

# Tugas 2 Ansible

Disusun oleh:

**Muhammad Amir Al Aqwa**



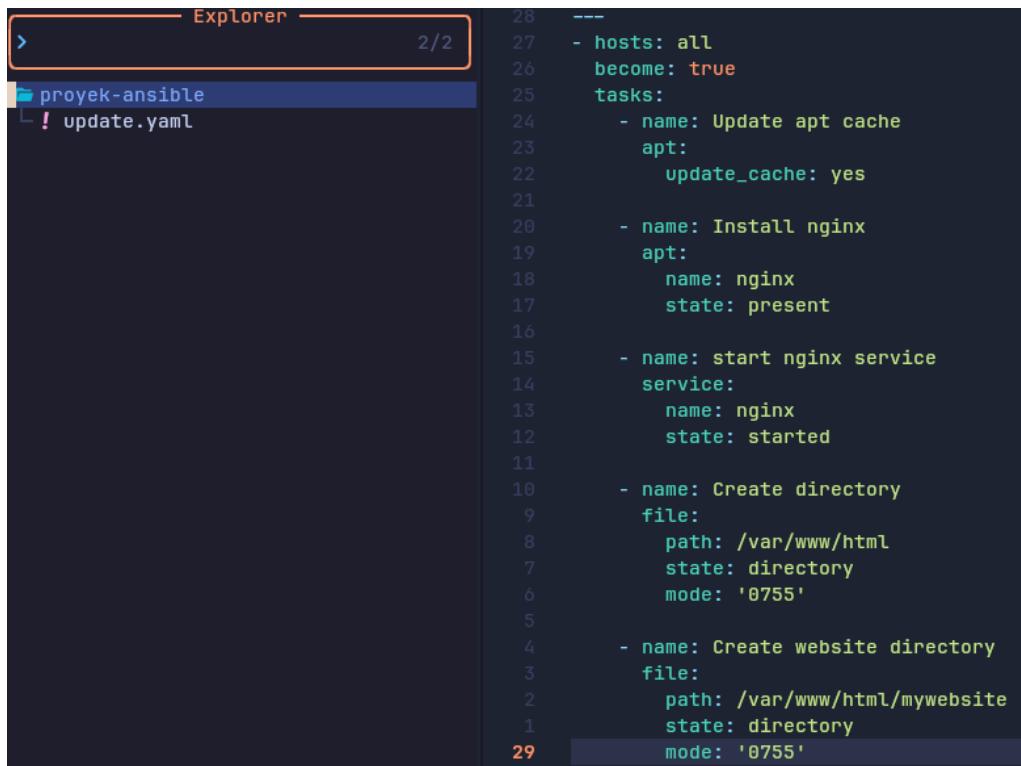
# Pendahuluan

Laporan ini dibuat untuk memenuhi tugas mata kuliah Jaringan Komputer, khususnya pada materi konfigurasi server menggunakan Ansible. Dokumen ini berisi dua bagian utama: (1) pembuatan playbook sederhana untuk menginstal nginx dan membuat direktori, serta (2) implementasi Ansible Roles yang bersifat reusable untuk konfigurasi web server. Setiap tahapan akan dilengkapi dengan screenshot sebagai bukti verifikasi keberhasilan eksekusi.

## Langkah Praktikum

### 1. Pembuatan playbook

1. Pada direktori penggeraan playbook, buat file playbook dengan nama webserver.yaml dengan isi file sebagai berikut.



```
---  
- hosts: all  
  become: true  
  tasks:  
    - name: Update apt cache  
      apt:  
        update_cache: yes  
  
    - name: Install nginx  
      apt:  
        name: nginx  
        state: present  
  
    - name: start nginx service  
      service:  
        name: nginx  
        state: started  
  
    - name: Create directory  
      file:  
        path: /var/www/html  
        state: directory  
        mode: '0755'  
  
    - name: Create website directory  
      file:  
        path: /var/www/html/mywebsite  
        state: directory  
        mode: '0755'
```

