

# Sybau Labubu Tracer

**um... do you want  
to... hop on cisco  
packet tracer**



*"Senpai, do you want to... setup IPv6 tunneling with me 🙌🙌"*

**Dokumen Spesifikasi Topologi Jaringan**

**Oleh: M. Rayhan Farrukh - 13523035**

*Seleksi Asisten Laboratorium Sistem Terdistribusi 2025*

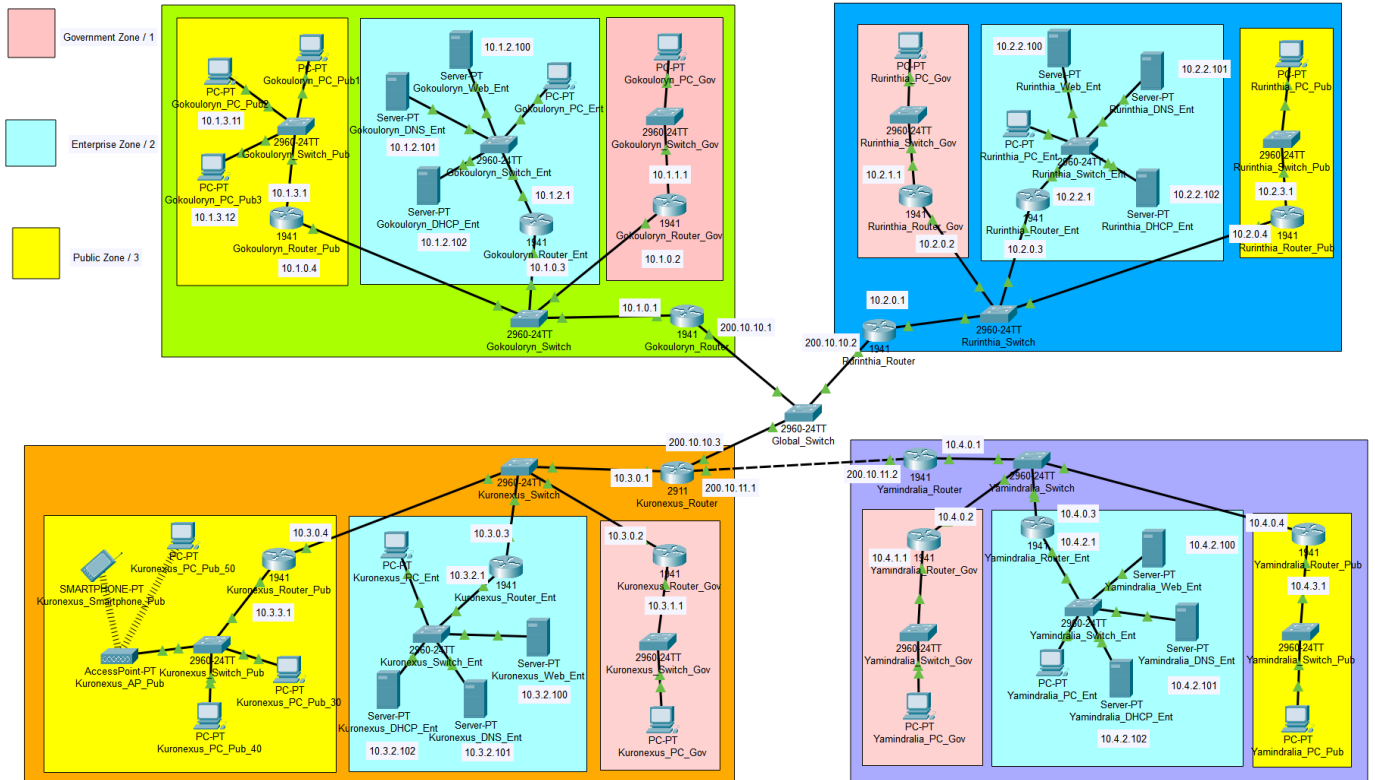
## Daftar Isi

Daftar Isi.....	1
Daftar Gambar.....	1
Overview.....	2
Implementasi.....	4
Konfigurasi Awal.....	4
Internal Routing.....	5
DHCP.....	6
External Routing.....	7
VLAN.....	8
Wireless Network.....	10
DNS.....	11
ACL.....	12
SSH/Telnet.....	14
NAT.....	15
IPv6.....	16

