

## 第1章 ansible介绍

- 1.什么是ansible
- 2.为什么需要ansible
- 3.如何学习ansible

## 第2章 Ansible安装部署

## 第3章 Ansible主机清单

- 1.什么是主机清单
- 2.主机分组执行
- 3.所有的主机都执行
- 4.SSH使用密码连接并且端口号不是22
- 5.同组主机SSH端口号不一样，账号密码也不一样
- 6.同一组连续的IP
- 7.同一组具有相同的变量
- 8.所有主机都生效的变量

## 第4章 Ansible常用模块

- 0.如何学习ansible模块
- 1.ping 测试连通性
- 2.command 简单命令模块
- 3.shell 万能模块
- 4.copy 拷贝文件
- 5.file 文件相关
- 10.yum 安装软件
- 11.systemd 服务启动

# 第1章 ansible介绍

## 1.什么是ansible

1. Python写的一套自动化运维工具
2. ansible基于SSH协议通讯

## 2.为什么需要ansible

1. 有状态管理
2. 批量部署,批量执行命令
3. 统一配置管理,模板管理
4. 批量收集主机信息
5. 批量分发文件

## 3.如何学习ansible

0. 每天第一件事,打开ansible官方文档,记得带个呕吐袋,不要吐到其他同学身上
1. 你所需要的命令都有专门的模块
2. 模块使用的语法是官方定义的
3. 尽量少用shell模块.当需要用shell模块的时候,停下来思考一下,是不是有专门的模块可以使用
4. 多看优秀同学的分享

# 第2章 Ansible安装部署

```
1 yum install ansible -y
2 ansible --version
```

## 第3章 Ansible主机清单

### 1.什么是主机清单

```
1 https://docs.ansible.com/ansible/latest/user_guide/intro_inventory.html
```

### 2.主机分组执行

主机清单配置：

```
1 [root@m01 ~]# vim /etc/ansible/hosts
2 [web]
3 172.16.1.7
4 172.16.1.8
5
6 [nfs]
7 172.16.1.31
8
9 [backup]
10 172.16.1.41
```

分组执行测试命令：

```
1 ansible web -m ping
2 ansible nfs -m ping
3 ansible backup -m ping
```

### 3.所有的主机都执行

两种方法：

1. 执行all就代表把所有主机全部执行
2. 主机清单里把所有主机划分到一个组里，注意，一个主机可以属于多个组

主机清单配置：

```
1 [web]
2 172.16.1.7
3 172.16.1.8
4
5 [nfs]
6 172.16.1.31
7
8 [backup]
9 172.16.1.41
10
11 [zabbix]
12 172.16.1.7
13 172.16.1.8
```

```
14 | 172.16.1.31
15 | 172.16.1.41
```

测试命令：

```
1 | ansible all -m ping
2 | ansible zabbix -m ping
```

## 4.SSH使用密码连接并且端口号不是22

主机清单配置：

```
1 | [web]
2 | 172.16.1.7 ansible_ssh_port=9527
3 | 172.16.1.8
```

测试命令：

```
1 | ansible web -m ping
```

## 5.同组主机SSH端口号不一样，账号密码也不一样

方法1: 修改主机清单配置：

前提条件,需要提前把主机信息加入到know\_host文件里

```
1 | [web]
2 | 172.16.1.7 ansible_ssh_port=9527 ansible_ssh_pass='12345678'
3 | 172.16.1.8 ansible_ssh_port=9528 ansible_ssh_pass='123456'
```

方法2: 修改ansible配置文件,打开取消认证的注释

```
1 | host_key_checking = False
```

测试命令：

```
1 | ansible web -m ping
```

## 6.同一组连续的IP

主机清单配置：

```
1 | [web]
2 | 172.16.1.[7:8]
```

测试命令：

```
1 | ansible web -m ping
```

## 7.同一组具有相同的变量

主机清单配置：

```
1 [web]
2 172.16.1.7 ansible_ssh_pass='12345678'
3 172.16.1.8 ansible_ssh_pass='123456'
4
5 [web:vars]
6 ansible_ssh_port=9527
```

测试命令：

```
1 ansible zabbix -m ping
```

## 8.所有主机都生效的变量

主机清单配置：

```
1 [web]
2 172.16.1.7 ansible_ssh_pass='12345678'
3 172.16.1.8 ansible_ssh_pass='123456'
4
5 [nfs]
6 172.16.1.31
7
8 [all:vars]
9 ansible_ssh_port=9527
```

# 第4章 Ansible常用模块

## 0.如何学习ansible模块

```
1 看官网 看官网 看官网
2 https://docs.ansible.com/ansible/latest/collections/ansible/builtin/index.html#plugins-in-ansible-builtin
```

