

ภาคผนวก

ภาคผนวก ก

คู่มือการติดตั้งระบบ

ก.1. ติดตั้ง Ubuntu Server 18.04

สามารถทำตามขั้นตอนจาก link ต่อไปนี้ <https://ubuntu.com/tutorials/tutorial-install-ubuntu-server#1-overview>

ก.2. ติดตั้ง Ansible AWX

1. สามารถทำตามขั้นตอนจาก link ต่อไปนี้ <https://computingforgeeks.com/how-to-install-ansible-awx-on-ubuntu-linux/>

2. ในขั้นตอนที่ 5 ให้แก้ไขไฟล์ inventory ตามด้านล่างนี้

dockerhub_base=ansible

awx_task_hostname=awx

awx_web_hostname=awxweb

postgres_data_dir=/var/awx/postgres/pgdocker/

host_port=80

host_port_ssl=443

docker_compose_dir=/var/awx/awxcompose

pg_username=awx

pg_password=awxpass

pg_database=awx

pg_port=5432

rabbitmq_password=awxpass

rabbitmq_erlang_cookie=cookiemonster

admin_user={user ที่จะใช้ตอน login Ansible AWX}

admin_password={password ที่จะใช้ตอน login Ansible AWX}

create_preload_data=True

secret_key={secret key ที่ generate จากขั้นตอนก่อนหน้า}

project_data_dir=/var/lib/awx/projects

3. ทำตามขั้นตอนการติดตั้งต่อจนติดตั้ง Ansible AWX เสร็จ

ก.3. เตรียมไฟล์ playbook

1. Clone repository ที่เก็บไฟล์ playbook ด้วยคำสั่ง

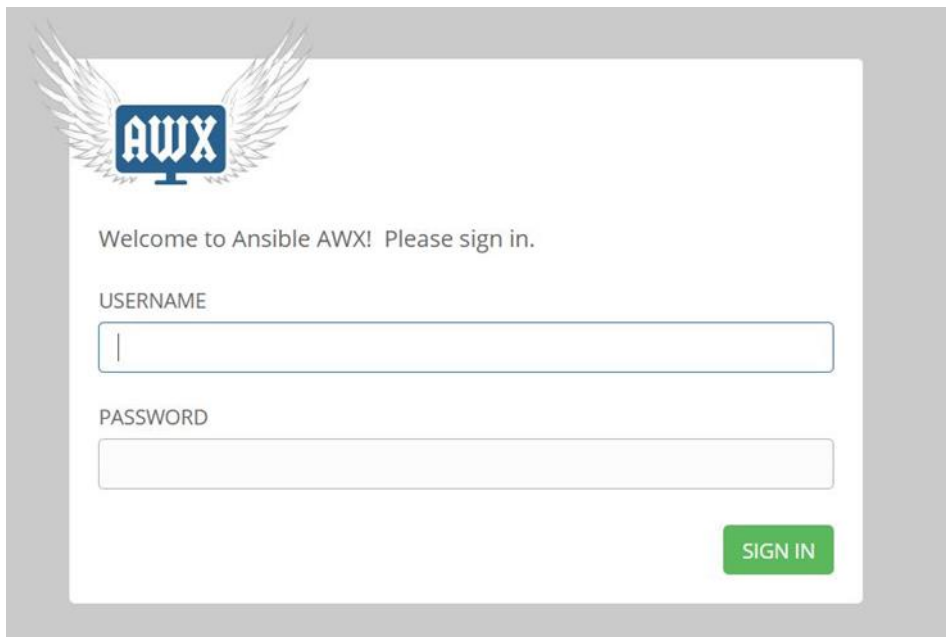
```
root@awx_test:~# git clone \
https://github.com/tinnawat91/Project_AWX
```

2. ย้าย folder ที่เก็บ playbook ไปไว้ใน folder /var/lib/awx/projects ด้วยคำสั่ง

```
root@awx_test:~# mv Project_AWX/* \
/var/lib/awx/projects/
```

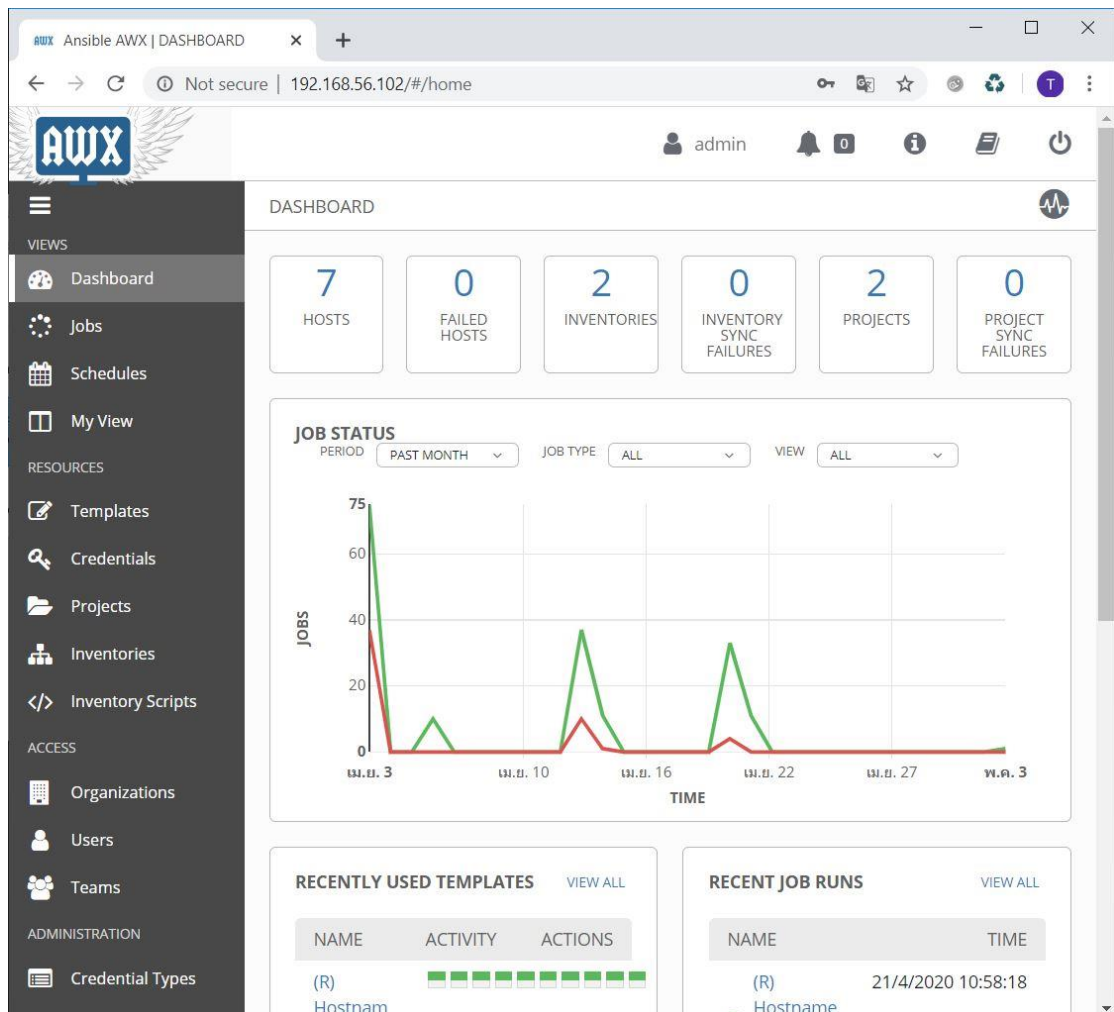
ก.4. Login เข้า Ansible AWX

ในการใช้งาน Ansible AWX ให้เปิดเบราว์เซอร์ ในช่อง URL ให้ใส่ IP address ของ server ที่ติดตั้ง Ansible AWX จากนั้นจะมีหน้าต่างให้ login โดยใช้ username และ password ที่ตั้งในไฟล์ inventory ในขั้นตอนการติดตั้ง ดังแสดงในรูปที่ ก.1



