ansible是什么

ansible是一种自动化运维工具，基于python开发

ansible主要模块

Ansible 核心组件

HostInventory 管理主机

Playbooks 定义多个文件在一个yaml格式文件，定义主机需要调用哪些模块来完成功能

CoreModules 核心模块

CustomModules 自定义模块

ConnectionPlugins 连接插件，ansible和host通信使用

工作流程

ansible命令执行-》读取ansible.cfg配置文件-》通过规则过滤inventory中定义主机列表-》加载task对应的模块文件-》通过ansible core将模块或者命令打包成python脚本文件-》将脚本文件传输至目标主机-》执行用户的home目录.ansible/tmp/xxx/xxx.py-》给文件添加执行权限-》执行py文件返回结果=》删除临时py文件

ansible安装方式

一 pip安装

yum install python-pip

pip install ansible

二、yum安装

yum install ansible -y

ansible目录结构

配置文件目录 /etc/ansible/

执行文件目录 /usr/bin/

Lib库以来目录 /usr/lib/pythonx.x/site-packages/ansible/

help文档目录

Man文档目录

配置文件查找顺序

1.检查环境变量export ANSIBLE\_CONFIG=/etc/annsible.cfg

2.~/.ansible.cfg

3./etc/ansible.cfg

常用命令

ansible 临时命令的执行

在各个主机执行命令，并获取结果

ansible all -m command -a 'ss -ntl' -i hosts

测试各个主机网络连通新

ansible all -m ping -i hosts

host为指定主机文件

执行shell命令

ansible all -m shell -a 'cat /etc/password|grep "keer"'

ansible-doc

ansible-galaxy

ansible-playbook 任务编排工具执行

ansible-pull

ansible-vault

ansible-console

playBook

playbook由ymal语言编写，格式如下

1、文件的第一行以---开头

2、同一行使用#表示注释

3、ymal中的列表元素以"-"开然后加一个空格，后面为元素内容

4、同一个列表中的元素保持相同的缩进。

5、playbook中hosts、variables、roles、tasks等对象的表示方法都是以：加一个空格

安装与运行mysql服务ymal文件示例

---

- hosts: node1

remote: root

tasks:

- name: install mysql-server package

yum: name=mysql-server state=present

- name: starting mysqld service

service: name=mysql state=started

核心内容

playbook的核心元素：

Hosts 主机

Tasks 任务

Variables 变量

Templates 模板

Handlers 特定条件触发的任务

variables部分

variables变量有四种定义方法

facts

自定义 变量

通过roles变量

host inventory

角色定制

roles就是通过分别将变量vars,files,tasks,modules,handlers放置单独的目录中，并可以便捷地include它们的一种机制。

角色集合定义在roles/目录下

files/:存储由copy或script等模块调用的文件

tasks/:此目录中至少有一个main.yml文件定义task

handlers/：此目录中至少有一个main.yml文件定义handler

vars/:此目录中至少有一个main.yml文件定义variable

templates/:存储模板

meta/:此目录中至少有一个main.yml文件定义当前橘色的特殊设定依赖关系

default/:此目录中至少有一个main.yml文件定义用于设定默认变量

<https://www.cnblogs.com/keerya/p/8004566.html>

hosts文件

vim hosts

hostname1 ansible\_ssh\_user=root aisible\_ssh\_pass=xxx zone=zone1 tags=role:management

hostname2 ansible\_ssh\_user=root aisible\_ssh\_pass=xxx zone=zone2 tags=role:management

hostname3 ansible\_ssh\_user=root aisible\_ssh\_pass=xxx zone=zone3 tags=role:management

[ece\_first\_node]

hostname1

mkdir roles

cd roles

midir -pv ./{nginx}/{files,templates,vars,tasks,handlers,meta,default}

tree

vim ./nginx/tasks/main.yml

---

- name: install {{ rpmname }}

yum: name={{ rpmname }} state=present

- name: copy {{ rpmname }}.conf

copy: src=/tmp/{{ rpmname }}.conf dest=/etc/{{ rpmname }}/{{ rpmname }}.conf backup=yes

notify: reload

tags: reload{{ rpmname }}

- name: start {{ rpmname }} service

service: name{{ rpmname }} state=started

tags: start{{ rpmname }}

vim ./nginx/handlers/main.yml

---

- name: reload

service: name={{ rpmname }} state=restarted

vim ./nginx/vars/main.yml

---

rpmname: nginx

当前目录

vim nginx-roles.yml

---

- hosts: all

remote\_user: root

roles:

- nginx