## 1.ping 测试连通性

命令解释：

```
1 ansible 主机组 -m 模块名称 [模块参数]
```

执行命令：

```
1 ansible zabbix -m ping
```

## 2.command 简单命令模块

命令解释：

```
1 ansible 主机组 -m command -a '需要批量执行的命令'
```

执行命令：

```
1 ansible web -m command -a 'ls /tmp'
```

## 3.shell 万能模块

命令解释：

```
1 ansible 主机组 -m shell -a '需要批量执行的命令'
```

执行命令：

```
1 ansible web -m shell -a 'ls /tmp|grep 123'
```

## 4.copy 拷贝文件

命令解释：

```
1 ansible web -m copy -a '参数'
```

简单发送文件：

```
1 ansible web -m copy -a 'src=/opt/m-61.txt dest=/opt/'
```

发送文件的同时指定文件权限和属性：属于www用户,并且权限为600

```
1 ansible web -m copy -a 'src=/opt/m-61.txt dest=/opt/ owner=www group=www mode=600'
```

发送文件的同时备份一份：

```
1 ansible web -m copy -a 'src=/opt/m-61.txt dest=/opt/ owner=www group=www mode=600 backup=yes'
```

写入一行文本到指定文件：

```
1 ansible backup -m copy -a 'content="rsync_backup:oldboy" dest=/etc/rsync.pass mode=600'
```

复制目录：

```
1 ansible backup -m copy -a 'src=/opt dest=/opt/'
```

复制目录下的文件：

```
1 ansible backup -m copy -a 'src=/opt/ dest=/opt/'
```

## 5.file 文件相关

命令解释：

- 1 请-滚去看官网
- 2 [https://docs.ansible.com/ansible/latest/modules/file\\_module.html#file-module](https://docs.ansible.com/ansible/latest/modules/file_module.html#file-module)

创建一个文件:

- 1 `ansible web -m file -a 'path=/opt/linux6.txt state=touch'`

创建一个目录:

- 1 `ansible web -m file -a 'path=/opt/linux6 state=directory'`

删除一个文件或目录

- 1 `ansible web -m file -a 'path=/opt/linux6 state=absent'`
- 2 `ansible web -m file -a 'path=/opt/linux6.txt state=absent'`

创建文件同时制定用户属主权限

- 1 `ansible web -m file -a 'path=/opt/linux6.txt state=touch owner=www group=www mode=600'`
- 2 `ansible web -m file -a 'path=/opt/linux6 state=directory owner=www group=www mode=777'`

对已经存在的文件修改属性

- 1 `ansible web -m file -a 'path=/opt/linux6.txt mode=644'`
- 2 `ansible web -m file -a 'path=/opt/yazhang mode=777 owner=www group=www'`
- 3 `ansible web -m file -a 'path=/opt/linux6.txt mode=644 owner=www group=www'`
- 4
- 5 `## 6.script 执行脚本`
- 6
- 7
- 8 命令解释:
- 9

[https://docs.ansible.com/ansible/latest/modules/shell\\_module.html#shell-module](https://docs.ansible.com/ansible/latest/modules/shell_module.html#shell-module)

- 1 编写脚本文件:
- 2

```
[root@m-61 ~]# cat > echo_hostname.sh << 'EOF'
#!/bin/bash
echo "${hostname}" > /tmp/hostname.txt
EOF
```

- 1
- 2 执行命令:
- 3

```
ansible web -m script -a '/opt/echo_hostname.sh'
```

```
1
2 查看主机生成的文件:
3
```

```
ansible web -m shell -a 'cat /tmp/hostname.txt'
```

```
1
2 查看详细输出过程
3
```

```
ansible web -vvv -m script -a '/opt/echo_hostname.sh'
```

```
1
2 ## 7.cron 定时任务
3
4
5 命令解释:
6
```

[https://docs.ansible.com/ansible/latest/modules/cron\\_module.html#cron-module](https://docs.ansible.com/ansible/latest/modules/cron_module.html#cron-module)

```
1 创建测试脚本:
2
```

```
[root@m01 ~]# cat echo_hostname.sh
#!/bin/bash
echo "$(date +%M:%S) $(hostname)" >> /tmp/hostname.txt
```

```
1
2 传统定时任务命令:
3
```

- ○ ■ ■ ■ /bin/bash /opt/echo\_hostname.sh

```
1
2 默认5颗星创建定时任务:
```

```
1
2 默认5颗星创建定时任务并指定任务名称:
```

```
1
2 修改指定名称的定时任务:
```

```
1
2 注释一条任务:
```

```
1
2 打开注释的任务:
```

```
1
2 删除定时任务：
```

```
1 ## 8.group 组相关
2
3
4 命令解释：
5
```

