Ansible Playbook简介

ansbile-playbook是一系列ansible命令的集合，利用yaml 语言编写。playbook命令根据自上而下的顺序依次执行。同时，playbook开创了很多特性,它可以允许你传输某个命令的状态到后面的指令,如你可以从一台机器的文件中抓取内容并附为变量,然后在另一台机器中使用,这使得你可以实现一些复杂的部署机制,这是ansible命令无法实现的。

playbook通过ansible-playbook命令使用,它的参数和ansible命令类似,如参数-k(–ask-pass) 和 -K (–ask-sudo) 来询问ssh密码和sudo密码,-u指定用户,这些指令也可以通过规定的单元写在playbook 。 ansible-playbook的简单使用方法: ansible-playbook example-play.yml 。

Playbook基本语法

下面是一个简单的ansible-playbook示例，可以了解其构成:

```
# cat user.yml
- name: create user
  hosts: all
  remote_user: root
  gather_facts: false
  vars:
    user:"test"
  tasks:
    - name: create  user
      user: name="{{ user }}"
```

配置项说明:

- `name`：对该playbook实现的功能做一个概述，后面执行过程中，会打印 name变量的值
- `hosts`：指定对哪些被管理机进行操作;
- `remote_user`：指定在远程被管理机上执行操作时使用什么用户，如不指定，则使用ansible.cfg中配置的remote_user

- `gather_facts`：指定在执行任务之前，是否先执行setup模块获取主机相关信息，如未用到，可不指定
- `vars`：定义后续任务中会使用到的变量，如未用到，可不指定
- `tasks`：定义具体需要执行的任务
  - name：对任务的描述，在执行过程中会打印出来。
  - user：指定调用user模块；
    - name：user模块里的一个参数，用于指定创建的用户名称

同样，如果想实现把这个新增的用户删除，只需将该playbook文件的最后一行替换为如下行再执行相应的playbook即可：

```
user: name="{{ user }}" state=absent remove=yes
```

Playbook简单示例

下面通过playbook管理一个httpd服务器来简单了解下playbook的应用：

1. 创建playbook

```
# cat manage_apache.yml
- name: play to setup web server
  hosts: all
  tasks:
    - name: latest httpd version installed
      yum:
        name: httpd
        state: latest

    - name: correct index.html is present
      copy:
        src: files/index.html
        dest: /var/www/html/index.html

    - name: start httpd service
      service:
        name: httpd
        state: started
        enabled: true
```

2. 执行playbook

```
# ansible-playbook  manage_apache.yml

PLAY [play to setup web server]
*****************************************************************************************
*************************************************

TASK [Gathering Facts]
```

```
*************************************************************************************************
************************************************************
ok: [10.1.61.187]

TASK [latest httpd version installed]
*************************************************************************************************
*******************************************
changed: [10.1.61.187]

TASK [correct index.html is present]
*************************************************************************************************
*********************************************
changed: [10.1.61.187]

TASK [start httpd service]
*************************************************************************************************
*****************************************************
changed: [10.1.61.187]

PLAY RECAP
*************************************************************************************************
*********************************************************************
10.1.61.187                : ok=4    changed=2    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0
```

ansible-playbook常用选项

1. 打印详细信息

- -v：打印任务运行结果
- -vv：打印任务运行结果以及任务的配置信息
- -vvv：包含了远程连接的一些信息
- -vvvv：Adds extra verbosity options to the connection plug-ins,including the users being used in the managed hosts to execute scripts, and what scripts have been executed

```
# ansible-playbook  manage_apache.yml  -vv
```

2. 校验playbook语法

```
# ansible-playbook --syntax-check  manage_apache.yml

playbook: manage_apache.yml
```

3. 测试运行playbook

通过-C选项可以测试playbook的执行情况，但不会真的执行：

```
# ansible-playbook -C  manage_apache.yml

PLAY [play to setup web server]
*************************************************************************************************
*************************************************

TASK [Gathering Facts]
*************************************************************************************************
*************************************************************
ok: [10.1.61.187]

TASK [latest httpd version installed]
*************************************************************************************************
*******************************************
ok: [10.1.61.187]

TASK [correct index.html is present]
*************************************************************************************************
*********************************************
ok: [10.1.61.187]

TASK [start httpd service]
*************************************************************************************************
*********************************************************
ok: [10.1.61.187]

PLAY RECAP
*************************************************************************************************
*************************************************************************
10.1.61.187                : ok=4    changed=0    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0
```

Multiple Plays

```
# This is a simple playbook with two plays

- name: first play
  hosts: web.example.com
  tasks:
    - name: first task
      yum:
        name: httpd
        status: present
    - name: second task
      service:
        name: httpd
```

```
        state: started

- name: second play
  hosts: db.example.com
  tasks:
    - name: first task
      yum:
        name: mariadb-server
        status: present
    - name: second task
      service:
        name: mariadb
        state: started
```