## Daftar Gambar

Gambar 1. Topologi Jaringan (sorry for the mess).....	2
Gambar 2. Contoh konfigurasi DHCP.....	5
Gambar 3 Konfigurasi pembuatan VLAN.....	7
Gambar 4. Contoh konfigurasi access port pada interface switch.....	7
Gambar 5. Konfigurasi port 1 access point.....	9
Gambar 6. Contoh konfigurasi A record dan NS record untuk Gokouloryn.....	10

## Overview



Gambar 1. Topologi Jaringan (sorry for the mess)

### Catatan

jika *rectangle* ada yang hilang, gerakan *rectangle* yang besar sedikit, lalu kembalikan seperti semula (CPT *moment*).

Topologi ini adalah simulasi internet kecil yang menghubungkan 4 negara (Gokoulorn, Rurinthia, Kuronexus, dan Yamindralia). Tiga negara pertama terhubung secara langsung melalui *switch*, sedangkan Yamindralia terhubung melalui Kuronexus. Seluruh *traffic* antarnegara diatur oleh BGP sebagai protokol *external routing*, sedangkan *traffic* intranegara diatur oleh OSPF.

Untuk jaringan dalam negara, keempat negara kurang lebih memiliki topologi fisik yang sama, kecuali untuk Kuronexus yang memiliki pembagian jaringan VLAN pada zona publiknya. Setiap negara memiliki zona *enterprise* yang memiliki *server* untuk layanan penting jaringan tersebut. IP *address* dari komputer dan *end device* lainnya diatur otomatis menggunakan DHCP melalui server DHCP yang ada di zona *enterprise*.

Setiap negara juga memiliki *web server* yang dapat diakses melalui *domain name* yang telah ditentukan untuk negara tersebut.

Jaringan ini juga memiliki beberapa fitur tambahan, seperti SSH dan Telnet untuk setiap *border router* negara, *wireless network* untuk salah satu VLAN pada Kuronexus, serta komunikasi IPv6 untuk Rurinthia-Yamindralia.

Bagian-bagian selanjutnya adalah langkah-langkah atau konfigurasi yang saya lakukan untuk mengimplementasikan setiap fitur/fungsi. Penjelasan cara implementasi akan saya tuliskan dalam gaya panduan agar mudah untuk diikuti (dan lebih mudah bagi saya untuk menulis dan memahaminya).

## Implementasi

### Konfigurasi Awal

Langkah pertama adalah membentuk topologi sesuai dengan spesifikasi soal. Buat topologi 4 negara menggunakan komponen yang dijelaskan pada soal, untuk *switch*, saya menggunakan model 2960. Koneksi *interface* untuk tiap komponen disambungkan menggunakan *tool Automatically choose connection*.

Kemudian atur *display name*, *hostname* untuk tiap *device*, dan IP *address* dari tiap *router* melalui *config* pada antarmuka *device*. IP address yang saya gunakan sebagai berikut:

- 200.10.10.x - untuk *public* IP, dimana x adalah nomor negara.
- 200.10.11.x - untuk Kuronexus-Yamindralia
- 10.x.y.z - untuk *private* IP, dimana x adalah nomor negara, y adalah nomor zona, z adalah nomor *device*.

**Catatan:** Nomor negara bebas, tidak ada aturan, saya menomorkan sesuai urutan nama negara disebutkan pada soal.

## Internal Routing

Selanjutnya, kita akan mengonfigurasi *internal routing* agar *end device* pada satu negara bisa berkomunikasi satu sama lain walaupun terletak pada *zone* yang berbeda. Untuk ini kita akan menggunakan *Multi-Area OSPF*. *Command* yang dipakai:

```
# Border Router
enable
configure terminal
router ospf 1
router-id 1.1.<x>.1
network 10.<x>.0.0 0.0.0.255 area 0
```

```
# Zone Router
enable
configure terminal
router ospf 1
router-id 1.1.<x>.<y+1>
network 10.<x>.0.0 0.0.0.255 area 0
network 10.<x>.<y>.0 0.0.0.255 area <y>
```

Lakukan ini untuk semua *router* pada semua negara.