[https://docs.ansible.com/ansible/latest/modules/group\\_module.html#group-module](https://docs.ansible.com/ansible/latest/modules/group_module.html#group-module)

```
1 创建组的同时指定gid：
2
```

`ansible all -m group -a 'name=www gid=1000'`

```
1
2 删除用户组
3
```

`ansible all -m group -a 'name=www state=absent'`

```
1
2 ## 9.user 用户相关
3
4 命令解释：
5
```

[https://docs.ansible.com/ansible/latest/modules/user\\_module.html#user-module](https://docs.ansible.com/ansible/latest/modules/user_module.html#user-module)

```
1 创建用户的同时指定uid和组id并且不允许登陆不创建家目录：
```

`useradd www -u 1000 -g 1000 -M -s /sbin/nologin`

`ansible all -m user -a 'name=www uid=1000 group=www shell=/sbin/nologin create_home=no'`

```
1
2 删除用户
3
```

`ansible all -m user -a 'name=www state=absent'`

## 10.yum 安装软件

命令解释：

```
1 https://docs.ansible.com/ansible/latest/modules/yum_module.html#yum-module
```

安装一个软件的最新版本：

```
1 ansible nfs -m yum -a 'name=screen state=installed'
```



卸载一个软件:

```
1 | ansible nfs -m yum -a 'name=screen state=removed'
```

## 11.systemd 服务启动

命令解释:

```
1 | https://docs.ansible.com/ansible/latest/collections/ansible/builtin/systemd_module.html#ansible-collections-ansible-builtin-systemd-module
```

启动一个服务:

```
1 | ansible nfs -m systemd -a 'name=nfs state=started'
```

停止一个服务:

```
1 | ansible nfs -m systemd -a 'name=nfs state=stopped'
```

重启一个服务:

```
1 | ansible nfs -m systemd -a 'name=nfs state=restarted'
2 |
3 | 设置一个服务开启自启动:
4 |
```

ansible nfs -m systemd -a 'name=nfs enabled=yes'

```
1 | 设置一个服务不要开机自启动:
```

ansible nfs -m systemd -a 'name=nfs enabled=no'

```
1 |
2 | ## 12.mount 挂载命令
3 |
4 | 命令解释:
5 |
```

[https://docs.ansible.com/ansible/latest/modules/mount\\_module.html#mount-module](https://docs.ansible.com/ansible/latest/modules/mount_module.html#mount-module)

```
1 | 挂载一个目录并且写入开机自启动文件fstab:
2 |
```

```
1 | 只写入fstab但是不挂载:
```

```
1 | 卸载已经挂载的目录并且删除fstab里的条目:
```

```
1 | 卸载已经挂载的目录, 但是不删除fstab里的条目:
```

1 | 挂载状态解释:

mounted     挂载上并且写入fstab  
present     仅写入fstab, 不挂载  
absent      卸载并且移除fstab条目  
unmounted    仅卸载, 不移除fstab条目

```
1  
2  ## 13.unarchive 解压缩  
3  
4  命令解释:  
5
```

[https://docs.ansible.com/ansible/latest/modules/unarchive\\_module.html](https://docs.ansible.com/ansible/latest/modules/unarchive_module.html)

1 | 把自己的压缩包解压到远程服务器的指定目录:

1 | 将远程服务器本身的压缩包解压到远程服务器的指定目录:

```
1  
2  ## 14.archive 压缩  
3  
4  命令解释:  
5
```

[https://docs.ansible.com/ansible/latest/modules/archive\\_module.html](https://docs.ansible.com/ansible/latest/modules/archive_module.html)

1 | 压缩文件到指定目录:

```
1  ## 15.setup 获取主机信息  
2  
3  命令解释:  
4
```

[https://docs.ansible.com/ansible/latest/modules/setup\\_module.html](https://docs.ansible.com/ansible/latest/modules/setup_module.html)

1 | 使用内置变量获取远程主机的IP地址:

ansible web -m setup

```
1  
2  ## 16. 查看帮助  
3  
4  命令解释:  
5
```

ansible-doc

```
1  执行命令:  
2
```

```
1  
2 # 第5章 Ansible颜色输出解释  
3
```

绿色: 代表执行成功,但是状态没有发生任何改变

黄色: 代表执行成功,状态并发生了改变

红色: 有报错,执行失败

紫色: 警告,建议使用专用的模块

蓝色: 详细的执行过程

```
1  
2  
3  
4  
5  
6  
7  
8
```

深圳教習  
老男教習