2. Jalankan playbook dengan perintah sebagai berikut.

```
ansible-playbook -i /etc/ansible/hosts webserver.yaml --ask-become-pass
```

3. Berikut tanda kalau sudah berhasil.

```

-----
< PLAY RECAP >
-----
\   ^__^
 \  (oo)\----_
  (__)\       )\/\
    ||----w |
    ||     ||

100.109.201.60      : ok=0    changed=0    unreachable=1    failed=0    skipped=0    rescued=0    ignored=0
192.168.100.215     : ok=1    changed=0    unreachable=0    failed=1    skipped=0    rescued=0    ignored=0

```

```

minkwaq@minkwaq-thinkpad ~$ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
  Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
  Active: active (running) since Wed 2025-09-24 13:17:33 WIB; 10s ago
  Invocation: 1ab520dcf04544e7a973fbcb1dbbbba3e
    Docs: man:nginx(8)
   Process: 86283 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0)
   Process: 86285 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0)
 Main PID: 86354 (nginx)
    Tasks: 5 (limit: 8580)
   Memory: 4.5M (peak: 11.1M)
      CPU: 61ms
     CGroup: /system.slice/nginx.service
             ├─86354 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─86356 "nginx: worker process"
             ├─86357 "nginx: worker process"
             ├─86358 "nginx: worker process"
             └─86359 "nginx: worker process"

Sep 24 13:17:33 minkwaq-thinkpad systemd[1]: Starting nginx.service - A high performance web server...
Sep 24 13:17:33 minkwaq-thinkpad systemd[1]: Started nginx.service - A high performance web server...
lines 1-20/20 (END)

```

## 2. Implementasi ansible roles

- Pada control nodes, buat folder kerja baru dan buat juga file dengan struktur berikut.

```

① 0s ~ └▶ ⌂ / ⌂ Kuliah/SIB/8#/proyek-ansible
② 0s ~ └▶ mkdir -p roles/nginx/{tasks,handlers,templates,files,vars,meta,defaults}
③ 0s ~ └▶ tree roles/
roles/
└── nginx
    ├── defaults
    ├── files
    ├── handlers
    ├── meta
    ├── tasks
    ├── templates
    └── vars

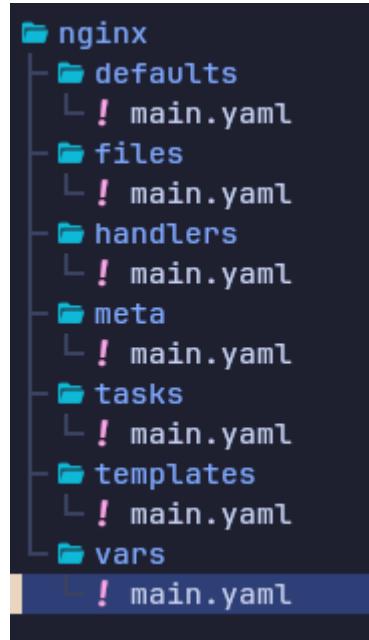
```

- Lalu buat file seperti ini

```

④ 0s ~ └▶ touch roles/nginx/tasks/main.yml roles/nginx/handlers/main.yml roles/nginx/templates/main.yml
      roles/nginx/meta/main.yml

```



3. Isi file main.yaml pada folder defaults,

```
Explorer 1  nginx_port: 80 | 15/15
```

```
nginx
├── defaults
│   └── ! main.yaml
├── files
│   └── ! main.yaml
├── handlers
│   └── ! main.yaml
├── meta
│   └── ! main.yaml
├── tasks
│   └── ! main.yaml
├── templates
│   └── ! main.yaml
└── vars
    └── ! main.yaml
```

4. Folder templates, nginx.conf.j2,

Explorer 10/10

```

nginx
├── defaults
├── files
├── handlers
├── meta
├── tasks
└── templates
    └── nginx.conf.j2
    └── vars
    └── ! site.yaml

22 worker_processes auto;
21
20 events {
19 |     worker_connections 1024;
18 }
17
16 http {
15 |     include mime.types;
14 |     default_type application/octet-stream;
13 |
12 |     sendfile on;
11 |     keepalive_timeout 65;
10 |
9 |     server {
8 |         listen 80;
7 |         server_name localhost;
6 |
5 |         location / {
4 |             root html;
3 |             index index.html index.htm;
2 |         }
1 |     }
23 }