## DHCP

Sekarang kita akan implementasi server DHCP agar tidak perlu *assign* IP pada PC secara manual. Sebelum itu, *assign static* IP untuk semua server, serta *default gateway* dan *address* server DNS-nya.

Setelah itu, untuk mengatur server DHCP, buka GUI server, lalu buka tab *Services* > DHCP. Pada halaman ini, hidupkan *service* (Service → on), lalu buat tiga *pool* untuk setiap zona. Atur *start IP address* dan *default gateway* sesuai zona, dan DNS *server* sesuai negara, dan abaikan saja pengaturan lainnya. **Catatan:** untuk zona *enterprise* ubah aturan *serverPool*, jangan menambahkan aturan baru. Contoh konfigurasi yang saya pakai sebagai berikut:

DHCP

Interface: FastEthernet0 Service: ☒ On ☐ Off

Pool Name: Goko\_Gov\_Pool

Default Gateway: 10.1.1.1

DNS Server: 10.1.2.101

Start IP Address : 10 1 1 10

Subnet Mask: 255 255 255 0

Maximum Number of Users : 246

TFTP Server: 0.0.0.0

WLC Address: 0.0.0.0

Add Save Remove

Gambar 2. Contoh konfigurasi DHCP

Kemudian, atur *config* semua PC untuk menggunakan DHCP pada *interface* dan *Gateway/DNS*. Pada saat ini, PC pada zona selain *enterprise* masih belum mendapat DHCP. Kita harus atur *router* zona tersebut agar dapat meneruskan permintaan DHCP ke server DHCP. Gunakan *command* berikut:

```
# Public/Government Router
enable
configure terminal
interface <nama interface inside>
ip helper-address <ip address dhcp server>
```

### Catatan

- *Interface inside* adalah *interface* yang terhubung ke *zone switch*
- Terkadang setelah *restart* Cisco Packet Tracer, beberapa PC berubah menjadi menggunakan *static IP*, atur ulang saja agar menggunakan DHCP

## External Routing

Setelah setiap zona bisa berkomunikasi satu sama lain, saatnya kita mengatur agar komunikasi antarnegara. Untuk ini, kita akan mengatur BGP pada setiap *border router*. Jalankan *command* berikut pada setiap *border router*:

```
# Border Router
enable
configure terminal
router bgp <AS number>
bgp router-id <bgp_router_id> # Samakan saja dengan OSPF router-id
neighbor <ip_border_router_tetangga> remote-as <ASN_tetangga>
network <ip_network> mask 255.255.255.0
```

### Catatan

- AS number pilih bebas, yang saya gunakan adalah 65001-65004
- Ulangi *command* neighbor sebanyak jumlah tetangga *router*
- Ulangi *command* network untuk semua *network* dimana *border router* berada, misalnya, 200.10.10.0 dan 200.10.11.0 untuk Kuronexus

Selanjutnya, kita akan mengatur redistribusi rute BGP ke OSPF agar jaringan internal tahu rute ke jaringan eksternal Dan OSPF ke BGP agar jaringan eksternal tahu rute ke jaringan internal.

```
# Border Router (OSPF → BGP)
enable
configure terminal
router bgp <ASN>
redistribute ospf 1
```

