

Pemanfaatan Ansible dalam Sistem Otomatisasi Dan Monitoring Jaringan Komputer

Loudry Tivvany Aijen¹, Rina Dewi Indahsari^{1*}

¹Prodi Teknik Informatika, Fakultas Teknologi dan desain Institut Teknologi dan Bisnis Asia
Malang
loudrytivvany14@gmail.com,²rinadewi@asia.ac.id

ABSTRAK

Perangkat jaringan yang telah terpasang terkadang perlu dikonfigurasi ulang agar dapat terus beroperasi dengan baik. Di suatu perusahaan dalam melakukan konfigurasi perangkat jaringan masih menggunakan cara konfigurasi manual. Melakukan konfigurasi manual membutuhkan waktu yang lebih lama jika perangkat yang digunakan dalam jumlah yang cukup banyak serta dapat menimbulkan kompleksitas yang tinggi dan potensi terjadinya human error. Metode yang lebih efisien untuk dapat mengkonfigurasi perangkat-perangkat tersebut secara otomatis yaitu dengan melakukan otomatisasi jaringan. Otomatisasi jaringan merupakan sebuah sistem yang dibuat untuk melakukan konfigurasi ke beberapa router secara bersamaan. Penelitian ini bertujuan untuk melakukan otomatisasi konfigurasi yang diimplementasi oleh Ansible yang hasilnya diuji pada aplikasi simulasi jaringan atau GNS3. Otomasi konfigurasi diimplementasikan melalui skrip pada Ansible yang dihubungkan dengan router melalui Secure Shell (SSH) sehingga konfigurasi dapat dimasukkan kedalam router. Router yang digunakan pada penelitian ini merupakan router virtual yang dibangun menggunakan GNS3. Hasil dari penelitian ini Ansible telah berhasil digunakan untuk mengotomasi konfigurasi router. Berapapun router yang akan dipasang pada jaringan, router tetap dapat di otomasi hanya dengan menggunakan skrip yang telah dibuat pada Ansible. Penerapan otomasi jaringan ini dapat memudahkan untuk konfigurasi perangkat jaringan dengan skala besar secara otomatis serta dapat meminimalisir human error.

Kata Kunci: *otomatisasi jaringan, monitoring jaringan, Ansible, Python.*

ABSTRACT

Installed network devices may require reconfiguration to continue functioning properly. Enterprises still use manual configuration methods when configuring network devices. With large numbers of devices, manual configuration can be time-consuming, highly complex, and prone to human error. A more efficient way to automatically configure these devices is network automation. Network automation is a system created for configuring multiple routers simultaneously. Installed network devices may require reconfiguration to continue functioning properly. Enterprises still use manual configuration methods when configuring network devices. With large numbers of devices, manual configuration can be time-consuming, highly complex, and prone to human error. A more efficient way to automatically configure these devices is network automation. Network automation is a system created for configuring multiple routers simultaneously. The results are tested in a network simulation application or GNS3. Configuration automation is implemented by an Ansible script that connects to the router via Secure Shell (SSH) so that the configuration can be entered into the router. The routers used in this study are virtual routers created with GNS3. As a result of this research, we have successfully used Ansible to automate router configuration. Regardless of the routers installed in your network, routers can only be automated using scripts written in Ansible. Applying network

automation facilitates the automatic configuration of large-scale network devices, minimizing human error.

Keywords: *Network automation, network monitoring, Ansible, Python*

A. PENDAHULUAN

Saat ini perkembangan jaringan komputer sangat pesat, sehingga untuk manajemen jaringan perlu dilakukan secara terus-menerus dan itu merupakan sebuah tantangan bagi suatu perusahaan. Semua perangkat jaringan yang terpasang perlu dilakukan pemeliharaan dan konfigurasi secara berkala, agar jaringan dapat berjalan dengan baik dan tidak menghambat suatu proses dalam perusahaan. Pada cara tradisional, administrator jaringan perlu masuk ke sistem perangkat secara satu persatu, sehingga akan memakan waktu yang lama dan kurang efisien[1]. Pemantauan jaringan sering dianggap sebagai kunci dalam memastikan jaringan yang berjalan secara efisien. Namun, seiring bisnis dan operasinya yang semakin berkembang. Pemantauan jaringan berubah jadi tanggung jawab yang menjadi suatu kebutuhan. Pada skala yang lebih besar, masalah jaringan dapat mengakibatkan hambatan yang signifikan untuk pengoperasiannya, maka dari itu terdapat beberapa solusi yang dapat digunakan untuk manajemen jaringan, salah satu diantaranya adalah otomatisasi jaringan komputer.

