MODUL PENGANTAR OTOMATISASI JARINGAN KOMPUTER

Dasar Teori, Konfigurasi Prasyarat, & Praktikum Manual Infrastruktur Multi-Vendor

BAGIAN 1: DASAR TEORI & PUSTAKA OTOMATISASI JARINGAN

1. Definisi & Pemahaman Otomatisasi Jaringan Komputer

Otomatisasi Jaringan Komputer adalah proses mengotomatiskan perencanaan, penyebaran (deployment), konfigurasi, pengelolaan, serta pengujian perangkat jaringan (Router, Switch, Firewall, dan Server) menggunakan perangkat lunak (script/playbook) alih-alih konfigurasi manual CLI satu per satu.

Tujuan Utama: Mengurangi kesalahan manusia (*human error*), mempercepat proses *provisioning*, dan memastikan konsistensi konfigurasi di seluruh infrastruktur multi-vendor.

2. Pemrograman Python, PIP, dan Virtual Environment (venv)

Definisi:

- **Python:** Bahasa pemrograman tingkat tinggi yang populer untuk otomatisasi jaringan karena sintaksnya yang mudah dibaca dan didukung oleh ribuan pustaka (*libraries*).
- **PIP (Package Installer for Python):** Manajer paket standar untuk Python yang digunakan untuk menginstal pustaka pihak ketiga.
- **Virtual Environment (venv):** Isolasi lingkungan eksekusi Python agar paket/pustaka suatu proyek tidak saling berbenturan (*conflict*) dengan pustaka sistem atau proyek lain.

Cara Instalasi pada Ubuntu / WSL:

```
# Update repositori dan install Python3, PIP, serta modul venv
sudo apt update && sudo apt install -y python3 python3-pip python3-venv

# Membuat virtual environment bernama 'env-otomasi'
python3 -m venv env-otomasi

# Mengaktifkan virtual environment
```

```
source env-otomasi/bin/activate
```

3. Metode Subprocess (Bypass Offline)

Definisi & Pemahaman: Metode `subprocess` adalah penggunaan pustaka bawaan Python (`subprocess`) untuk mengeksekusi perintah shell sistem (seperti `sshpass`, `ssh`, `ping`) langsung dari dalam script Python.

Mengapa Menggunakan Metode Subprocess?

1. **Bypass Offline / No External Library Dependency:** Tidak memerlukan instalasi pustaka Python tingkat tinggi eksternal seperti Netmiko/Paramiko jika pustaka tersebut sulit diinstal di lingkungan tanpa internet (*air-gapped*).
2. **Sederhana:** Langsung memanfaatkan fungsi CLI bawaan OS lokal.

4. Netmiko (Pustaka Python Multi-Vendor)

Definisi & Pemahaman: Netmiko adalah pustaka Python berbasis Paramiko yang dirancang khusus untuk mempermudah koneksi SSH ke perangkat jaringan (Cisco, MikroTik, Juniper, dll.) dengan menangani abstraksi terminal, *prompting*, dan pembuatan perintah khusus.

Instalasi & Contoh Script Sederhana:

```
pip install netmiko

from netmiko import ConnectHandler


# Definisi Parameter Perangkat Cisco

cisco_device = {

    'device_type': 'cisco_ios',

    'host': '192.168.1.10',

    'username': 'admin',

    'password': 'password123',

    'port': 22,

}


# Membuat Koneksi & Mengirim Perintah Show

net_connect = ConnectHandler(**cisco_device)

output = net_connect.send_command('show ip interface brief')
```

```
print(output)

net_connect.disconnect()
```

5. Ansible, Paramiko, & Koleksi Multi-Vendor (community.routeros)

Definisi & Pemahaman:

- **Paramiko:** Pustaka dasar Python untuk protokol SSHv2.
- **Ansible:** *Automation engine* berbasis *agentless* yang menggunakan sintaks YAML untuk mengelola infrastruktur secara masif.
- **Koleksi Multi-Vendor (community.routeros):** Modul resmi dari komunitas Ansible untuk mengelola RouterOS MikroTik via API / CLI.

