname* dan terminal aktif.

```

Router>en
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
resume      Resume an active network connection
rmdir       Remove existing directory
send        Send a message to other tty lines
setup       Run the SETUP command facility
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
undebg      Disable debugging functions (see also 'debug')
vlan        Configure VLAN parameters
write       Write running configuration to memory, network, or terminal
Router#|

```

*Gambar 4. Command pada Mode Privileged EXEC Cisco ISR2811*

Pada gambar di atas, mengetikkan 'en' memiliki efek seperti mengetikkan 'enable' secara lengkap. Cisco CLI dapat mengeksekusi singkatan *command* ketika hanya ada satu *command*



yang diawali dengan singkatan tersebut. Pada contoh di atas, tidak terdapat *command* lain yang diawali dengan 'en' pada mode *User EXEC*.

Terdapat banyak *command* yang dapat dijalankan pada mode *Privileged EXEC*, namun konfigurasi tertentu diakses dengan mengaktifkan mode konfigurasi spesifik. Konfigurasi umum perangkat dapat dilakukan pada mode *Global Configuration* yang diakses dengan *command* 'configure terminal' pada mode *Privileged EXEC*. Mode *Global Configuration* ditandai dengan '(config)#' sebagai pemisah antara *hostname* dan terminal aktif.

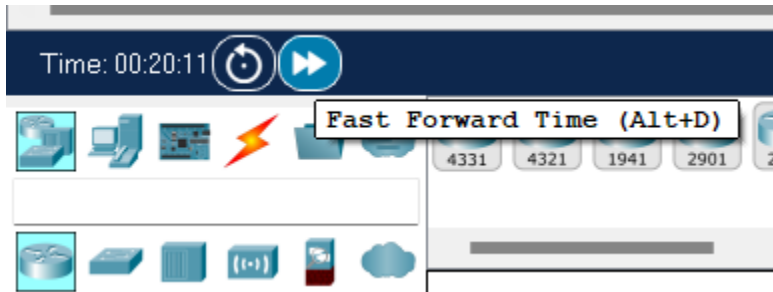
```
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#?
Configure commands:
aaa                Authentication, Authorization and Accounting.
access-list        Add an access list entry
banner             Define a login banner
bba-group          Configure BBA Group
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
interface         Select an interface to configure
ip              Global IP configuration subcommands
ipv6            Global IPv6 configuration commands
key             Key management
line            Configure a terminal line
lldp           Global LLDP configuration subcommands
logging        Modify message logging facilities
login         Enable secure login checking
mac-address-table Configure the MAC address table
no           Negate a command or set its defaults
ntp          Configure NTP
parameter-map parameter map
parser       Configure parser
policy-map   Configure QoS Policy Map
port-channel EtherChannel configuration
priority-list Build a priority list
privilege    Command privilege parameters
queue-list   Build a custom queue list
router       Enable a routing process
secure       Secure image and configuration archival commands
security     Infra Security CLIs
service      Modify use of network based services
snmp-server  Modify SNMP engine parameters
spanning-tree Spanning Tree Subsystem
tacacs-server Modify TACACS query parameters
telephony-service Configure Cisco Unified Communications Manager Express
username     Establish User Name Authentication
vpdn         Virtual Private Dialup Network
vpdn-group   VPDN group configuration
zone         FW with zoning
zone-pair    Zone pair command
Router(config)#
```

*Gambar 5. Global Configuration Mode commands on Cisco ISR2811*

Terdapat beberapa mode konfigurasi lain, beberapa di antaranya adalah mode *line configuration*, mode *interface configuration*, mode *multi-interface configuration*. Sebagian mode-mode tersebut akan digunakan pada aktivitas lab ke depannya. Anda dapat mengakses penjelasan lebih lengkap terkait CLI pada [Cisco IOS official CLI reference book](#).

**Catatan:** Ketika anda memasukkan *command* yang tidak dikenali oleh Cisco CLI, CLI akan melakukan *DNS lookup* (DNS akan dipelajari di kelas kedepannya), dan mengakibatkan pengguna harus menunggu cukup lama (yang cukup mengganggu jika anda banyak melakukan kesalahan *command* (yang telah dialami oleh para asisten dahulu (yang tentu saja maklum terjadi ketika baru saja memulai menggunakan CLI Cisco))) Untuk mengatasi hal ini, terdapat beberapa hal yang dapat anda lakukan:

- Batalkan *command*  
Gunakan Ctrl+shift+6
- Gunakan fitur Fast Forward Time (or Alt+D) di simulator



- Nonaktifkan *DNS lookup* pada perangkat  
Aktifkan mode *Global Configuration* (telah dibahas di bagian sebelumnya) dan gunakan *command*  
`no ip domain-lookup`

Kami **sangat menyarankan** anda untuk menggunakan “?” setiap kali anda menggunakan *command* baru untuk mendapatkan intuisi dan pengertian lebih terkait sintaks CLI dari sekarang.

Tugas 2	
Q	<p>Konfigurasi switch yang telah anda tambahkan pada <a href="#">Tugas 1</a>:</p> <ul style="list-style-type: none"> <li>- Set <i>hostname</i> menjadi nama belakang anda</li> <li>- <a href="#">Set unencrypted password untuk menghubungkan perangkat</a></li> <li>- <a href="#">Set unencrypted password untuk mengakses mode privileged EXEC</a></li> <li>- <a href="#">Encrypt the password to enter privileged mode using MD5 algorithm</a></li> <li>- Save konfigurasi</li> </ul> <p>Lampirkan <i>screenshot-screenshot</i> berikut pada kolom jawaban di bawah:</p> <ul style="list-style-type: none"> <li>- Prompt password saat menghubungkan CLI</li> <li>- Prompt password saat memasuki mode <i>privileged EXEC</i></li> <li>- Konfigurasi perangkat sebelum dan sesudah password dienkripsi</li> <li>- <i>Startup-config</i> sebelum dan sesudah <i>saving</i> (cukup <i>highlight</i> sebagian perubahan)</li> </ul>

## A Prompt password saat menghubungkan CLI

```
Andini(config)#enable password wigglo923
Andini(config)#
```

Copy

Paste

☐ Top

```
Andini#exit
```

```
Andini con0 is now available
```

```
Press RETURN to get started.
```

```
User Access Verification
```

```
Password:
```

Copy

Paste

☐ Top

## Prompt password saat memasuki mode *privileged EXEC*

```
Andini(config)#line console 0
Andini(config-line)#login
% Login disabled on line 0, until 'password' is set
Andini(config-line)#password wigglo923
```

```
User Access Verification
```

```
Password:
```

```
Andini>en
```

```
Password:
```

## Konfigurasi perangkat sebelum password dienkripsi

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

Current configuration : 1127 bytes
!
version 15.0
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Andini
!
enable password wig23
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
!
interface FastEthernet0/2
!
interface FastEthernet0/3
!
interface FastEthernet0/4
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
!
interface FastEthernet0/8
!
interface FastEthernet0/9
!
interface FastEthernet0/10
!
interface FastEthernet0/11
!
interface FastEthernet0/12
!
```

```
interface FastEthernet0/14
!
interface FastEthernet0/15
!
interface FastEthernet0/16
!
interface FastEthernet0/17
!
interface FastEthernet0/18
!
interface FastEthernet0/19
!
interface FastEthernet0/20
!
interface FastEthernet0/21
!
interface FastEthernet0/22
!
interface FastEthernet0/23
!
interface FastEthernet0/24
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
!
interface Vlan1
  no ip address
  shutdown
!
!
!
!
line con 0
  password wig23
  login
!
line vty 0 4
  login
line vty 5 15
  login
!
!
!
end
```

## Konfigurasi perangkat sesudah password dienkripsi

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

Current configuration : 1152 bytes
!
version 15.0
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Andini
!
enable secret 5 $1$mERr$TT.iZgwwm9rmixiZqW2gV0
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
!
interface FastEthernet0/2
!
interface FastEthernet0/3
!
interface FastEthernet0/4
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
!
interface FastEthernet0/8
!
interface FastEthernet0/9
!
interface FastEthernet0/10
!
interface FastEthernet0/11
!
interface FastEthernet0/12
!
```

```
interface FastEthernet0/14
!
interface FastEthernet0/15
!
interface FastEthernet0/16
!
interface FastEthernet0/17
!
interface FastEthernet0/18
!
interface FastEthernet0/19
!
interface FastEthernet0/20
!
interface FastEthernet0/21
!
interface FastEthernet0/22
!
interface FastEthernet0/23
!
interface FastEthernet0/24
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
!
interface Vlan1
no ip address
shutdown
!
!
!
!
line con 0
password wig23
login
!
line vty 0 4
login
line vty 5 15
login
!
!
!
!
end
```

**Startup-config sebelum saving** (cukup *highlight* sebagian perubahan)

```
Andini#show startup-config
startup-config is not present
```

**Startup-config** sesudah *saving* (cukup *highlight* sebagian perubahan)

```
Andini#show startup-config
Using 1152 bytes
!
version 15.0
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Andini
!
enable secret 5 $1$mERr$TT.iZgwm9rmixiZqW2gV0
!
!
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
!
interface FastEthernet0/1
!
interface FastEthernet0/2
!
interface FastEthernet0/3
!
interface FastEthernet0/4
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
!
interface FastEthernet0/8
!
interface FastEthernet0/9
!
interface FastEthernet0/10
!
interface FastEthernet0/11
!
interface FastEthernet0/12
!
interface FastEthernet0/13
!
interface FastEthernet0/14
!
interface FastEthernet0/15
!
interface FastEthernet0/16
!
interface FastEthernet0/17
!
```



```
interface FastEthernet0/18
!
interface FastEthernet0/19
!
interface FastEthernet0/20
!
interface FastEthernet0/21
!
interface FastEthernet0/22
!
interface FastEthernet0/23
!
interface FastEthernet0/24
!
interface GigabitEthernet0/1
!
interface GigabitEthernet0/2
!
interface Vlan1
no ip address
shutdown
!
!
!
!
line con 0
password wig23
login
!
line vty 0 4
login
line vty 5 15
login
!
!
!
!
end
```

## IV.2. Frame Switching

Sebuah frame adalah [PDU \(Protocol Data Unit\)](#) pada *layer* ke-2 [OSI Model](#), *data link layer*. *Data link layer* menyediakan *transfer* data antar *node* yang terhubung secara langsung. Selain itu, *layer* ini mendeteksi dan bisa saja mengoreksi *error* yang muncul pada *physical layer* dan mendefinisikan protokol untuk memulai dan menutup koneksi antar dua perangkat yang terhubung secara fisik. *Layer* ini juga menyediakan protokol untuk mengatur *flow* antar node.

