



บทความเผยแพร่ความรู้

เรื่อง Python for Network Engineer

ผู้จัดทำ

อาจารย์ ดร.ศรัณย์ นาคถนอม

สาขาวิชาวิทยาศาสตร์และเทคโนโลยี

โครงการนี้ได้รับการสนับสนุนจากทุนพัฒนาบุคลากรเพื่อการศึกษาทางไกล

ประเภทรายบุคคล ประจำปีงบประมาณ 2565

ความรู้เกี่ยวกับ Python for Network Engineer

2.4.1 Paramiko คือ SSH library บน Python มีความสามารถในการทำงานที่ดีและสามารถพัฒนาได้มากกว่า Netmiko สำหรับการทำ SSH ไปหาอุปกรณ์ปลายทาง หรือการใช้ transfer ไฟล์ด้วย SFTP

ซึ่งข้อเสียของ Paramiko ก็คือความสามารถพัฒนาของ Paramiko เพราะทำให้เราต้องเขียน code มากขึ้นเพื่อให้ตอบโจทย์การทำงาน เช่น การทำ SSH ไปยัง Cisco Router เพื่อที่จะเข้า config mode ต้องเขียน code เข้าถึง privilege mode ให้เจอ prompt เป็น # แสดงว่าไม่ได้อยู่ใน config mode ก็ให้ส่ง configure terminal ไปยังอุปกรณ์

ฟังก์ชันของ Paramiko เบื้องต้น

`set_missing_host_key_policy(paramiko.AutoAddPolicy())` คือ เพิ่ม SSH key ไปที่ไฟล์ `known_hosts`

`client.connect()` คือ เริ่ม connect ไปที่ SSH server

`client.close()` คือ ปิด SSH connection

`client.invoke_shell()` คือ สร้าง session

`session.send()` คือ ส่ง command ผ่าน session จาก `invoke_shell()`

`session.recv()` คือ รับ response จาก server ผ่าน session จาก `invoke_shell()`

2.4.2 Netmiko คือ SSH library บน Python จะมีฟังก์ชันที่เน้นไปที่การ SSH ไปยังอุปกรณ์เครือข่าย และรองรับกับอุปกรณ์ต่างๆ มีอ้างอิงใน Document ของ Netmiko

โดย Netmiko จะทำงานภายใต้ Paramiko ซึ่งเพิ่มความสะดวกในการใช้งานให้กับผู้พัฒนา ด้วยความที่ Netmiko มีรูปแบบ high level มากกว่า Paramiko ทำให้ใช้ได้ง่ายเพราะการพัฒนาในการเขียน code น้อยกว่า ยกเว้นไม่รองรับกับอุปกรณ์เครือข่าย ซึ่งจะกลายเป็นข้อเสียทันที