รูปที่ ก.1 หน้าต่างการ login

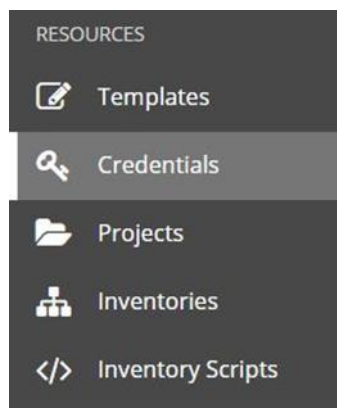
เมื่อ login เข้ามาแล้วจะพบหน้าต่าง dashboard ดังรูปที่ ก.2



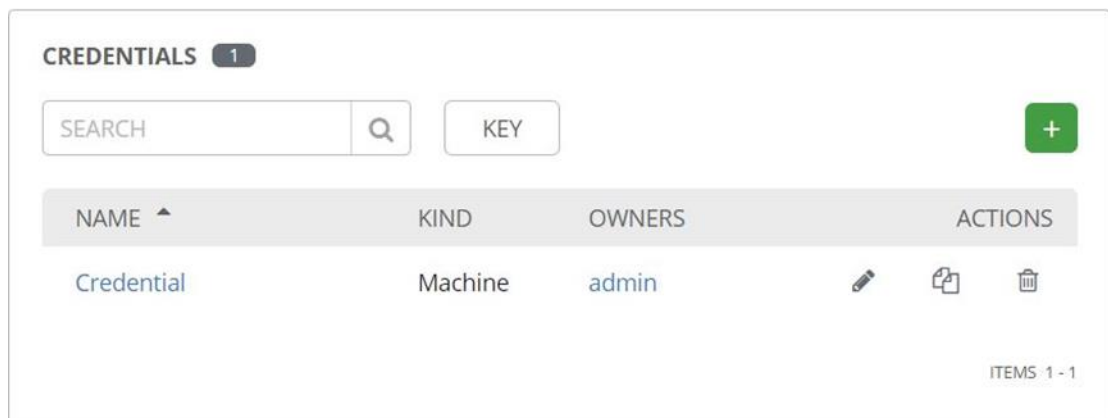
รูปที่ ก.2 หน้าต่าง dashboard

ก.5. สร้าง credentials

เมื่อเข้ามาที่ Ansible AWX แล้วให้ไปเลือก Credentials ที่แถบเมนูด้านข้าง ดังแสดงในรูปที่ ก.3 จะปรากฏรายชื่อ credential ที่ได้สร้างไว้ จากนั้นให้เลือกที่เครื่องหมายบวกสีเขียวเพื่อสร้าง credential ใหม่ ดังแสดงในรูปที่ ก.4



รูปที่ ก.3 แถบเมนู Credentials



รูปที่ ก.4 หน้าต่างแสดงรายชื่อ credentials

จากนั้นจะมีหน้าต่างให้กรอกข้อมูลดังแสดงในรูปที่ ก.5 - ก.6 โดยให้กรอกข้อมูลดังรายละเอียดต่อไปนี้

1. NAME: ชื่อของ credential
2. CREDENTIAL TYPE: เลือกประเภท Machine
3. USERNAME: username ของอุปกรณ์ ในการทดลองบนอุปกรณ์เครือข่ายใช้
username: 261336
4. รหัสผ่าน: รหัสผ่านสำหรับการใช้งานอุปกรณ์ ในการทดลองบนอุปกรณ์เครือข่ายใช้
รหัสผ่าน: netlab
5. PRIVILEGE ESCALATION METHOD: เลือกประเภท enable
6. PRIVILEGE ESCALATION PASSWORD: password สำหรับเข้าสู่โหมด enable ใน
การทดลองบนอุปกรณ์เครือข่ายใช้ password: netlab

NEW CREDENTIAL

DETAILS

PERMISSIONS

* NAME

DESCRIPTION

ORGANIZATION

SELECT AN ORGANIZ

* CREDENTIAL TYPE

Machine

รูปที่ ก.5 หน้าต่างการสร้าง credential

TYPE DETAILS

USERNAME

รหัสผ่าน

☐ Prompt on launch

261336

ENCRYPTED

SSH PRIVATE KEY

HINT: Drag and drop private file on the field below.

SIGNED SSH CERTIFICATE

HINT: Drag and drop private file on the field below.

PRIVATE KEY PASSPHRASE

☐ Prompt on launch

PRIVILEGE ESCALATION

METHOD

enable

PRIVILEGE ESCALATION

USERNAME

PRIVILEGE ESCALATION

PASSWORD

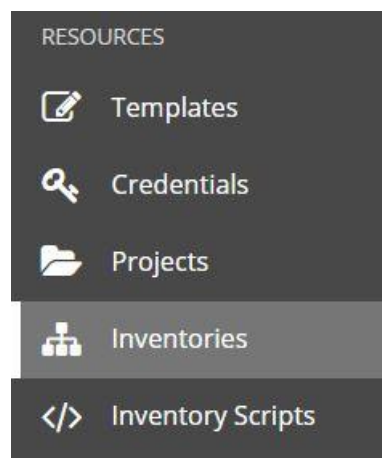
☐ Prompt on launch

ENCRYPTED

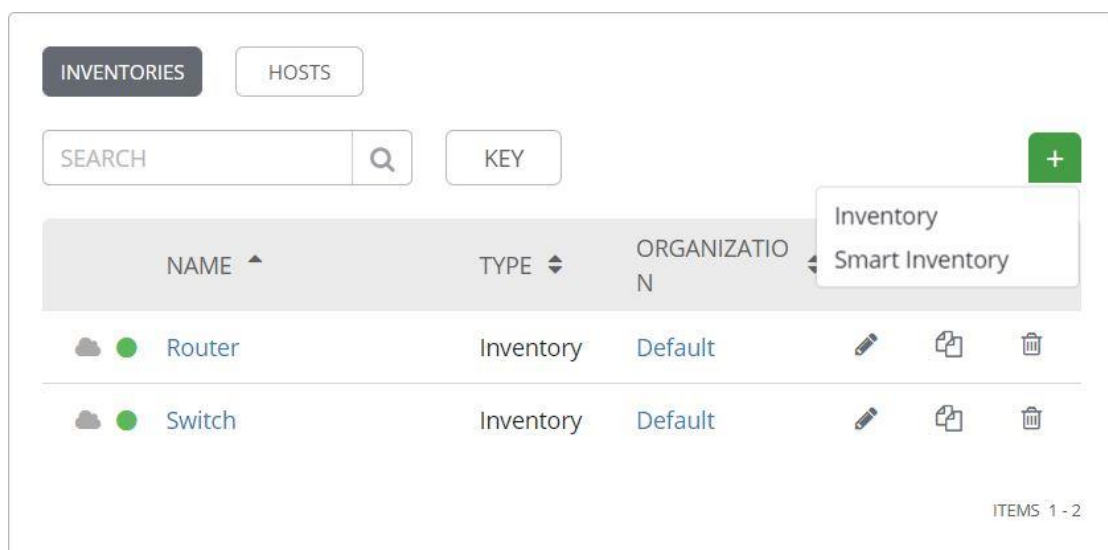
รูปที่ ก.6 หน้าต่างการสร้าง credential

ก.6. สร้าง inventories

เมื่อสร้าง credentials แล้วให้ไปเลือก Inventories ที่แถบเมนูด้านข้าง ดังแสดงในรูปที่ ก.7 จะปรากฏรายชื่อ inventory ที่ได้สร้างไว้ จากนั้นให้เลือกที่เครื่องหมายบวกสีเขียวแล้วเลือก Inventory เพื่อสร้าง inventory ใหม่ ดังแสดงในรูปที่ ก.8 โดยให้สร้าง 2 inventory คือ Router และ Switch



รูปที่ ก.7 แถบเมนู Inventories



รูปที่ ก.8 หน้าต่างแสดงรายชื่อ inventories

จากนั้นจะมีหน้าต่างให้กรอกข้อมูลดังแสดงรูปที่ ก.9 โดยให้กรอกข้อมูลดังรายละเอียดต่อไปนี้

1. NAME: ชื่อของ inventory

NEW INVENTORY

DETAILS PERMISSIONS GROUPS HOSTS SOURCES

COMPLETED JOBS

* NAME DESCRIPTION * ORGANIZATION

INSIGHTS CREDENTIAL INSTANCE GROUPS ?