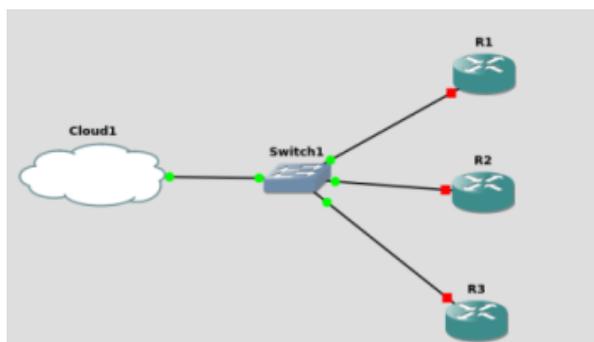
Otomatisasi jaringan komputer adalah sebuah sistem yang melakukan otomatisasi konfigurasi terhadap router yang terhubung atau IP router yang terdaftar pada sistem dengan menggunakan protokol SSH. Otomatisasi jaringan komputer dilakukan untuk mempermudah manajemen perangkat jaringan, dan mempercepat proses konfigurasi perangkat. Dengan adanya otomatisasi jaringan maka untuk melakukan monitoring jaringan akan menjadi lebih mudah dan efektif. Otomatisasi jaringan merupakan solusi untuk melakukan pekerjaan yang rumit dan repetitive. Pekerjaan yang bersifat repetitive seperti melakukan konfigurasi secara otomatis, backup konfigurasi, dan restore konfigurasi. Sistem otomatisasi jaringan dibuat dengan menggunakan Ansible dengan bahasa pemrograman python dan SSH (*Secure Shell Protokol*) sebagai penghubung antara perangkat jaringan dengan server. Ansible merupakan perangkat lunak komputer yang dapat membantu seorang DevOps atau sistem administrator dalam melakukan pekerjaannya seperti instalasi aplikasi, deployment, bahkan mengupdate server, selain itu juga dapat digunakan untuk melakukan otomasi server lainnya[2].

Di suatu perusahaan tertentu dalam melakukan konfigurasi perangkat router masih menggunakan cara manual, yaitu melakukan konfigurasi dengan masuk ke sistem perangkat secara satu persatu.[3] Apabila jumlah perangkat yang harus dikonfigurasi banyak, maka akan membutuhkan waktu yang relatif lama. Jika untuk meminimalisir waktu tentunya akan memerlukan lebih banyak administrator jaringan, namun akan menambah biaya yang dikeluarkan. Untuk mengatasi hal tersebut maka diperlukan sebuah program yang dapat melakukan otomatisasi jaringan. Otomatisasi jaringan dapat dibuat menggunakan Ansible. Dengan adanya otomatisasi jaringan, maka melakukan konfigurasi pada banyak perangkat hanya perlu menjalankan sebuah perintah otomatisasi jaringan, maka semua perangkat yang terhubung akan terkonfigurasi secara otomatis.

B. PEMBAHASAN

Konfigurasi jaringan adalah kegiatan yang berhubungan dengan membangun dan mempertahankan jaringan data. Konfigurasi jaringan mencakup isu-isu yang berkaitan dengan memungkinkan protokol dari perspektif perangkat lunak, dan isu-isu yang berkaitan dengan router, switch, dan firewall dari perspektif hardware. Secara umum untuk melakukan konfigurasi jaringan LAN melalui langkah sebagai berikut:

1. Menentukan perangkat jaringan yang digunakan
2. Melakukan setting IP Address
3. Membuat desain jaringan
4. Konfigurasi Router (IP Address dan Subnet mask)
5. Konfigurasi server
6. Setting router server pada setiap Komputer Client
7. Konfigurasi RIP Routing pada setiap Router
8. Pengujian jaringan



Gambar 1. Topologi Jaringan 3 Router dengan 1 Swicth [4]

Pada gambar 1 digambarkan topologi jaringan dalam sebuah gedung terdapat LAN dengan beberapa perangkat jaringan yaitu 3 router mikrotik dan 1 swicth. Akan dilakukan konfigurasi setiap router dengan menggunakan routing static. Selanjutnya akan dilakukan setting IP Address untuk setiap perangkat jaringan yang digunakan. Dalam hal ini IP Address akan ditentukan untuk 3 Router yang digunakan yaitu R1, R2 dan R3. Setting IP Address ditunjukkan pada tabel 1.

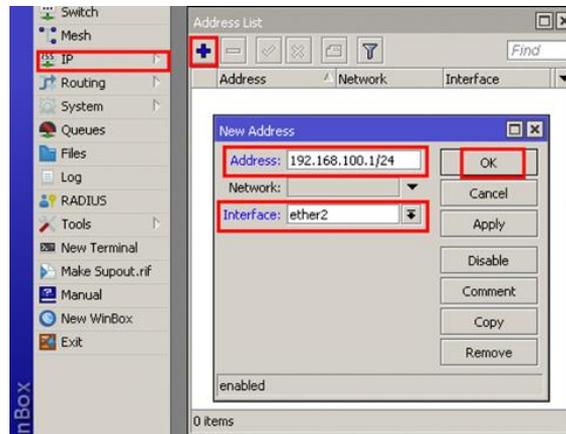
Tabel 1. Setting IP Address untuk R1, R2 dan R3

Gambar	Keterangan	IP
R1	Router 1	192.168.56.2
R2	Router 2	192.168.10.2
R3	Router 3	192.168.1.2

