Ansible

https://docs.ansible.com/ansible/latest/index.html #документация

Требует python и sshpass

sudo pip install -U $(pip freeze | cut -d '=' -f 1) обновляем все пакеты pip

pip install ansible

ansible - - version

ansible all -i inventory -m ping #пингуем все хосты указанные в inventory

вместо all можно указать определенные группы

ansible all -i inventory -m setup #выдает всю информацию о хостах

ansible all -m shell -a "uptime" #выполнить команду, например: «shell»

ansible all -i inventory -m file -a «path=/root/file.txt status=touch» #создать файл

ansible all -i inventory -m copy -a «src=.. dest=/etc mode=777» -b #скопировать файл и назначить права

**Конфигурационный файл**

создаем ansible.cfg

пример

[defaults]

inventory = ~/ansible/hosts #указываем местоположение inventory по умолчанию

host\_key\_checking = false #отключить key\_cheking

**inventory**

**ex**

host0.example.org ansible\_ssh\_host=192.168.1.2 ansible\_ssh\_user=root ansible\_password=123

192.168.1.1 ansible\_user=root ansible\_ssh\_private\_key\_file=/ #подключаемся по ключу, предварительно отправить добавить ключ в файл authorized\_keys на хосте

**Группы и children**

**[test]**

host0.example.org ansible\_ssh\_host=192.168.1.2 ansible\_ssh\_user=root ansible\_password=123

**[group2]**

host1.example.org ansible\_ssh\_host=192.168.1.3 ansible\_ssh\_user=root ansible\_password=123

**[all\_groups:children]**

**group2**

**group3**

**Group Vars**

1. mkdir group\_vars
2. touch test #создаем файл с названием группы

touch group2

3) добавляем в файл

ansible\_host: 192.168.1.1

ansible\_user: root

ansible\_password: 123

4) редактируем inventory

[test]

host1

**Playbooks**

ex. пинг хостов

создаем файл ping.yaml

- name: Ping hosts

hosts: all

become: yes

tasks:

- name: Task Ping

ping:

ansible-playbook -i hosts ping.yaml

Пример для centos 8

- name: Update Hosts and install apache

hosts: all

become: yes

tasks:

- name: Update cache

dnf:

update\_cache: yes

- name: Upgrade all packages

dnf:

name: "\*"

state: latest

- name: Uninstall unneeded packages

dnf:

autoremove: yes

- name: Install apache

dnf:

name: httpd

state: latest

создат директорию

- name: Mkdir test

file:

path: /root/test

state: directory

mode: 0777

**переменные**

vars:

file\_src: ./ping.yaml

file\_dest: /root/test

task

- name: Copy

copy:

src: "{{file\_src}}"

dest: "{{file\_dest}}"

mode: 0777

**циклы**

- name: loops

hosts: all

become: yes

tasks:

- name: Create folder

file:

path: "/root/{{item}}"

state: directory

loop:

- dir1

- dir2

**создаем группы и пользователей**

- name: Create groups

group:

name: "{{item}}"

state: present

loop:

- dev

- test

- name: Create users

user:

name: "{{item.clientname}}" #создаем на каждой машине по 3 новых пользователя с 3 домашними директориями

shell: /bin/bash

groups: dev, test #добавлям пользователей в 2 группы

append: yes

home: "/home/{{item.homedir}}"

with\_items:

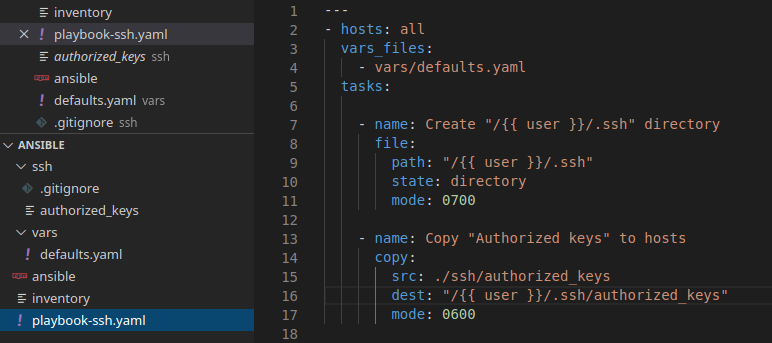
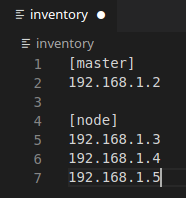
- {clientname: client01, homedir: client5}

- {clientname: client02, homedir: client6}

- {clientname: client03, homedir: client7}

можно создать переменную в inventory для каждого пользователя, например «user\_client=client1» и добавить в name «{{user.client}}», для каждой машины будет создан свой пользователь

пример добавления авторизации по ключу



ansible-playbook -i inventory playbook-ssh.yaml --user=root --ask-pass