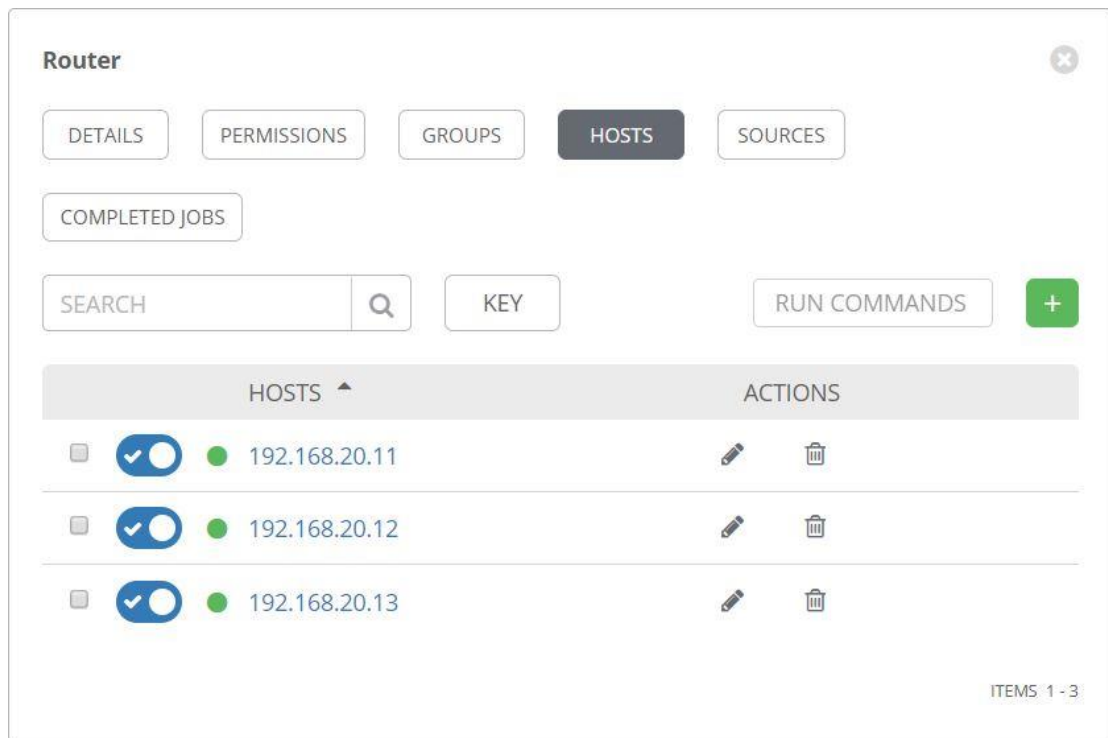
VARIABLES ? YAML JSON EXPAND

1 ---

CANCEL SAVE

รูปที่ ก.9 หน้าต่างการสร้าง inventory

จากนั้นให้กด SAVE แล้วเข้าไปเลือกที่ tab HOSTS จะปรากฏหน้าต่างแสดงรายชื่อ host ที่ได้สร้างไว้ ดังแสดงในรูปที่ ก.10



รูปที่ ก.10 หน้าต่างแสดงรายชื่อ hosts

จากนั้นให้กดที่เครื่องหมายบวกสีเขียวเพื่อสร้าง host ใหม่ โดยจะมีช่องให้กรอกข้อมูลดังแสดงในรูปที่ ก.11 ซึ่งมีรายละเอียดดังนี้

1. HOST NAME: IP address ของอุปกรณ์ปลายทาง
2. DESCRIPTION: คำอธิบาย host
3. VARIABLES: ข้อมูลตัวแปรที่จำเป็นของ host โดยในการทดลองได้ระบุประเภทของการเชื่อมต่อและระบบปฏิบัติการของอุปกรณ์ปลายทาง

192.168.20.11 ✓

DETAILS FACTS GROUPS COMPLETED JOBS

* HOST NAME ? DESCRIPTION

192.168.20.11 R1

VARIABLES ? YAML JSON EXPAND

```

1 ---
2 ansible_host: 192.168.20.11
3 ansible_connection: network_cli
4 ansible_network_os: ios

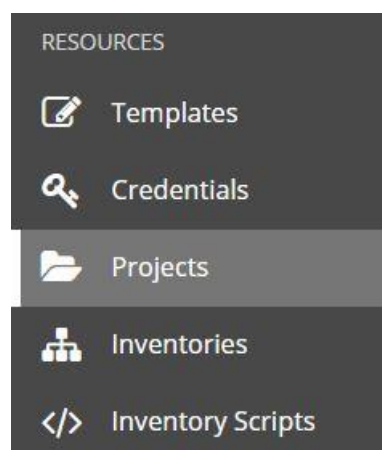
```

CANCEL SAVE

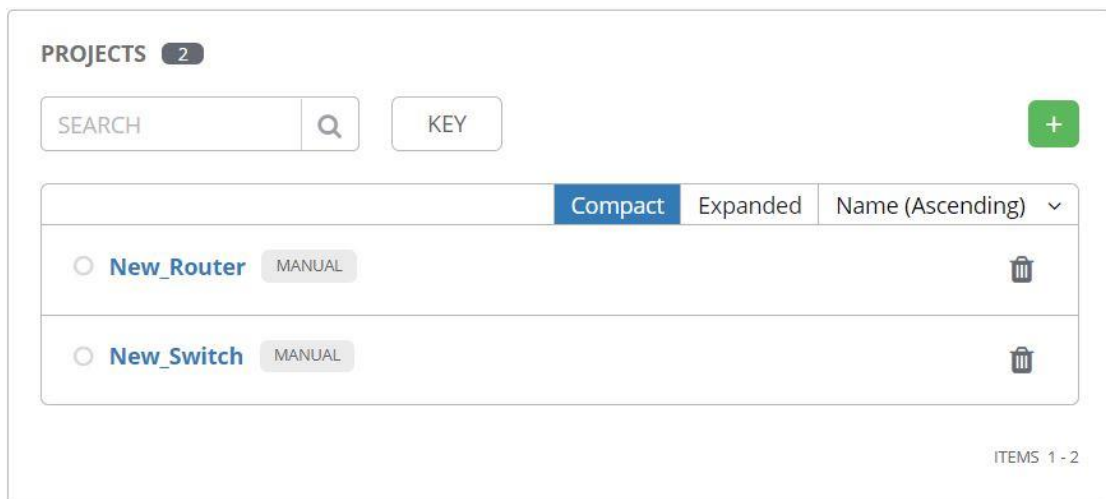
รูปที่ ก.11 หน้าต่างการสร้าง host

ก.7. สร้าง projects

เมื่อสร้าง inventory แล้วให้ไปเลือก Projects ที่แถบเมนูด้านข้าง ดังแสดงในรูปที่ ก.12 จะปรากฏรายชื่อ project ที่ได้สร้างไว้ จากนั้นให้เลือกที่เครื่องหมายบวกสีเขียวเพื่อสร้าง project ใหม่ ดังแสดงในรูปที่ ก.13 โดยให้สร้าง 2 project คือ New_Router และ New_Switch