Cara Instalasi:

```
# Install Ansible via PIP

pip install ansible paramiko


# Install Koleksi MikroTik dari Ansible Galaxy

ansible-galaxy collection install community.routeros
```

6. Ansible Playbook

Definisi & Pemahaman: Playbook adalah berkas blueprint otomasi Ansible yang ditulis dalam format YAML. Playbook mendefinisikan *tasks* (tugas) yang akan dieksekusi pada grup *hosts* tertentu yang terdaftar di dalam file `inventory`.

7. YAML & Script Konfigurasi IP Address Static (Ubuntu)

Definisi & Pemahaman: YAML (YAML Ain't Markup Language) adalah format serialisasi data yang ramah manusia dan digunakan secara luas pada Ansible serta Netplan di Ubuntu.

Contoh YAML Konfigurasi IP Static Netplan (`/etc/netplan/01-netcfg.yaml`):

```
network:

  version: 2

  renderer: networkd

  ethernets:

    eth0:
```

```
      dhcp4: no
      addresses:
        - 192.168.10.2/24
      routes:
        - to: default
          via: 192.168.10.1
      nameservers:
        addresses: [8.8.8.8, 1.1.1.1]
```

Terapkan konfigurasi dengan perintah: `sudo netplan apply`

8. WSL (Windows Subsystem for Linux) Ubuntu

Definisi & Pemahaman: WSL adalah fitur Microsoft Windows yang memungkinkan pengembang menjalankan lingkungan Linux (seperti Ubuntu Server) langsung di atas Windows tanpa *overhead* mesin virtual (VMware/VirtualBox) yang berat.

Cara Instalasi Dasar (Windows PowerShell Admin):

```
# Jalankan perintah instalasi WSL Ubuntu
wsl --install -d Ubuntu


# Setelah restart komputer, buka terminal Ubuntu dan buat pengguna baru:
# Username : wslcontrol
# Password : passwd123
```

BAGIAN 2: PRASYARAT KONEKSI & PROTOKOL AKSEBILITAS

1. Prasyarat Konfigurasi Perangkat Jaringan & End-Device

- **Router Core:** IP Gateway, IP Pool/DHCP, Direct Static Routing/OSPF, & NAT DST-NAT Forwarding.
- **Switch Access:** IP SVI Management, AAA Credentials (Username/Secret), Telnet/SSH Line VTY, serta Port Trunking/Access VLAN.
- **End-Device / Server:** IP Address Static, Default Gateway, WinRM (Windows) atau SSH Daemon (Linux).

2. Protokol SSH & Keamanan Switch (KexAlgorithms & Diffie-Hellman)

Definisi: SSH (Secure Shell) adalah protokol jaringan kriptografi untuk mengoperasikan layanan jaringan secara aman di atas jaringan yang tidak aman.

Masalah Algoritma Kriptografi pada Switch Lama / Emulasi (vIOS-L2): Switch Cisco versi lama sering kali hanya mendukung algoritma pertukaran kunci tua (*legacy*) seperti `diffie-hellman-group1-sha1`. Host Ubuntu/Windows modern secara default menolak algoritma ini demi keamanan.

Cara Kerja Menyutujui (Bypass) di Host Linux (Ubuntu/WSL):

```
ssh -o KexAlgorithms=+diffie-hellman-group1-sha1 -o Ciphers=+3des-cbc
admin@192.168.1.10 -p 2222
```

3. Protokol WinRM (Windows Remote Management)

Definisi: WinRM adalah protokol manajemen berbasis SOAP yang memungkinkan OS Windows menerima perintah eksekusi skrip jarak jauh dari sistem eksternal (termasuk Ansible).

Cara Mengaktifkan pada Windows Server (PowerShell Admin):

```
# Mengaktifkan layanan WinRM dasar

winrm quickconfig -q


# Membuka otentikasi Unencrypted/Basic & port WinRM HTTPS (5986) / HTTP (5985)

Set-Item WSMan:\localhost\Service\AllowUnencrypted -Value true

Set-Item WSMan:\localhost\Service\Auth\Basic -Value true

New-NetFirewallRule -Name "WinRM-HTTP" -DisplayName "WinRM HTTP" -Enabled True -
Direction Inbound -Protocol TCP -LocalPort 5985
```

4. OSPF Routing Dinamis pada MikroTik

Definisi: Open Shortest Path First (OSPF) adalah protokol routing Link-State dinamis yang digunakan untuk mendistribusikan informasi rute dalam satu Autonomous System (AS).

