Provisioning Server AWS dengan Ansible

Estu Fardani 20240209

Hello,

Nama saya

Estu Fardani

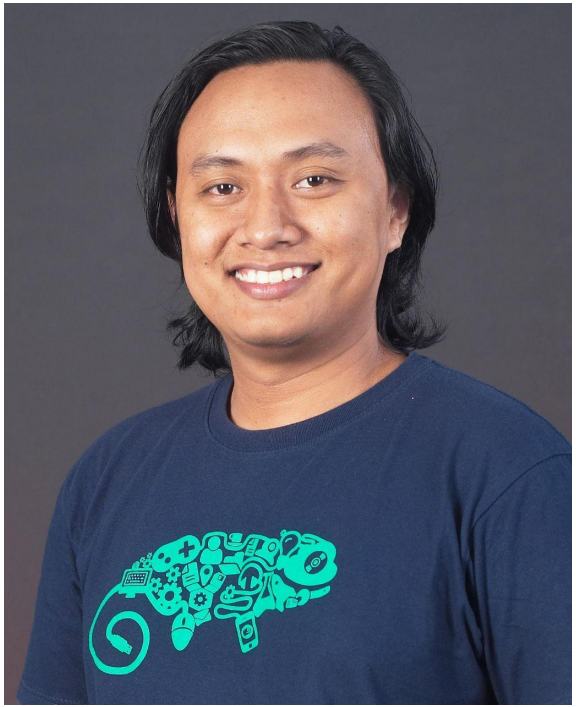
openSUSE.ID

DevOps Consultant

Cloud Platform Engineer

@tuanpembual

linkedin.com/in/tuanpembual/



Masa lalu

TIF UIN Sunan Kalijaga
Former

- sysadmin di BTech (2014-2015)
- DevOps di GOJEK (2015-2016)
- DevOps di Peentar (2016-2017)
- DevOps at GTech (2019-2020)

Contributor at

- openSUSE Asia
- GNOME Asia
- BlankOn Project
- KPLI Jogja (2011-2014)
- KSL Kusuka

Tech Stack

- Cloud Computing
  - AWS
  - GCP
  - Alibaba
  - DO
- OS
  - openSUSE
  - Arch
  - Debian/Ubuntu
  - Slackware
  - BlankOn
- Automation
  - Bash
  - Ansible
  - Terraform
- CICD
  - GitLab
  - Jenkins
- Monitoring
  - Grafana
- Container
  - Docker
  - Kubernetes

Apa itu provisioning

Provisioning adalah proses penyediaan dan konfigurasi sumber daya komputasi dan jaringan yang diperlukan untuk mendukung aplikasi, layanan, atau beban kerja tertentu. Ini melibatkan alokasi dan penyiapan sumber daya seperti server, penyimpanan, jaringan, dan perangkat lunak, serta konfigurasi mereka sesuai dengan kebutuhan spesifik.

Melakukan Provisioning

Provisioning dapat dilakukan secara manual oleh administrator sistem atau secara otomatis menggunakan alat otomatisasi infrastruktur seperti Ansible, Chef, Puppet, atau Terraform.

Pendekatan otomatisasi memberikan keuntungan dalam hal konsistensi konfigurasi, efisiensi waktu, dan pengurangan kesalahan manusia.

Provisioning = Setup

Manual dan otomatis. Berawal dari

- cmd
- Bash script
- Tools (1000 vm)

Ansible Intro

Ansible adalah sebuah tool automation open-source yang disponsori oleh Red Hat untuk membantu melakukan instalasi, konfigurasi sistem, deployment software, bahkan melakukan update server.

Ansible ini bersifat agentless, yang artinya ia dapat berjalan hanya dengan koneksi SSH dan nggak perlu agent atau software tambahan di client.

Python stuff ~

Agentless?