OSI model				
Layer		Protocol data unit (PDU)	Function <sup>[26]</sup>	
Host layers	7	Application	Data	High-level protocols such as for resource sharing or remote file access, e.g. HTTP.
	6	Presentation		Translation of data between a networking service and an application; including character encoding, data compression and encryption/decryption
	5	Session		Managing communication sessions, i.e., continuous exchange of information in the form of multiple back-and-forth transmissions between two nodes
	4	Transport	Segment, Datagram	Reliable transmission of data segments between points on a network, including segmentation, acknowledgement and multiplexing
Media layers	3	Network	Packet	Structuring and managing a multi-node network, including addressing, routing and traffic control
	2	Data link	Frame	Transmission of data frames between two nodes connected by a physical layer
	1	Physical	Bit, Symbol	Transmission and reception of raw bit streams over a physical medium

Gambar 6. Tabel OSI Model

Switch bekerja pada layer ini dengan mencatat *identifier* perangkat yang bernama [MAC \(Media Access Control\)](#) dan memetakan *identifier* tersebut dengan antarmuka *port* yang terhubung pada *MAC Table*. Meskipun demikian, alamat yang digunakan pada antarmuka jaringan komputer tidak menggunakan *MAC address* melainkan [IP address](#), karena *MAC address* ditujukan sebagai *identifier* dari *network interface controller* dan tidak ditujukan sebagai alamat untuk *routing* komunikasi antar *node*. [ARP \(Address Resolution Protocol\)](#) digunakan untuk menentukan *MAC address* dari *IP address*, karena transmisi antar *node* menggunakan *MAC address*.

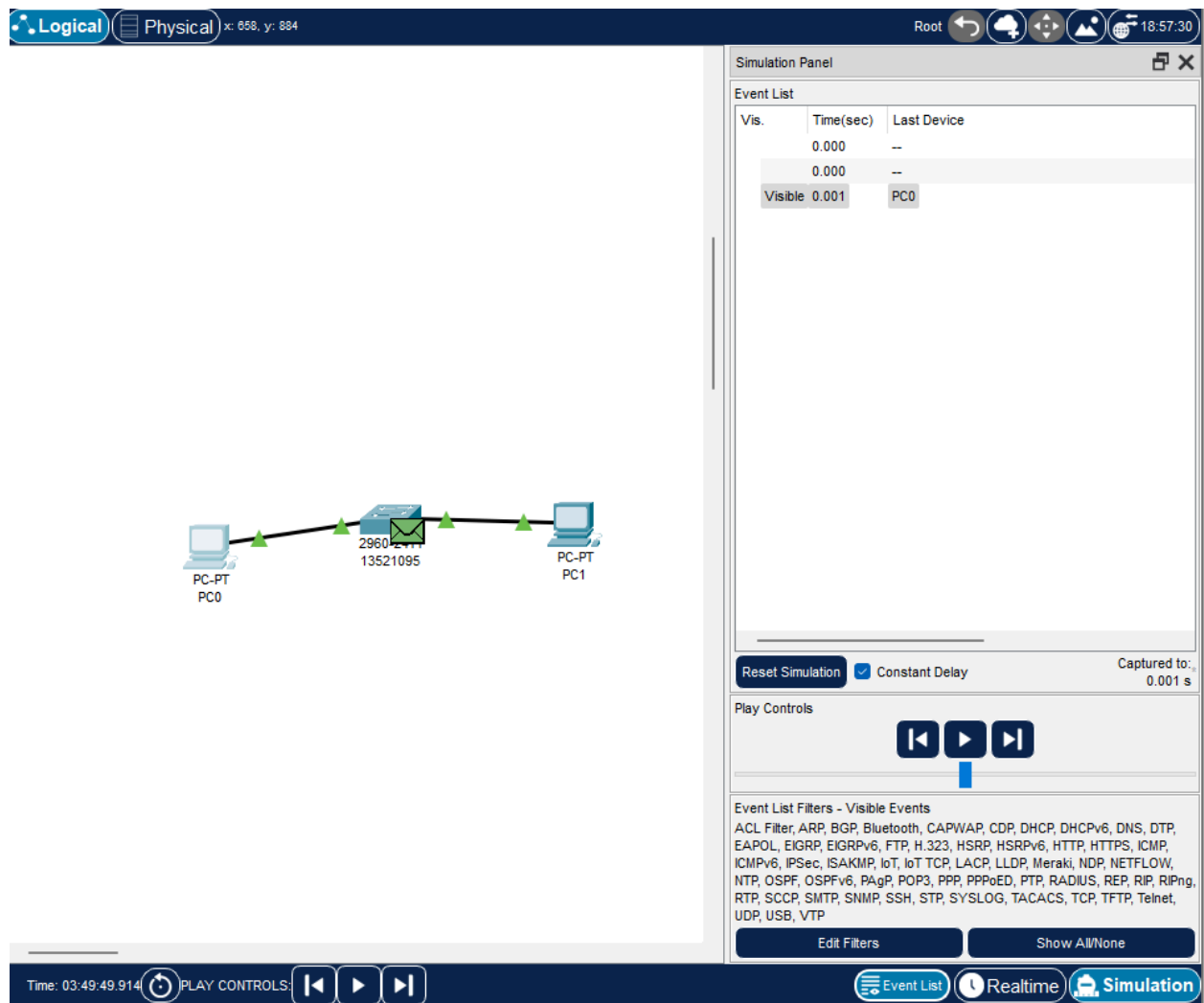
[Ping](#) adalah kaskas jaringan komputer yang digunakan untuk memeriksa keterjangkauan perangkat dalam jaringan melalui *internet protocol* dengan menggunakan [ICMP](#). Kaskas ini akan sangat sering digunakan dalam aktivitas-aktivitas lab ke depannya. Untuk memahami *frame switching*, kita dapat menggunakan *ping* sebagai contoh. *Ping* biasa dijalankan dengan perintah berikut

```
ping <IP_Address>
```

Misalkan terdapat dua komputer yang terhubung dengan satu sama lain melalui *switch* yang baru dinyalakan dengan *MAC address table* yang kosong, PC0 terhubung dengan *IP address* 192.168.1.1 dan *subnet mask* 255.255.255.0 serta PC1 terhubung dengan *IP address* 192.168.1.2 dan *subnet mask* 255.255.255.0, *ping* akan dijalankan dengan langkah-langkah berikut:

Step	Explanation
<i>ARP Request</i> PC0 ke Switch	PC0 hanya memiliki informasi <i>IP address</i> PC1, jadi PC0 mengirimkan <i>ARP request</i> dengan tujuan <i>MAC address</i> ffff:ffff:ffff pada <a href="#"><i>frame header</i></a> .
<i>ARP Request Flooding</i> Switch ke semua PC kecuali PC0	Karena switch memiliki <i>MAC address table</i> yang kosong, tidak ada informasi mengenai tujuan paket tersebut, sehingga switch mengirimkan paket tersebut ke semua <i>port</i> kecuali <i>port</i> yang menerima paket tersebut (hal ini dinamakan <i>flooding</i> ). Switch juga mencatat <i>MAC address</i> PC0 yang terdapat pada <i>frame header</i> yang diterimanya beserta <i>port</i> yang menerimanya.
<i>ARP Reply</i> PC1 ke Switch	PC1 mengenali <i>IP address</i> pada <i>ARP payload</i> dan membalas <i>request</i> tersebut ke switch.
<i>ARP Reply</i> Switch ke PC0	Switch telah mencatat <i>MAC address</i> PC0 dan meneruskan paket menuju <i>port</i> yang sesuai. Switch juga mencatat <i>MAC address</i> PC1 yang terdapat pada <i>frame header</i> beserta <i>port</i> yang sesuai.
<i>ICMP Echo Request</i> PC0 ke Switch	Setelah mengetahui <i>MAC address</i> PC1, PC0 mengirimkan paket ICMP.
<i>ICMP Echo Request</i> Switch ke PC1	Switch melakukan <a href="#"><i>decapsulation</i></a> paket ICMP hingga terdapat <i>MAC address</i> tujuan, mengenkapsulasi ulang paket tersebut, dan meneruskan paket ICMP menuju <i>port</i> tujuannya.
<i>ICMP Echo Reply</i> PC1 ke Switch	PC1 membalas <i>request</i> ICMP PC0, mengkonfirmasi <i>liveness</i> dan <i>reachability</i> PC1.
<i>ICMP Echo Reply</i> Switch ke PC0	Switch melakukan <i>decapsulation</i> paket ICMP hingga terdapat <i>MAC address</i> tujuan, mengenkapsulasi ulang paket tersebut, dan meneruskan paket ICMP menuju <i>port</i> tujuannya.

Cisco Packet Tracer memiliki fitur untuk memperlambat aktivitas perangkat-perangkat. Fitur ini dapat digunakan untuk memvisualisasikan cara kerja *ping request* seperti yang telah dijelaskan di atas. Selain itu, anda juga dapat melihat isi dari data komunikasi antar *node* dengan meng-klik ikon surat atau melalui entri pada *event list*.



Gambar 7. Fitur Simulasi Cisco Packet Tracer

### Tugas 3

Q Pada Cisco Packet Tracer, buatlah topologi *star* yang terdiri dari satu 2960 Switch yang terhubung pada 4 komputer.

