**DESAIN & IMPLEMENTASI JARINGAN SKALA KECIL (SMALL NETWORK) DENGAN KONSEP NETWROK PROGRAMMABILTY**

Di ajukan untuk memenuhi persyaratan kelulusan

Program MSIB Network Programmabilty DTS-TSA



Disusun oleh :

**Ahmad Romadhani**

**NIM : 1921500006**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK UNIVERSITAS NURUL JADID**

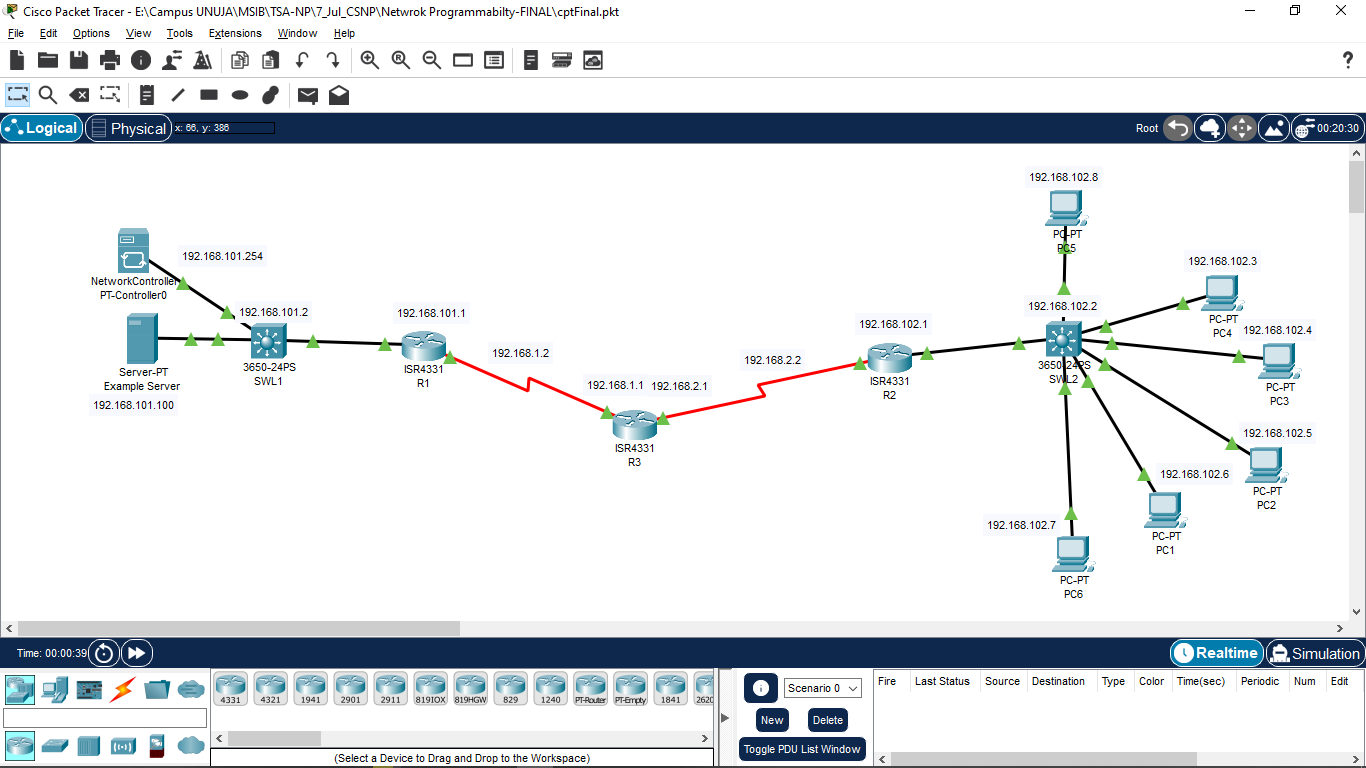
**PAITON PROBOLINGGO**

**2022**

**STUDI KASUS**



1. Desain memuat minimal 3 (tiga) unit **Cisco 4331 routers**, 2 (dua) unit **Cisco** **3650 Switches**, 6 (enam) unit PC client, 1 (satu) Server dan **wajib menggunakan** 1 (satu) **Network Controller**.