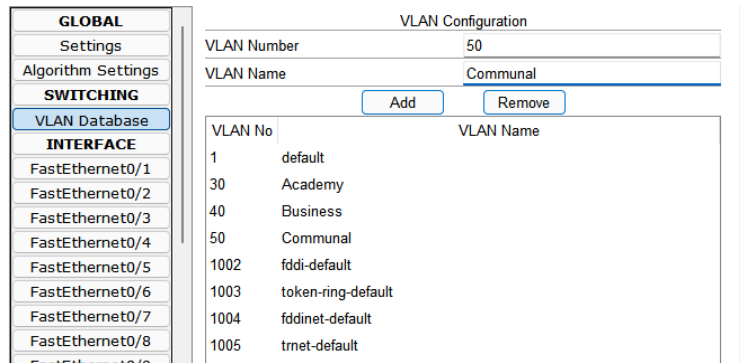
```
# Border Router (BGP → OSPF)
enable
configure terminal
router ospf 1
redistribute bgp <ASN> subnets
Redistribute connected subnetsp
```

Setelah ini, saya menemukan masalah untuk koneksi Yamindralia dengan negara lain (selain Kuronexus), untuk itu, jalankan perintah berikut untuk memperbaikinya



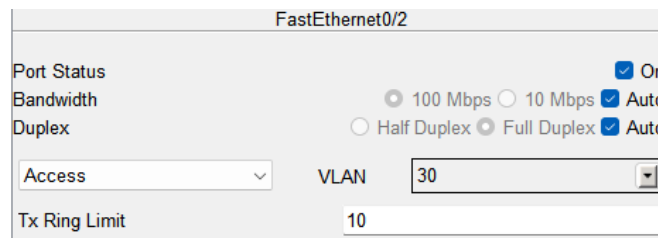
## VLAN

Selanjutnya, kita akan memisahkan *public zone* pada negara Kuronexus menjadi tiga buah VLAN. Untuk melakukan ini, pertama atur *switch* pada zona *public* untuk membuat VLAN tersebut. Buka GUI *config* pada *switch* dan tambahkan VLAN 30, 40, dan 50 seperti berikut:



Gambar 3 Konfigurasi pembuatan VLAN

Kemudian, atur *access port* pada *interface switch* untuk mengakses masing-masing VLAN, lakukan ini untuk VLAN 30, 40 dan 50. Lalu pada *interface* yang terhubung ke *router*, atur menjadi *trunk port*.



Gambar 4. Contoh konfigurasi *access port* pada *interface switch*

Setelah ini, kita akan mengatur *router* dari zona *public*, bagian ini akan dilakukan melalui CLI *router*.

Pertama, hapus *IP address* lama dari *interface* dalam *router*, matikan lalu nyalakan kembali. Kemudian, buat *sub-interface* virtual untuk masing-masing VLAN menggunakan *command* berikut:

```
# Kuronexus Public Zone Router (Catatan: v adalah nomor VLAN)
enable
configure terminal
interface Gi0/1.<v>                                     # gunakan interface dalam
encapsulation dot1q <v>
```

```
ip address 10.3.<v>.1 255.255.255.0  
ip helper-address 10.3.2.102          # server DHCP
```

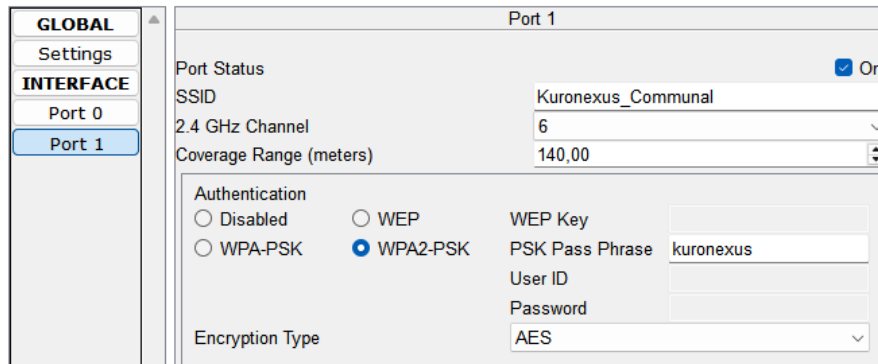
Lalu kita akan meng-*update routing protocols* OSPF dan BGP pada *public zone router* Kuronexus.

```
# Kuronexus Public Zone Router (OSPF)  
enable  
configure terminal  
no network 10.3.3.0 0.0.0.255 area 3  
network 10.3.<v>.0 0.0.0.255 area 0
```

```
# Kuronexus Public Zone Router (BGP)  
enable  
configure terminal  
router bgp 65003  
no network 10.3.3.0 mask 255.255.255.0  
network 10.3.<v>.0 mask 255.255.255.0
```