Agentless provisioning adalah pendekatan dalam otomatisasi infrastruktur dimana sumber daya atau perangkat yang diatur tidak memerlukan instalasi atau pemasangan perangkat lunak tambahan (atau "agen") di perangkat target. Dalam konteks provisioning, ini berarti bahwa sistem atau perangkat yang akan dikonfigurasi dapat diakses dan dikendalikan tanpa memerlukan agen yang berjalan di dalamnya.

Perkakas Serupa

- Puppet: https://www.puppet.com/docs/puppet/6/puppet_overview.html
- Chef (master, node) https://www.chef.io/
- Salt (master, client) http://saltstack.com/

Provisioning tools - Elements

Contain similar elements:

– Directive

– Directive Script

– Master Node

– Slave Nodes

Ansible elements

The specific elements in Ansible are known as...

Directive => Task

Directive Script => Playbook

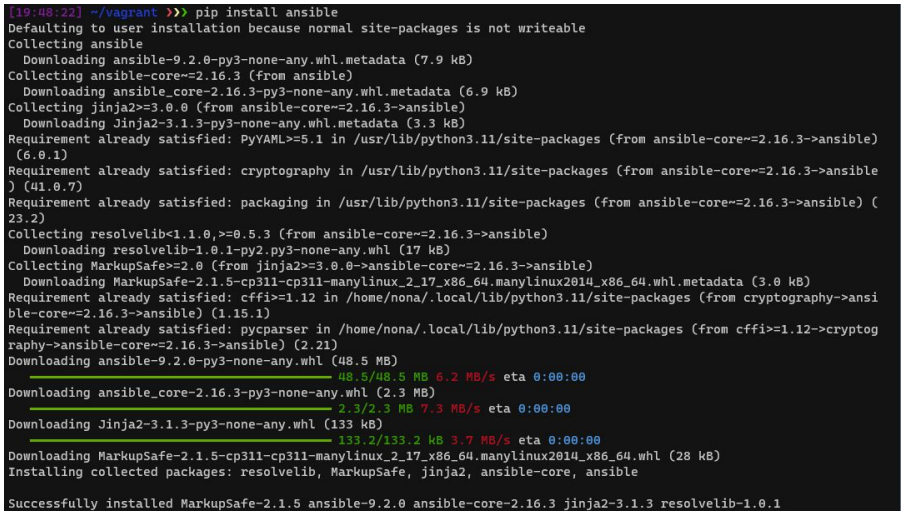
Master Node => Your own machine

Client Node => Any remote server

Master node installation

pip install ansible

sudo apt install ansible



Client node requirements

SSH

(that's it !)

(..and a bit of Python)

bagian-bagian Ansible

Inventory

Inventory File

Tells the master node about the child nodes

Contains list of all servers, and defines groups of servers

Defaults to /etc/ansible/hosts but you can use your own or pass as argument

cat inventories/aws-malioboro/inventory.ini

```
[mesin1]
54.226.249.234 ansible_connection=ssh ansible_ssh_private_key_file=../malioboro.pem ansible_ssh_user=ubuntu