1. Desain jaringan menerapkan skema pengalamatan **Variable Length Subnet Mask (VLSM)** dan menerapkan **Virtual Local Area Network (VLAN)** serta menggunakan **routing protocol**.

|  |  |  |
| --- | --- | --- |
| **Devices** | **Interfaces** | **IP Addreses** |
| R1 | G0/0/0 | 192.168.101.1/24 |
| S0/1/0 | 192.168.1.2 |
| R2 | G0/0/0 | 192.168.102.1 |
| S0/1/1 | 192.168.2.2 |
| R3 | G0/0/0 | 10.0.1.1 |
| G0/0/1 | 10.0.2.1 |
| S0/1/0 | 192.168.1.1 |
| S0/1/1 | 192.168.2.1 |
| SWL1 | VLAN1 | 192.168.101.2 |
| SWL2 | VLAN1 | 192.168.102.2 |
| PC1 | NIC | 192.168.102.6 |
| PC2 | NIC | 192.168.102.5 |
| PC3 | NIC | 192.168.102.4 |
| PC4 | NIC | 192.168.102.3 |
| PC5 | NIC | 192.168.102.8 |
| PC6 | NIC | 192.168.102.7 |
| Example Server | NIC | 192.168.101.100 |
| PT-Controller | NIC | 192.168.101.254 |

1. Konfigurasi dasar pengamanan perangkat jaringan meliputi menerapkan **password privilege**, **console** dan **remote access** hanya **melalui Secure Shell (SSH)**.

Cisco Packet Tracer PC Command Line 1.0

C:\>SSH -l cisco 192.168.101.1

Password:

R1>en

R1>enable

Password:

R1#exit

[Connection to 192.168.101.1 closed by foreign host]

C:\>SSH -l cisco 192.168.101.2

Password:

SWL1#ex

[Connection to 192.168.101.2 closed by foreign host]

C:\>SSH -l cisco 192.168.102.1

Password:

R2>exit

[Connection to 192.168.102.1 closed by foreign host]

C:\>SSH -l cisco 192.168.102.2

Password:

SWL2#exit

[Connection to 192.168.102.2 closed by foreign host]

C:\>

1. Kode program yang dibuat mengimplementasikan fitur **REST-API** dari

**Network Controller**.

Bagian 1 : Verifikasi pengaturan Packet Tracer untuk akses eksternal.

a. Klik Opsi > Preferensi > Lain-lain. Di bawah Akses Jaringan Eksternal, verifikasi bahwa Aktifkan

Akses Eksternal untuk Pengontrol Jaringan REST API dicentang.

b. Tutup jendela Preferensi.

c. Klik PT-Controller0 > Konfigurasi.

d. Di t kiri, di bawah DUNIA NYATA, klik Pengontrol.

e. Centang Access Enabled dan catat nomor port, yang kemungkinan besar adalah 58000. Ini adalah nomor port yang Anda perlukan saat mengakses aktivitas Packet Tracer secara eksternal dari Chromium, VS Code, dan Postman nanti dalam aktivitas dan pastikan Anda dapat mengakses Packet Tracer dari program lain di DEVASC VM.

Buka Chromium dan navigasikan ke http://localhost:58000/api/v1/host

Anda akan mendapatkan respons berikut. Langkah ini memverifikasi bahwa Anda dapat mengakses Packet Tracer dan PT-Controller0 secara eksternal. Perhatikan bahwa otorisasi memerlukan tiket. Anda akan mendapatkan token otorisasi di Bagian selanjutnya.

{

"tanggapan": {

"detail": " Security Authentication Failure", "errorCode": "REST\_API\_EXTERNAL\_ACCESS",

"message": " Ticket-based authorization: empty ticket. "

},

"versi": "1.0"

} {

Bagian 2 : Meminta Token Otentikasi dengan Tukang Pos

Di Bagian ini, Anda akan menyelidiki dokumentasi REST API di Packet Tracer dan menggunakan Postman untuk meminta token otentikasi dari PT-Controller0. Anda juga dapat melakukan ini dalam Kode VS dengan skrip Python.

Langkah 1: Selidiki dokumentasi REST API untuk Network Controller.

Untuk melihat dokumentasi REST API untuk atau PT-Controller0, selesaikan langkah-langkah berikut:

sebuah. Klik PC > Desktop > Browser Web.

b. Masukkan 192.168.101.254

c. Login ke PT-Controller0 dengan user dan password.

d. Klik menu di sebelah log Cisco dan pilih API Docs. API ini memberikan dokumentasi yang sama yang Anda temukan di PT-Controller0.

Langkah 2: Buat permintaan POST baru.

sebuah. Setelah meninjau dokumentasi AddTicket REST API Method Postman. Di area Luncurkan, klik tanda plus untuk membuat Permintaan Tanpa Judul baru.

a. Klik panah bawah dan ubah jenis dari GET menjadi POST.

b. Masukkan URL http://localhost:58000/api/v1/ticket

c. Di bawah bidang URL, klik Isi. Ubah jenisnya menjadi mentah.

d. Klik panah bawah di sebelah Teks dan ubah ke JSON. Perubahan ini juga akan menyetel Header HTTP “Content-type” menjadi “application/json” yang diperlukan untuk panggilan API ini.

e. Rekatkan objek JSON berikut ke bidang Body. Pastikan kode Anda diformat dengan benar

{

"username": "What is your username ? ",

"password": "Enter your password !"