## Wireless Network

Selanjutnya, kita akan menambah *wireless network* untuk VLAN 50. Tambahkan *access point* (model AP-PT), PC dan *smartphone*. Hubungkan *access point* ke *switch* pada *interface* yang diatur untuk VLAN 50. Kemudian atur *port 1* pada AP melalui *configi* GUI, dan atur seperti pada gambar 6.



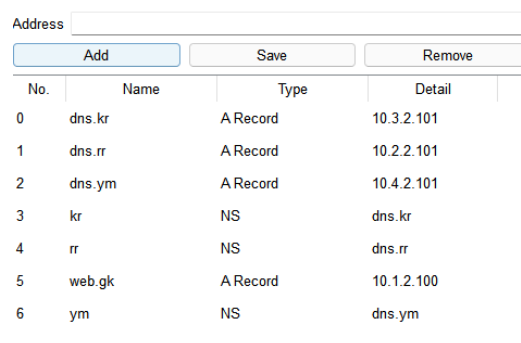
Gambar 5. Konfigurasi *port 1 access point*

Untuk menghubungkan PC ke *access point*, kita perlu mengubah modul pada PC. Pada GUI, buka *tab physical*, lalu tekan tombol *power* pada PC. Cabut modul *ethernet* (berada pada bagian bawah PC), lalu masukkan modul *wireless* (WMP300N), dan kembali PC tersebut. Setelah itu, untuk menghubungkan PC ke Wi-Fi, buka tab Desktop → PC Wireless. Buka *tab* Connect, tunggu sesaat hingga jaringan “Kuronexus\_Communal” muncul. Pilih jaringan tersebut dan klik Connect, lalu masukkan *passphrase* yang sudah diatur sebelumnya. Untuk *smartphone*, cukup atur *interface* Wireless0 saja, dan secara otomatis akan terhubung ke *access point*.

## DNS

Pada tahap ini kita akan *setup* server DNS pada setiap negara agar kita dapat menggunakan *domain name* untuk mengakses *web server* negara tersebut. Pertama, ubah file *index.html* pada *web server* setiap negara, dengan membuka *tab* Services → HTTP. Berikan *header* yang berisi nama negara dan *font color* yang berbeda dari negara lainnya. Lalu, pada DNS server, buka *tab* Services → DNS, hidupkan dan tambah *domain name* beserta *address* dari *web server* negara tersebut (misal, web.gk), lakukan ini pada DNS server tiap negara.

Sekarang, *device* di dapat mengakses *web server* dari negaranya sendiri menggunakan *domain name*. Selanjutnya, kita akan membuat agar *device* dapat mengakses *web server* seluruh negara lainnya melalui *domain name*. Untuk itu, kita perlu menambahkan *address (A) record* dan *name server (NS) record* untuk memberi tahu DNS server harus bertanya ke DNS server mana untuk menemukan alamat suatu *domain name*. Contoh *records* yang harus ditambahkan terdapat pada gambar 7



No.	Name	Type	Detail
0	dns.kr	A Record	10.3.2.101
1	dns.rr	A Record	10.2.2.101
2	dns.ym	A Record	10.4.2.101
3	kr	NS	dns.kr
4	rr	NS	dns.rr
5	web.gk	A Record	10.1.2.100
6	ym	NS	dns.ym

Gambar 6. Contoh konfigurasi A record dan NS record untuk Gokouloryn

NS record menunjukkan server mana yang harus ditanyakan untuk TLD tertentu, misal NS record dengan nama rr, dan menuju ke dns.rr berarti “untuk domain dengan TLD .rr, tanyakan alamatnya pada server dengan domain name dns.rr”. Agar ini dapat bekerja, kita juga harus memberi tahu alamat IP dari domain dns.rr, maka dari itu, tambahkan A record yang berisikan alamatnya.

