

```

graph TD
    subgraph Control_Node [Control Node]
        subgraph Default_Settings [## Default Settings]
            D1[ansible_connection: ssh (ssh.py)]
            D2[ansible_port: 22]
            D3[ansible_user: root (variable file)]
            D4[## Remote_user=devops (ansible_cfg)]
            D5[ansible_password: redhat (SSH_KEY) [--ask-pass for Ad-hoc]]
        end

        subgraph Windows_Settings [## Custom Windows Settings]
            W1[ansible_connection: winrm (winrm.py)]
            W2[ansible_port: 5986]
            W3[ansible_user: devops (remote_user)]
            W4[ansible_password: redhat (SSH_KEY)]
        end

        subgraph Network_Device_Settings [## Custom Network Device Settings]
            N1[ansible_connection: network_cli/netconf (network_cli.py/netconf.py)]
            N2[ansible_port: 22 (Depend)]
            N3[ansible_user: admin/vyos (remote_user)]
            N4[ansible_password: student/vyos (SSH_KEY)]
        end
    end

    subgraph Execution_Machine [Execution Machine]
        subgraph Execution_Node [Execution Node]
            E1((thread))
            E2((thread))
            E3((thread))
            E4((thread))
            E5[Image 1  
python 3.9]
        end

        subgraph Execution_Node_2 [Execution Node]
            E6((thread))
            E7((thread))
            E8((thread))
            E9((thread))
            E10[Image 2  
python 3.6]
        end
    end

```

from Control Node to Managed Host (Temp Folder)  
 ion of module.py to executable via chmod u+x module.py  
 using "python module.py" and make changes(if not DESIRED)  
 which was copied (rm -rf Module\_path)

## Default Settings  
 ansible\_connection: ssh (ssh.py)  
 ansible\_port: 22  
 ansible\_user: root (variable file)  
 ## Remote\_user=devops (ansible\_cfg)  
 ansible\_password: redhat (SSH\_KEY) [--ask-pass for Ad-hoc]

## Custom Windows Settings  
 ansible\_connection: winrm (winrm.py)  
 ansible\_port: 5986  
 ansible\_user: devops (remote\_user)  
 ansible\_password: redhat (SSH\_KEY)

## Custom Network Device Settings  
 ansible\_connection: network\_cli/netconf (network\_cli.py/netconf.py)  
 ansible\_port: 22 (Depend)  
 ansible\_user: admin/vyos (remote\_user)  
 ansible\_password: student/vyos (SSH\_KEY)

**Execution Node**

thread  
 thread  
 thread  
 thread  
 Image 1  
 python 3.9

thread  
 thread  
 thread  
 thread  
 Image 2  
 python 3.6

**Execution Machine**

1. Graphical tool to work with Ansible
2. Easy based on Browser
3. Schedule your play's to run as specific time
4. Logs of plays run
5. RBAC

```
yum install ansible-core ansible-navigator python -y
```

Dynamic [Active Directory, LDAP, Satellite  
CMDB, Openstack,AWS,VMWare,  
VirtualBox] via python script

1. Ad-Hoc Command (Development/Testing) One time NOT reusable  
ansible hostname -m Module\_name -a 'option1=value1 option2=value2'
2. Playbook (Production) Reusable as many time  
ansible-playbook play1.yml

Ansible CORE 2.11  
builtin Collecti

```
vim play.yaml
- name: Playbook Purpose
  hosts: list_host
  tasks:
    - name: Install httpd
      yum:
        name: httpd
        state: latest
    - name: Create user
      Module_name:
        Module_options:
```

```
control 1 cisco 1 modules python 2.7
control 2 cisco 2 modules python 3.6
```

## control 2 cisco 2 modules python 3.8

```
## Default Settings
ansible_connection: ssh (ssh.py)
ansible_port: 22
ansible_user: root (variable file)
# remote_user=devops (ansible.cfg)
ansible_password: redhat (SSH_KEY) [--ask-pass for Ad-hoc]
```

```
# Custom Windows Settings
ansible_connection: winrm (winrm.py)
ansible_port: 5986
ansible_user: devops (remote_user)
ansible_password: redhat (SSH_KEY)
```

```
# Custom Network Device Settings
ansible_connection: network_cli/netconf
                    (network_cli.py/netconf.py)
ansible_port: 22 (Depend)
ansible_user: admin/vyos (remote_user)
ansible_password: student/vyos (SSH_KEY)
```

```
Python Installed Linux
[privilege_escalation]
become: true
become_user: root
become_method: sudo/su
become_ask_pass: true/false
```

```
become: true
become_user: administrator
become_method: runas
become_ask_pass: true/false
```

```
become: true
become_user: admin
become_method: enable
become_ask_pass: true/false
```

Linux

OWS

network  
service



user

Execution Node




Diagram illustrating a process containing four threads, each labeled 'thread'.

## Execution Machine