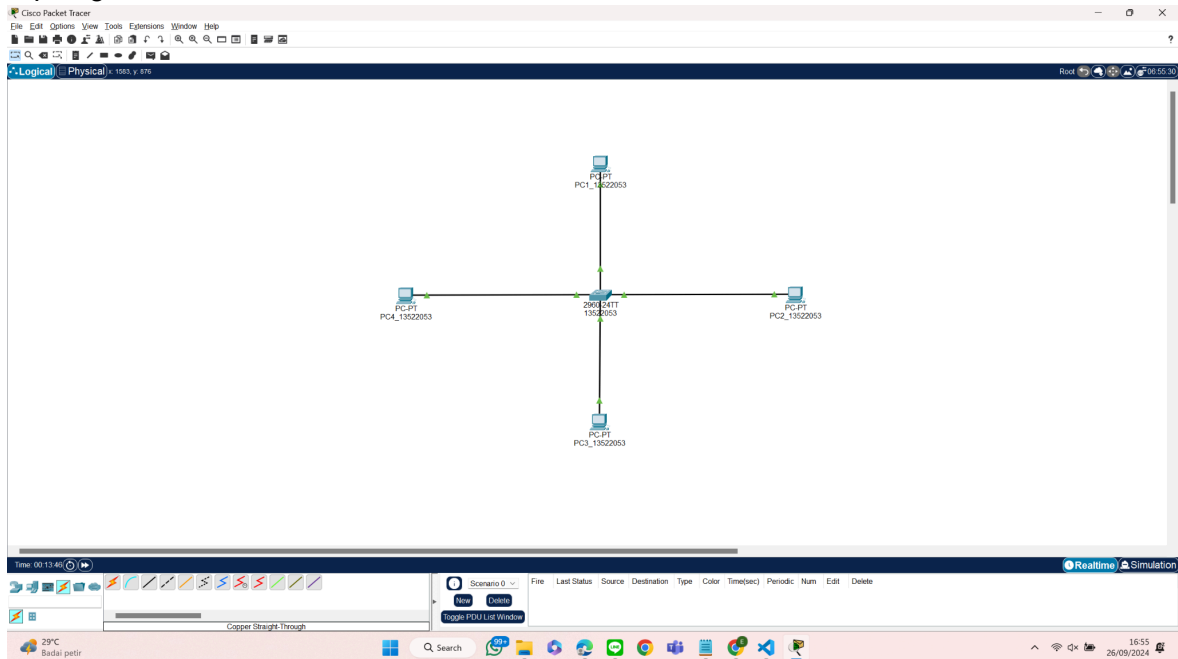
Gunakan *display name* dan *IP address* pada setiap perangkat, urutan penomoran PC dibebaskan:

- Switch\_<NIM>: -
- PC1\_<NIM>: 192.168.1.1, subnet mask 255.255.255.0
- PC2\_<NIM>: 192.168.1.2, subnet mask 255.255.255.0
- PC3\_<NIM>: 192.168.1.3, subnet mask 255.255.255.0
- PC4\_<NIM>: 192.168.1.4, subnet mask 255.255.255.0

Kemudian coba lakukan *ping* PC4 dari PC1 dan amati langkah-langkah yang terjadi seperti penjelasan di atas.

Lampirkan topologi dan langkah-langkah *ping* pada kolom di bawah (gunakan [fitur simulasi packet tracer](#)!). Selain itu, jelaskan apa yang terjadi pada setiap langkah pada *ping*.

## A Topologi Star



Langkah-langkah ping:

1. Buka command prompt PC1
2. Run simulation
3. Ketik "ping 192.168.1.4"

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:

Reply from 192.168.1.4: bytes=32 time=8ms TTL=128
Reply from 192.168.1.4: bytes=32 time=4ms TTL=128
Reply from 192.168.1.4: bytes=32 time=4ms TTL=128
Reply from 192.168.1.4: bytes=32 time=4ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 8ms, Average = 5ms