ฟังก์ชันของ Netmiko เบื้องต้น

`ConnectHandler()` คือ สร้าง object ของ connection/session

`enable()` คือ เข้าสู่ privilege mode

`config_mode()` คือ เข้าสู่ configuration mode

`exit_enable_mode()` คือ ออกจาก privilege mode

`exit_config_mode()` คือ ออกจาก configuration mode

`send_command()` คือ ส่ง command และ return ค่า output

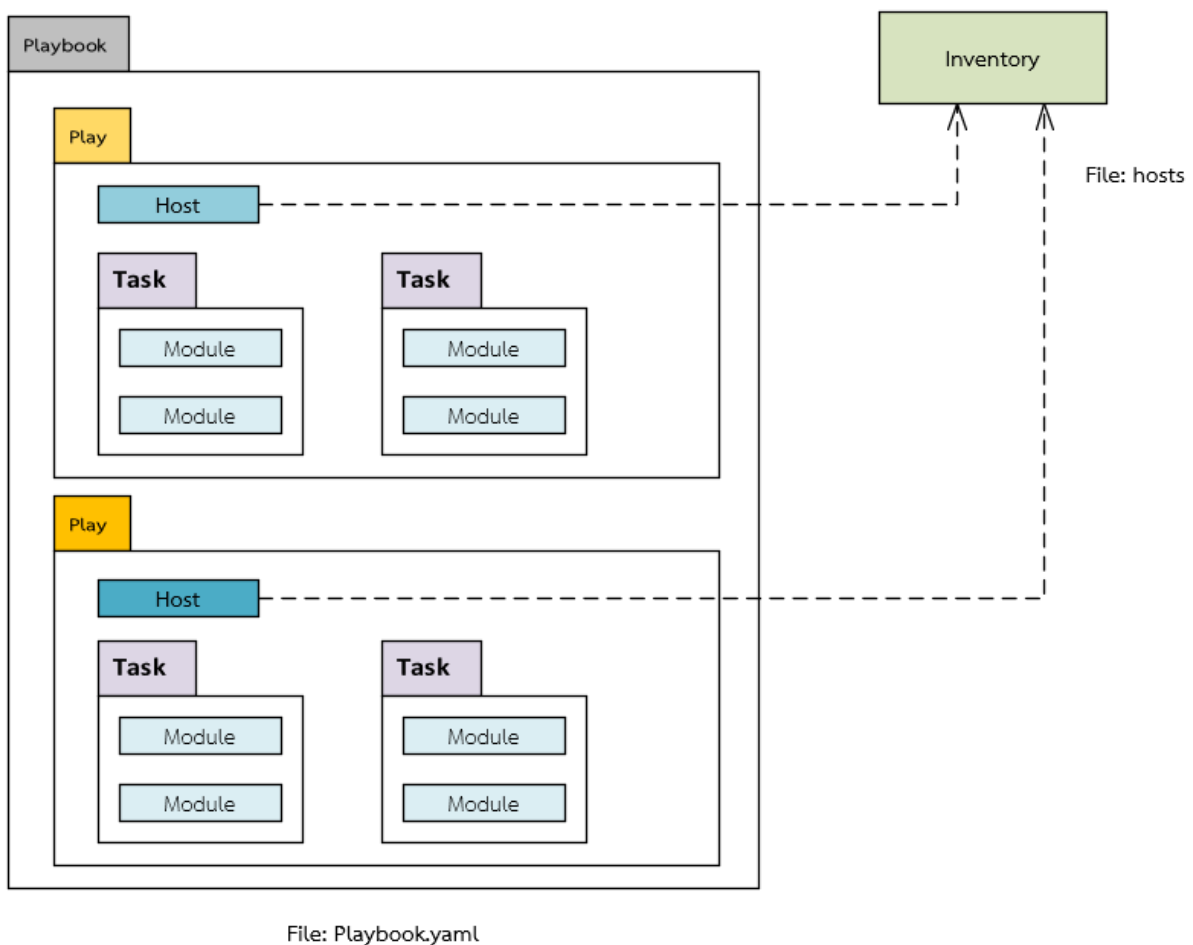
send_config_set() คือ ส่ง command เป็น list และ return ค่า output

disconnect() คือ ตัด SSH connection

2.4.3 Ansible

1) ภาพรวมโครงสร้างของ Ansible Playbook

การทำงาน Ansible นั้น จะต้องทำงานผ่าน playbook ซึ่งประกอบด้วยส่วนต่าง ๆ หลายส่วน ดังรูปด้านล่างนี้



ภาพที่ 1 โครงสร้างของ Ansible Playbook

ภาพแสดงไฟล์ playbook ที่เขียนอยู่ในรูปแบบของ YAML

```

1 ---
2 - hosts: all
3   tasks:
4     - name: Check IP Connectivity
5       ping:
6     - name: Install NGINX
7       apt: name=nginx update_cache=yes

```

ภาพที่ 2 ไฟล์ playbook.yaml

ซึ่งเมื่อเขียนอธิบายส่วนประกอบต่าง ๆ ได้ดังนี้

```

1 ---
2 - hosts: all
3   tasks:
4     - name: Check IP Connectivity
5       ping:
6     - name: Install NGINX
7       apt: name=nginx update_cache=yes

```

Diagram illustrating the components of the playbook.yaml file:

- Play**: The entire configuration block (lines 2-7) is enclosed in a green box.
- Host (ซึ่งไม่หาไฟล์ Inventory)**: The `hosts: all` line (line 2) is highlighted with a red box.
- Task**: The task blocks are highlighted with yellow boxes.
 - Task 1 (lines 4-5): `- name: Check IP Connectivity` and `ping:`. The `ping:` line is further highlighted with a pink box and labeled **Module**.
 - Task 2 (lines 6-7): `- name: Install NGINX` and `apt: name=nginx update_cache=yes`. The `apt:` line is further highlighted with a pink box and labeled **Module**.

