

nsd1905_devops_day03

ansible应用

```
(nsd1905) [root@room8pc16 ~]# pip install zzg_pypkgs/ansible_pkg/*
# 或在线安装
(nsd1905) [root@room8pc16 day03]# pip install ansible==2.7.2

# 创建工作目录
(nsd1905) [root@room8pc16 day03]# mkdir myansible
(nsd1905) [root@room8pc16 day03]# cd myansible
# 创建配置文件
(nsd1905) [root@room8pc16 myansible]# vim ansible.cfg
[defaults]
inventory = hosts
remote_user = root
# 创建主机清单文件
(nsd1905) [root@room8pc16 myansible]# vim hosts
[dbservers]
node4

[webservers]
node5
node6
# 准备三台虚拟机
node4: 192.168.4.4
node5: 192.168.4.5
node6: 192.168.4.6
配置可以对其免密登陆
(nsd1905) [root@room8pc16 myansible]# for i in {4..6}
> do
> ssh-copy-id root@192.168.4.$i
> done
配置名称解析
(nsd1905) [root@room8pc16 day01]# for i in {1..254}
> do
> echo -e "192.168.4.$i\tnode$i.tedu.cn\tnode$i" >> /etc/hosts
> done
收集主机密钥
(nsd1905) [root@room8pc16 day01]# ssh-keyscan node{4..6} node{4..6}.tedu.cn 192.168.4.
{4..6} >> ~/.ssh/known_hosts

# 测试到远程主机的通信
(nsd1905) [root@room8pc16 myansible]# ansible all -m ping

# 配置vim, 使编写yaml文件时, 更方便
(nsd1905) [root@room8pc16 day01]# vim ~/.vimrc
autocmd FileType yaml setlocal sw=2 ts=2 et ai
```

通过ansible配置远程服务器的yum

```
(nsd1905) [root@room8pc16 myansible]# vim yum.yml
```

```
---
```

```
- name: configure yum repo
```

```
hosts: all
```

```
tasks:
```

```
  - name: make yum client config
```

```
    yum_repository:
```

```
      file: server
```

```
      name: server
```

```
      description: centos 7.4 repo
```

```
      baseurl: ftp://192.168.4.254/centos7.4
```

```
      enabled: yes
```

```
      gpgcheck: no
```

```
(nsd1905) [root@room8pc16 myansible]# ansible-playbook --syntax-check yum.yml
```

```
(nsd1905) [root@room8pc16 myansible]# ansible-playbook yum.yml
```

编写playbook，完成对web服务器和db服务器的配置

webserver上安装httpd、php、php-mysql并启动服务

dbserver上安装mariadb-server并启动服务

```
(nsd1905) [root@room8pc16 myansible]# vim lamp.yml
```

```
---
```

```
- name: config webserver
```

```
hosts: webserver
```

```
tasks:
```

```
  - name: install web pkgs
```

```
    yum:
```

```
      name:
```

```
        - httpd
```

```
        - php
```

```
        - php-mysql
```

```
      state: present
```

```
  - name: config web service
```

```
    service:
```

```
      name: httpd
```

```
      state: started
```

```
      enabled: yes
```

```
- name: config dbserver
```

```
hosts: dbserver
```

```
tasks:
```

```
  - name:
```

```
    yum:
```

```
      name: mariadb-server
```

```
      state: present
```

```
  - name: config db service
```

```
    service:
```

```
      name: mariadb
```

```
      state: started
```

```
enabled: yes
(nsd1905) [root@room8pc16 myansible]# ansible-playbook lamp.yml
```

ansible编程

找到ansible的官方手册：<https://docs.ansible.com/> -> ansible documentation -> 2.7 -> 搜索python api。将python api页面中的example拷贝过来。

命名元组

- 本质上还是元组
- 为元组的下标起名

```
>>> from collections import namedtuple
>>> Point = namedtuple('Point', ('x', 'y', 'z'))
>>> p1 = Point(5, 10, 20)
>>> type(p1)
<class '__main__.Point'>
>>> p1[0]
5
>>> p1[1:]
(10, 20)
>>> p1.x
5
>>> p1.y
10
>>> p1.z
20
```

将playbook转换成python数据类型

将lamp.yml转成python数据类型

```
[
  {
    name: config webservers,
    hosts: webservers,
    tasks: [
      {
        name: install web pkgs,
        yum: {
          name: [httpd, php, php-mysql],
          state: present
        }
      },
      {
        name: config web service,
        service: {
          name: httpd,
```

```

        state: started,
        enabled: yes
    }
},
]
},
{
    name: config dbservers,
    hosts: dbservers,
    tasks: [
        {
            name: install db server,
            yum: {
                name: mariadb-server,
                state: present
            }
        },
        {
            name: config db service,
            service: {
                name: mariadb,
                state: started,
                enabled: yes
            }
        }
    ]
},
]

```

编写ansible模块

- 创建自定义模块目录
- 设置Ansible查找模块的环境变量

```

(nsd1905) [root@room8pc16 day03]# mkdir /tmp/mymodules
(nsd1905) [root@room8pc16 day03]# export ANSIBLE_LIBRARY=/tmp/mymodules

```

编写模块，模块用于在远程主机上将一个文件拷贝到其他位置

```

(nsd1905) [root@room8pc16 day03]# vim /tmp/mymodules/rcopy.py
#!/root/nsd1905/bin/python
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()

# 执行任务
(nsd1905) [root@room8pc16 myansible]# ansible webservers -m rcopy -a "yuan=/etc/hosts
mubiao=/tmp/zhuji.txt"

```

编写实现下载功能的模块

在目标主机上安装python软件包

```

[root@node4 ~]# unzip wget-3.2.zip
[root@node4 ~]# cd wget-3.2/
[root@node4 wget-3.2]# python setup.py install

```

```

(nsd1905) [root@room8pc16 myansible]# vim /tmp/mymodules/download.py
#!/root/nsd1905/bin/python
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()

(nsd1905) [root@room8pc16 myansible]# ansible dbbservers -m download -a
"url=http://192.168.4.254/zabbix.png dest=/tmp/"

```

ansible-cmdb : 将ansible收集到的主机信息以web方式呈现

安装

```
(nsd1905) [root@room8pc16 ~]# pip install zzg_pypkgs/ansible-cmdb_pkgs/*
```

通过setup模块收集远程主机信息，保存到/tmp/info

```
(nsd1905) [root@room8pc16 myansible]# ansible all -m setup --tree /tmp/info
```

使用ansible-cmdb分析输出内容，并生成网页

```
(nsd1905) [root@room8pc16 myansible]# ansible-cmdb /tmp/info/ > /tmp/hosts.html
```