C:\>
```

#### 4. Amati pergerakan message “ping” dari PC1 hingga mencapai PC4

The image displays two screenshots of the Cisco Packet Tracer interface, illustrating the movement of a ping message from PC1 to PC4.

**Top Screenshot:**

- Network Topology:** A central switch (Switch\_13522053) is connected to four PCs: PC1\_13522053 (top), PC2\_13522053 (right), PC3\_13522053 (bottom), and PC4\_13522053 (left).
- Event List:** The list shows the progression of the ping message from PC1 to PC4, with the last event occurring at 3.999 seconds.
- Simulation Panel:** The simulation is running at 3.999 seconds.

**Bottom Screenshot:**

- Network Topology:** The same network topology is shown.
- Event List:** The list shows the completion of the ping message, with the last event occurring at 4.000 seconds.
- Simulation Panel:** The simulation is running at 4.000 seconds.

Cisco Packet Tracer

File Edit Options View Tools Extensions Window Help

Logical Physical

Simulation Panel

Event List

Via	Time(sec)	Last Device	All Device	Type
1.997	-	-	Switch_13522053	STP
1.998	Switch_13522053	PC1_13522053	PC1_13522053	STP
1.998	Switch_13522053	PC2_13522053	PC2_13522053	STP
1.998	Switch_13522053	PC3_13522053	PC3_13522053	STP
1.998	Switch_13522053	PC4_13522053	PC4_13522053	STP
3.996	-	-	Switch_13522053	STP
3.996	-	-	PC1_13522053	ICMP
3.996	-	-	PC1_13522053	ARP
3.997	Switch_13522053	PC1_13522053	PC1_13522053	STP
3.997	Switch_13522053	PC2_13522053	PC2_13522053	STP
3.997	Switch_13522053	PC3_13522053	PC3_13522053	STP
3.997	Switch_13522053	PC4_13522053	PC4_13522053	STP
3.997	PC1_13522053	Switch_13522053	Switch_13522053	ARP
3.998	Switch_13522053	PC2_13522053	PC2_13522053	ARP
3.998	Switch_13522053	PC3_13522053	PC3_13522053	ARP
3.998	Switch_13522053	PC4_13522053	PC4_13522053	ARP
3.999	PC4_13522053	Switch_13522053	Switch_13522053	ARP
4.000	Switch_13522053	PC1_13522053	PC1_13522053	ARP
4.000	-	-	PC1_13522053	ICMP
4.001	PC1_13522053	Switch_13522053	Switch_13522053	ICMP

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, PING, SANAP, SET, BT, LACP, LLDP, MMR, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAPI, POP3, PPP, PPPoE, PTP, RADIUS, REP, RFP, RFPing, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TFTP, TFTP, Tunnel, UDP, USB, VTY

Event List

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, PING, SANAP, SET, BT, LACP, LLDP, MMR, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAPI, POP3, PPP, PPPoE, PTP, RADIUS, REP, RFP, RFPing, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TFTP, TFTP, Tunnel, UDP, USB, VTY

Cisco Packet Tracer

File Edit Options View Tools Extensions Window Help

Logical Physical

Simulation Panel

Event List

Via	Time(sec)	Last Device	All Device	Type
1.997	-	-	Switch_13522053	STP
1.998	Switch_13522053	PC1_13522053	PC1_13522053	STP
1.998	Switch_13522053	PC2_13522053	PC2_13522053	STP
1.998	Switch_13522053	PC3_13522053	PC3_13522053	STP
1.998	Switch_13522053	PC4_13522053	PC4_13522053	STP
3.996	-	-	Switch_13522053	STP
3.996	-	-	PC1_13522053	ICMP
3.996	-	-	PC1_13522053	ARP
3.997	Switch_13522053	PC1_13522053	PC1_13522053	STP
3.997	Switch_13522053	PC2_13522053	PC2_13522053	STP
3.997	Switch_13522053	PC3_13522053	PC3_13522053	STP
3.997	Switch_13522053	PC4_13522053	PC4_13522053	STP
3.997	PC1_13522053	Switch_13522053	Switch_13522053	ARP
3.998	Switch_13522053	PC2_13522053	PC2_13522053	ARP
3.998	Switch_13522053	PC3_13522053	PC3_13522053	ARP
3.998	Switch_13522053	PC4_13522053	PC4_13522053	ARP
3.999	PC4_13522053	Switch_13522053	Switch_13522053	ARP
4.000	Switch_13522053	PC1_13522053	PC1_13522053	ARP
4.000	-	-	PC1_13522053	ICMP
4.001	PC1_13522053	Switch_13522053	Switch_13522053	ICMP
4.002	Switch_13522053	PC4_13522053	PC4_13522053	ICMP

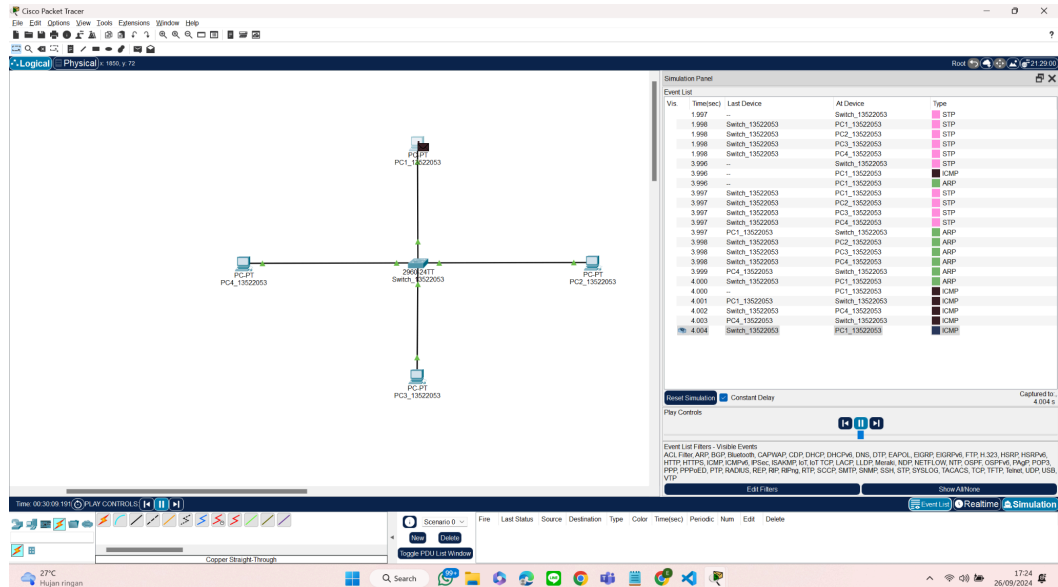
Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, PING, SANAP, SET, BT, LACP, LLDP, MMR, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAPI, POP3, PPP, PPPoE, PTP, RADIUS, REP, RFP, RFPing, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TFTP, TFTP, Tunnel, UDP, USB, VTY

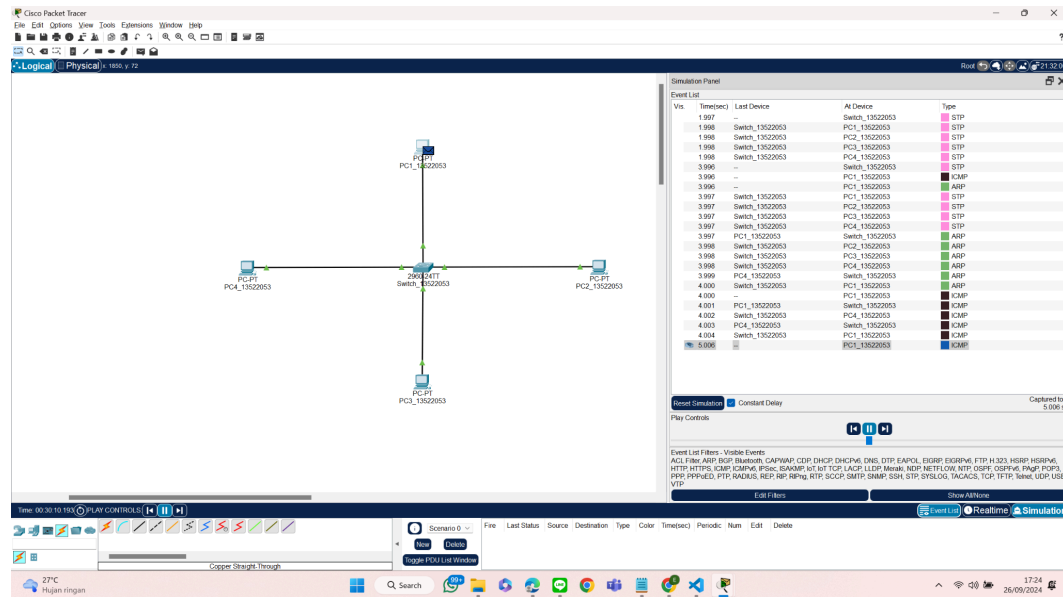
Event List

Event List Filters - Visible Events

ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, PING, SANAP, SET, BT, LACP, LLDP, MMR, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAPI, POP3, PPP, PPPoE, PTP, RADIUS, REP, RFP, RFPing, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TFTP, TFTP, Tunnel, UDP, USB, VTY







5. Lihat list tipe network protocol pada list yang tertera di tabel panel simulasi

Event List				
Vis	Time(sec)	Last Device	At Device	Type
	3.997	PC1_13522053	Switch_13522053	ARP
	3.998	Switch_13522053	PC2_13522053	ARP
	3.998	Switch_13522053	PC3_13522053	ARP
	3.998	Switch_13522053	PC4_13522053	ARP
	3.999	PC4_13522053	Switch_13522053	ARP
	4.000	Switch_13522053	PC1_13522053	ARP
	4.000	--	PC1_13522053	ICMP
	4.001	PC1_13522053	Switch_13522053	ICMP
	4.002	Switch_13522053	PC4_13522053	ICMP
	4.003	PC4_13522053	Switch_13522053	ICMP
	4.004	Switch_13522053	PC1_13522053	ICMP

### IV.3. LAN dan Subnetting

LAN adalah singkatan dari *Local Area Network*. Secara formal, LAN didefinisikan sebagai jaringan dengan satu [broadcast domain](#), yang merupakan area atau grup dari *nodes* yang menerima *broadcast* yang sama dari perangkat jaringan pada **data link layer**. Berdasarkan definisi tersebut, **perangkat dalam sebuah jaringan tidak dapat berkomunikasi dengan perangkat dalam jaringan lain tanpa [routing](#)**. Semua jaringan memiliki beberapa nomor-nomor penting, salah satu yang terutama adalah **network address** dan **subnet mask**.

**Network address** terdiri dari **IP address** yang digunakan perangkat-perangkat dan jaringan lain untuk mengidentifikasi sebuah jaringan. Namun, *IP address* tidak memberikan informasi yang cukup terkait "bagian" jaringan mana yang dirujuk oleh nomor tersebut, sebuah informasi tambahan diperlukan untuk membedakan *network address* dari *device address*. Sumber informasi tambahan ini dikenal sebagai **subnet mask**. Sederhananya, **subnet mask** adalah

nomor dengan panjang 32-bit (pada IPv4) yang memungkinkan perangkat untuk membedakan bagian dari *IP address* yang merupakan bagian dari jaringan, dan yang merupakan bagian dari perangkat dalam jaringan tersebut.

Sebagai contoh, sebuah komputer yang terhubung pada jaringan memiliki *IP address* 192.168.100.1, dan *subnet mask* 255.255.255.0. Untuk mempermudah penulisan, digunakan notasi *IP address* **192.168.100.1/24** yang disebut *CIDR notation*. Untuk mendapatkan *network address* dari alamat tersebut, gunakan **operator AND** pada kedua nomor. Operasi tersebut menghasilkan 192.168.100.0. Semua alamat, **dan hanya** alamat dari 192.168.100.1 hingga 192.168.100.254 merupakan perangkat yang terhubung pada jaringan tersebut (bagaimana dengan 192.168.100.0/24 & 192.168.100.255/24?).

Terdapat nomor penting lain, yaitu ***broadcast address***, yang dapat digunakan untuk mengirimkan pesan kepada semua perangkat lain dalam jaringan yang sama, yaitu pada *broadcast domain* yang sama. Nomor ini adalah alamat terakhir dalam jaringan. Pada contoh di atas, *broadcast address* jaringan tersebut adalah 192.168.100.255.

#### Tugas 4

Q Pada Cisco Packet Tracer, buat dua topologi *star*, masing-masing terdiri dari satu 2960 Switch yang terhubung ke tiga komputer. Hint: gunakan ulang dengan meng-*copy* dan sesuaikan hasil dari tugas sebelumnya.

Untuk topologi pertama, gunakan *display name* dan *IP address* berikut untuk masing-masing perangkat. Urutan penomoran PC dibebaskan: (**IP address terakhir bukan salah ketik**)

- SwitchA\_<NIM>: -
- PC1A\_<NIM>: 192.168.1.1, subnet mask 255.255.255.0
- PC2A\_<NIM>: 192.168.1.2, subnet mask 255.255.255.0
- PC3A\_<NIM>: 192.168.2.3, subnet mask 255.255.255.0

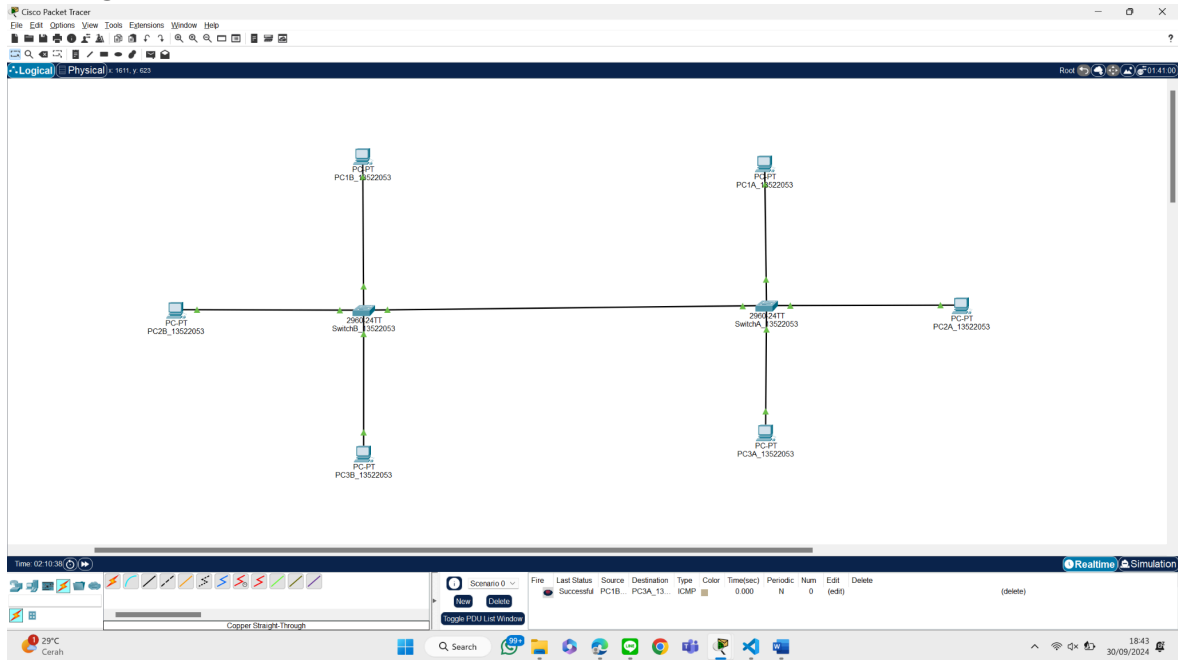
Untuk topologi ke-dua, gunakan *display name* dan *IP address* berikut untuk masing-masing perangkat. Urutan penomoran PC dibebaskan: (**IP address terakhir bukan salah ketik**)

- SwitchB\_<NIM>: -
- PC1B\_<NIM>: 192.168.2.1, subnet mask 255.255.255.0
- PC2B\_<NIM>: 192.168.2.2, subnet mask 255.255.255.0
- PC3B\_<NIM>: 192.168.1.3, subnet mask 255.255.255.0

Kemudian, hubungkan SwitchA dan SwitchB pada *port* FastEthernet masing-masing.

Lampirkan topologinya, kemudian coba ping semua PC dari PC1A dan PC1B. Apa yang terjadi? Lampirkan hasilnya dan jelaskan!

## A Topologi



### PC1A

#### 1. PC1A → PC2A

```
PC1A_13522053
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time=9ms TTL=128
Reply from 192.168.1.2: bytes=32 time=4ms TTL=128
Reply from 192.168.1.2: bytes=32 time=4ms TTL=128
Reply from 192.168.1.2: bytes=32 time=4ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 9ms, Average = 5ms
```

#### 2. PC1A → PC3A

```
C:\>ping 192.168.2.3

Pinging 192.168.2.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

#### 3. PC1A → PC1B

```
C:\>ping 192.168.2.1

Pinging 192.168.2.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.2.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

#### 4. PC1A → PC2B

```
C:\>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

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

#### 5. PC1A → PC3B

```
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
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

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

### PC1B

#### 1. PC1B → PC2B

```
C:\>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time<1ms TTL=128
Reply from 192.168.2.2: bytes=32 time=1ms TTL=128
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

#### 2. PC1B → PC3B

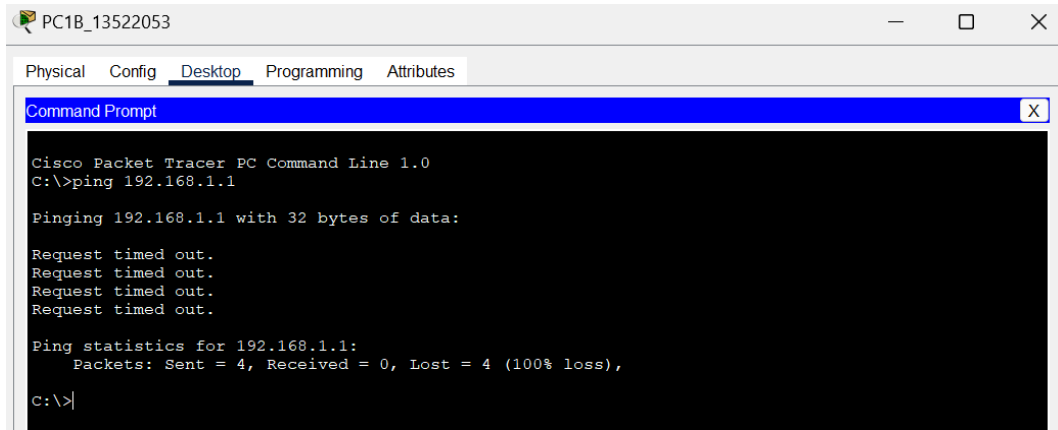
```
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

#### 3. PC1B → PC1A