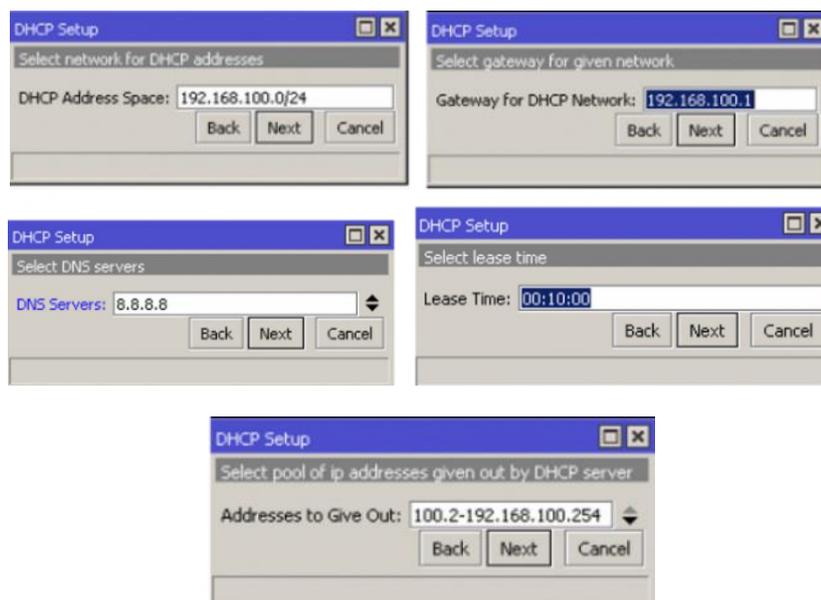
Konfigurasi jaringan secara manual dapat dilakukan dengan cara menggunakan aplikasi WinBox dan SSH. Dalam penelitian ini digunakan Winbox (64 bit) versi 3.27. Aplikasi ini dapat berjalan pada windows, Linux dan MAC OS. Langkah pertama yang harus dilakukan untuk masuk ke dalam konfigurasi router adalah memilih mac address router yang akan dikonfigurasi, masukan user dan password dari router tersebut. Dalam penelitian ini dipilih mac address 08:00:27:30:78:F3. Setiap router dikonfigurasi sendiri-sendiri, sehingga dalam penelitian ini akan dilakukan 3 kali konfigurasi untuk masing-masing router. Winbox sudah memiliki konfigurasi bawaan pabrik atau lebih dikenal dengan RouterOS Default Configuration, tiap routerboard beda beda tergantung perangkat nya tipe apa. Sebelum melakukan konfigurasi manual, maka konfigurasi bawaan pabrik harus di reset (hapus)[5].

Gambar 2 menunjukkan proses setting IP Address di Winbox, tuliskan IP Address yang mengarah ke jaringan LAN, dalam penelitian ini digunakan ether 2. Selanjutkan

membuat DHCP server untuk jaringan lokal. Hal ini bertujuan agar client mendapatkan IP Address secara otomatis dari mikrotik. Ada beberapa step pada DHCP Server yang perlu diperhatikan, yaitu Network Address, Gateway, IP Pool. Setting DHCP Server ditunjukkan pada gambar 3.



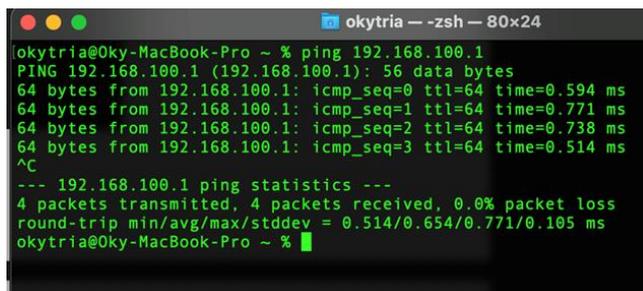
Gambar 2. Setting IP di WinBox



Gambar 3. Setting DHCP Server

DHCP Address Space otomatis akan terisi Network Address nya. Gateway for DHCP Network akan otomatis terisi IP Router (192.168.100.1). Addresses to Give Out adalah Range IP yang akan diberikan oleh client, misalnya 192.168.100.2-254. DNS Servers diisi dengan salah satu DNS Server yang akan digunakan, bisa DNS Provider, DNS Google, dan sebagainya. Lease Time adalah Waktu “sewa” yang akan diberikan oleh client, jika client tidak mengakses jaringan lebih dari waktu itu, maka ip tersebut bisa diberikan ke client lain.

Pada sisi client dipastikan menggunakan IP Otomatis/Obtain/DHCP Client, sehingga mendapatkan IP secara otomatis. Setelah mendapatkan IP pada laptop, pastikan laptop bisa ping ke router untuk verifikasi seperti ditunjukkan pada gambar 4.



```

okytria@oky-MacBook-Pro ~ % ping 192.168.100.1
PING 192.168.100.1 (192.168.100.1): 56 data bytes
64 bytes from 192.168.100.1: icmp_seq=0 ttl=64 time=0.594 ms
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=0.771 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.738 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.514 ms
^C
--- 192.168.100.1 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.514/0.654/0.771/0.105 ms
okytria@oky-MacBook-Pro ~ %

```

Gambar 4. Verifikasi Ping Client ke Router

Otomatisasi Jaringan Menggunakan Ansible

Analisis sistem yang akan dilakukan terhadap manajemen konfigurasi didalam hal proses deploy ke router adalah untuk melakukan konfigurasi dan deployment Router MikroTik dengan menggunakan simulator GNS3 pada topologi jaringan [6]. Dalam penelitian ini dilakukan studi kasus di Laboratorium Komputer Institut Asia Malang. Analisis sistem ini bekerja terhadap proses otomasi yang dilakukan menggunakan Ansible dengan Playbook dan Shell Script dengan Bash. Analisis perbandingan akan dilakukan meliputi beberapa parameter yaitu, Kecepatan Deployment, Penggunaan CPU dan Memory.

