

Playbook 进阶

- 高级Playbook技巧



高效运维社区
GreatOps Community



ANSIBLE

Playbook高级技巧

- with_items : 任务循环
- when : 条件判断
- register : 变量注册
- template : jinja2模板
- include : 任务引用
- tags : 任务标签



高效运维社区
GreatOps Community

with_items

在一个任务task中，遍历with_items内列表中的元素（由纯量或对象集合组成），循环执行任务，直到遍历结束

纯量

```
1 with_items:
2   - a
3   - b
```

对象集合

```
1 with_items:
2   - { varA: good, varB: boy }
3   - { varA: bad, varB: girl }
```

变量引用：

- 纯量：{{ item }}
- 对象：{{ item.vars }}



高效运维社区
GreatOps

纯量遍历：

```
1 ---
2 - hosts: all
3   tasks:
4     - name: install zip
5       yum:
6         name: zip
7         state: present
8     - name: install make
9       yum:
10        name: make
11        state: present
```



```
1 ---
2 - hosts: all
3   tasks:
4     - name: install tools
5       yum:
6         name: {{ item }}
7         state: present
8       with_items:
9         - zip
10        - make
11
```



对象遍历：

```
1  ---
2  - hosts: all
3    tasks:
4      - name: create user
5        user:
6          name: {{ item.name }}
7          group: {{ item.group }}
8          state: present
9        with_items:
10       - { name: ftp, group: ftp }
11       - { name: samba, group: samba }
```



高效运维社区
High Efficiency Operations Community

register

将任务运行结果注册为变量，多用于将shell执行结果注册

- 命令：`1 register: vars`

- 任务结果是字典，其中包含：

- changed 状态是否改变
- failed 是否失败
- delta 运行耗费时间
- stdout 标准输出
- stderr 标准错误



高效运维社区
GreatOps Community

when

根据变量判断任务是否执行
和register是最佳搭配

- 变量是否等于某个值

```
1 when: "vars == 'str'"
```

- 变量中是否有某个值

```
1 when: "'str' in vars"
```

- 变量是否存在

```
1 when: vars
```



高效运维社区
GreatOps Community

```
1 ---
2 - hosts: all
3   tasks:
4     - name: touch file
5       file:
6         dest: /tmp/123.txt
7         state: touch
8         when: "ansible_hostname == 'apple_home'"
9         -----
10        when: "'apple' in ansible_hostname"
```



template

使用jinja2模板，将模板文件从ansible控制机发送到目标机器，并对变量赋值

模板文件名：xxx.j2

```
vars:  
  apt_php_packages:  
    - php7.0  
    - php7.0-cli  
    .....
```

Variables

+

Templates



Remote Files



GreatPS Community

参数：

- src：本地Jinja2模版的template文件位置
- dest：远程节点上的绝对路径，用于放置template文件
- mode/owner/group/backup

示例：

```
1  - template:
2      src: /mytemplates/foo.conf.j2
3      dest: /etc/foo.conf
4      mode: 644
```



include

调用外部任务文件，便于任务集统一更新，维护

```
1 - hosts: all
2   tasks:
3     - include: /ansible/tasks/change_pwd.yml
```

任务文件只有task section

```
1 - name: aaaa
2   module 1:
3     ..... #module 1 args
4 - name: bbbb
5   module 2:
6     ..... #module 2 args
7 - name: cccc
8   module 3:
9     ..... #module 3 args
```



高效运维社区
GreatOps Community

tags

为任务打标签，执行playbook可以选择或跳过指定tag的任务

```
1  - name: print sth
2    shell: echo 'hello world'
3    tags: aaa
```

- 只执行带aaa标签的任务
ansible-playbook test.yml --tags aaa
- 只执行带aaa,bbb标签的任务
ansible-playbook test.yml --tags aaa,bbb
- 跳过带aaa标签的任务
ansible-playbook test.yml --skip-tags aaa



高效运维社区
GreatOps Community

实战高级 Playbook

- 运用高级功能搭建HTTP&NFS服务器



高效运维社区
GreatOps Community



ANSIBLE

实战高级playbook

搭建HTTP,NFS服务器

- 要求：在目标机器中安装apache&nfs服务，修改主页使其能显示自己的IP，OS版本，主机名以及创建时间，在目标服务器建立两rw和ro属性的文件夹并将其通过nfs共享，启动apache&nfs服务
- 步骤：
 1. 针对不同系统，安装apache
 2. 将当前时间注册成变量
 3. 使用模板文件在目标机器生成index.html
 4. 启动apache服务
 5. Include nfs安装yaml文件
 6. 在nfs yaml文件里针对不同系统安装nfs服务
 7. 新建不同权限的共享文件夹
 8. 添加文件夹到nfs exports
 9. 启动nfs服务