```
PC1B_13522053
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.1

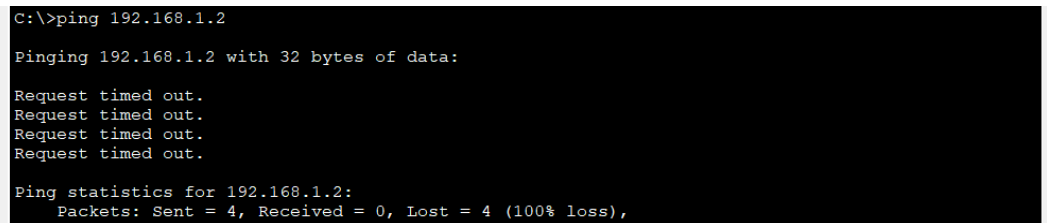
Pinging 192.168.1.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

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

C:\>|
```

#### 4. PC1B → PC2A



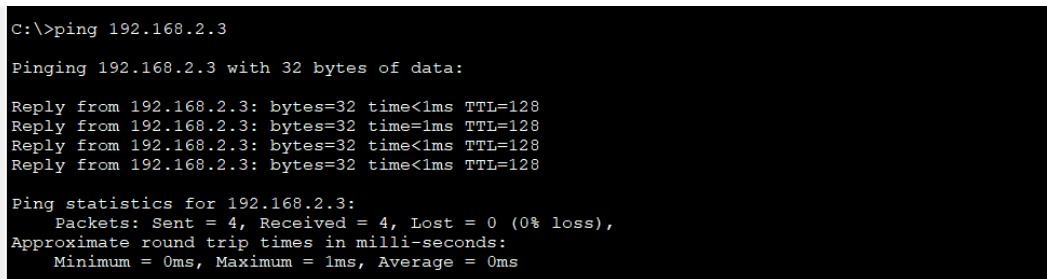
```
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

#### 5. PC1B → PC3A



```
C:\>ping 192.168.2.3

Pinging 192.168.2.3 with 32 bytes of data:

Reply from 192.168.2.3: bytes=32 time<1ms TTL=128
Reply from 192.168.2.3: bytes=32 time<1ms TTL=128
Reply from 192.168.2.3: bytes=32 time<1ms TTL=128
Reply from 192.168.2.3: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

PC1A dengan IP Address 192.168.1.1 hanya bisa berkomunikasi dengan PC yang memiliki IP Address 192.168.1.X. Yang bisa berubah-ubah hanya 8 bit terakhir saja karena terdapat /24 dimana batas IP Address ada di 24 bit yaitu 192.168.1. Maka PC1A tidak bisa berkomunikasi dengan PC yang memiliki IP Address 192.168.2.X karena sudah berbeda subnet. PC yang bisa berkomunikasi dengan PC1A adalah PC2A (192.168.1.2) dan PC3B (192.168.1.3).

PC1B dengan IP Address 192.168.2.1 hanya bisa berkomunikasi dengan PC yang memiliki IP Address 192.168.2.X. Yang bisa berubah-ubah hanya 8 bit terakhir saja karena terdapat /24 dimana batas IP Address ada di 24 bit yaitu 192.168.2. Maka PC1B tidak bisa berkomunikasi dengan PC yang memiliki IP Address 192.168.1.X karena sudah berbeda subnet. PC yang bisa berkomunikasi dengan PC1B adalah PC3A (192.168.2.3) dan PC2B (192.168.2.2).

Dengan memanfaatkan *isolation property* dari jaringan dan *subnet mask*, rentang *IP address* dapat dibagi menjadi jaringan-jaringan yang lebih kecil melalui **subnetting**. Sebagai contoh, dengan rentang *IP addresses* 192.168.100.0-192.168.100.255, dapat digunakan **subnet mask**

untuk membagi jaringan tersebut menjadi dua jaringan terpisah yang lebih kecil. seperti *sub network* 192.168.100.0/25 dan 192.168.100.128/25.

### Tugas 5

Q Seperti tugas sebelumnya, buat dua topologi *star*, masing-masing terdiri dari satu 2960 Switch yang terhubung ke tiga komputer. Hint: gunakan ulang dan sesuaikan hasil dari tugas sebelumnya.

Untuk topologi pertama, gunakan *display name* dan *IP address* berikut untuk masing-masing perangkat. Urutan penomoran PC dibebaskan: (**IP address terakhir bukan salah ketik**)

- SwitchA\_<NIM>: -
- PC1A\_<NIM>: 192.168.1.1, subnet mask 255.255.255.0
- PC2A\_<NIM>: 192.168.1.2, subnet mask 255.255.255.0
- PC3A\_<NIM>: 192.168.1.129, subnet mask 255.255.255.0

Untuk topologi ke-dua, gunakan *display name* dan *IP address* berikut untuk masing-masing perangkat. Urutan penomoran PC dibebaskan: (**IP address terakhir bukan salah ketik**)

- SwitchB\_<NIM>: -
- PC1B\_<NIM>: 192.168.1.3, subnet mask 255.255.255.0
- PC2B\_<NIM>: 192.168.1.4, subnet mask 255.255.255.0
- PC3B\_<NIM>: 192.168.1.130, subnet mask 255.255.255.0

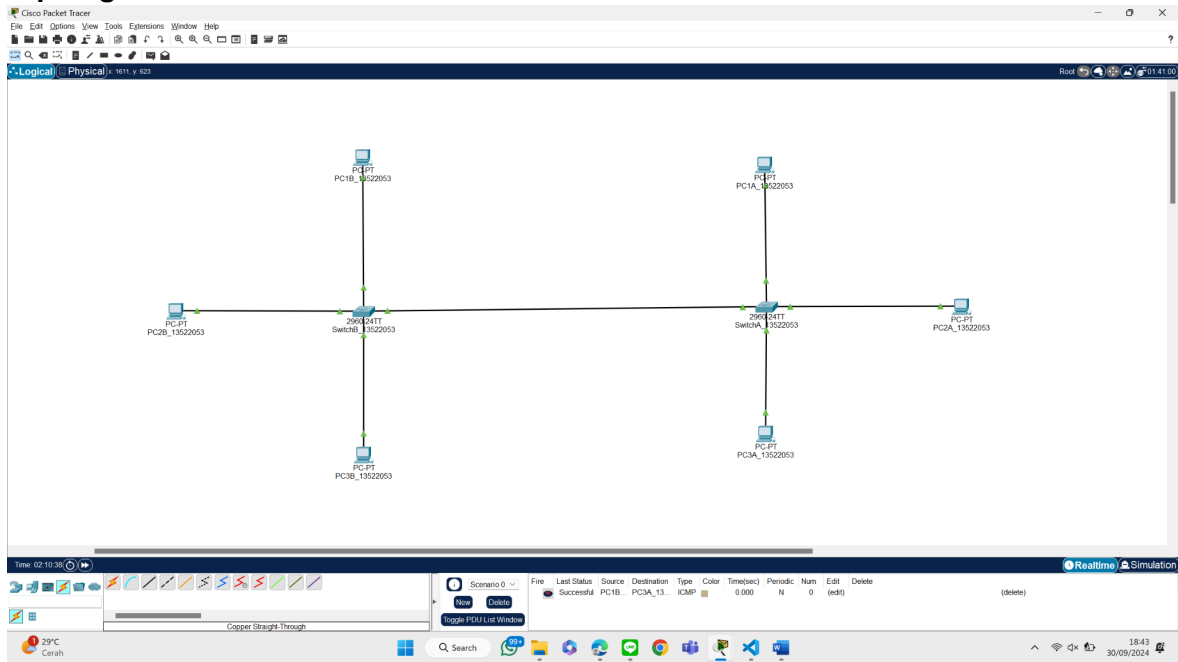
Kemudian, hubungkan SwitchA dan SwitchB pada *port* FastEthernet masing-masing.

Lampirkan topologinya, kemudian coba lakukan ping semua PC dari PC1A dan PC1B. Apa yang terjadi? Lampirkan hasilnya dan jelaskan!

Kemudian, ganti *subnet mask* semua PC menjadi 255.255.255.128.

Coba lakukan ping semua PC dari PC1A dan PC1B. Apa yang terjadi? Lampirkan hasilnya dan jelaskan!

## A Topologi



**Subnet mask 255.255.255.0**

**PC1A**

1. PC1A → PC2A

```
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

2. PC1A → PC3A

```
C:\>ping 192.168.1.129

Pinging 192.168.1.129 with 32 bytes of data:

Reply from 192.168.1.129: bytes=32 time<1ms TTL=128
Reply from 192.168.1.129: bytes=32 time<1ms TTL=128
Reply from 192.168.1.129: bytes=32 time<1ms TTL=128
Reply from 192.168.1.129: bytes=32 time=8ms TTL=128

Ping statistics for 192.168.1.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 8ms, Average = 2ms
```

### 3. PC1A → PC1B

```
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
Reply from 192.168.1.3: bytes=32 time=17ms 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

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 17ms, Average = 4ms
```

### 4. PC1A → PC2B

```
C:\>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:

Reply from 192.168.1.4: bytes=32 time=1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

### 5. PC1A → PC3B

```
C:\>ping 192.168.1.130

Pinging 192.168.1.130 with 32 bytes of data:

Reply from 192.168.1.130: bytes=32 time<1ms TTL=128
Reply from 192.168.1.130: bytes=32 time<1ms TTL=128
Reply from 192.168.1.130: bytes=32 time=1ms TTL=128
Reply from 192.168.1.130: bytes=32 time=1ms TTL=128

Ping statistics for 192.168.1.130:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

## PC1B

### 1. PC1B → PC2B

```
C:\>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:

Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

### 2. PC1B → PC3B

```
C:\>ping 192.168.1.130

Pinging 192.168.1.130 with 32 bytes of data:

Reply from 192.168.1.130: bytes=32 time<1ms TTL=128
Reply from 192.168.1.130: bytes=32 time<1ms TTL=128
Reply from 192.168.1.130: bytes=32 time=17ms TTL=128
Reply from 192.168.1.130: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.130:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 17ms, Average = 4ms
```



### 3. PC1B → PC1A

```
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=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
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 = 0ms, Average = 0ms
```

### 4. PC1B → PC2A