```

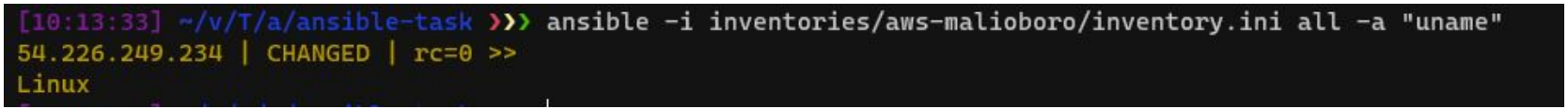
Run! Ping the machines

ansible -i inventories/aws-malioboro/inventory.ini all -m ping

```
[10:12:06] ~/v/T/a/ansible-task ››› ansible -i inventories/aws-malioboro/inventory.ini all -m ping
54.226.249.234 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

Execute commands

ansible -i inventories/aws-malioboro/inventory.ini all -a "uname"



Connectivity options

What if you don't want to SSH in as your current user ?

- --user use different username for SSH
- --ask-pass ask for SSH password
- --sudo run commands through sudo
- --ask-sudo-pass ask for sudo password
- --sudo-user sudo to different non-root user
- ansible -u estu --sudo --sudo-user batman ...

Modules

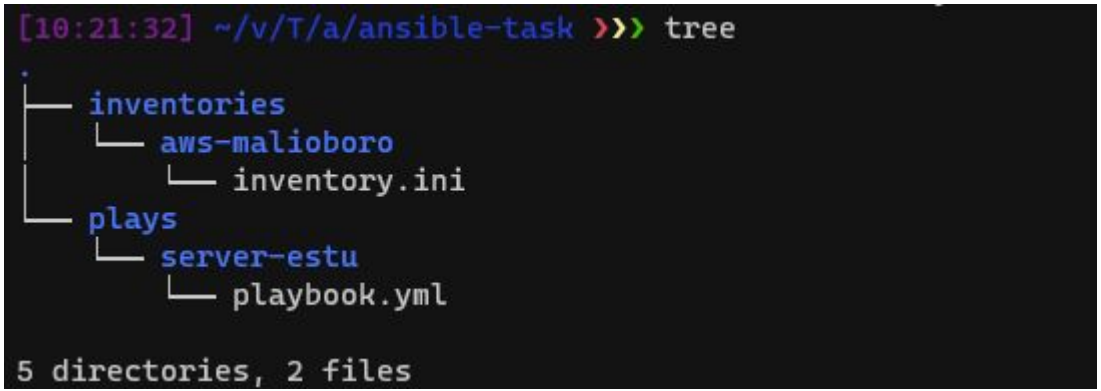
Modules - Resep General

- Perintah yang sudah bisa digunakan:
  - Apt
  - Files
  - Users
  - Etc
- https://docs.ansible.com/ansible/2.9/modules/list_of_all_modules.html

Skelethon

Ansible Skelethon Folder

Sample:



Playbooks

Playbooks are automation blueprints, in YAML format, that Ansible uses to deploy and configure managed nodes.

Sebuah cetak biru, apa yang akan kita lakukan kepada server yg terdapat di inventory

Hendaknya mencerminkan kondisi server atau penggunaan akhir.

Memulai menulis “Task”

Pada berkas plays/server-estu/playbook.yaml

ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/playbook.yml

```yaml
- name: My first play
  hosts: mesin1
  tasks:
    - name: Ping my hosts
      ansible.builtin.ping:

    - name: Print message
      ansible.builtin.debug:
        msg: Hello world
```

```
[10:33:55] ~/v/t/a/ansible-task ❯❯❯ ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/playbook.yml

PLAY [My first play] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [54.226.249.234]

TASK [Ping my hosts] ***********************************************************************************************
ok: [54.226.249.234]

TASK [Print message] ***********************************************************************************************
ok: [54.226.249.234] => {
    "msg": "Hello world"
}

PLAY RECAP *********************************************************************************************************
54.226.249.234             : ok=3    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Recapture command

- Buat folder ansible-task/inventories/aws-malioboro ansible-task/plays/server-estu
  - mkdir -p ansible-task/inventories/aws-malioboro ansible-task/plays/server-estu
  - cd ansible-task
- Isi berkas inventory.ini
  - vim inventories/aws-malioboro/inventory.ini
- Isi berkas playbook.yml
  - vim plays/server-estu/playbook.yaml
- Run!

Bagian susahnya

Improve ~

- Know cmd
- Know bash
- Then convert to ansible

Cara masang paket apt (vim, htop dst) di ansible

Ref:
https://docs.ansible.com/ansible/latest/collections/ansible/builtin/apt_module.html