1. Analisis Perbandingan Deployment yaitu dengan konfigurasi secara manual dan konfigurasi secara automation menggunakan Ansible.
2. Analisis perbandingan penggunaan CPU dan Memory akan dilakukan dengan menggunakan Software Netdata.

Hasil analisis dari kebutuhan perangkat jaringan untuk topologi jaringan komputer pada laboratorium komputer Institut Asia Malang ditunjukkan pada tabel 2.

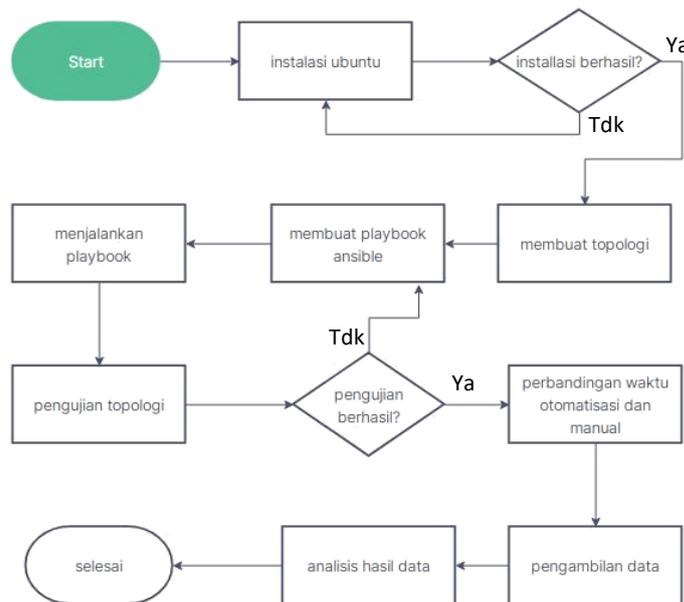
Tabel 2 Kebutuhan Perangkat Jaringan

No	Nama Barang	Jumlah
1	PC Server	1
2	Switch	1
3	Kabel RJ45	5
4	Router Mikrotik	3

Skenario Simulasi Otomasi Jaringan

Simulasi akan dilakukan dalam beberapa tahap yang mengacu pada diagram alur gambar 5. Tahap pertama yaitu instalasi appliance Ubuntu Network Docker dan yang mendukung Automation pada perangkat lunak GNS3. Dengan Appliance tersebut, playbook ansible akan dijalankan. Jika terjadi error atau gagal, maka proses instalasi akan diulang, Jika berhasil maka akan ke tahap pembuatan topologi jaringan. Salah satu interface dari router dihubungkan ke Router. Penjelasan topologi akan dijelaskan pada uraian pada pembahasan berikutnya.

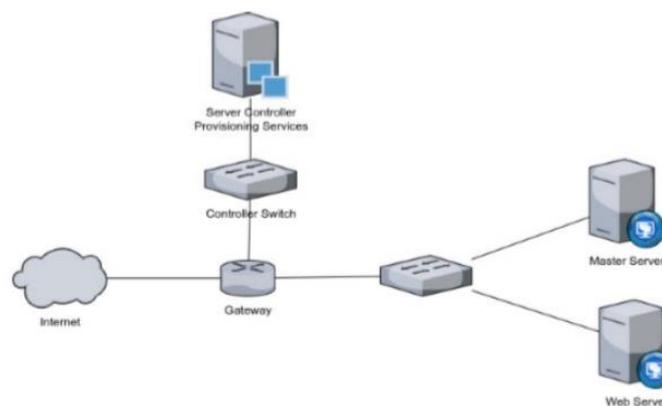
Proses selanjutnya adalah membuat script otomasi jaringan sesuai dengan topologi dengan menggunakan shell script (BASH). Script tersebut dibuat di Ubuntu Network Automation yang mendukung network automation. Setelah pembuatan Script selesai maka tahap selanjutnya adalah melakukan pengujian pada Playbook, jika gagal maka akan kembali ke pengecekan pada Script didalam Playbook, namun jika berhasil maka ke tahap selanjutnya yaitu pengambilan data dengan melakukan perbandingan waktu antara otomasi dan manual dan juga akan dihitung penggunaan pemakaian cpu dan memory pada saat otomasi, setelah itu akan dianalisis dan ditarik suatu kesimpulan [7].



Gambar 5. Alur Simulasi Otomatisasi Jaringan

Desain Topologi Jaringan untuk Simulasi

Desain topologi jaringan sebelum di otomasi belum memiliki server otomatisasi sehingga dalam pengelolaan dibutuhkan untuk mengkonfigurasi pada masing-masing perangkat jaringan komputer. Pada gambar 6 ditunjukkan topologi jaringan setelah di otomasi. Topologi ditambahkan server otomatisasi yang berfungsi sebagai pusat konfigurasi untuk mengirimkan konfigurasi pada perangkat jaringan yang dituju.



Gambar 6. Topologi Jaringan setelah di otomasi [8]

Tahap simulasi dilakukan menggunakan Virtualbox. Spesifikasi node untuk simulasi digunakan master server (4GB RAM, 40 GB dan 2 Processor) dan Router Mikrotik (64MB RAM, 20GB dan 1 Processor). Adapun spesifikasi komputer yang digunakan untuk pengujian simulasi melalui VirtualBox ini mempunyai spesifikasi 16 GB RAM.