```
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time=1ms TTL=128
Reply from 192.168.1.2: bytes=32 time=1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

### 5. PC1B → PC3A

```
C:\>ping 192.168.1.129

Pinging 192.168.1.129 with 32 bytes of data:

Reply from 192.168.1.129: bytes=32 time<1ms TTL=128
Reply from 192.168.1.129: bytes=32 time=2ms TTL=128
Reply from 192.168.1.129: bytes=32 time=1ms TTL=128
Reply from 192.168.1.129: bytes=32 time<1ms TTL=128

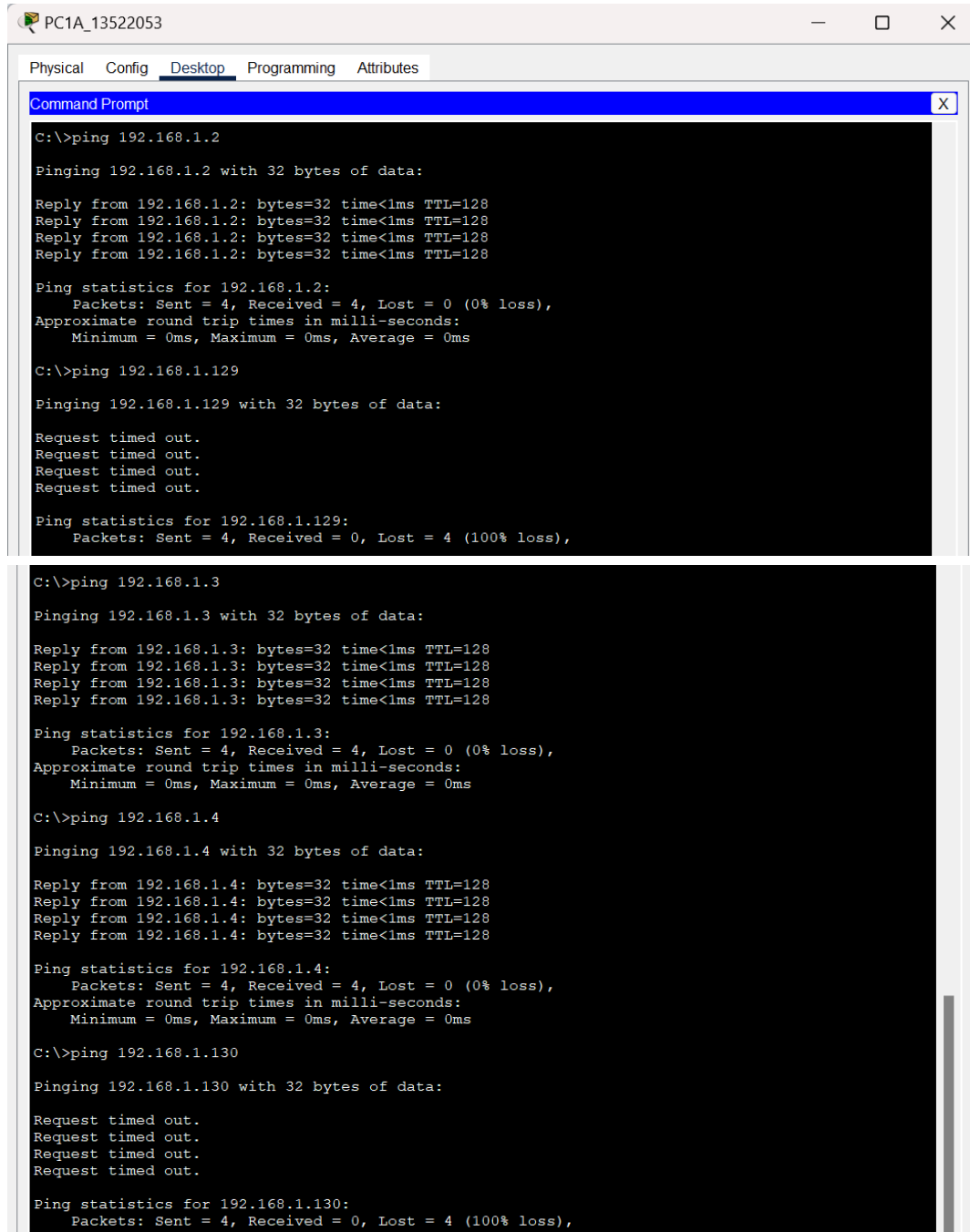
Ping statistics for 192.168.1.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms
```

Baik dari PC1A maupun PC1B dapat berkomunikasi dengan PC manapun karena semua PC memiliki subnet yang sama yaitu 192.168.1.X

### Subnet mask 255.255.255.128

#### PC1A (dokumentasi sesuai urutan)

1. PC1A → PC2A
2. PC1A → PC3A
3. PC1A → PC1B
4. PC1A → PC2B
5. PC1A → PC3B



The screenshot shows a Windows Command Prompt window titled "PC1A\_13522053" with tabs for Physical, Config, Desktop, Programming, and Attributes. The "Desktop" tab is active, and a "Command Prompt" window is open. The command prompt shows the following output:

```
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.129

Pinging 192.168.1.129 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.129:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

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
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

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.4

Pinging 192.168.1.4 with 32 bytes of data:

Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.130

Pinging 192.168.1.130 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.130:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

### PC1B (dokumentasi sesuai urutan)

1. PC1B → PC2B
2. PC1B → PC3B
3. PC1B → PC1A
4. PC1B → PC2A
5. PC1B → PC3A

```
PC1B_13522053
Physical Config Desktop Programming Attributes
Command Prompt
C:\>ping 192.168.1.4
Pinging 192.168.1.4 with 32 bytes of data:

Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128
Reply from 192.168.1.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 192.168.1.130
Pinging 192.168.1.130 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.130:
    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<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
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 = 1ms, Average = 0ms

C:\>ping 192.168.1.2
Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 192.168.1.129
Pinging 192.168.1.129 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.129:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Baik PC1A maupun PC1B dapat berkomunikasi dengan semua PC lain kecuali PC3A dan PC3B karena subnet terbesar untuk subnet mask 255.255.255.128 pada kasus ini adalah 192.168.1.127. Karena PC3A memiliki IP Address 192.168.1.129 dan PC3B memiliki IP Address 192.168.1.130 yang berarti subnet nya sudah berbeda dengan subnet PC lain sehingga PC1A dan PC1B tidak bisa berkomunikasi dengan kedua PC tersebut.

## IV.4. VLAN

VLAN adalah singkatan dari *Virtual Local Area Network*, yang berfungsi persis seperti LAN. Dengan VLAN, pada satu infrastruktur perangkat keras jaringan yang sama, yang bisa saja terletak pada lokasi fisik yang sama dapat dibagi menjadi beberapa LAN yang terpisah secara *logical*. Hal ini dapat digunakan untuk menggabungkan perangkat-perangkat dengan lokasi fisik berbeda-beda ke dalam sebuah jaringan tertentu tanpa harus membuat infrastruktur jaringan yang rumit (terdapat beberapa *use case* lain dari VLAN - silakan cari tahu).

VLAN dijalankan pada **data link layer**, dan bekerja dengan memberi **tag** pada **network frames**. Untuk membuat VLAN, dibutuhkan *managed switch* (seperti Cisco 2950 dan 2960). Manajemen *tag* dari *frame* dilakukan oleh switch, dan pengguna hanya perlu melakukan konfigurasi *grouping port* pada switch.

### Tugas 6

Q Pada Cisco Packet Tracer, buat sebuah topologi *star*, masing-masing terdiri dari satu 2960 Switch yang terhubung ke empat komputer. Hint: gunakan ulang dan sesuaikan hasil dari tugas sebelumnya.(cek [Tugas 3](#))

Gunakan *display name* dan *IP address* pada setiap perangkat, urutan penomoran PC dibebaskan:

- Switch\_<NIM>: -
- PC1\_<NIM>: 192.168.1.1, subnet mask 255.255.255.0
- PC2\_<NIM>: 192.168.1.2, subnet mask 255.255.255.0
- PC3\_<NIM>: 192.168.1.3, subnet mask 255.255.255.0
- PC4\_<NIM>: 192.168.1.4, subnet mask 255.255.255.0

Kali ini, hubungkan PC pada *port* FastEthernet switch sesuai dengan nomornya masing-masing (misal PC1 menuju *port* FastEthernet0/1). **Ini hanya membantu mengurangi kebingungan pada langkah-langkah selanjutnya. Silakan gunakan *port* lain sesuka hati!**

Kemudian, kosongkan *MAC address table* pada **switch** menggunakan `clear mac-address-table.`

Setelah mengosongkan *MAC address table*, tampilkan *MAC address table* saat ini menggunakan `show mac-address-table.`

**Tampilkan *MAC address table* dan jelaskan apa yang terjadi pada kolom di bawah ini.**

Kemudian, lakukan *ping* setiap PC dari PC1, kemudian tampilkan *MAC address table*.

Tampilkan *MAC address table* dan jelaskan apa yang terjadi pada kolom di bawah ini.

A MAC address table saat awal

```
Andini#clear mac-address-table
Andini#show mac-address-table
      Mac Address Table
-----
Vlan    Mac Address      Type    Ports
----    -
Andini#
```

Copy

Paste

Tabel mac address dalam posisi kosong karena telah diclear sebelumnya

MAC address table setelah melakukan ping dari PC1 ke semua PC lain

```
Andini#show mac-address-table
      Mac Address Table
-----
Vlan    Mac Address      Type    Ports
----    -
      1    0001.641d.9ea7    DYNAMIC Fa0/3
      1    0002.17d8.edb2    DYNAMIC Fa0/1
      1    000c.cf1e.883b    DYNAMIC Fa0/4
      1    0040.0b83.8459    DYNAMIC Fa0/2
Andini#
```

Copy

Paste

MAC address dari tiap PC sudah terdaftar di dalam MAC address table karena saat PC1 melakukan ping ke masing-masing PC, switch mendaftarkan MAC address dari masing-masing PC sehingga MAC address masing-masing muncul di table

Q Setelah menyiapkan topologi perangkat keras, saatnya mengkonfigurasi VLAN. Konfigurasi VLAN untuk memisahkan PC1 & PC2 ke dalam satu VLAN, dan PC3 & PC4 ke dalam VLAN lain.

Lakukan ini dengan mengkonfigurasi *interface (port)* yang akan dikelompokkan ke dalam VLAN.

Hint: akses mode *interface configuration* menggunakan  
interface [range] ... dan gunakan konfigurasi switchport

Tampilkan konfigurasi VLAN!