Perintah Dasar OSPF MikroTik (RouterOS v7):

```
# 1. Buat OSPF Instance

/routing ospf instance add name=ospf-instance-1 router-id=1.1.1.1
```

```
# 2. Buat OSPF Area (Area 0 / Backbone)
/routing ospf area add name=backbone-area instance=ospf-instance-1 area-id=0.0.0.0

# 3. Mendaftarkan Interface Transit ke OSPF
/routing ospf interface-template add area=backbone-area interfaces=vlan99-transit
```

5. Strategi Backup Sistem

Sebelum membangun otomasi, 3 jenis backup utama yang wajib diterapkan:

1. **Configuration Backup (Text/Script):** File konfigurasi mentah yang mudah dibaca (`/export` pada MikroTik, `show run` pada Cisco).
2. **Binary System Backup:** File cadangan *state* sistem lengkap (`.backup` MikroTik).
3. **State Snapshot:** Menyimpan *point-in-time* dari OS Server/VM secara keseluruhan.

BAGIAN 3: MODUL PRAKTEK DASAR (MANUAL DEPLOYMENT)

STAGE 1: Aksesibilitas Jarak Jauh ke PNETLab (via OpenVPN)

1. **Download Sertifikat & Client OpenVPN:**
   - Sertifikat Config: https://download.elitbmti.com/OpenVPN/2026.zip
   - OpenVPN Installer: https://download.elitbmti.com/OpenVPN/openvpn-install-2.4.11-I602-Win10.exe
2. **Langkah Instalasi & Ekstraksi:**
   - Install OpenVPN Client pada Windows Anda hingga selesai.
   - Ekstrak seluruh isi file `2026.zip` ke folder: `C:\Program Files\OpenVPN\config`
3. **Koneksi:** Jalankan OpenVPN GUI sebagai **Administrator** -> Klik kanan ikon OpenVPN di tray -> **Connect**. Masukkan Password jika diminta: **`manage123`**
4. **Akses Lab PNETLab:** Buka browser dan ketik alamat IP PNETLab: **`http://192.168.101.56`**

STAGE 2: Konfigurasi Router MikroTik (via CLI / Winbox)

```
# ============================================================
# 1. KONFIGURASI INTERFACE & VIRTUAL LAN (VLAN)
# ============================================================
/interface vlan
```

```
add name=vlan10-tkj vlan-id=10 interface=ether2

add name=vlan20-informatika vlan-id=20 interface=ether2

add name=vlan30-elektronika vlan-id=30 interface=ether2

add name=vlan99-transit vlan-id=99 interface=ether2


# ============================================================

# 2. KONFIGURASI IP ADDRESS

# ============================================================

/ip dhcp-client add interface=ether1 disabled=no

/ip address

add address=192.168.1.1/24 interface=ether2

add address=192.168.10.1/24 interface=vlan10-tkj

add address=192.168.20.1/24 interface=vlan20-informatika

add address=192.168.30.1/24 interface=vlan30-elektronika

add address=10.10.10.1/24 interface=vlan99-transit


# ============================================================

# 3. KONFIGURASI DHCP SERVER POOL

# ============================================================

/ip pool

add name=pool-ether2 ranges=192.168.1.10-192.168.1.20

add name=pool-vlan10 ranges=192.168.10.10-192.168.10.20

add name=pool-vlan20 ranges=192.168.20.10-192.168.20.20

add name=pool-vlan30 ranges=192.168.30.10-192.168.30.20

add name=pool-vlan99 ranges=10.10.10.10-10.10.10.20


/ip dhcp-server

add name=dhcp-ether2 interface=ether2 address-pool=pool-ether2 disabled=no

add name=dhcp-vlan10 interface=vlan10-tkj address-pool=pool-vlan10 disabled=no

add name=dhcp-vlan20 interface=vlan20-informatika address-pool=pool-vlan20
disabled=no

add name=dhcp-vlan30 interface=vlan30-elektronika address-pool=pool-vlan30
disabled=no

add name=dhcp-vlan99 interface=vlan99-transit address-pool=pool-vlan99 disabled=no
```

```
/ip dhcp-server network

add address=192.168.1.0/24 gateway=192.168.1.1

add address=192.168.10.0/24 gateway=192.168.10.1

add address=192.168.20.0/24 gateway=192.168.20.1

add address=192.168.30.0/24 gateway=192.168.30.1

add address=10.10.10.0/24 gateway=10.10.10.1


# ========================================================

# 4. FIREWALL NAT & PORT FORWARDING (DST-NAT)

# ========================================================

/ip firewall nat

# NAT Internet

add chain=srcnat out-interface=ether1 action=masquerade comment="NAT Internet"


# Forward SSH Switch Cisco (2222 -> 22 @ 192.168.1.10)

add chain=dstnat protocol=tcp dst-port=2222 action=dst-nat to-
addresses=192.168.1.10 to-ports=22 comment="SSH Cisco Switch"


# Forward SSH Ubuntu Server (2211 -> 22 @ 192.168.10.2)

add chain=dstnat protocol=tcp dst-port=2211 action=dst-nat to-
addresses=192.168.10.2 to-ports=22 comment="SSH Ubuntu Server"


# Forward HTTP Ubuntu Nginx Server (8080 -> 80 @ 192.168.10.2)

add chain=dstnat protocol=tcp dst-port=8080 action=dst-nat to-
addresses=192.168.10.2 to-ports=80 comment="HTTP Nginx Server"


# Forward WinRM Windows Server (5986 -> 5986 @ 192.168.20.2)

add chain=dstnat protocol=tcp dst-port=5986 action=dst-nat to-
addresses=192.168.20.2 to-ports=5986 comment="WinRM Windows Server"


# Forward HTTP Windows IIS Server (8081 -> 80 @ 192.168.20.2)

add chain=dstnat protocol=tcp dst-port=8081 action=dst-nat to-
addresses=192.168.20.2 to-ports=80 comment="HTTP IIS Server"
```

```
# Forward RDP Windows Server (3389 -> 3389 @ 192.168.20.2)

add chain=dstnat protocol=tcp dst-port=3389 action=dst-nat to-
addresses=192.168.20.2 to-ports=3389 comment="RDP Windows Server"


# ============================================================

# 5. ROUTING OSPF DINAMIS

# ============================================================

/routing ospf instance add name=ospf-instance-1 router-id=1.1.1.1

/routing ospf area add name=backbone-area instance=ospf-instance-1 area-id=0.0.0.0

/routing ospf interface-template add area=backbone-area interfaces=vlan99-transit
```