vim plays/server-estu/playbook.yaml

```yaml
- name: My first play
  hosts: mesin1
  become: yes
  tasks:
    - name: pasang paket
      ansible.builtin.apt:
        name: vim
        update_cache: yes
```

```
[11:11:49] ~/v/T/a/ansible-task ››› ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/playbook.yml

PLAY [My first play] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [54.226.249.234]

TASK [Ping my hosts] ***********************************************************************************************
ok: [54.226.249.234]

TASK [Print message] ***********************************************************************************************
ok: [54.226.249.234] => {
    "msg": "Hello World"
}

TASK [pasang paket] ************************************************************************************************
ok: [54.226.249.234]

PLAY RECAP *********************************************************************************************************
54.226.249.234             : ok=4    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Files

Ref: https://docs.ansible.com/ansible/latest/collections/ansible/builtin/copy_module.html

mkdir -p plays/server-estu/files
vim plays/server-estu/files/index.html
vim plays/server-estu/playbook.yaml

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to Malioboro</title>
<style>
            body {
            width: 35em;
            margin: 0 auto;
            font-family: Tahoma, Verdana, Arial, sans-serif;
            }
</style>
</head>

<body>

<h1>Welcome to Malioboro</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>
```

```
- name: My first play
  hosts: mesin1
  become: yes
  tasks:
    - name: pasang paket
      ansible.builtin.apt:
        name: nginx
        update_cache: yes
    - name: salin berkas
      ansible.builtin.copy:
        src: files/index.html
        dest: /var/www/html/index.html
```

Run! files

```
[11:41:37] ~/v/T/a/ansible-task ››› ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/p
aybook.yml

PLAY [My first play] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com]

TASK [pasang nginx] ************************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com]

TASK [salin berkas] ************************************************************************************************
changed: [ec2-54-226-249-234.compute-1.amazonaws.com]
```

Handlers

Handlers: running operations on change

Sometimes you want a task to run only when a change is made on a machine. For example, you may want to restart a service if a task updates the configuration of that service, but not if the configuration is unchanged.

Ansible uses handlers to address this use case. Handlers are tasks that only run when notified.

```
- name: My first play
  hosts: mesin1
  become: yes
  tasks:
    - name: pasang paket
      ansible.builtin.apt:
        name: vim
        update_cache: yes
      notify:
        - restart nginx
  handlers:
    - name: restart nginx
      ansible.builtin.service:
        name: nginx
        state: restarted
```

Run handlers and files

```
[11:41:37] ~/v/T/a/ansible-task ››› ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/pl
aybook.yml

PLAY [My first play] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com]

TASK [pasang nginx] ************************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com]

TASK [salin berkas] ************************************************************************************************
changed: [ec2-54-226-249-234.compute-1.amazonaws.com]

RUNNING HANDLER [restart nginx] ************************************************************************************
changed: [ec2-54-226-249-234.compute-1.amazonaws.com]

PLAY RECAP *********************************************************************************************************
ec2-54-226-249-234.compute-1.amazonaws.com : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
```

Welcome to Malioboro

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Template dan Variable

Template and Vars

mkdir -p plays/server-estu/templates
vim plays/server-estu/files/index.html.j2
vim plays/server-estu/playbook.yaml

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to {{ sites }}</title>
<style>
            body {
            width: 35em;
            margin: 0 auto;
            font-family: Tahoma, Verdana, Arial, sans-serif;
            }
</style>
</head>

<body>

<h1>Welcome to {{ sites }} {{ os }}</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>
```

```
- name: My first play
  hosts: mesin1
  vars:
    sites: Malioboro
    os: Ubuntu
  become: yes
  tasks:
    - name: Print message
      ansible.builtin.debug:
        msg: Hello world {{ os }}

    - name: salin template
      ansible.builtin.template:
        src: templates/index.html.j2
        dest: /var/www/html/index.html
      notify:
        - restart nginx

  handlers:
    - name: restart nginx
      ansible.builtin.service:
        name: nginx
        state: restarted
```

Run templates and vars

```
[12:29:39] ~/v/T/a/ansible-task ❯❯❯ ansible-playbook -i inventories/aws-malioboro/inventory.ini plays/server-estu/pl
aybook.yml

PLAY [My first play] ***********************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com]

TASK [Print message] ***********************************************************************************************
ok: [ec2-54-226-249-234.compute-1.amazonaws.com] => {
    "msg": "Hello world Ubuntu"
}