รูปที่ ก.12 แถบเมนู Projects



รูปที่ ก.13 หน้าต่างแสดงรายชื่อ projects

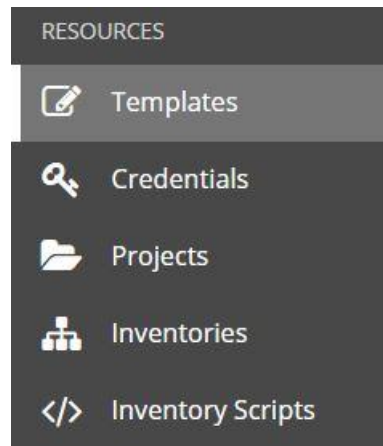
จากนั้นจะมีหน้าต่างให้กรอกข้อมูลดังแสดงในรูปที่ ก.14 โดยให้กรอกข้อมูลดังรายละเอียดต่อไปนี้

1. NAME: ชื่อของ project
2. SCM TYPE: แหล่งที่มาของไฟล์ playbook ในการทดลองได้สร้างไฟล์ playbook บน server ที่รัน Ansible AWX จึงเลือก Manual
3. PLAYBOOK DIRECTORY: เลือก folder ที่ได้สร้างไว้ (New_Router เลือก folder New_Router, New_Switch เลือก folder New_Switch)

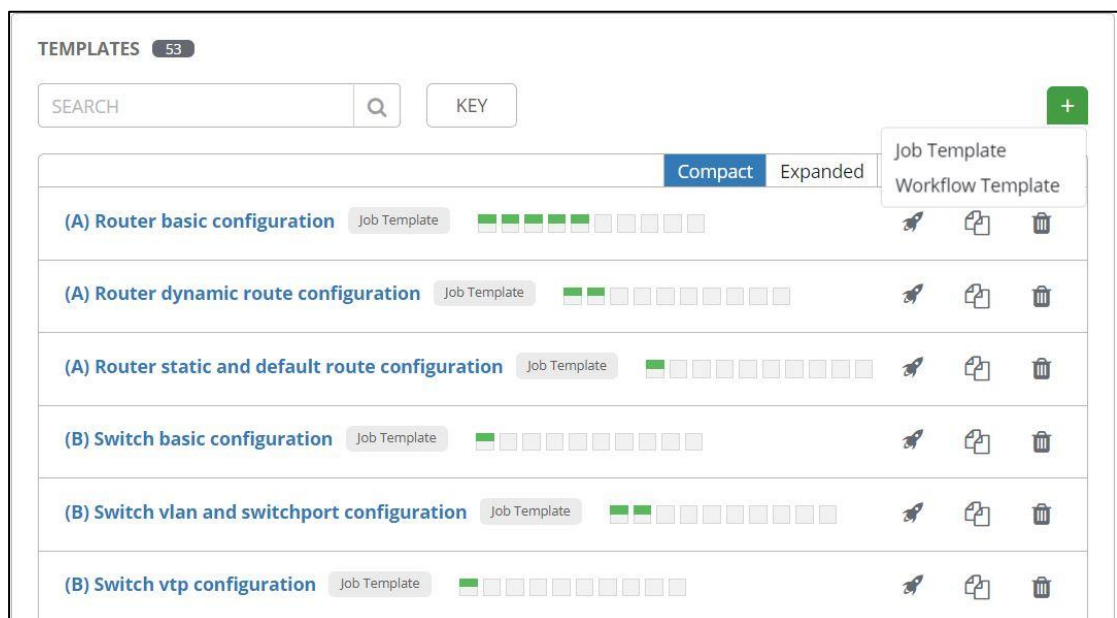
รูปที่ ก.14 หน้าต่างการสร้าง project

ก.8. สร้าง templates

เมื่อสร้าง project แล้วให้ไปเลือก Templates ที่แถบเมนูด้านข้าง ดังแสดงในรูปที่ ก.15 จะปรากฏรายชื่อ template ที่ได้สร้างไว้ จากนั้นให้เลือกที่เครื่องหมายบวกสีเขียวแล้วเลือก Job Template เพื่อสร้าง template ใหม่ ดังแสดงในรูปที่ ก.16



รูปที่ ก.15 แถบเมนู Templates



รูปที่ ก.16 หน้าต่างแสดงรายชื่อ templates

จากนั้นจะมีหน้าต่างให้กรอกข้อมูลดังแสดงในรูปที่ ก.17 โดยให้กรอกข้อมูลดังรายละเอียดต่อไปนี้

1. NAME: ชื่อ template
2. JOB TYPE: เลือกประเภท run
3. INVENTORY: เลือก inventory ที่ได้สร้างไว้
4. PROJECT: เลือก project ที่ได้สร้างไว้
5. PLAYBOOK: เลือกไฟล์ playbook ที่ได้สร้างไว้บน server
6. CREDENTIAL: เลือก credential ที่ได้สร้างไว้
7. VERBOSITY: เลือกประเภท 0(Normal)
8. OPTIONS: เลือก ENABLE PRIVILEGE ESCALATION

(A) Router basic configuration

DETAILS PERMISSIONS NOTIFICATIONS COMPLETED JOBS SCHEDULES EDIT SURVEY

* NAME: (A) Router basic configuration DESCRIPTION: PROMPT ON LAUNCH: ☐

* INVENTORY: Router PROMPT ON LAUNCH: ☐ * PROJECT: New_Router * PLAYBOOK: Config/basic.yml

CREDENTIALS: Credential PROMPT ON LAUNCH: ☐ FORKS: 0 LIMIT: PROMPT ON LAUNCH: ☐

* VERBOSITY: 0 (Normal) PROMPT ON LAUNCH: ☐ JOB TAGS: PROMPT ON LAUNCH: ☐ SKIP TAGS: PROMPT ON LAUNCH: ☐

LABELS: INSTANCE GROUPS: JOB SLICING: 1

TIMEOUT: 0 SHOW CHANGES: ☐ PROMPT ON LAUNCH: ☐ OPTIONS:

- ☒ ENABLE PRIVILEGE ESCALATION
- ☐ ENABLE PROVISIONING CALLBACKS
- ☐ ENABLE WEBHOOK
- ☐ ENABLE CONCURRENT JOBS
- ☐ ENABLE FACT CACHE

รูปที่ ก.17 หน้าต่างการสร้าง template

โดยให้กำหนดชื่อ template ของ switch ดังแสดงในตารางที่ ก.1

ตารางที่ ก.1 ความสัมพันธ์ระหว่างชื่อ template กับไฟล์ playbook (✕ คือ ไม่มีไฟล์ playbook)

Name	Template Name		
	Configuration	Verification mode 1	Verification mode 2
Device name	(SW) Hostname Configuration	(SW) Hostname Verification mode 1	(SW) Hostname Verification mode 2
ไฟล์ playbook	Config/hostname.yml	Verify/hostname.yml	Verify_mode2/hostname.yml
Interface status	(SW) Interface status configuration	(SW) Interface status verification mode 1	(SW) Interface status verification mode 2
ไฟล์ playbook	Config/Interface_status.yml	Verify/Interface_status.yml	Verify_mode2/Interface_status.yml
Interface switchport	(SW) Switchport configuration	(SW) Switchport verification mode 1	(SW) Switchport verification mode 2
ไฟล์ playbook	Config/Interface_mode.yml	Verify/Interface_mode.yml	Verify_mode2/Interface_mode.yml
VLAN	(SW) Vlan configuration	(SW) Vlan verification mode 1	(SW) Vlan verification mode 2
ไฟล์ playbook	Config/vlan.yml	Verify/vlan.yml	Verify_mode2/vlan.yml
VTP mode & domain	(SW) VTP mode and domain configuration	(SW) VTP mode and domain verification mode 1	(SW) VTP mode and domain verification mode 2
ไฟล์ playbook	Config/vtp_mode_domain.yml	Verify/vtp_mode_domain.yml	Verify_mode2/vtp_mode_domain.yml

