nsd_1908_devops_day03

ansible

```
# 安装
(nsd1908) [root@room8pc16 day03]# pip install /var/ftp/pub/zzg_pypkgs/ansible_pkg/*

# 配置
(nsd1908) [root@room8pc16 day03]# mkdir myansible
(nsd1908) [root@room8pc16 day03]# cd myansible
##############################
# 如果远程用户是普通用户，使用此方式
(nsd1908) [root@room8pc16 myansible]# vim ansible.cfg
[defaults]
inventory = hosts
remote_user = tom

[privilege_escalation]
become = True
become_method = sudo
become_user = root
become_ask_pass = False
# 被管理的所有主机需要授权tom可以执行任何管理员命令
[root@room8pc16 pub]# visudo
tom    ALL=(ALL)   NOPASSWD: ALL
######################################
# 直接使用root用户对远程主机进行管理
(nsd1908) [root@room8pc16 myansible]# vim ansible.cfg
[defaults]
inventory = hosts
remote_user = root

(nsd1908) [root@room8pc16 myansible]# vim hosts
[dbservers]
node4

[webservers]
node5
node6

# 配置名称解析
(nsd1908) [root@room8pc16 myansible]# for i in {1..254}
> do
> echo -e "192.168.4.$i\tnode$i.tedu.cn\tnode$i" >> /etc/hosts
> done

# 收集密钥
(nsd1908) [root@room8pc16 myansible]# ssh-keyscan node{4..6} >> ~/.ssh/known_hosts
```

```
# 配置免密登陆
(nsd1908) [root@room8pc16 myansible]# for host in node{4..6}; do ssh-copy-id $host; done

# 配置免密登陆的playbook
# 该playbook的执行需要sshpass软件包
# (nsd1908) [root@room8pc16 myansible]# yum install sshpass
(nsd1908) [root@room8pc16 myansible]# vim myssh_key.yml
---
- name: configure ssh pubkey
  hosts: all
  tasks:
    - name: upload pub key
      authorized_key:
        user: root
        state: present
        key: "{{ lookup('file', '/root/.ssh/id_rsa.pub') }}"
(nsd1908) [root@room8pc16 myansible]# ansible-playbook myssh_key.yml -k

# 配置vim，使它支持yaml的特点
(nsd1908) [root@room8pc16 myansible]# vim ~/.vimrc
autocmd FileType yaml setlocal sw=2 ts=2 et ai

# 配置yum
(nsd1908) [root@room8pc16 myansible]# vim yum.yml
- name: configure yum repo
  hosts: all
  tasks:
    - name: configure yum client
      yum_repository:
        name: centos7
        description: centos 7.4
        baseurl: ftp://192.168.4.254/centos7.4
        file: centos7
        gpgcheck: no
        enabled: yes
# 检查语法
(nsd1908) [root@room8pc16 myansible]# ansible-playbook --syntax-check yum.yml
# 执行playbook
(nsd1908) [root@room8pc16 myansible]# ansible-playbook yum.yml


# lamp配置
(nsd1908) [root@room8pc16 myansible]# vim lamp.yml
---
- name: configure dbservers
  hosts: dbservers
  tasks:
    - name: install db pkgs
      yum:
        name: mariadb-server
        state: present
    - name: configure db serivce
```

```
      service:
        name: mariadb
        state: started
        enabled: yes

- name: configure webservers
  hosts: webservers
  tasks:
    - name: install web pkgs
      yum:
        name: [httpd, php, php-mysql]
        state: present
    - name: configure web serivce
      service:
        name: httpd
        state: started
        enabled: yes
(nsd1908) [root@room8pc16 myansible]# ansible-playbook --syntax-check lamp.yml
(nsd1908) [root@room8pc16 myansible]# ansible-playbook lamp.yml
```

ansible编程

- 官方文档：http://docs.ansible.com/ -> Ansible Documentation -> 选2.7版本 -> 搜索python api，将example中的代码复制、执行

命名元组

- 仍然是元组，只是扩展了元组的功能
- 命名元组，为每个下标命名

```
>>> from collections import namedtuple
>>> Point = namedtuple('Point', ['x', 'y', 'z'])
>>> p1 = Point(10, 20, 12)
>>> p1[-1]
12
>>> p1[:2]
(10, 20)
>>> p1.x
10
>>> p1.y
20
>>> p1.z
12
```