高效运维社区
GreatOps Community

Ansible Roles

- 什么是Roles
- Roles结构
- Ansible-galaxy



高效运维社区
GreatOps Community



ANSIBLE

什么是Roles

- 用以实现某特定功能
- 是变量，模板，文件，任务的集合
- 强依赖文件目录结构

- 默认目录：/etc/ansible/roles

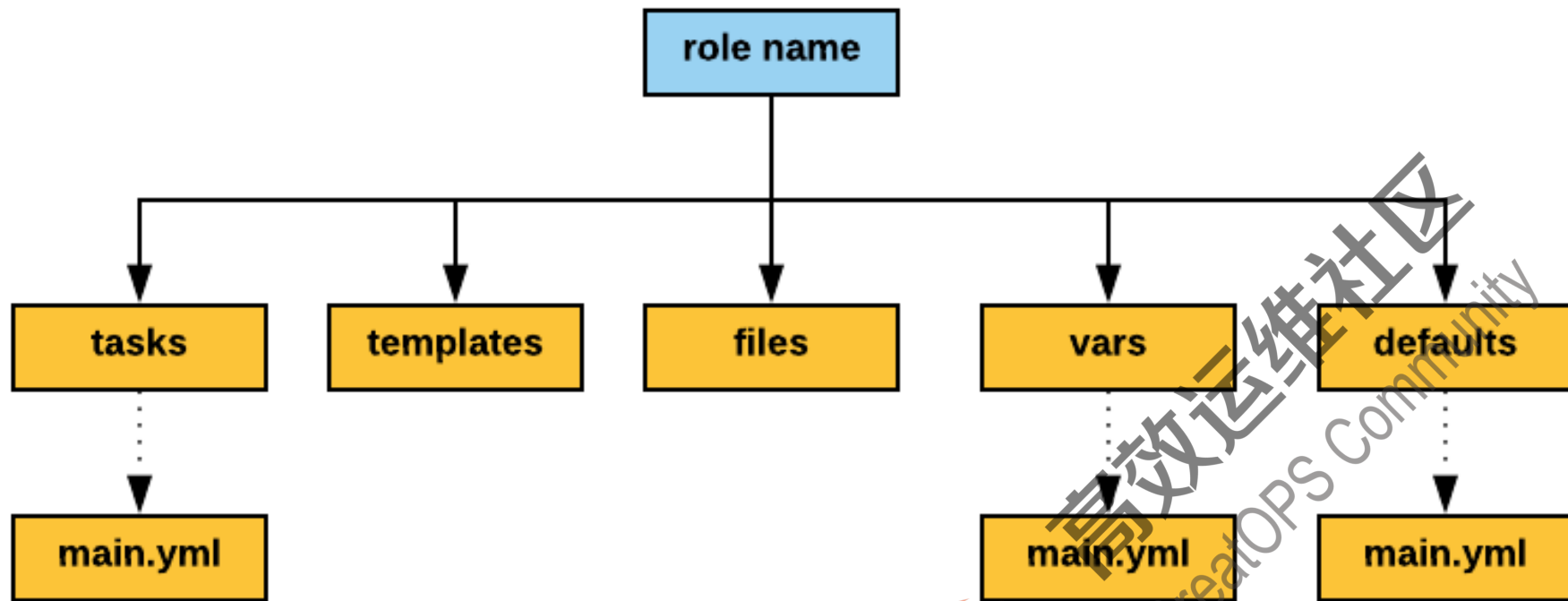
- 使用：

```
1  ---
2  - hosts: all
3    roles:
4      - docker
5      - k8s
```



高效运维社区
GreatOps Community

Roles结构



Ansible-galaxy

从Ansible Galaxy 网站下载role

<https://galaxy.ansible.com/>

下载一个role

`ansible-galaxy install -p ./roles username.rolename`

查看role的具体信息

`ansible-galaxy info username.rolename`



高效运维社区
GreatOps Community