Name	Template Name		
	Configuration	Verification mode 1	Verification mode 2
VTP password	(SW) VTP password configuration	(SW) VTP password verification mode 1	(SW) VTP password verification mode 2
ไฟล์ playbook	Config/vtp_password.yml	Verify/vtp_password.yml	Verify_mode2/vtp_password.yml
Default gateway	(SW) Default-gateway configuration	(SW) Default-gateway verification mode 1	(SW) Default-gateway verification mode 2
ไฟล์ playbook	Config/gateway.yml	Verify/gateway.yml	Verify_mode2/gateway.yml
Spanning-tree priority	(SW) STP priority configuration	(SW) STP priority verification mode 1	(SW) STP priority verification mode 2
ไฟล์ playbook	Config/stp.yml	Verify/stp.yml	Verify_mode2/stp.yml
Basic configuration (Device name + Interface status)	(SW) Switch basic configuration	✗	✗
ไฟล์ playbook	Config/basic.yml	✗	✗
VLAN all (VLAN + Interface switchport)	(SW) Switch vlan and switchport configuration	✗	✗
ไฟล์ playbook	Config/vlan_all.yml	✗	✗
VTP (VTP mode & domain + VTP password)	(SW) Switch vtp configuration	✗	✗
ไฟล์ playbook	Config/vtp.yml	✗	✗

Name	Template Name		
	Configuration	Verification mode 1	Verification mode 2
Spanning-tree port	✕	✕	(SW) STP port verification mode 2
ไฟล์ playbook	✕	✕	Verify_mode2/stp_port.yml

โดยให้กำหนดชื่อ template ของ router ดังแสดงในตารางที่ ก.2

ตารางที่ ก.2 ความสัมพันธ์ระหว่างชื่อ template กับไฟล์ playbook (✕ คือ ไม่มีไฟล์ playbook)

Name	Template Name		
	Configuration	Verification mode 1	Verification mode 2
Device name	(R) Hostname Configuration	(R) Hostname Verification mode 1	(R) Hostname Verification mode 2
ไฟล์ playbook	Config/hostname.yml	Verify/hostname.yml	Verify_mode2/hostname.yml
Interface status	(R) Interface status configuration	(R) Interface status verification mode 1	(R) Interface status verification mode 2
ไฟล์ playbook	Config/Interface_status.yml	Verify/Interface_status.yml	Verify_mode2/Interface_status.yml
Interface IP address	(R) Interface IP address configuration	(R) Interface IP address verification mode 1	(R) Interface IP address verification mode 2
ไฟล์ playbook	Config/Interface_ip.yml	Verify/Interface_ip.yml	Verify_mode2/Interface_ip.yml
Static route	(R) Static route configuration	(R) Static route verification mode 1	(R) Static route verification mode 2
ไฟล์ playbook	Config/static.yml	Verify/ static.yml	Verify_mode2/ static.yml
Default route	(R) Default route configuration	(R) Default route verification mode 1	(R) Default route verification mode 2
ไฟล์ playbook	Config/default_route.yml	Verify/ default_route.yml	Verify_mode2/ default_route.yml

Name	Template Name		
	Configuration	Verification mode 1	Verification mode 2
RIP	(R) RIP configuration	(R) RIP verification mode 1	(R) RIP verification mode 2
ไฟล์ playbook	Config/rip.yml	Verify/ rip.yml	Verify_mode2/ rip.yml
OSPF	(R) OSPF configuration	(R) OSPF verification mode 1	(R) OSPF verification mode 2
ไฟล์ playbook	Config/ospf.yml	Verify/ ospf.yml	Verify_mode2/ ospf.yml
Basic configuration (Device name + Interface status + Interface IP address)	(R) Switch basic configuration	✗	✗
ไฟล์ playbook	Config/basic.yml	✗	✗
Static & Default route (Static route + Default route)	(R) Switch vlan and switchport configuration	✗	✗
ไฟล์ playbook	Config/static_and_default_route.yml	✗	✗
Dynamic route (RIP + OSPF)	(R) Switch vtp configuration	✗	✗
ไฟล์ playbook	Config/dynamic_route.yml	✗	✗
Routing table	✗	✗	(R) Routing table verification mode 2
ไฟล์ playbook	✗	✗	Verify_mode2/routing_table.yml

ก.9. สร้าง surveys

เมื่อสร้าง template แล้ว tab survey จะปรากฏขึ้น ให้กดเข้าไปแล้วจะมีหน้าต่างให้กรอกข้อมูลดังแสดงในรูปที่ ก.18 โดยมีรายละเอียดดังนี้

1. PROMPT: คำอธิบายช่องกรอกข้อมูล
2. DESCRIPTION: คำอธิบายรอง
3. ANSWER VARIABLE NAME: ชื่อตัวแปร โดยจะต้องตั้งให้ตรงกับในไฟล์ playbook
4. ANSWER TYPE: ประเภทของการตอบโดยสามารถเลือกได้หลายประเภทเช่น TEXT, INT, Multiple Choice (single select) เป็นต้น

EDIT SURVEY PROMPT

* PROMPT

Enter management IP address(es) of device(s).

DESCRIPTION

For configure device(s). Separate by comma(s) e.g. 192.168.20.11,192.168.

* ANSWER VARIABLE NAME ?

host_ip

* ANSWER TYPE ?

Text

MINIMUM LENGTH

0

MAXIMUM LENGTH

1024

DEFAULT ANSWER

รูปที่ ก.18 หน้าต่างการสร้าง survey

โดยที่ทุก template จะมี survey 1 อันที่เหมือนกันทุก template คือ survey ที่ให้ผู้ใช้กรอก IP address ของเครื่องปลายทาง โดย survey นี้จะกำหนดชื่อตัวแปรว่า host_ip

โดยที่การสร้าง survey ต้องกำหนดชื่อตัวแปรให้ตรงกับในไฟล์ playbook โดยสังเกตได้จากตัวแปรในไฟล์ playbook จะมี {{ }} ครอบอยู่ดังแสดงในรูปที่ ก.19

```
- name: Output(status up or down)
  debug:
    msg:
      - Status of interface {{ interface_name }} is
      - "{{ int.stdout_lines[0][0].split()[4] }}"
    when: int.stdout_lines[0][0].split()[4] != "administratively"
```

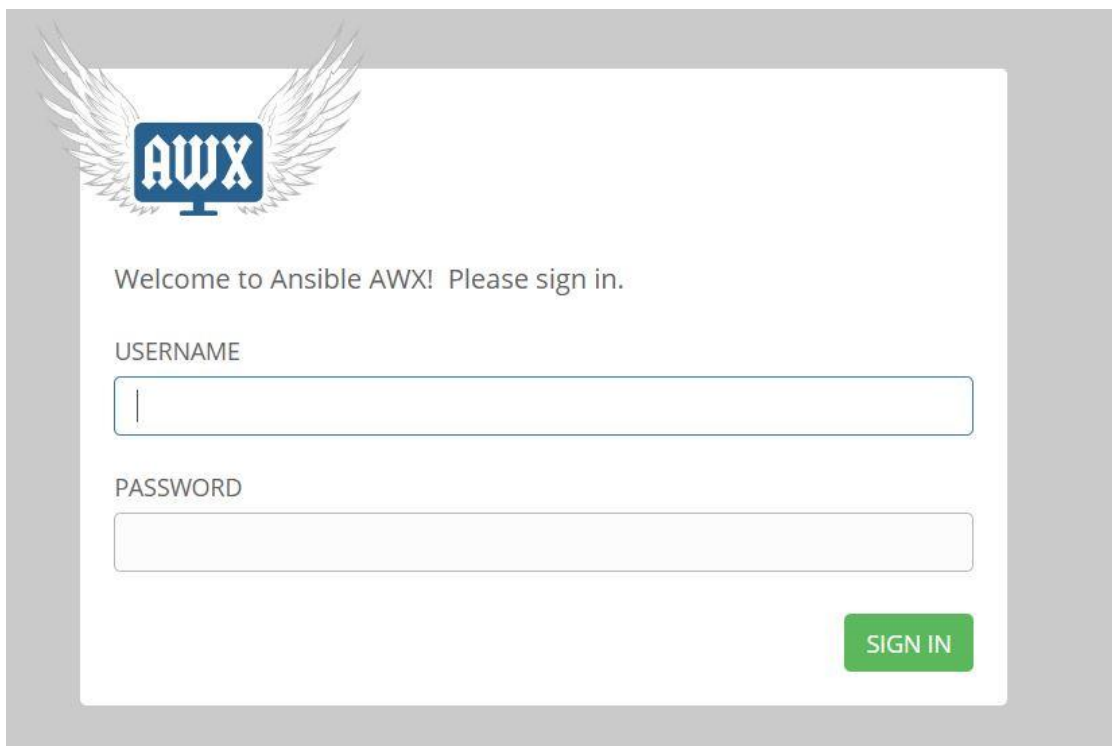
รูปที่ ก.19 ตัวอย่างตัวแปรในไฟล์ playbook โดยในรูปคือตัวแปรชื่อ interface_name