Implementasi Otomatisasi Jaringan Menggunakan Ansible

Implementasi dimulai ketika tahap simulasi prototipe tidak mengalami kegagalan dan gangguan sama sekali. Pada tahap ini PC server menggunakan Ubuntu 18.04 sebagai sistem operasi default untuk menjalankan Ansible, dengan memanfaatkan 1 PC di laboratorium

komputer Institut Asia Malang untuk dijadikan sebagai server. Server yang ada diinstall Ubuntu 18.04 minimal install. Proses install ini membutuhkan sekitar 30 menit ditahap instalasi. Topologi yang dipakai sama dengan yang telah direncanakan sebelumnya pada gambar 6, setelah semua alat – alat yang dibutuhkan terpenuhi maka tahap implementasi dapat dijalankan [9].

Ansible diinstall pada server master kemudian dilakukan tahap instalasi ssh key pair untuk menghubungkan antara server master dan server lain, tak lupa juga memberikan ssh key pair ke router mikrotik agar router dapat terhubung ke server master. Setelah siap semua tahap otomatisasi dengan Ansible dapat dilaksanakan. Otomatisasi perangkat jaringan ini menggunakan Ansible Playbook untuk mengkonfigurasi masing–masing host. Berikut contoh Ansible Playbook sederhana[10].

```
---
- hosts: r1
  connection: network_cli
  gather_facts: no
  tasks:
    - name: Menambahkan IP Address Ether2
      routeros_command:
        commands: /ip address add address=192.168.10.1/24
        interface=ether2
```

Pada bagian “routeros command” terdapat perintah untuk menambahkan ip address pada router mikrotik, apabila ingin menambah command hal tersebut sangat mungkin dilakukan dengan menambahkan beberapa command mikrotik yang diinginkan. Pada bagian server terdapat pula Ansible Playbook yang harus dijalankan, untuk melakukan instalasi paket – paket untuk web server seperti contoh berikut.

```
---
- name: install and start httpd
  hosts: web
  tasks:
    - name: install epel repo
      yum: name=epel-release state=latest
    - name: install httpd
      yum: name=httpd state=latest
      notify: restart httpd
    - name: start httpd service
      service: name=httpd state=started enabled=yes
```

Pada bagian hosts terdapat keterangan “web”, hal tersebut dimaksudkan untuk mengkonfigurasi host web dan host lainnya tidak ikut terkonfigurasi.

Tahap implementasi pemanfaatan ansible untuk melakukan otomatisasi yang akan dilakukan pada penelitian ini dengan menggunakan mesin simulator GNS3[11] dan akan diimplementasi dengan menggunakan topologi pada laboratorium komputer Institut Asia Malang. Tahap–tahap yang akan dilakukan adalah melakukan pengujian kecepatan deployment antara lain adalah melakukan konfigurasi secara manual atau dengan konfigurasi menggunakan CLI dan menggunakan tools Ansible dengan cara otomatisasi setelah itu dibuat perbandingan dengan menggunakan tabel komparasi dan dari situlah akan ditarik kesimpulan mana yang lebih baik antara manualisasi atau otomatisasi. Tahapan–tahapannya sebagai berikut:

1. Melakukan Instalasi GNS3

GNS3 adalah aplikasi simulator jaringan (*Graphic Simulator Network*) berbasis GUI software yang bisa mensimulasikan perangkat asli baik dengan bantuan emulator ataupun teknologi virtualisasi. Namun untuk melakukan simulasi perlu membuat server GNS3 sendiri agar saat simulasi perangkat tidak membebani komputer. Maka

pada penelitian ini akan ada server terpisah yang akan digunakan untuk menjalankan GNS VMWARE dan juga ada GNS3 yang menjadi Client.

Tahap pertama adalah melakukan update dan upgrade dengan cara mengetikkan perintah berikut ini di terminal linux

```
sudo apt update
sudo apt upgrade
```

Setelah itu mengetikkan perintah reboot

```
sudo reboot
```

Install software yang diperlukan untuk penginstalan GNS3

```
sudo apt install -y python3-pip python3-pyqt5 python3-pyqt5.qtsvg python3-pyqt5.qtwebsockets qemu qemu-kvm qemu-utils libvirt-clients libvirt-daemon-system virtinst wireshark xtightvncviewer apt-transport-https ca-certificates curl gnupg2 software-properties-common
```

Install GNS3

```
sudo pip3 install gns3-server
sudo pip3 install gns3-gui
```

Untuk membuka aplikasi simulator GNS3 dapat menggunakan terminal kali linux atau melalui pencarian di kali linux.