### Catatan

- Ketika mencoba mengakses web melalui domain, untuk percobaan pertama dapat memakan waktu yang cukup lama, meskipun sudah menggunakan *fast forward*.

## ACL

### ACL untuk *Government Zone*

Kita perlu mengimplementasikan *access control list* agar jaringan *government* hanya dapat diakses oleh *government* lainnya. Sekilas ini tampak mudah, ini dapat dilakukan dengan *deny all* kemudian *allow traffic* dari *network* IP *government* lainnya. Namun, dengan melakukan itu, *government* tidak akan dapat mengakses jaringan zona lainnya karena *traffic* balasan dari zona tersebut akan ditolak oleh jaringan *government*. Untuk itu, kita perlu langkah khusus. *Command* yang digunakan sebagai berikut:

```
# Government Zone Router Negara <x>
enable
configure terminal

# Allow OSPF
access-list 101 permit ospf any any

# Allow traffic untuk DNS dan DHCP (yang menggunakan UDP)
access-list 101 permit udp any eq 53 any
access-list 101 permit udp any any eq 67
access-list 101 permit udp any any eq 68

# Allow traffic yang merupakan balasan (TCP dan ICMP)
access-list 101 permit tcp any any established
access-list 101 permit icmp any any echo-reply

# Allow traffic government zone
# y adalah negara asing, jalankan perintah ini 3 kali dengan y yang berbeda
access-list 101 permit ip 10.<y>.1.0 0.0.0.255 10.<x>.1.0 0.0.0.255

interface Gi0/<0/1>          # interface "luar"
ip access-group 101 in
```

### ACL untuk *Enterprise Zone*

Untuk *enterprise zone*, tidak akan ada masalah seperti pada *government zone*, jadi ACL dibuat biasa saja sesuai yang dituliskan pada spesifikasi soal.

```
# Enterprise Zone Router Negara <x>
access-list 102 permit ospf any any
access-list 102 permit ip 10.<x>.1.0 0.0.0.255 10.<x>.2.0 0.0.0.255
```

**# Allow HTTPS, DNS, dan DHCP**

```
access-list 102 permit tcp any 10.<x>.2.0 0.0.0.255 eq 443
access-list 102 permit udp any 10.<x>.2.0 0.0.0.255 eq 53
access-list 102 permit tcp any 10.<x>.2.0 0.0.0.255 eq 53
access-list 102 permit udp any eq 53 any                # DNS Forward Reply

access-list 102 permit udp any eq 67 any eq 68          # DHCP Reply
access-list 102 permit udp any any eq 67                # DHCP Request
```

**# Allow traffic yang merupakan balasan (TCP dan ICMP)**

```
access-list 102 permit tcp any any established
access-list 102 permit icmp any any echo-reply
```

```
interface Gi0/<z>          # interface "luar"
ip access-group 102 in
```

## SSH/Telnet

Sekarang kita akan implementasikan SSH serta Telnet pada *border router* tiap negara. Untuk ini kita perlu *setup* beberapa hal agar SSH dapat digunakan, serta menambahkan DNS *entry* agar SSH dan Telnet mudah diakses. *Setup* dan aktivasi SSH dan Telnet dengan perintah berikut:

```
# Border Router
enable
config terminal
ip domain-name <border.gk>          # Bebas memilih domain name apa saja
crypto key generate rsa              # Anda akan diminta memilih key size
                                     # pilih saja 512

username <name> secret <password>
line vty 0 4                         # Maksimum 5 SSH bersamaan
login local
transport input all
```

Jalankan perintah diatas pada setiap *border router*. Setelah itu, tambahkan entri DNS (misal, `border.gk`) pada masing-masing negara yang mengarah ke alamat publik *border router* negara tersebut, dengan cara yang sama pada Bagian [DNS](#). Untuk memasuki SSH, gunakan perintah berikut:

- SSH: `ssh -l <username> <domain_name/address>`
- Telnet: `telnet <domain_name/address>`.