}

Langkah 3: Kirim permintaan POST

sebuah. Klik Send untuk mengirim permintaan POST ke PT-Controller0.

Anda harus mendapatkan tanggapan yang serupa dengan berikut ini. Namun, your\_serviceTicket akan menjadi nilai aktual.

{

"response": {

"idleTimeout": 900,

"serviceTicket": "NC-34-447a43ff9adc466db188-nbi",

"sessionTimeout": 3600

},

"version": "1.0"

}

b. Salin nilai serviceTicket tanpa tanda kutip ke file teks untuk digunakan atau yang lebih baru.

Bagian 3: Kirim Permintaan REST dengan Tukang Pos

Di Bagian ini, Anda akan menggunakan tiket layanan Anda untuk mengirim tiga permintaan REST ke PT-Controller0.

Langkah 1: Buat permintaan GET baru untuk semua perangkat jaringan di jaringan.

sebuah. Di Tukang Pos, klik tanda plus untuk membuat Permintaan Tanpa Judul baru.

a. Masukkan URL http://localhost:58000/api/v1/host

b. Di bawah bidang URL, klik Header.

c. Di bawah KUNCI terakhir, klik bidang Kunci dan masukkan X-Auth-Token.

Bagian 4: Kirim Permintaan REST Di Dalam Packet Tracer (Opsional)

Di Bagian ini, Anda akan menggunakan skrip yang sama dengan satu pengeditan kecil untuk mengirim permintaan API yang sama di dalam Paket

Tracer yang Anda kirim dari VS Code.

Langkah 1: Buat Proyek di Packet Tracer

sebuah.

a. Di Packet Tracer klik PC

b. Klik tab Pemrograman.

c. Saat ini tidak ada proyek. Klik Baru.

d. Masukkan REST APIs sebagai Name dan pilih Empty - Python sebagai template.

e. Klik Buat.

Proyek REST API (Python) sekarang dibuat dengan skrip main.py kosong.

Langkah 2: Ubah skrip untuk dijalankan di dalam Packet Tracer.

Akses dari satu aplikasi ke aplikasi lain pada mesin host yang sama mengharuskan nomor port ditentukan dalam URL. Namun, Packet Tracer mensimulasikan jaringan nyata. Di dunia nyata, kamu tidak biasanya

tentukan nomor port saat membuat permintaan API. Selain itu, Anda akan menggunakan nama domain atau alamat IP

di URL-nya.

sebuah. Dalam Kode VS,

salin kode get\_your\_host.py.

b. Di tab Admin > Pemrograman, klik dua kali skrip main.py untuk membukanya.

c. Tempelkan kode di skrip main.py.

d. Ubah api\_url. Ganti localhost:58000/api/v1/host dengan 192.168.101.254/api/v1/host

e. Hasil edit disimpan secara otomatis. Klik Jalankan. Output Packet Tracer tidak persis mensimulasikan apa yang Anda lihat di baris perintah Linux. Namun, Anda akan melihat output serupa seperti yang ditunjukkan di bawah ini.

Memulai REST API (Python)... ('Status permintaan: ', 200)

('PC4', '\t', '192.168.102.3', '\t', '00E0.F96C.155B', '\t', 'GigabitEthernet1/0/24')

('PC3', '\t', '10.0.2.129', '\t', '0004.9A42.C245', '\t', 'GigabitEthernet1/0/24') ('PC1', '\t' , '10.0.1.129', '\t', '00E0.A330.3359', '\t', 'GigabitEthernet1/0/22') ('PC2', '\t', '10.0.2.130', ' \t', '0060.47C1.A4DB', '\t', 'GigabitEthernet1/0/23') ('Admin', '\t', '10.0.1.130', '\t', '0050.0FCE.B095 ', '\t', 'GigabitEthernet1/0/21') ('Contoh Server', '\t', '192.168.101.100', '\t', '000A.413D.D793', '\t',

'GigabitEthernet1/0/3')

REST API (Python) selesai berjalan.

1. Link YouTube :

Link Google Drive : https://drive.google.com/drive/folders/11\_fbVybS0cDAF-yicPbYPShiy25RWQsE

Link Github : https://github.com/ahmadromadhani007/Netwrok-Programmabilty-MBKM