STAGE 3: Konfigurasi Switch Cisco (vIOS-L2) via CLI

```
enable

configure terminal


! 1. Pembuatan Database VLAN

vlan 10

 name tkj

vlan 20

 name informatika

vlan 30

 name elektronika

vlan 99

 name transit

exit


! 2. Konfigurasi SVI Management (VLAN 1 Native)

interface vlan 1

 ip address 192.168.1.10 255.255.255.0

 no shutdown

exit

ip default-gateway 192.168.1.1
```

```
! 3. Kredensial, Domain Name, RSA Key, & SSH v2
ip domain-name bmti.local
crypto key generate rsa modulus 2048
username admin privilege 15 secret password123
ip ssh version 2

line vty 0 4
 login local
 transport input ssh
exit

! 4. Konfigurasi Mode Interface Trunk & Access
! Port Gi0/0 (Trunk to MikroTik)
interface GigabitEthernet0/0
 switchport trunk encapsulation dot1q
 switchport mode trunk
 switchport trunk native vlan 1
 switchport trunk allowed vlan 1,10,20,30,99
 no shutdown
exit

! Port Access VLAN 10 (tkj)
interface GigabitEthernet0/1
 switchport mode access
 switchport access vlan 10
 no shutdown
exit

interface GigabitEthernet1/0
 switchport mode access
 switchport access vlan 10
 no shutdown
```

```
exit

! Port Access VLAN 20 (informatika)
interface GigabitEthernet0/2
 switchport mode access
 switchport access vlan 20
 no shutdown
exit

interface GigabitEthernet1/1
 switchport mode access
 switchport access vlan 20
 no shutdown
exit

! Port Access VLAN 30 (elektronika)
interface GigabitEthernet0/3
 switchport mode access
 switchport access vlan 30
 no shutdown
exit

! Port Access VLAN 99 (transit)
interface GigabitEthernet1/3
 switchport mode access
 switchport access vlan 99
 no shutdown
exit

end
write memory
```