ภาคผนวก ข

คู่มือการใช้งานระบบ

ข.1. Login

เมื่อเข้าไปที่หน้าเว็บของ Ansible AWX จะพบกับหน้าต่าง login ดังแสดงในรูปที่ ข.1 โดยให้ใช้ username และ password ที่ได้สร้างไว้ในขั้นตอนการติดตั้ง



AWX

Welcome to Ansible AWX! Please sign in.

USERNAME

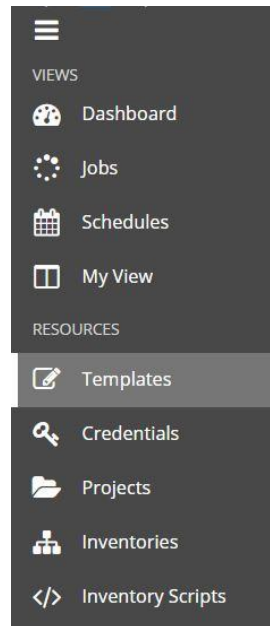
PASSWORD

SIGN IN

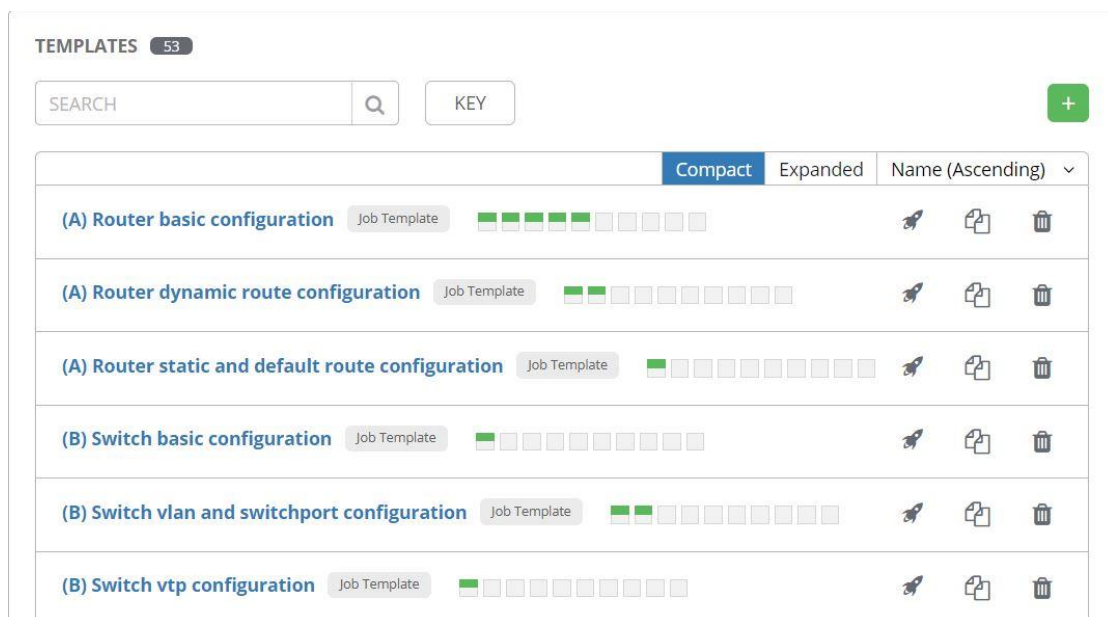
รูปที่ ข.1 หน้าต่างการ login

ข.2. การใช้งาน

เมื่อเข้ามาแล้วให้ไปเลือกเมนู Templates จากแถบเมนูด้านข้าง โดยจะมี template หัวข้อ การตั้งค่าหรือการตรวจสอบการตั้งค่าให้เลือก การใช้งานจะกดที่ไอคอนรูปจรวดด้านหลัง template ดังแสดงในรูปที่ ข.2 – ข.3



รูปที่ ข.2 แถบเมนู



รูปที่ ข.3 รายชื่อ templates

ข.3. การกรอกข้อมูล

เมื่อกดที่ไอคอนรูปจรวดหลัง template แล้ว จะมีหน้าต่างให้ผู้ใช้กรอกข้อมูล โดยจะเรียกว่า survey ในส่วนข้อมูลที่ต้องกรอกช่องแรกนั้นจะเหมือนกันทุก template โดยจะต้องกรอก IP address ของเครื่องปลายทาง (ในการทดลองบนอุปกรณ์ switch ใช้ IP address 192.168.20.21 – 192.168.20.24 และบนอุปกรณ์ router ใช้ IP address 192.168.20.11 – 192.168.20.13) เมื่อผู้ใช้กรอกเรียบร้อยแล้วให้กด NEXT ดังแสดงในรูปที่ ข.4

(R) OSPF CONFIGURATION

SURVEY **PREVIEW**

* ENTER MANAGEMENT IP ADDRESS(ES) OF DEVICE(S). SEPARATE BY COMMA(S) E.G.
192.168.20.11,192.168.20.12
For configure device(s)

192.168.20.11

* ENTER OSPF PROCESS ID

1

ENTER IP ADDRESS OF INTERFACE TO BE ACTIVATED

192.168.100.1

ENTER OSPF AREA

1

* CHOOSE MODE TO CONFIGURE OR DELETE

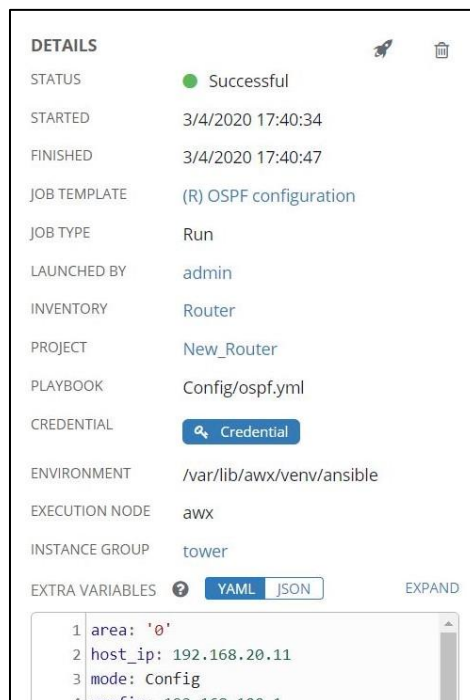
Config

CANCEL NEXT

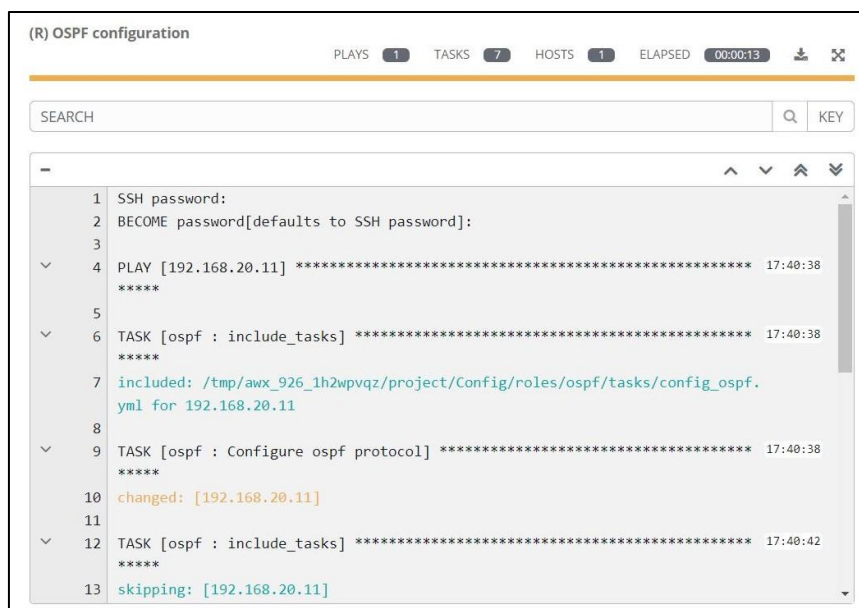
รูปที่ ข.4 หน้าต่างการกรอกข้อมูล (survey)