ภาพที่ 3 ส่วนประกอบ ไฟล์ playbook.yaml

2) Playbook

Playbook คือ ใบสั่งงานที่เป็นไฟล์ที่ใช้เรียกว่า Ansible จะต้องทำงานอย่างไร และทำงานกับ host ไตบ้าง ซึ่งสามารถเขียนโดยใช้รูปแบบของ YAML ข้อดีของ Ansible ที่ดีกว่า Configuration Management Tool เครื่องมืออื่น ๆ คือไม่ต้องเสียเวลาในการเรียนรู้ภาษา เฉพาะในการเขียนสั่งการทำงานแบบ Chef หรือ Puppet เนื่องจาก YAML เป็น Data Format ที่ใช้งานกันอย่างแพร่หลายในปัจจุบัน

2.1) Play

ภายใน playbook จะต้องมียี่ play เป็นอย่างน้อย ซึ่งใน play ก็จะเป็นการกำหนดการทำงานบางอย่างให้กับ host แต่ในหนึ่ง playbook ของสามารถมีได้หลาย play ขึ้นอยู่กับการจัดการของผู้ใช้งาน โดยมีขั้นตอนดังนี้

- กำหนด host ใน play จะทำงานกับ host ไດบ้าง
- มี task เพื่อทำงานกับ host ที่ระกำหนดไว้

2.2) Task

Task คือ งาน ที่ต้องทำจบการทำงาน ซึ่งอาจมีหลาย task ภายในหนึ่ง play ได้ เช่น play ที่ติดตั้ง NGINX ซึ่งการติดตั้งอาจมีหลาย task ประกอบกัน

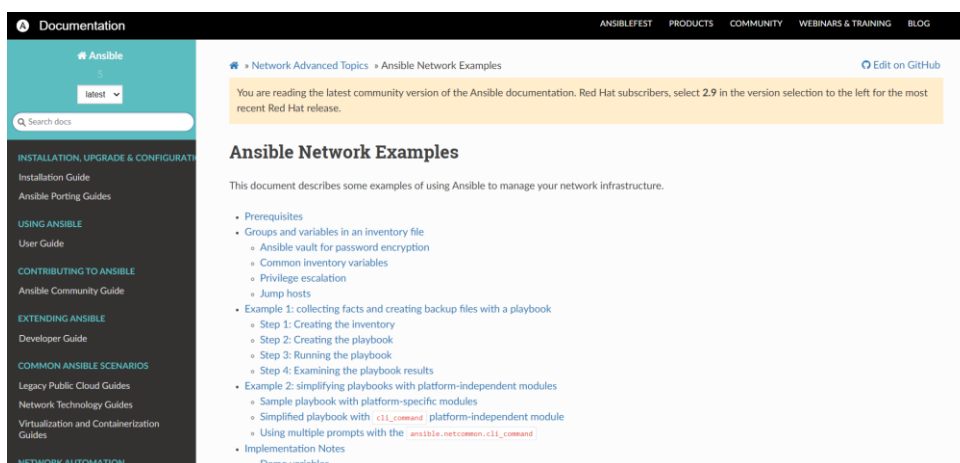
2.3) Module

Module คือ script หรือ code สามารถเรียกใช้เพราะมี built-in เรียบร้อย เพื่อให้ Ansible นั้นสามารถทำงานร่วมกับ host ปลายทางได้ ส่วนนี้ไม่ต้องเขียน Code วัตถุประสงค์เพื่อทำ Automation เช่น ถ้าต้องการ copy ไฟล์ sarun.txt จากเครื่องทำงาน โดยใช้ Ansible ไปยังไดเรกทอรี opt ของ host ทั้งหมด จะเรียกใช้ copy module ดังตัวอย่าง

```
- name: Copy sarun.txt to destination host(s)
  copy:
    src: /files/sarun.txt
    dest: /opt/sarun.txt
```

ซึ่งใน Ansible มี module ที่หลากหลาย สามารถศึกษาได้จาก URL Link

https://docs.ansible.com/ansible/2.9/modules/modules_by_category.html



ภาพที่ 4 Ansible Documentation