Hint: tampilkan konfigurasi VLAN menggunakan  
show VLAN

Lakukan *ping* ke semua PC dari PC1, kemudian lakukan hal yang sama dari PC4. **Apa yang terjadi? Jelaskan pada kolom jawaban di bawah. Kemudian, tampilkan *MAC address table* dari switch!**

## A Konfigurasi VLAN saat awal setelah disetting

```
Andini#show vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/5, Fa0/6, 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
2	VLAN0002	active	Fa0/1, Fa0/2
3	VLAN0003	active	Fa0/3, Fa0/4
1002	fddi-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
1	enet	100001	1500	-	-	-	-	-	0	0
2	enet	100002	1500	-	-	-	-	-	0	0
3	enet	100003	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0

--More--

Copy

Paste

### Ping dari PC1 ke PC2, PC3, PC4

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1_...	PC2_135...	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC1_...	PC3_135...	ICMP		0.000	N	1	(edit)	(delete)
	Failed	PC1_...	PC4_135...	ICMP		0.000	N	2	(edit)	(delete)

Ping dari PC1 hanya berhasil ke PC2 karena PC1 dan PC2 berada di 1 VLAN yaitu VLAN0002, sedangkan PC3 dan PC4 berada di VLAN yang berbeda dengan PC1 sehingga PC1 tidak bisa berkomunikasi dengan P3 dan P4.

### Ping dari PC4 ke PC1, PC2, PC3

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Failed	PC4_...	PC1_135...	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC4_...	PC2_135...	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC4_...	PC3_135...	ICMP		0.000	N	2	(edit)	(delete)

Ping dari PC4 hanya berhasil ke PC3 karena PC4 dan PC3 berada di 1 VLAN yaitu VLAN0003, sedangkan PC1 dan PC2 berada di VLAN yang berbeda dengan PC4 sehingga PC4 tidak bisa berkomunikasi dengan P1 dan P2.

### MAC Address Table setelah proses ping

```
Andini#show mac-address-table
Mac Address Table
```

Vlan	Mac Address	Type	Ports
2	0002.17d8.edb2	DYNAMIC	Fa0/1
2	0040.0b83.8459	DYNAMIC	Fa0/2
3	0001.641d.9ea7	DYNAMIC	Fa0/3
3	000c.cf1e.883b	DYNAMIC	Fa0/4

```
Andini#
```

Copy

Paste

Terkadang, infrastruktur sebuah jaringan terdiri dari berbagai *host* yang terhubung pada sebuah infrastruktur jaringan, dan sebuah switch tidak mampu menangani jumlah port yang dibutuhkan untuk menghubungkan perangkat-perangkat yang ada. Untuk mengatasi ini, dapat digunakan *bridge* (yang tidak akan digunakan untuk kegiatan *lab*), *backbone* switch (yang merupakan switch yang digunakan untuk menghubungkan switch-switch lain) yang menghubungkan *edge* switch, atau menghubungkan switch-switch yang ada (contohnya ring topology).

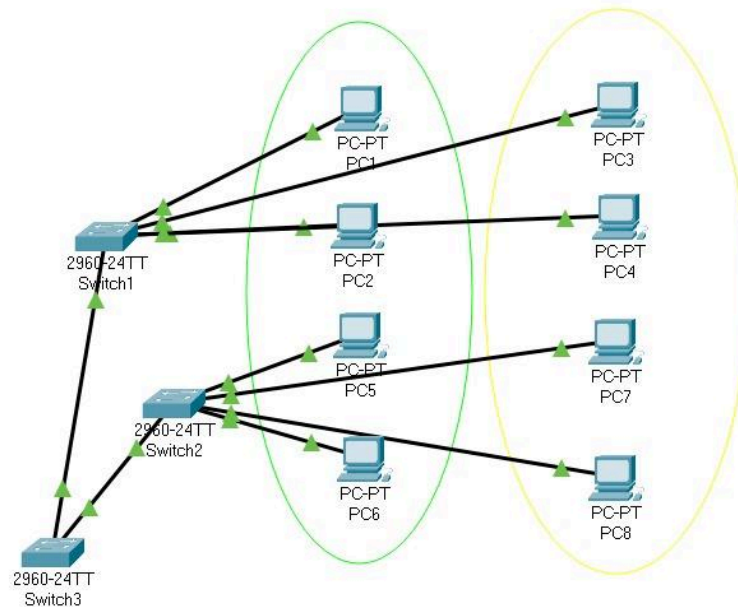
Karena tag VLAN diletakkan pada **frames**, kita dapat melakukan ini dengan mengkonfigurasi grup LAN pada switch yang memungkinkan PDU dengan tag tersebut agar dapat melalui *interface* yang tepat (belum tentu *port* karena akan diperkenalkan *trunk*)

### Tugas 7

Q Pada Cisco Packet Tracer, buat dua topologi *star*, masing-masing terdiri dari satu 2960 Switch yang terhubung ke empat komputer.(seperti pada [Tugas 6](#)), kemudian hubungkan kedua switch dengan 2960 Switch lain.

Kali ini, gunakan *display name* dan *IP address* berikut pada setiap perangkat, urutan penomoran PC dibebaskan:

- Switch1\_<NIM>: -
- PC1\_<NIM>: 192.168.1.1, subnet mask 255.255.255.0
- PC2\_<NIM>: 192.168.1.2, subnet mask 255.255.255.0
- PC3\_<NIM>: 192.168.1.3, subnet mask 255.255.255.0
- PC4\_<NIM>: 192.168.1.4, subnet mask 255.255.255.0
  
- Switch2\_<NIM>: -
- PC5\_<NIM>: 192.168.1.5, subnet mask 255.255.255.0
- PC6\_<NIM>: 192.168.1.6, subnet mask 255.255.255.0
- PC7\_<NIM>: 192.168.1.7, subnet mask 255.255.255.0
- PC8\_<NIM>: 192.168.1.8, subnet mask 255.255.255.0
  
- Switch3\_<NIM>: -



Sama seperti tugas sebelumnya, hubungkan PC1-4 menuju *port* FastEthernet Switch1 (PC1 menuju *port* FastEthernet0/1). Kemudian lakukan hal yang serupa pada PC5-8 di Switch2 (PC5 to *port* FastEthernet0/1, dan seterusnya) **Seperti pada tugas sebelumnya, penggunaan port dibebaskan**. Setelah menghubungkan PCs pada masing-masing switchnya, hubungkan kedua switch yang menghubungkan PC-PC ke Switch3,

Setelah menyiapkan topologi perangkat keras, saatnya mengkonfigurasi VLAN. Konfigurasi VLAN untuk memisahkan PC1, PC2, PC5, & PC6 ke dalam satu VLAN, dan PC3, PC4, PC7, & PC8 ke dalam VLAN lain. Silakan lakukan konfigurasi pada switch manapun yang anda perlukan.

Menghubungkan switch dengan switch lain dapat dilakukan dengan menghubungkan beberapa kabel untuk setiap grup VLAN, atau menggunakan [trunking](#) (ini akan sangat membantu pada aktivitas lab-lab berikutnya, sehingga **sangat disarankan** untuk dicoba)

Lakukan ini dengan mengkonfigurasi *interface (port)* yang akan dikelompokkan ke dalam VLAN.

Hint: akses mode *interface configuration* menggunakan  
`interface [range] ...` dan gunakan konfigurasi `switchport`

**Tampilkan konfigurasi VLAN! (jika menggunakan trunking, tampilkan juga konfigurasi trunking)**



Hint: tampilkan konfigurasi VLAN menggunakan

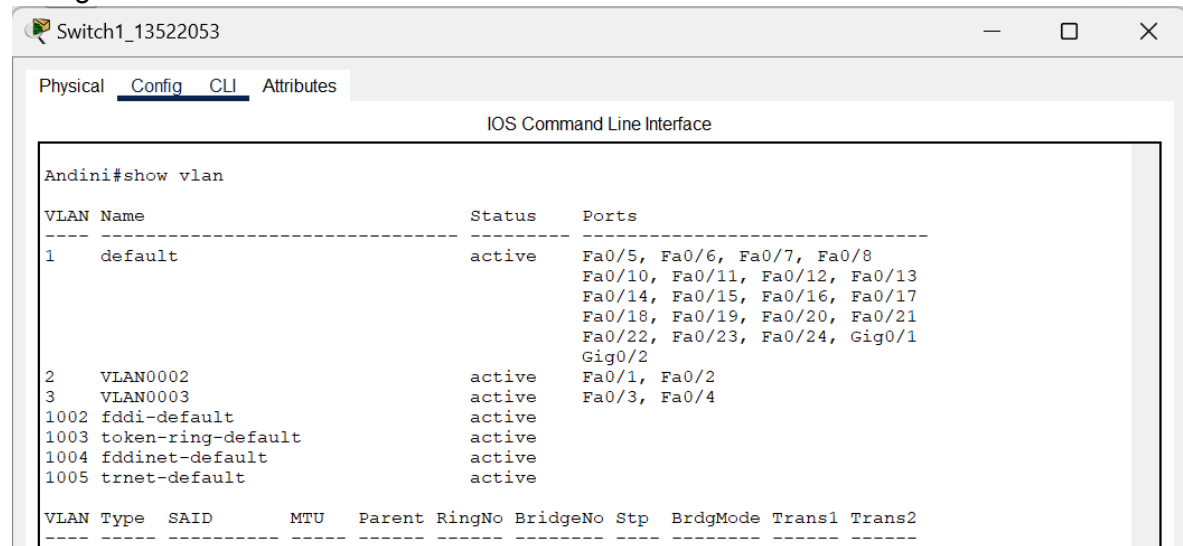
`show VLAN`

Hint: tampilkan konfigurasi trunking menggunakan

`show interfaces trunk`

Lakukan *ping* ke semua PC dari PC1, kemudian lakukan hal yang sama dari PC8. **Apa yang terjadi? Jelaskan pada kolom jawaban di bawah. Kemudian, tampilkan *MAC address table* dari semua switch!**

#### A Konfigurasi VLAN Switch1



Switch1\_13522053

Physical Config CLI Attributes

IOS Command Line Interface

