# 安装步骤

1. yum install epel-release,然后 yum install ansible

# 概述

ansible不需要安装客户端,只需要装一台服务端就可以了,想要和客户端通信,则只需要配置从服务端免密钥到客户端

配置面密钥过程:

ssh-copy-id -i id\_rsa.pub [root@192.168.200.146](mailto:root@192.168.200.146)

# 使用

服务端:192.168.200.239

客户端1:192.168.200.146

客户端2:192.168.200.149

1. 配置服务端免密钥到两个客户端
2. vim /etc/ansible/hosts

添加:

[web]

192.168.200.146

192.168.200.149

3. ansible all -m ping

all:可以替换为web 或者ip或者主机名

安装了ansible后就有了ansible-playbook,一般用ansible-playbook比ansible功能强大

# 4.最佳规范

在/etc/ansible下的roles

|  |
| --- |
| roles/  └── nginx  ├── defaults  │   └── main.yml  ├── files  │   ├── install\_nginx.sh  │   ├── nginx  │   └── nginx-1.8.1-1.el6.x86\_64.rpm  ├── handlers  │   └── main.yml  ├── meta  ├── tasks  │   ├── install.yml  │   └── main.yml  ├── templates  │   └── nginx.conf.j2  └── vars |

defaults/main.yml:

|  |
| --- |
| user: nginx  group: nginx  rpmname: nginx-1.8.1-1.el6.x86\_64.rpm  nginx\_port: 8888  nginx\_conf: nginx.conf.j2 |

files下的的install\_nginx.sh:

|  |
| --- |
| #!/bin/bash  which nginx  result=$?  if [ $result != 0 ];then  rpm -i ./nginx-1.8.1-1.el6.x86\_64.rpm  fi |

files下nginx(将此文件放在/etc/init.d目录中,nginx可由service管理,不然无法在ansible中使用service模块)

|  |
| --- |
| #!/bin/sh  #  # nginx - this script starts and stops the nginx daemon  #  # chkconfig: - 85 15  # description: NGINX is an HTTP(S) server, HTTP(S) reverse \  # proxy and IMAP/POP3 proxy server  # processname: nginx  # config: /etc/nginx/nginx.conf  # config: /etc/sysconfig/nginx  # pidfile: /var/run/nginx.pid  # Source function library.  . /etc/rc.d/init.d/functions  # Source networking configuration.  . /etc/sysconfig/network  # Check that networking is up.  [ "$NETWORKING" = "no" ] && exit 0  nginx="/usr/sbin/nginx"  prog=$(basename $nginx)  NGINX\_CONF\_FILE="/etc/nginx/nginx.conf"  [ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx  lockfile=/var/lock/subsys/nginx  make\_dirs() {  # make required directories  user=`$nginx -V 2>&1 | grep "configure arguments:" | sed 's/[^\*]\*--user=\([^ ]\*\).\*/\1/g' -`  if [ -z "`grep $user /etc/passwd`" ]; then  useradd -M -s /bin/nologin $user  fi  options=`$nginx -V 2>&1 | grep 'configure arguments:'`  for opt in $options; do  if [ `echo $opt | grep '.\*-temp-path'` ]; then  value=`echo $opt | cut -d "=" -f 2`  if [ ! -d "$value" ]; then  # echo "creating" $value  mkdir -p $value && chown -R $user $value  fi  fi  done  }  start() {  [ -x $nginx ] || exit 5  [ -f $NGINX\_CONF\_FILE ] || exit 6  make\_dirs  echo -n $"Starting $prog: "  daemon $nginx -c $NGINX\_CONF\_FILE  retval=$?  echo  [ $retval -eq 0 ] && touch $lockfile  return $retval  }  stop() {  echo -n $"Stopping $prog: "  killproc $prog -QUIT  retval=$?  echo  [ $retval -eq 0 ] && rm -f $lockfile  return $retval  }  restart() {  configtest || return $?  stop  sleep 1  start  }  reload() {  configtest || return $?  echo -n $"Reloading $prog: "  killproc $nginx -HUP  RETVAL=$?  echo  }  force\_reload() {  restart  }  configtest() {  $nginx -t -c $NGINX\_CONF\_FILE  }  rh\_status() {  status $prog  }  rh\_status\_q() {  rh\_status >/dev/null 2>&1  }  case "$1" in  start)  rh\_status\_q && exit 0  $1  ;;  stop)  rh\_status\_q || exit 0  $1  ;;  restart|configtest)  $1  ;;  reload)  rh\_status\_q || exit 7  $1  ;;  force-reload)  force\_reload  ;;  status)  rh\_status  ;;  condrestart|try-restart)  rh\_status\_q || exit 0  ;;  \*)  echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-reload|configtest}"  exit 2  esac |

tasks/install.yml:

|  |
| --- |
| ---  - name: groupadd nginx #创建nginx组,存在则忽略  group:  name: "{{group}}"  - name: useadd nginx  user:  name: "{{user}}"  group: "{{group}}"  createhome: no  shell: /sbin/nologin  - name: add epel #添加epel源  yum:  name: epel-release  state: installed  - name: yum install dependecy #安装依赖  package:  name: "{{ item }}" #注意要加双引号  state: latest  with\_items:  - pcre-devel  - openssl-devel  - geoip-devel    - name: install nginx #开始安装nginx  script: install\_nginx.sh    - name: write the nginx config #配置文件  template:  src: "{{nginx\_conf}}"  dest: "/etc/nginx/nginx.conf"  backup: yes  notify:  - reload nginx  - name: use service manage nginx #使用service来管理nginx  copy:  src: nginx  dest: /etc/init.d/nginx  mode: u=rwx,g=rx,o=rx  - name: use chkconfig manage nginx #有chkconfig来管理nginx  command: chkconfig --add /etc/init.d/nginx  - name: start nginx  service: name=nginx state=started enabled=yes |

tasks/main.yml:

|  |
| --- |
| ---  - import\_tasks: install.yml |

handlers/main.yml:

|  |
| --- |
| ---  - name: restart nginx  service: name=nginx state=restarted  - name: validate nginx configuration  command: nginx -t -c /etc/nginx/nginx.conf  changed\_when: false    - name: reload nginx  service: name=nginx state=reloaded |

templates/ nginx.conf.j2(在这个文件中要使用ansible变量,直接{{变量名称}}就可以了):

|  |
| --- |
| ....  server{  ....  listen {{nginx\_port}};  .....  }  ..... |

最后在/etc/ansible下新建一个nginx.yml:

|  |
| --- |
| ---  - name: deploy nginx  hosts: all  remote\_user: root  roles:  - {role: nginx} |

执行ansible-playbook nginx.yml 就ok了

# 5.学习网站

<http://www.ansible.com.cn/>

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

<<奔跑吧ansible>>

# 6.命令

#查看模块的使用信息

ansible-doc 模块