2. Instalasi Virtualbox

Virtualbox atau perangkat lunak virtualisasi adalah perangkat lunak yang dapat digunakan untuk melakukan install sistem operasi tambahan di dalam sistem operasi utama. Virtualbox juga dapat digunakan untuk melakukan simulasi dan uji coba instalasi suatu sistem tanpa harus kehilangan sistem yang ada. Sebelum melakukan install virtualbox, pastikan kali linux sudah up-to-date, setelah itu lakukan reboot.

```
Sudo apt update
Sudo apt full-upgrade -y
[ -f /var/run/reboot-required ] && sudo reboot -f
```

Pertama-tama yang harus dilakukan adalah mengimport virtualbox repository key

```
wget -q https://www.virtualbox.org/download/oracle_vbox_2016.asc -O- | gpg --dearmor | sudo tee /usr/share/keyrings/virtualbox-archive-keyring.gpg
```

Kemudian menambahkan repositori virtualbox. Lalu menambahkan ke file terpisah, sehingga tidak mengganggu repositori utama kali linux.

```
echo "deb [arch=amd64 signed-by=/usr/share/keyrings/virtualbox-archive-keyring.gpg]
http://download.virtualbox.org/virtualbox/debian buster contrib" \ | sudo tee /etc/apt/sources.list.d/virtualbox.list
```

Karena VirtualBox memiliki berbagai modul kernel (mis. vboxdrv, vboxnetflt dan vboxnetadp), perlu memastikan modul-modul tersebut tetap mutakhir ketika kernel Kali Linux diperbarui. Ini dapat dicapai dengan menggunakan dkms. Cara install dkms menggunakan kode `sudo apt install -y dkms`

Tahap terakhir yaitu install virtualbox dengan menuliskan kode `Sudo apt install -y virtualbox virtualbox-ext-pack`

3. Install mikrotik di virtualbox

Untuk melakukan simulasi pada GNS3 perlu dilakukan penginstalan mikrotik pada virtualbox.

4. Integrasi Router Dengan Ansible

Untuk menggunakan Ansible automation yang perlu dilakukan adalah memastikan agar mesin otomatisasi dapat terhubung dengan host. Cara memastikan mesin telah terhubung atau tidak dengan menggunakan tools ping dengan cara ping ke setiap host, namun dikarenakan pada penelitian ini topologi existing belum ada maka harus dilakukan konfigurasi dari awal agar setiap host bisa terhubung dengan mesin otomatisasi.

5. Konfigurasi Router

Pada Router yang harus dilakukan pertama adalah konfigurasi Hostname Default dari Router, setelah itu memasang Ip Address pada tiap interface yang terhubung dan daftar routing static untuk semua device. Gambar 7 menunjukkan menunjukkan proses konfigurasi Router.

```
[admin@RouterFora] > ip route add dst-address=192.168.3.0/24 gateway=192.168.2.2
[admin@RouterFora] > ip route add dst-address=192.168.4.0/24 gateway=192.168.2.2
[admin@RouterFora] > ip route add dst-address=192.168.5.0/24 gateway=192.168.2.2
[admin@RouterFora] > ip route add dst-address=192.168.6.0/24 gateway=192.168.2.2
[admin@RouterFora] > ip route add dst-address=192.168.7.0/24 gateway=192.168.2.2

[admin@RouterFora] > ip route ps
Flags: X - disabled, A - active, D - dynamic,
C - connect, S - static, r - rip, b - bgp, o - ospf, m - mme,
B - blackhole, U - unreachable, P - prohibit
# DST-ADDRESS    PREF-SRC  GATEWAY    DISTANCE
0 ADC 192.168.1.0/24  192.168.1.1 ether1      0
1 ADC 192.168.2.0/24  192.168.2.1 ether2      0
2 A S 192.168.3.0/24  192.168.2.2 1
3 A S 192.168.4.0/24  192.168.2.2 1
4 A S 192.168.5.0/24  192.168.2.2 1
5 A S 192.168.6.0/24  192.168.2.2 1
6 A S 192.168.7.0/24  192.168.2.2 1
[admin@RouterFora] >

[admin@RouterFora] > password
old-password: **
new-password: ****
confirm-new-password: ****
[admin@RouterFora] >
```

Gambar 7. Konfigurasi Router

6. Integrasi Ansible

Pada tahap integrasi Ansible ada beberapa point penting yang harus dilakukan di dalam Mesin Ansible yang pertama yaitu memastikan koneksi antara Mesin Ansible dengan Router Smile Project sudah saling terhubung, untuk memastikan Mesin Ansible sudah terhubung adalah dengan melakukan test ping kepada alamat-alamat ip dari setiap Router jika berhasil selanjutnya Manage Machines Ansible. Manajemen mesin meliputi membuat Inventory, Public keys, import public key, dan playbook setelah proses itu selesai maka selanjutnya tahap Deploy.

Pengujian Fungsionalitas Otomatisasi Jaringan

Tahap uji fungsionalitas ini didasarkan pada setting dasar mikrotik dan setting dasar penginstallan paket web server, adapun yang membedakan dengan setting mikrotik dan penginstallan paket web server pada umumnya ialah pada saat proses konfigurasi, jika pada umumnya setting dasar mikrotik dan penginstallan web server dengan cara tradisional yakni mikrotik menggunakan winbox dan web server dengan langsung melakukan konfigurasi

melalui mesin yang diinstall dengan web server tersebut. Hal ini tidak dilakukan sepenuhnya dalam otomatisasi perangkat jaringan. Beberapa hal yang menjadi acuan untuk pengujian keberhasilan fungsional pada perangkat jaringan otomatisasi menjadi 15 aspek penilaian yang ditunjukkan pada tabel 3.