```

## 5. Folder tasks,

Explorer 15/15

```

nginx
├── defaults
│   └── ! main.yaml
├── files
├── handlers
│   └── ! main.yaml
├── meta
│   └── ! main.yaml
├── tasks
│   └── ! main.yaml
└── templates
    └── ! main.yaml
    └── vars
        └── ! main.yaml
    └── ! site.yaml

! templates/main.yaml ● ! tasks/main.yaml ✘
9 - name: Install Nginx
8 | apt:
7 | | name: nginx
6 | | state: present
5
4 - name: Copy Nginx configuration files
3 | template:
2 | | src: nginx.conf.j2
1 | | dest: /etc/nginx/nginx.conf
10 | notify: Restart Nginx

```

## 6. Folder handlers,

Explorer 15/15

```

nginx
├── defaults
│   └── ! main.yaml
├── files
├── handlers
│   └── ! main.yaml
├── meta
│   └── ! main.yaml
├── tasks
│   └── ! main.yaml
└── templates
    └── ! main.yaml
    └── vars
        └── ! main.yaml
    └── ! site.yaml

! defaults/main.yaml ✘ ! templates/main.yaml ● | ! handlers/main.yaml ✘
8 - name: Restart Nginx
7 | service:
6 | | name: nginx
5 | | state: restarted
4
3 - name: Start Nginx
2 | service:
1 | | name: nginx
9 | | state: started

```

## 7. Folder meta,

The screenshot shows the VS Code interface with the following details:

- Explorer** pane on the left, titled "nginx". It lists several directory and file entries:
  - defaults
  - files
  - handlers
  - meta
  - tasks
  - templates
  - vars
  - site.yaml
- Editor** pane on the right, titled "templates/main.yaml". The content is as follows:

```
! templates/main.yaml • | ! meta/main.yaml x
1 ---[!--]
1   dependencies: []
```

8. Selanjutnya buat playbook dengan isi seperti ini.

The terminal window displays the file structure of an Ansible project. The left pane shows the directory tree:

- ! defaults
- ! files
- ! handlers
- ! meta
- ! tasks
- ! templates
- ! vars
- ! site.yaml

The right pane shows the content of the `site.yaml` file:

```
! defaults/main.yaml x ! templates/main.yaml ● ! tasks/main.yaml  
3 - hosts: minkwaq-thinkpad  
2 | become: yes  
1 | roles:  
4 |   - nginx  
1
```

```
• ► tree
.
└── roles
    └── nginx
        ├── defaults
        │   └── main.yaml
        ├── files
        ├── handlers
        │   └── main.yaml
        ├── meta
        │   └── main.yaml
        ├── tasks
        │   └── main.yaml
        ├── templates
        │   └── nginx.conf.j2
        └── vars
            └── main.yaml
    └── site.yaml
    └── update.yaml
    └── webserver.yaml
```

9. Lalu jalankan playbook.

```
• ► ansible-playbook -i /etc/ansible/hosts site.yaml --ask-become-pass
BECOME password:
```

```
100.109.201.60 : ok=0    changed=0    unreachable=1    failed=0    skipped=0    rescued=0
                  ignored=0
192.168.100.215 : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=0
                  ignored=0
```

10. Lakukan perintah berikut untuk verifikasi.

```
minkwaq@minkwaq-thinkpad ~> curl http://localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
minkwaq@minkwaq-thinkpad ~> |
```

## Penutup

Dengan demikian, laporan ini telah berhasil mendemonstrasikan kemampuan Ansible dalam mengotomatisasi konfigurasi server, mulai dari playbook sederhana hingga penggunaan Ansible Roles untuk pengelolaan yang lebih terstruktur dan efisien. Seluruh task yang diberikan telah berhasil dieksekusi dengan baik, sebagaimana dibuktikan oleh screenshot yang disertakan.