

Python-ansible

NSD python

docker



docker

- Docker 是一个开源项目，诞生于 2013 年初，最初是 dotCloud 公司内部的一个业余项目。它基于 Google 公司推出的 Go 语言实现。项目后来加入了 Linux 基金会，遵从了 Apache 2.0 协议，项目代码在 GitHub 上进行维护。
- Docker 自开源后受到广泛的关注和讨论，以至于 dotCloud 公司后来都改名为 Docker Inc。Redhat 已经在其 RHEL6.5 中集中支持 Docker；Google 也在其 PaaS 产品中广泛应用。
- Docker 项目的目标是实现轻量级的操作系统虚拟化解决方案。Docker 的基础是 Linux 容器（LXC）等技术。
- 在 LXC 的基础上 Docker 进行了进一步的封装，让用户不需要去关心容器的管理，使得操作更为简便。用户操作 Docker 的容器就像操作一个快速轻量级的虚拟机一样简单。



Docker安装 –centos

- `yum install -y yum-utils device-mapper-persistent-data lvm2`
- `yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo`
- `yum install docker-ce`



Docker 名词解释

- 镜像 images
- 容器 container
- 仓库 registry



导入镜像

- `docker import`
- `docker images`
- `docker load < xxx.tar.gz`
- `docker tag idxxx namexxxx:tag`



启动容器

```
docker run -it -d -h hostname - -name containername image:tag  
docker ps  
docker exec -it containerid bash
```



常用命令

- docker [run|start|stop|restart|kill|rm|pause|unpause] # 容器生命周期管理
- docker [ps|inspect|top|attach|events|logs|wait|export|port] # 容器操作运维
- docker [login|pull|push|search] # 镜像仓库
- docker [images|rm|tag|build|history|save|import] # 本地镜像管理
- docker [info|version] # 查看docker信息



练习

- 导入课程使用的镜像
- 使用ansible的镜像启动一个容器
- 使用centos6的镜像启动两个容器



ansible



什么是ansible

Ansible是一个部署一群远程主机的工具。远程的主机可以是远程虚拟机或物理机，也可以是本地主机。

同类产品：

saltstack

chef

Puppet



Ansible架构

- Ansilbe管理员节点和远程主机节点通过ssh协议进行通信。所以Ansible配置的时候只需要保证从Ansible管理节点通过SSH能够连接到被管理的远程的远程节点即可。。但是SSH必须配置为公钥认证登录方式，而非密码认证
- 只要机器上安装了 Python 2.6 或 Python 2.7 (windows系统不可以做控制主机),都可以运行Ansible.



Ansible安装

redhat/CentOS Linux上, Ansible目前放在的epel源中

```
yum install epel-release
```

```
yum install ansible -y
```

仅在管理端安装



Host Inventory(主机清单)

/etc/ansible/hosts

简单配置

192.168.1.50
aserver.example.org
bserver.example.org

带分类配置

[webservers]
foo.example.com ansible_ssh_port=22 ansible_ssh_user=root ansible_ssh_pass=pass
bar.example.com



命令行管理主机

检查所有的远程主机，是否以user1用户创建了ansible主机可以访问的环境。

```
$ansible all -m ping -u user1
```



执行命令

在所有的远程主机上，以当前bash的同名用户，在远程主机执行“echo bash”

```
$ansible all -a "/bin/echo hello"
```



拷贝文件

拷贝文件/etc/host到远程主机（组）web，位置为/tmp/hosts
\$ **ansible web -m copy -a "src=/etc/hosts dest=/tmp/hosts"**



安装软件包

远程主机（组）web安装yum包acme

```
$ ansible web -m yum -a "name=acme state=present"
```



添加用户

添加用户

```
$ ansible all -m user -a "name=foo password=<crypted password here>"
```



下载git

```
ansible web -m git -a "repo=git://foo.example.org/repo.git dest=/srv/  
myapp version=HEAD"
```



启动服务

```
ansible web -m service -a "name=httpd state=started"
```



查看远程主机的所有信息

```
ansible all -m setup
```



ansible模块查看

ansible-doc -l:获取列表

ansible-doc -s module_name



练习

- 给远程主机传送文件，并设置权限为644
- 查看远程主机信息



ansible-playbook



什么是playbooks

playbooks是ansible更强大的配置管理组件，实现基于文本文件编排执行的多个任务，且多次重复执行。

(1)核心元素

Tasks任务、Variables变量、Templates模板、Handlers处理器、Roles角色

(2)playbooks中定义任务：

- name: task description 注释描述信息
module_name: module_args 声明模块：定义ansible模块参数

(3)ansible-playbook执行命令：

ansible-playbook <filename.yml> ... [options]



Playbooks样例

```
---
- hosts: web
  vars:
    http_port: 80
    max_clients: 200
    remote_user: root
  tasks:
    - name: ensure apache is at the latest version
      yum: pkg=httpd state=latest
      when: ansible_os_family == "RedHat"
    - name: Write the configuration file
      template: src=templates/httpd.conf.j2 dest=/etc/httpd/conf/httpd.conf
      notify:
        - restart apache

    - name: Write the default index.html file
      template: src=templates/index.html.j2 dest=/var/www/html/index.html

    - name: ensure apache is running
      service: name=httpd state=started
  handlers:
    - name: restart apache
      service: name=httpd state=restarted
```



Httpd.conf 模板 (templates目录下)

```
port: {{ http_port }}
```



index.html 模板 (templates目录下)

```
<html>
<title>Demo</title>
<body>
<div class="block" style="height: 99%;">
  <div class="centered">
    <h1>#46 Demo {{ max_clients }}</h1>
    <p>Served by {{ ansible_hostname }} ({{ ansible_default_ipv4.address }}).</p>
  </div>
</div>
</body>
</html>
```



练习

- 使用palybooks配置软件



Ansible api