Tabel 3. Pengujian

No	Aspek Pengujian	Hasil pengujian
1	Semua server dapat terkoneksi ke Internet	Berhasil
2	Ping antar server berhasil server master dapat menjalankan role Ansible	Berhasil
3	server master dapat menjalankan role Ansible	Berhasil
4	SSH antar server berhasil	Berhasil
5	Server ter-cluster dengan server lain	Berhasil
6	Generate SSH dari server master ke slave server	Berhasil
7	Menginstall paket web server dari server master menggunakan Ansible	Berhasil
8	Menjalankan Playbook	Berhasil
9	Menjalankan paket web server dari server master	Berhasil
10	Cek konfigurasi server lain melalui server master	Berhasil
11	Ping router dan server master	Berhasil
12	Konfigurasi IP Address pada Mikrotik dapat dilakukan melalui server master	Berhasil
13	Konfigurasi DHCP server via server master	Berhasil
14	Konfigurasi NAT pada router mikrotik melalui server master	Berhasil
15	Menambahkan User baru melalui server master	Berhasil

Dari hasil pengujian fungsional yang dilakukan baik untuk otomatisasi router dan server berjalan dengan baik. Ansible telah melakukan tugasnya untuk mengotomatisasi perangkat jaringan komputer di laboratorium komputer tanpa mengalami kegagalan. Bukan hanya pada server saja, akan tetapi otomatisasi dapat dilakukan pada router mikrotik yang telah dikonfigurasi sedemikian rupa sehingga bisa diotomatisasi dengan Ansible.

Pengujian Kecepatan Otomatisasi Jaringan

Pada pengujian kecepatan ini akan dibandingkan dengan dua metode yaitu dengan menggunakan Ansible Automation dan dengan mencoba dengan tanpa Ansible atau Manualisasi. Pada kasus ini yang akan dilakukan pada router adalah menambah IP Address, Membuat DHCP, dan membuat Security, IP Address dan DHCP akan ditambahkan pada RouterSiko, RouterGafi, RouterKayoa, RouterGuraici, RouterLelei dan untuk security akan di buat pada semua Router.

A. Pengujian Tanpa Ansible (Manualisasi)

Untuk pengujian tanpa menggunakan ansible ini memakan waktu 5-10 menit per router. Untuk menyelesaikan semua konfigurasi tersebut yang harus dilakukan adalah membuat ip address, ip dhcp-server, dan membuat security proses tersebut dilakukan berulang-ulang di setiap router oleh karena itu memakan waktu yang cukup lama. Pada penelitian ini router yang digunakan ada tujuh router namun jika ada kasus yang lebih dari tujuh router atau ratusan router yang harus di konfigurasi maka akan lebih memakan waktu lebih lama.

B. Pengujian dengan Ansible (Automation)

Pengujian yang dilakukan menggunakan Ansible Automation kali ini dengan konfigurasi yang telah dijelaskan waktu yang dibutuhkan untuk konfigurasi semua Router hanya membutuhkan waktu 2 menit. Berikut dibawah adalah Script yang di jalankan pada Ansible [12].

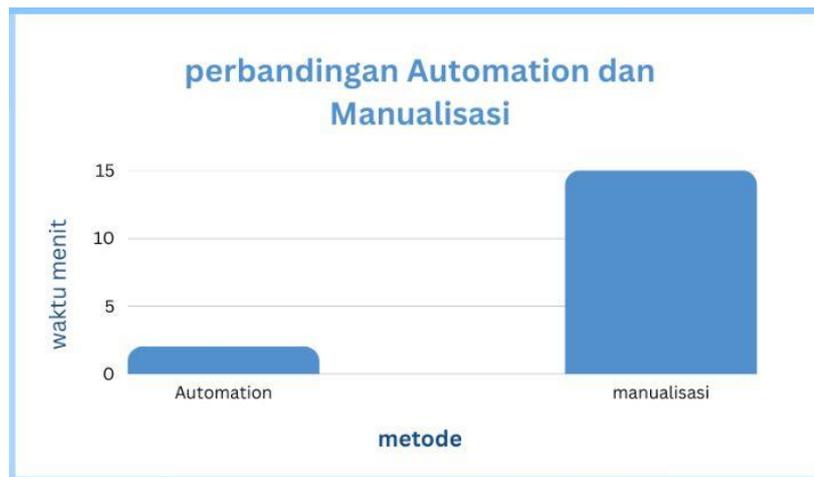
```
- name : menambah ip dan membuat DHCP server
hosts : RouterSiko, RouterGafi, RouterGuraici, RouterKayoa,
RouterLelei
tasks :
- name : Menambah Ip address
routeros_command :
commands : /ip address add address=10.10.10.1/24 interface=ether2
- name : membuat ip pool
routeros_command :
commands : /ip pool add ranges=10.10.10.2-10.10.10.50 name=pool2
- name : menambah DHCP-Server
routeros_command :
commads : /ip dhcp-server
add interface=ether2
address-pool=pool2
name=dhcp1
- name : mengaktifkan dhcp-server
routeros_command :
commands : /ip dhcp-server enable numbers=0[13]
```