STAGE 4: Konfigurasi Server Linux & Windows Server

1. Linux Ubuntu Server

Kredensial Default: **Username:** `adminlocal` | **Password:** `adminpnet`

A. Konfigurasi IP Address Static, Hostname, & DNS Resolv:

Buka file Netplan: `sudo nano /etc/netplan/00-installer-config.yaml`

```
network:
  version: 2
  renderer: networkd
  ethernets:
    ens3:
      dhcp4: no
      addresses:
        - 192.168.10.2/24
      routes:
        - to: default
          via: 192.168.10.1
      nameservers:
        addresses: [8.8.8.8, 1.1.1.1]
```

Terapkan dan ubah hostname:

```
sudo netplan apply

sudo hostnamectl set-hostname ubuntusrv
```

B. Instalasi & Konfigurasi Nginx Web Server:

```
sudo apt update && sudo apt install -y nginx

echo "<h1>Linux Ubuntu Server - Otomatisasi BMTI</h1>" | sudo tee
/var/www/html/index.html

sudo systemctl restart nginx
```

2. Windows Server 2019

Kredensial Default: **Username:** `Administrator` | **Password:** `Manager@123`

A. Konfigurasi IP Static & Hostname (PowerShell Admin):

```
# Set IP Address Static & Gateway

New-NetIPAddress -InterfaceAlias "Ethernet" -IPAddress 192.168.20.2 -PrefixLength
24 -DefaultGateway 192.168.20.1

Set-DnsClientServerAddress -InterfaceAlias "Ethernet" -ServerAddresses
("8.8.8.8","1.1.1.1")



# Ganti Hostname menjadi winsrv

Rename-Computer -NewName "winsrv" -Force
```

B. Instalasi Web Server IIS & Custom Index Page:

```
# Install Feature Web-Server (IIS)

Install-WindowsFeature -name Web-Server -IncludeManagementTools



# Buat Konten HTML Custom

Set-Content -Path "C:\inetpub\wwwroot\index.html" -Value "<h1> Windows 2019 Server
IIS - Otomatisasi BMTI</h1>"
```

STAGE 5: Setup WSL-Ubuntu & Manual System Backup

1. Setup WSL-Ubuntu Kontroller

Jalankan di terminal WSL Ubuntu lokal Komputer Peserta:

- **Username:** `wslcontrol`
- **Password:** `passwd123`

2. Eksekusi Manual Backup Perangkat & Simpan di WSL

Jalankan perintah-perintah berikut langsung di dalam terminal WSL Ubuntu untuk mencadangkan konfigurasi seluruh sistem:

```
# Buat Direktori Tempat Backup

mkdir -p ~/backup_infrastruktur && cd ~/backup_infrastruktur
```

```
# A. Backup MikroTik Configuration via SSH

sshpass -p "admin" ssh -o StrictHostKeyChecking=no
admin@192.168.101.xx(WAN/Internet) -p 22 "/export file=backup_mikrotik"

sshpass -p "admin" scp -P 22 admin@192.168.101.56:backup_mikrotik.rsc
./mikrotik_core.rsc



# B. Backup Cisco Switch Configuration via SSH (Bypass Algoritma Kripto)

sshpass -p "password123" ssh -o KexAlgorithms=+diffie-hellman-group1-sha1 -o
Ciphers=+3des-cbc -o StrictHostKeyChecking=no admin@192.168.101.xx -p 2222 "show
running-config" > ./cisco_switch.cfg



# C. Backup HTML Content Ubuntu Server via SCP

sshpass -p "adminpnet" scp -P 2211 -o StrictHostKeyChecking=no
adminlocal@192.168.101.xx:/var/www/html/index.html ./index_ubuntu_backup.html
```

3. Verifikasi Pengujian Layanan Web dari WSL-Ubuntu

Lakukan *testing* akhir dari terminal WSL-Ubuntu untuk memastikan seluruh NAT Forwarding dan Service berjalan sempurna:

```
# Cek Nginx Server Ubuntu (Port 8080 via NAT Router)

curl http://192.168.101.56:8080



# Cek IIS Server Windows (Port 8081 via NAT Router)

curl http://192.168.101.56:8081
```

Status Modul: Praktikum Manual Berhasil. Seluruh infrastruktur telah terkonfigurasi dan teruji siap untuk dilanjutkan ke tahap Otomatisasi Playbook/Script Ansible.