ข.4. ผลการทำงาน

เมื่อกดสั่งใช้งาน template แล้ว จะมีหน้าต่างแสดงการทำงาน โดยจะประกอบไปด้วย หน้าต่างแสดงสถานะและหน้าต่างแสดงผลการทำงาน ดังแสดงในรูปที่ ข.5 - ข.6



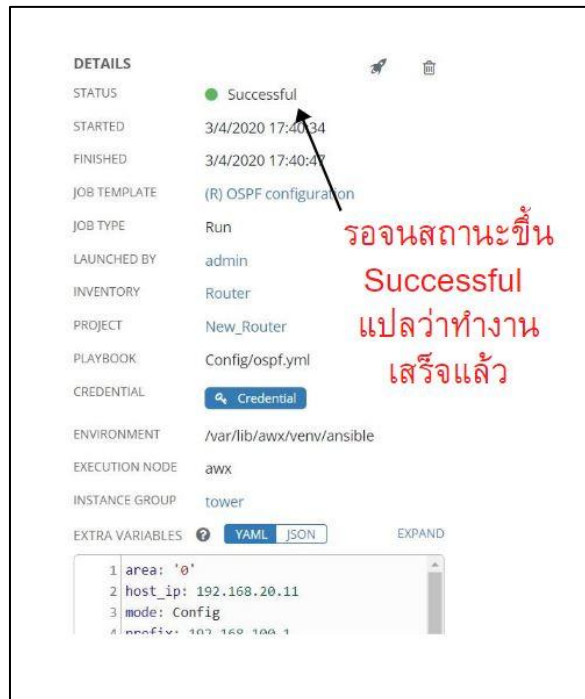
รูปที่ ข.5 หน้าต่างแสดงสถานะ



รูปที่ ข.6 หน้าต่างแสดงผลการทำงาน

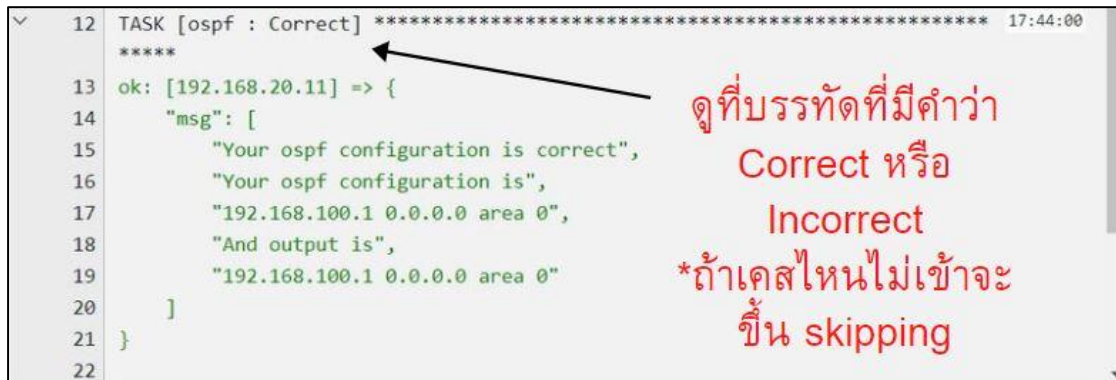
โดยจะมีวิธีการดูผลการทำงานดังนี้

1. ส่วนการตั้งค่า (Configuration) : ให้ดูที่สถานะ (STATUS) ถ้าขึ้นว่า Successful แสดงว่าระบบทำการตั้งค่าบนอุปกรณ์ปลายทางเรียบร้อยแล้ว ดังแสดงในรูปที่ ข.7



รูปที่ ข.7 หน้าต่างแสดงสถานะว่าทำงานเสร็จแล้ว

2. ส่วนการตรวจสอบการตั้งค่าแบบที่หนึ่ง (Verification mode 1) : ให้รอจนสถานะ (STATUS) ขึ้นว่า Successful ก่อน จากนั้นจึงไปดูที่หน้าต่างแสดงผลการทำงาน โดยดูที่บรรทัด Correct หรือ Incorrect ขึ้นอยู่กับว่าจะเข้าเงื่อนไขใด ดังแสดงในรูปที่ ข.8



```
12 TASK [ospf : Correct] ***** 17:44:00
13 ok: [192.168.20.11] => {
14     "msg": [
15         "Your ospf configuration is correct",
16         "Your ospf configuration is",
17         "192.168.100.1 0.0.0.0 area 0",
18         "And output is",
19         "192.168.100.1 0.0.0.0 area 0"
20     ]
21 }
22
```

ดูที่บรรทัดที่มีคำว่า
Correct หรือ
Incorrect
*ถ้าเคสไหนไม่เข้าจะ
ขึ้น skipping

รูปที่ ข.8 ตัวอย่างในส่วนการตรวจสอบการตั้งค่าแบบที่หนึ่ง (Verification mode 1)

3. ส่วนการตรวจสอบการตั้งค่าแบบที่สอง (Verification mode 2) : ให้รอจนสถานะ (STATUS) ขึ้นว่า Successful ก่อน จากนั้นจึงไปดูที่หน้าต่างแสดงผลการทำงาน โดยดูที่บรรทัด Output ดังแสดงในรูปที่ ข.9



```
13 TASK [ospf : Output] ***** 17:45:31
14 ok: [192.168.20.11] => {
15     "msg": [
16         "Network activated by ospf are",
17         [
18             "192.168.100.1 0.0.0.0 area 0"
19         ]
20     ]
21 }
22 ok: [192.168.20.12] => {
23     "msg": [
24         "Network activated by ospf are",
25         [
26             ]
27     ]
28 }
```

ดูที่บรรทัดที่มีคำว่า
Output

รูปที่ ข.9 ตัวอย่างในส่วนการตรวจสอบการตั้งค่าแบบที่สอง (Verification mode 2)

ภาคผนวก ค

คู่มือการพัฒนาระบบ

ค.1. เข้าไปที่ Folder ของไฟล์ playbook

บน server ที่ติดตั้ง Ansible AWX ให้เข้าไปที่ `/var/lib/awx/projects` จากนั้นเลือก folder ของอุปกรณ์ที่ต้องการ ดังแสดงในรูปที่ ค.1 โดยภายในจะมี 3 folder ย่อย คือ Config, Verify และ Verify_mode2

```
root@awx_test:/var/lib/awx/projects# tree -d -L 1 New_*
New_Router
├── Config
├── Verify
└── Verify_mode2
New_Switch
├── Config
├── Verify
└── Verify_mode2
6 directories
```

รูปที่ ค.1 folder ภายใน `/var/lib/awx/projects`

ค.2. สร้างไฟล์ playbook

เมื่อเข้าไปที่ folder ที่ต้องการแล้วให้สร้างไฟล์ใน format YAML (.yaml) ซึ่งมีรูปแบบการสร้าง ดังแสดงในรูปที่ ค.2 โดยมีรายละเอียดดังนี้