### Catatan

- Kredensial yang saya pakai adalah `<TLD>:<TLD123>` (misal, `gk:gk123`)

## NAT

Pada bagian ini kita akan mengimplementasikan NAT pada Gokouloryn untuk alamat IP yang genap. Dengan ini, *border router* akan menyembunyikan alamat IP sebenarnya dari *device* ketika melakukan koneksi ke luar negara.

Sebelum implementasi, tambahkan dua PC pada *public zone* Gokouloryn, dan tambahkan alamat statis dengan satu PC memiliki IP genap dan satunya lagi ganjil. Lalu pada *border router*, buat *access list* untuk memfilter alamat IP genap yang berasal dari zona publik, gunakan perintah berikut:

```
# Border Router
enable
configure terminal
access-list 1 permit 10.1.3.0 0.0.0.254
```

Kemudian konfigurasi NAT dan hubungkan dengan *access list* yang dibuat dengan perintah berikut:

```
# Border Router
enable
configure terminal

interface <interface_dalam>
ip nat inside

interface <interface_luar>
ip nat outside

ip nat inside source list 1 interface <interface_luar> overload
```



## IPv6

Jalankan perintah `ipv6 unicast-routing` (di dalam *config mode*) untuk semua *router* pada negara Rurinthia dan Yamindralia. Setelah itu, *setup tunnel* untuk menghubungkan kedua negara, sehingga dapat menggunakan IPv6 dalam jaringan IPv4. Gunakan perintah berikut untuk Rurinthia dan Yamindralia:

```
# Border Router (alamat tunnel 2001:DB8:A::<x>/64)
enable
config terminal
interface Tunnel0
ipv6 address <ipv6_address_tunnel>
tunnel source <address_router_ini>
tunnel destination <address_router_tujuan>
tunnel mode ipv6ip
no shutdown
```

Setelah itu, pasangkan alamat IPv6 untuk semua *interface* pada semua *router* di dua negara tersebut. Berikut tabel *assignment IPv6 address* yang saya gunakan (tidak harus seperti ini):

**Catatan:** Gi0/0 adalah *interface* ke luar, sedangkan Gi0/1 adalah *interface* ke dalam

Border_Rurinthia	Gi0/1	2001:DB8:2:0::1/64
Gov_Rurinthia	Gi0/0	2001:DB8:2:0::2/64
	Gi0/1	2001:DB8:2:1::1/64
Ent_Rurinthia	Gi0/0	2001:DB8:2:0::3/64
	Gi0/1	2001:DB8:2:2::1/64
Pub_Rurinthia	Gi0/0	2001:DB8:2:0::4/64
	Gi0/1	2001:DB8:2:3::1/64
Border_Yamindralia	Gi0/1	2001:DB8:4:0::1/64
Gov_Yamindralia	Gi0/0	2001:DB8:4:0::2/64
	Gi0/1	2001:DB8:4:1::1/64
Ent_Yamindralia	Gi0/0	2001:DB8:4:0::3/64
	Gi0/1	2001:DB8:4:2::1/64
Pub_Yamindralia	Gi0/0	2001:DB8:4:0::4/64
	Gi0/1	2001:DB8:4:3::1/64

Kemudian, *set* semua *device* lainnya untuk menggunakan IPv6 *automatic* agar mendapat alamat dengan sendirinya. Setelah itu, *setup routing* dengan perintah berikut:

```
# Router (Border dan Zone)
enable
config terminal
ipv6 router ospf 1
router-id <router_id>      # Samakan saja dengan id untuk IPv4 sebelumnya

# Masuk ke semua interface (termasuk Tunnel0) dan jalankan ini
ipv6 ospf 1 area 0
```

Setelah ini, Rurinthia dan Yamindralia dapat berkomunikasi satu sama lain menggunakan alamat IPv6.