TASK [salin template] **********************************************************************************************
changed: [ec2-54-226-249-234.compute-1.amazonaws.com]

RUNNING HANDLER [restart nginx] ************************************************************************************
changed: [ec2-54-226-249-234.compute-1.amazonaws.com]

PLAY RECAP *********************************************************************************************************
ec2-54-226-249-234.compute-1.amazonaws.com : ok=4    changed=2    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0
```

Welcome to Malioboro Ubuntu

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Roles

Roles

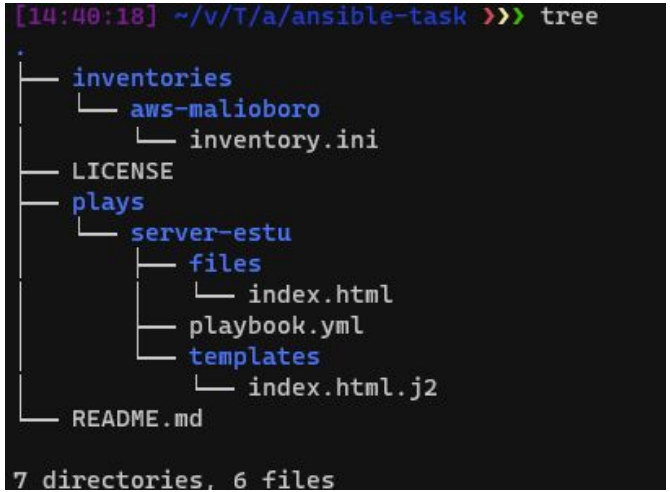
A limited distribution of reusable Ansible content (tasks, handlers, variables, plugins, templates and files) for use inside of a Play.

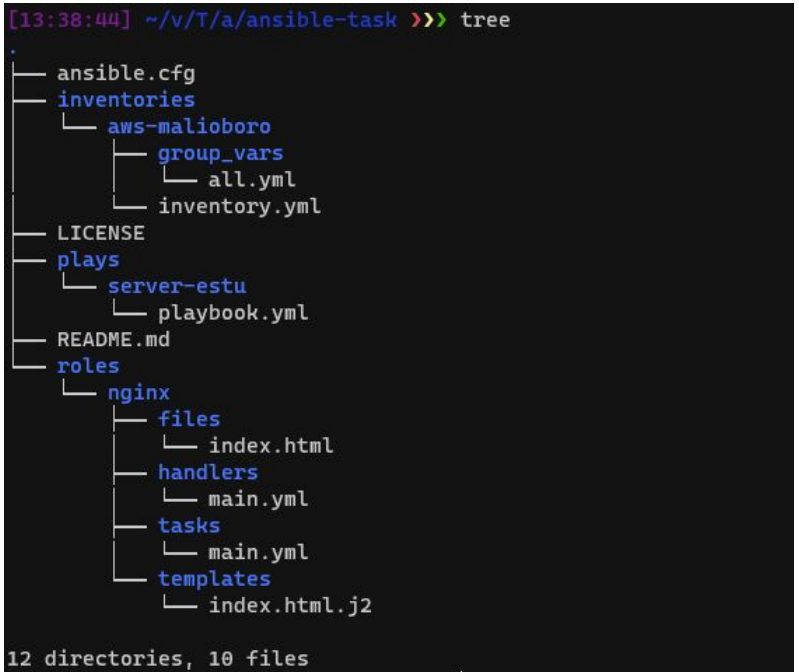
To use any Role resource, the Role itself must be imported into the Play.

Set ansible.cfg