```
Andini#show vlan
```

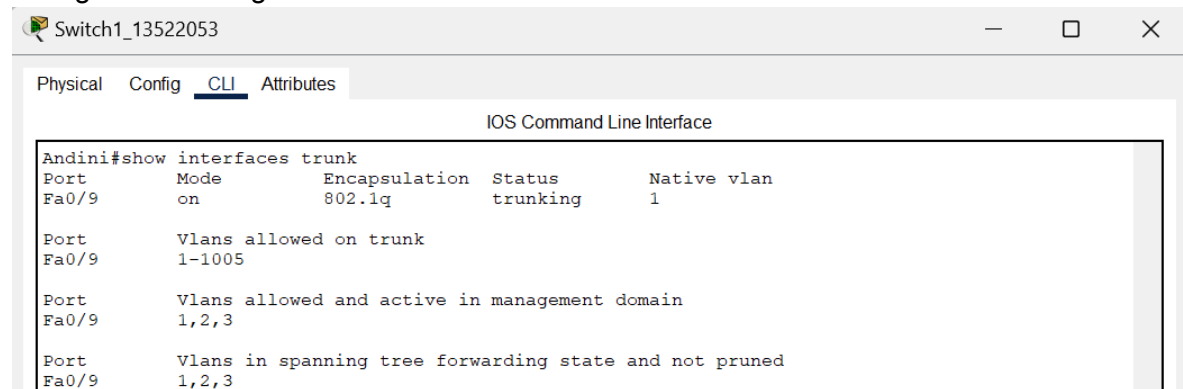
VLAN	Name	Status	Ports
1	default	active	Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/10, Fa0/11, Fa0/12, Fa0/13 Fa0/14, Fa0/15, Fa0/16, Fa0/17 Fa0/18, Fa0/19, Fa0/20, Fa0/21 Fa0/22, Fa0/23, Fa0/24, Gig0/1 Gig0/2
2	VLAN0002	active	Fa0/1, Fa0/2
3	VLAN0003	active	Fa0/3, Fa0/4
1002	fddi-default	active	
1003	token-ring-default	active	
1004	fddinet-default	active	
1005	trnet-default	active	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1										
2										
3										
1002										
1003										
1004										
1005										

VLAN0002 berisi port untuk PC1 dan PC2

VLAN0003 berisi port untuk PC3 dan PC4

#### Konfigurasi trunking Switch1



Switch1\_13522053

Physical Config CLI Attributes

IOS Command Line Interface

```
Andini#show interfaces trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/9	on	802.1q	trunking	1

Port Fa0/9 Vlans allowed on trunk 1-1005

Port Fa0/9 Vlans allowed and active in management domain 1,2,3

Port Fa0/9 Vlans in spanning tree forwarding state and not pruned 1,2,3

Mode trunking pada Switch1 sudah menjadi on yang berarti Switch1 sudah dapat menyambung dengan Switch3 melalui port Fa0/9

## Konfigurasi VLAN Switch2

```
Switch2_13522053
```

Physical Config CLI Attributes

IOS Command Line Interface

```
Andini#show vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/5, Fa0/6, Fa0/7, Fa0/8 Fa0/9, 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
2	VLAN0002	active	Fa0/1, Fa0/2
3	VLAN0003	active	Fa0/3, Fa0/4
1002	fddi-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
1	enet	100001	1500	-	-	-	-	-	0	0
2	enet	100002	1500	-	-	-	-	-	0	0
3	enet	100003	1500	-	-	-	-	-	0	0

VLAN0002 berisi port untuk PC5 dan PC6

VLAN0003 berisi port untuk PC7 dan PC8

## Konfigurasi trunking Switch2

```
Switch2_13522053
```

Physical Config CLI Attributes

IOS Command Line Interface

```
Andini#show interfaces trunk
```

Port	Mode	Encapsulation	Status	Native vlan
Fa0/10	on	802.1q	trunking	1

```
Andini#show interfaces trunk
```

Port	Vlans allowed on trunk
Fa0/10	1-1005

```
Andini#show interfaces trunk
```

Port	Vlans allowed and active in management domain
Fa0/10	1,2,3

```
Andini#show interfaces trunk
```

Port	Vlans in spanning tree forwarding state and not pruned
Fa0/10	1,2,3

Mode trunking pada Switch2 sudah menjadi on yang berarti Switch2 sudah dapat menyambung dengan Switch3 melalui port Fa0/10

### Konfigurasi VLAN Switch3

```
Andini#show vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/3, Fa0/4 Fa0/5, Fa0/6, Fa0/7, Fa0/8 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
2	VLAN0002	active	
3	VLAN0003	active	
1002	fddi-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
1	enet	100001	1500	-	-	-	-	-	0	0
2	enet	100002	1500	-	-	-	-	-	0	0
3	enet	100003	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0

--More--

Copy

Paste

Sudah ditambahkan VLAN0002 dan VLAN0003 pada Switch3 agar dapat menyambungkan komunikasi dari PC VLAN0002 pada Switch1 ke PC VLAN0002 Switch2 serta PC VLAN0003 pada Switch1 ke PC VLAN0003 Switch2.

### Konfigurasi trunking Switch3

```
Andini#show interfaces trunk
```

```
Port      Mode      Encapsulation  Status      Native vlan
Fa0/9     on        802.1q         trunking    1
Fa0/10    on        802.1q         trunking    1

Port      Vlans allowed on trunk
Fa0/9     1-1005
Fa0/10    1-1005

Port      Vlans allowed and active in management domain
Fa0/9     1,2,3
Fa0/10    1,2,3

Port      Vlans in spanning tree forwarding state and not pruned
Fa0/9     1,2,3
Fa0/10    1,2,3
```

```
Andini#
```

Trunking bertujuan untuk menyambungkan antar switch agar PC VLAN0002 pada Switch1 dan PC VLAN0002 Switch2 serta PC VLAN0003 pada Switch1 dan PC VLAN0003 Switch2 dapat tersambung untuk berkomunikasi.

### Ping dari PC1 ke PC2, PC3, PC4, PC5, PC6, PC7, PC8

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1_...	PC2_135...	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC1_...	PC3_135...	ICMP		0.000	N	1	(edit)	(delete)
	Failed	PC1_...	PC4_135...	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC1_...	PC5_135...	ICMP		0.000	N	3	(edit)	(delete)

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1_...	PC6_135...	ICMP		0.000	N	4	(edit)	(delete)
	Failed	PC1_...	PC7_135...	ICMP		0.000	N	5	(edit)	(delete)
	Failed	PC1_...	PC8_135...	ICMP		0.000	N	6	(edit)	(delete)

Ping dari PC1 hanya bisa sampai ke PC2, PC5, PC6 karena keempat PC itu berada di satu VLAN yang sama yaitu VLAN0002 dan telah dihubungkan oleh trunking Switch1, Switch2, dan Switch3. PC1 tidak dapat berkomunikasi dengan P3, P4, P7, P8 karena tidak berada di satu VLAN yang sama walaupun Switch1 dan Switch2 sudah terhubung oleh Switch3 melalui trunking.

### Ping dari PC8 ke PC1, PC2, PC3, PC4, PC5, PC6, PC7

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Failed	PC8_...	PC1_135...	ICMP		0.000	N	0	(edit)	(delete)
	Failed	PC8_...	PC2_135...	ICMP		0.000	N	1	(edit)	(delete)
	Successful	PC8_...	PC3_135...	ICMP		0.000	N	2	(edit)	(delete)
	Successful	PC8_...	PC4_135...	ICMP		0.000	N	3	(edit)	(delete)

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Failed	PC8_...	PC5_135...	ICMP		0.000	N	4	(edit)	(delete)
	Failed	PC8_...	PC6_135...	ICMP		0.000	N	5	(edit)	(delete)
	Successful	PC8_...	PC7_135...	ICMP		0.000	N	6	(edit)	(delete)

Ping dari PC8 hanya bisa sampai ke PC3, PC4, PC7 karena keempat PC itu berada di satu VLAN yang sama yaitu VLAN0003 dan telah dihubungkan oleh trunking Switch1, Switch2, dan Switch3. PC8 tidak dapat berkomunikasi dengan P1, P2, P5, P6 karena tidak berada di satu VLAN yang sama walaupun Switch1 dan Switch2 sudah terhubung oleh Switch3 melalui trunking.

### Mac Address Switch1

Switch1\_13522053
 —
□
×


Physical
 Config
 CLI
Attributes

IOS Command Line Interface

```

Andini#show mac-address-table
Mac Address Table
-----
Vlan    Mac Address      Type      Ports
----
1       0001.6473.e809   DYNAMIC   Fa0/9
2       0001.6473.e809   DYNAMIC   Fa0/9
2       0002.1764.5b04   DYNAMIC   Fa0/9
2       0002.17d8.edb2   DYNAMIC   Fa0/1
2       0009.7c67.3307   DYNAMIC   Fa0/9
2       0040.0b83.8459   DYNAMIC   Fa0/2
3       0001.641d.9ea7   DYNAMIC   Fa0/3
3       0001.6473.e809   DYNAMIC   Fa0/9
3       0004.9a47.bea1   DYNAMIC   Fa0/9
3       000c.cfi1e.883b   DYNAMIC   Fa0/4
    
```

## Mac Address Switch2

 Switch2\_13522053

Physical

Config

CLI


Attributes

IOS Command Line Interface

Andini#show mac-address-table  
Mac Address Table

Vlan	Mac Address	Type	Ports
1	0001.6473.e80a	DYNAMIC	Fa0/10
2	0002.1764.5b04	DYNAMIC	Fa0/1
2	0002.17d8.edb2	DYNAMIC	Fa0/10
2	0009.7c67.3307	DYNAMIC	Fa0/2
3	0001.641d.9ea7	DYNAMIC	Fa0/10
3	0004.9a47.bea1	DYNAMIC	Fa0/4
3	0007.ec37.a836	DYNAMIC	Fa0/3
3	000c.cf1e.883b	DYNAMIC	Fa0/10

## Mac Address Switch3

 Switch3\_13522053

Physical

Config

CLI

Attributes

IOS Command Line Interface

Andini#show mac-address-table  
Mac Address Table

Vlan	Mac Address	Type	Ports
1	0030.a34e.5409	DYNAMIC	Fa0/9
1	0090.2b6e.c9ce	DYNAMIC	Fa0/10
2	0002.1764.5b04	DYNAMIC	Fa0/10
2	0002.17d8.edb2	DYNAMIC	Fa0/9
2	0009.7c67.3307	DYNAMIC	Fa0/10
2	0090.2b6e.c9ce	DYNAMIC	Fa0/10
3	0001.641d.9ea7	DYNAMIC	Fa0/9
3	0004.9a47.bea1	DYNAMIC	Fa0/10
3	000c.cf1e.883b	DYNAMIC	Fa0/9
3	0090.2b6e.c9ce	DYNAMIC	Fa0/10

# Referensi

Cisco. (n.d.). *Cisco Networking Academy*. <https://www.netacad.com>

Lammle, T. (2020). *CCNA certification study guide: Exam 200-301*. Sybex.