将yaml文件转成python的数据类型

```
---
- name: configure dbservers
  hosts: dbservers
  tasks:
    - name: install db pkgs
      yum:
        name: mariadb-server
        state: present
    - name: configure db serivce
      service:
        name: mariadb
        state: started
        enabled: yes

- name: configure webservers
  hosts: webservers
  tasks:
    - name: install web pkgs
      yum:
        name: [httpd, php, php-mysql]
        state: present
    - name: configure web serivce
      service:
        name: httpd
        state: started
        enabled: yes
```

```
[
    {
        name: configure dbservers,
        hosts: dbservers,
        tasks: [
            {
                name: install db pkgs,
                yum: {
                    name: mariadb-server,
                    state: present
                }
            },
            {
                name: configure db serivce,
                service: {
                    name: mariadb,
                    state: started
                    enabled: yes
                }
            }
        ]
    },
    {
        name: configure webservers,
        hosts: webservers,
        tasks: [
            {
                name: install web pkgs,
                yum: {
                    name: [httpd, php, php-mysql],
                    state: present
                }
            },
            {
                name: configure web serivce,
                service: {
                    name: httpd,
                    state: started,
                    enabled: yes
                }
            }
        ]
    }
]
```

ansible加密文件

```
(nsd1908) [root@room8pc16 myansible]# cp /etc/hosts /tmp/
# 加密文件
(nsd1908) [root@room8pc16 myansible]# ansible-vault encrypt /tmp/hosts
New Vault password:
Confirm New Vault password:
(nsd1908) [root@room8pc16 myansible]# cat /tmp/hosts

# 解密
(nsd1908) [root@room8pc16 myansible]# ansible-vault decrypt /tmp/hosts
Vault password:
(nsd1908) [root@room8pc16 myansible]# cat /tmp/hosts
```

编写ansible模块

```
# 设定ansible查找自定义模块的路径
(nsd1908) [root@room8pc16 day03]# export ANSIBLE_LIBRARY=/tmp/mylibs

# 编写模块
(nsd1908) [root@room8pc16 day03]# vim /tmp/mylibs/rcopy.py
import shutil
from ansible.module_utils.basic import AnsibleModule

def main():
    module = AnsibleModule(
        argument_spec=dict(
            yuan=dict(required=True, type='str'),
            mubiao=dict(required=True, type='str')
        )
    )
    shutil.copy(module.params['yuan'], module.params['mubiao'])
    module.exit_json(changed=True)

if __name__ == '__main__':
    main()
```

```
(nsd1908) [root@room8pc16 day03]# cd myansible/
(nsd1908) [root@room8pc16 myansible]# ansible dbservers -m rcopy -a "yuan=/etc/hosts mubiao=/tmp/zhuji.txt"
```

实现下载功能的模块

```
(nsd1908) [root@room8pc16 myansible]# vim /tmp/mylibs/download.py
import wget
from ansible.module_utils.basic import AnsibleModule

def main():
    module = AnsibleModule(
        argument_spec=dict(
            url=dict(required=True, type='str'),
            dest=dict(required=True, type='str')
        )
    )
    wget.download(module.params['url'], module.params['dest'])
    module.exit_json(changed=True)

if __name__ == '__main__':
    main()

(nsd1908) [root@room8pc16 myansible]# ansible dbservers -m download -a "url=http://192.168.4.254/zabbix.png dest=/tmp/"
# 上一步将会出现ImportError: No module named wget。这是因为远程主机没有wget模块，不是ansible主机没有该模块
```

安装python模块

```
# 安装pip(安装其他python模块也是一样的方法)
# https://pypi.org上下载pip
# 在目标主机上安装pip
[root@node4 ~]# tar xf pip-19.3.1.tar.gz
[root@node4 ~]# cd pip-19.3.1/
[root@node4 pip-19.3.1]# python setup.py install

# 安装wget，可以使用与安装pip一样的方法
# 既然有了pip，可以使用pip安装wget
[root@node4 ~]# pip install wget-3.2.zip
```

ansible-cmdb插件

```
# 收集远程主机的信息
(nsd1908) [root@room8pc16 myansible]# ansible all -m setup --tree /tmp/nsd1908out
(nsd1908) [root@room8pc16 myansible]# ls /tmp/nsd1908out/
node4  node5  node6

# 安装ansible-cmdb
(nsd1908) [root@room8pc16 myansible]# pip install /var/ftp/pub/zzg_pypkgs/ansible-cmdb_pkgs/*

# 使用ansible-cmdb将远程主机信息生成html文件
(nsd1908) [root@room8pc16 myansible]# ansible-cmdb /tmp/nsd1908out/ > /tmp/hosts.html
(nsd1908) [root@room8pc16 myansible]# firefox /tmp/hosts.html
```