1. hosts: ระบุ IP address ของเครื่องปลายทาง ในระบบได้สร้างเป็นตัวแปรในชื่อ `host_ip`
2. roles: เปรียบเสมือนการเรียกใช้ฟังก์ชัน โดยระบุชื่อ folder ที่อยู่ภายใต้ `./role` ในตัวอย่างเป็นการเรียกใช้ฟังก์ชัน `rip` โดยจะเป็นการเรียกไฟล์ `main.yml` ใน `/role/rip/task/main.yml`

```
---
- hosts: "{{ host_ip }}"
  gather_facts: no

  roles:
    - {role: rip}
```

รูปที่ ค.2 ตัวอย่างไฟล์ playbook

ค.3. สร้าง role

ใน folder /role ให้สร้าง folder ย่อยสำหรับการทำงานที่เราต้องการ โดยภายใน folder นั้น ให้สร้าง folder ย่อย ชื่อว่า task โดยภายใน /task จะเก็บไฟล์ playbook สำหรับการทำงานไว้ ซึ่งจำเป็นจะต้องมีไฟล์ main.yml ดังแสดงในรูปที่ ค.3

```
root@aux_test:/var/lib/awx/projects/New_Router/Config/roles# tree rip
rip
└── tasks
    ├── config_rip.yml
    ├── create_verify.yml
    ├── delete_rip.yml
    └── main.yml
```

รูปที่ ค.3 การสร้าง folder ภายใน /role

ค.4. สร้างไฟล์ main.yml

ในไฟล์ main.yml จะมีรายละเอียดดังนี้

1. include_tasks: จะเป็นการเรียกไฟล์ playbook อื่น ๆ ที่อยู่ภายใต้ /task
2. when: เป็นการตรวจสอบเงื่อนไข ทำงานเหมือนกับ if

โดยถ้าเป็นส่วนการตรวจสอบการตั้งค่าทั้งสองแบบ (Verification mode 1,2) ภายในไฟล์ main.yml จะเหมือนกัน โดยจะเรียกแค่ไฟล์ Verify.yml ดังแสดงในรูปที่ ค.4 - ค.5

```
- include_tasks: config_rip.yml
  when: mode == "Config"
- include_tasks: delete_rip.yml
  when: mode == "Delete"
- include_tasks: create_verify.yml
```

รูปที่ ค.4 ตัวอย่างไฟล์ main.yml ในส่วนการตั้งค่า (Configuration)

```
- include_tasks: Verify.yml
```

รูปที่ ค.5 ตัวอย่างไฟล์ main.yml ในส่วนการตรวจสอบการตั้งค่า (Verification mode 1,2)

ค.5. ตัวอย่างการสร้างไฟล์ config_rip.yml

ในส่วนการตั้งค่า (Configuration) จะใช้โมดูล ios_config และใช้โมดูลย่อย lines โดยจะเปรียบเสมือนการตั้งค่าในโหมด configure terminal บนอุปกรณ์เครือข่าย ดังแสดงในรูปที่ ค.6.

```
---
- name: Configure rip protocol
  ios_config:
    lines:
      - network {{ network }}
      - no auto
      - version 2
    parents:
      - router rip
```

รูปที่ ค.6 ตัวอย่างไฟล์ config_rip.yml

ค.6. การสร้างไฟล์ create_verify.yml สำหรับส่วนการตั้งค่า (Configuration)

สำหรับส่วนการตั้งค่า (Configuration) จะมีการสร้างไฟล์ playbook สำหรับส่วนการตรวจสอบการตั้งค่าแบบที่หนึ่ง (Verification mode 1) ดังแสดงในรูปที่ ค.7 - ค.8 โดยจะเก็บการตั้งค่าจากผู้ใช้ครั้งสุดท้ายไว้ โดยมีรายละเอียดดังนี้

1. ทำการลบไฟล์ Verify.yml อันเก่าทิ้ง โดยทำการลบไฟล์ที่ระบุใน path:
2. ทำการสร้างไฟล์ Verify.yml เปล่า ๆ ในตำแหน่งที่ระบุใน path:
3. ทำการคัดลอกข้อมูลที่จะใส่ในไฟล์ Verify.yml โดยมีรายละเอียดดังนี้
 - a. ภายในโมดูล content จะต้องเริ่มต้นด้วย “ และปิดท้ายด้วย ” เสมอ
 - b. ถ้าประกาศตัวแปรในรูปแบบ {{ ชื่อตัวแปร }} จะเป็นการนำค่าที่ได้จากผู้ใช้มาใส่
 - c. ถ้าต้องการคงรูปแบบ {{ ชื่อตัวแปร }} ในไฟล์ Verify.yml ให้ครอบด้วย {% raw %} ... {% endraw %}

```

---
- name: Remove Verify.yml
  file:
    path: /var/lib/awx/projects/New_Router/Verify/roles/rip/tasks/Verify.yml
    state: absent

- name: Touch Verify.yml
  file:
    path: /var/lib/awx/projects/New_Router/Verify/roles/rip/tasks/Verify.yml
    state: touch
    mode: u=rw,g=r,o=r

- name: Copy contents to Verify.yml
  copy:
    content:
      "---
      \n- name: Verify IP address
      \n  ios_command:
      \n    commands:
      \n      - show run | include {{ network }}
      \n  register: ip
      \n- name: Verify network to activate
      \n  ios_command:
      \n    commands:
      \n      - show ip protocol | section rip
      \n  register: rip
      \n- name: Get subnet mask to change IP address to classful
      \n  checkClassful:
      \n    ip: {% raw %}{{ ip.stdout[0].split()[2] }}{% endraw %}
      \n  register: classful
      \n- name: Change IP address to classful

```

รูปที่ ค.7 ตัวอย่างไฟล์ create_verify.yml

```

\n- name: Correct
\n  debug:
\n    msg:
\n      - Your rip configuration is correct
\n      - You input IP address {{ network }} to activate
\n      - IP address {{ network }} convert to classful to {% raw %}{{ result.msg }}{% endraw %}
\n      - IP classful to activated by rip is
\n      - {% raw %}{{ rip.stdout[0] | regex_search(result.msg) }}{% endraw %}
\n  when: {% raw %}ip.stdout[0] {% endraw %} != \"\" and result.msg == rip.stdout[0] | regex_s$
\n- name: Incorrect
\n  debug:
\n    msg:
\n      - Your rip configuration is incorrect
\n      - You input IP address {{ network }} to activate
\n      - IP address {{ network }} convert to classful to {% raw %}{{ result.msg }}{% endraw %}
\n      - IP classful to activated by rip is
\n      - {% raw %}{{ rip.stdout[0] | regex_search(result.msg) }}{% endraw %}
\n  when: ({% raw %}ip.stdout[0] {% endraw %} != \"\") and ((rip.stdout[0] == \"\") or (result$
\n- name: Incorrect(no IP address {{ network }})
\n  debug:
\n    msg:
\n      - IP address {{ network }} is not on the router interface
\n  when: {% raw %}ip.stdout[0] {% endraw %} == \"\"
  dest: /var/lib/awx/projects/New_Router/Verify/roles/rip/tasks/Verify.yml
  when: mode == "Config"

```

รูปที่ ค.8 ตัวอย่างไฟล์ create_verify.yml

ค.7. การสร้างไฟล์ Verify.yml สำหรับส่วนการตรวจสอบการตั้งค่าแบบที่สอง (Verification mode 2)

ในส่วนการตรวจสอบการตั้งค่าแบบที่สอง (Verification mode 2) จะใช้โมดูล ios_command และใช้โมดูลย่อย commands ดังแสดงในรูปที่ ค.9

```
---
- name: Verify
  ios_command:
    commands:
      - show run | section rip
  register: rip
- name: Output
  debug:
    msg:
      - Network activated by rip are
      - "{{ rip.stdout[0] | regex_findall('\b(?:[0-9]{1,3}\.){3}[0-9]{1,3}\b') }}"
```

รูปที่ ค.9 ตัวอย่างไฟล์ Verify.yml ในส่วนการตรวจสอบการตั้งค่าแบบที่สอง(Verification mode 2)