Script yang dijalankan di Ansible dengan nama Playbook Management_Router.yml di dalam playbook tersebut ada dua host yang digunakan yaitu host RouterSiko, RouterGafi, RouterGuraici, RouterKayoa, RouterLelei, dan yang kedua adalah host routers dengan menggunakan variable group pada file inventory di ansible yang telah dibuat sebelumnya. Tujuan dari menggunakan dua host agar saat ingin deployment Router konfigurasi tidak akan dipush kesetiap router karena pada kasus kali ini host yang pertama akan hanya dikonfigurasi dengan menambah ip address dan membuat dhcp-server sedangkan pada host ke dua untuk security akan di jalankan ke semua router oleh karena itu pada host ke dua hanya menuliskan host dengan nama group yang telah terdaftar pada inventory. Berikut script yang dijalankan dengan nama playbook.

```
- name : security
hosts : routers
tasks :
- name : menonaktifkan RouterOS
Mac-Access routeros_command :
commands :
- /tool mac-server
set allowed-interface-list=none
- /tool mac-server
mac-winbox set allowed-interfacelist=none
- /tool mac-server
ping set enable=no
- name : menonaktifkan client service
routeros_command :
commands :
- /ip proxy set enable=no
- /ip socks set set enable=no
- /ip cloud set ddns-enable=no
update-time=no
- name : menonaktifkan ip service
routeros_command :
commands: /ip service disable
```

```
{{item}} with_items:  
- api  
- api  
-ssl  
- ftp
```

Beda dengan cara manual dengan melakukan konfigurasi berulang-ulang dan itu akan memakan waktu yang lama. Grafik komprasi dibawah adalah perbandingan antara dua metode tersebut bisa dilihat perbedaan antara dua metode tersebut berbeda jauh[14]. Grafik perbandingan waktu untuk konfigurasi secara manual dan secara otomatis ditunjukkan pada gambar 8.



Gambar 8. Komparasi Konfigurasi Manual dan Otomatis

Pada gambar 8 terlihat perbandingan waktu konfigurasi dari automation dan manualisasi yang sangat berbeda. Untuk automation untuk melakukan routing ke 3 router memerlukan waktu 2 menit, sedangkan untuk manualisasi memerlukan waktu 15 menit untuk melakukan konfigurasi ke 3 router.

C. KESIMPULAN

Otomatisasi jaringan dengan memanfaatkan Ansible sangat memungkinkan untuk dilakukan dengan tujuan untuk mengotomatisasi perangkat jaringan seperti router, server dan lain-lain. Ansible sudah melaksanakan tugasnya dengan baik tanpa adanya kesalahan dalam pengoperasiannya. Dengan memanfaatkan Ansible pekerjaan yang harus dilakukan secara berulang tentu akan menghilang dan dilakukan dengan satu skrip untuk menyelesaikan pekerjaan tersebut. Antara dua metode Automation dan Manual untuk lebih efisiennya pada administrator jaringan yaitu dengan menggunakan cara automation karena untuk melakukan konfigurasi pada setiap touter tidak akan memakan waktu yang lama dan juga terpusat.

Konfigurasi ini perlu dikembangkan sistem otomasi untuk mendukung berbagai tipe perangkat Router Board. Pengembangan Sistem Otomasi dengan Ansible ini bisa dilakukan dengan menggunakan metode QoS pada Router. Melihat kebutuhan dan perkembangan Teknologi, maka Ansible bisa dimanfaatkan lebih luas lagi seperti mengkonfigurasi melalui web atau mobile.

REFERENSI

- [1] A. Hadi, *Administrasi Jaringan Komputer*. 2016.
- [2] R. Abhishek, "Practical Network Automation: A beginner's guide to automating and optimizing networks using Python, Ansible, and more, 2nd Edition," 2018.
- [3] A. R. Komarudin, *Otomstisasi Administrasi Jaringan dengan Script Python*. 2018.
- [4] I. GmbH, "Virtualbox," <https://www.virtualbox.org/>, 2008.
- [5] A. H. Muttaqin, *Teknologi Jaringan Komputer*. 2022.
- [6] A. Affandi, M. R, Hatta, P, Efendi, "Otomatiasi Perangkat Jaringan Komputer Menggunakan Ansible Pada Laboratorium Komputer," *SMARTICS*, vol. 6, 2020.
- [7] A. B. Chaudhuri, "Flowchart and Algorithm Basics - The Art of Programming," 2020.
- [8] J. Forcier, "Paramiko," <http://www.paramiko.org/>.
- [9] I. Kukuh, N., Anggi, D. A., Syariful, "Perbandingan Kinerja Library Paramiko dan Netmiko Dalam Proses Otomatisasi Jaringan," *J. Nas. Inform. dan Teknol. Jar.*, vol. 5, 2020.
- [10] J. Enterprise, *Otodidak Pemrograman Python*. 2017.
- [11] J. Grossmann, "Gns3," <https://www.gns3.com/>, 2008.
- [12] I. Ginting, E. S., Suroso, Hadi, "Pengujian Konfigurasi Otomatis Penambahan Gateway Pada Virtual Router Menggunakan Aplikasi Otomatisasi Jaringan Berbasis Web," *J. Media Inform. Budidarma*, vol. 4, 2020.
- [13] F. Mahaila, P., Balan, T., Curpen, R.Sandu, "Network Automation and Abstraction using Python Programming Methods. Electronic and Computers Department Science," 2017.
- [14] B. A. Sakti, "Uji Kelayakan Implementasi SSH sebagai Pengaman FTP Server dengan Penetration Testing," *J. Teknol. dan Informasi.*, 2013.