Roles

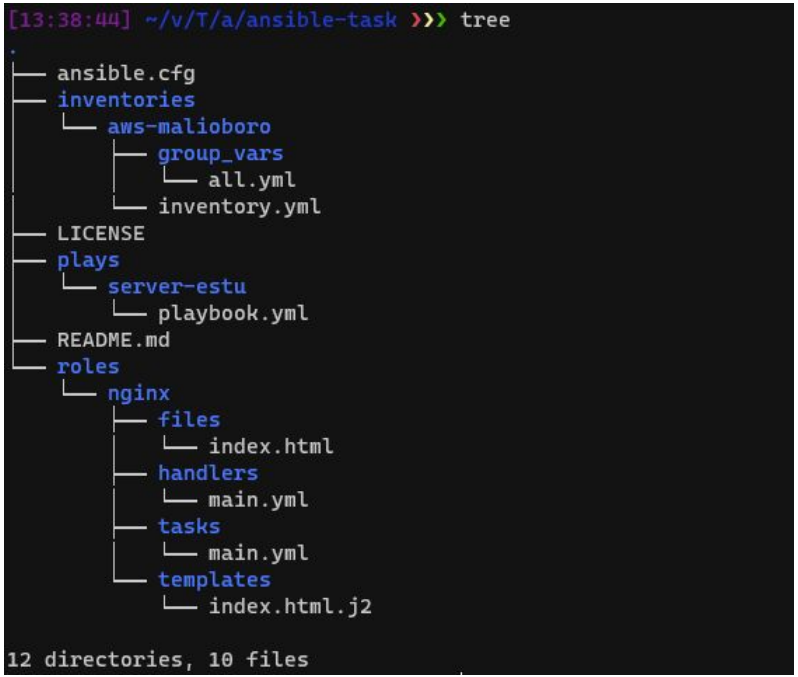
Roles are ways of automatically loading certain variables, tasks, and handlers based on a known file structure. (on folder server-estu)

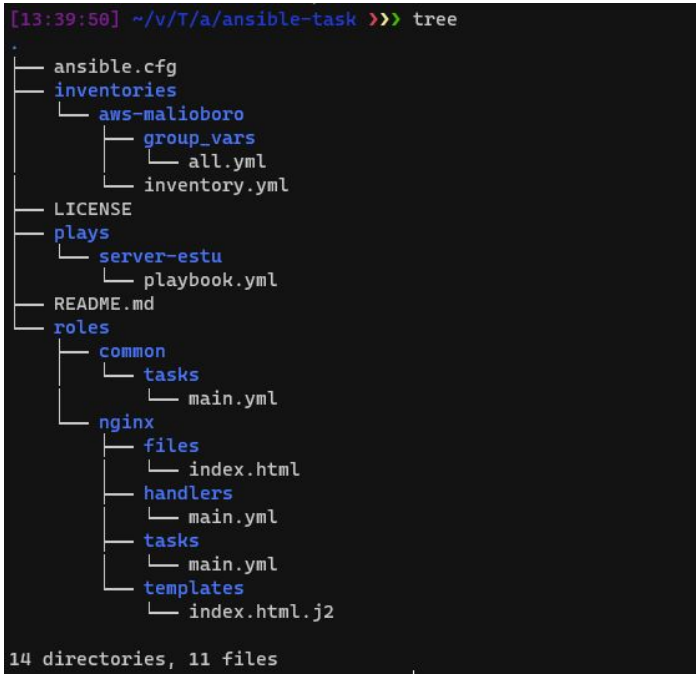




Integrate

New Skelethon





Referensi Repository

- https://github.com/tuanpembual/ansible-task

Ansible Vault

Untuk data-data sensitif:

- Username Password (user, db dst)
- SSL, Etc

Encryption with Ansible Vault ONLY protects ‘data at rest’. Once the content is decrypted (‘data in use’), play and plugin authors are responsible for avoiding any secret disclosure

Praktek

- Buat modul
  - Sysctl tuning
  - User
- Buat playbook
  - 2 playbook

Pertanyaan dan diskusi

References

- https://docs.ansible.com/ansible/latest/getting_started/index.html
- https://docs.ansible.com/ansible/2.9/modules/list_of_all_modules.html
- https://speakerdeck.com/phantomwhale/ansible-your-first-step-into-server-provisioning
- https://github.com/tuanpembual/ansible-task