Ansible初始化文档

准备工作

1. 服务器最小化安装Centos7系统[内存1G]，root初始密码[123/123456]
2. 虚拟机使用桥接模式，如果网络不通，配置虚拟机网络编辑器中的桥接设备 [wifi]
3. 按照网络拓扑，配置每台服务器的主机名，ip地址，网关信息[192.168.1.1]
4. 服务器追加hosts解析

```
192.168.1.110      ansible

192.168.1.99       lb-vip
192.168.1.11       lb-nginx1
192.168.1.12       lb-nginx2

192.168.1.21       web-nginx1
192.168.1.22       web-nginx2

192.168.1.31       tomcat1
192.168.1.32       tomcat2

192.168.1.41       redis-cluster1
192.168.1.42       redis-cluster2
192.168.1.43       redis-cluster3

192.168.1.150      mycat-vip
192.168.1.51       mycat1
192.168.1.52       mycat2

192.168.1.61       mha-mysql-manager
192.168.1.62       mha-mysql-master
192.168.1.63       mha-mysql-master-bak
```

```
192.168.1.64        mha-mysql-slave

192.168.1.200       dfs-client-vip
192.168.1.71        dfs-client1
192.168.1.72        dfs-client2
192.168.1.73        dfs-tracker1
192.168.1.74        dfs-tracker2
192.168.1.75        dfs-storage1
192.168.1.76        dfs-storage2


192.168.1.81        db-back
```

安装Ansible

安装ansible

```
# 1、安装yum epel源
curl -o /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.aliyun.com/repo/Centos-7.repo
curl -o /etc/yum.repos.d/epel.repo
http://mirrors.aliyun.com/repo/epel-7.repo

# 2. 安装ansible
yum install -y ansible
# ansible --version        查看版本信息
# ansible-doc -s command    查看 command 模块的文档帮助
```

修改主配置文件

```
# 3. 修改配置文件
vim /etc/ansible/ansible.cfg

# 开启日志，去除前面井号
log_path = /var/log/ansible.log
# ssh第一次登录检测去除
host_key_checking = False
# 在剧本中不使用ansible模块时候，不输出警告信息
command_warnings=False
```

修改hosts文件

```
# 4. 修改Ansible hosts文件
vim /etc/ansible/hosts

[ansible]
192.168.1.110

[lb]
192.168.1.11
192.168.1.12

[web]
192.168.1.21
192.168.1.22

[tomcat]
192.168.1.31
192.168.1.32

[redis]
192.168.1.41
192.168.1.42
192.168.1.43

[mycat]
192.168.1.51
192.168.1.52

[mha]
192.168.1.61
192.168.1.62
192.168.1.63
192.168.1.64

[dfs]
192.168.1.71
192.168.1.72
192.168.1.73
192.168.1.74
192.168.1.75
192.168.1.76
```

```
[db-bak]
192.168.1.81
```

ssh免密

编辑 host_ip.txt

```
# 编辑 host_ip.txt

192.168.1.110    root    123


192.168.1.11     root    123
192.168.1.12     root    123

192.168.1.21     root    123
192.168.1.22     root    123

192.168.1.31     root    123
192.168.1.32     root    123

192.168.1.41     root    123456
192.168.1.42     root    123456
192.168.1.43     root    123456


192.168.1.51     root    123456
192.168.1.52     root    123456


192.168.1.61     root    123456
192.168.1.62     root    123456
192.168.1.63     root    123456
192.168.1.64     root    123456
```

编辑免密脚本

```
# 1. 安装expect
```

```bash
yum install -y expect

# 2．编辑一键免密脚本
vim yy_ssh.sh

#!/bin/bash
#yum -y install expect
#hostip=$1
#name=$2
#PASSWORD=$3
key_generate() {
    expect -c "set timeout -1;
        spawn ssh-keygen -t rsa;
        expect {
            {Enter file in which to save the key*} {send --
\r;exp_continue}
            {Enter passphrase*} {send -- \r;exp_continue}
            {Enter same passphrase again:} {send --
\r;exp_continue}
            {Overwrite (y/n)*} {send -- n\r;exp_continue}
            eof             {exit 0;}
    };"
}
auto_ssh_copy_id () {
    expect -c "set timeout -1;
        spawn ssh-copy-id -i $HOME/.ssh/id_rsa.pub $2@$1;
            expect {
                {Are you sure you want to continue connecting *}
{send -- yes\r;exp_continue;}
                {*password:} {send -- $3\r;exp_continue;}
                eof {exit 0;}
            };"
}

check_cmd () {
    if [[ $? == 0 ]];then
        echo "$1 SUCCESS"
    else
        echo "$1 ERROR"
        exit
    fi
}
```

```
if [[ ! -e $HOME/.ssh/id_rsa.pub ]]; then
  key_generate && check_cmd "ssh key creat"
fi
cat host_ip.txt | while read myline
do
    hostip=`echo $myline | cut -d " " -f1`
    name=`echo $myline | cut -d " " -f2`
    PASSWORD=`echo $myline | cut -d " " -f3`
    auto_ssh_copy_id $hostip  $name $PASSWORD
done
check_cmd "ssh scripts run cmd"
```

执行脚本并测试

```
# 执行免密脚本
sh yj_ssh.sh
```

```
# 测试连接
ansible lb -m ping
```

```
# 可以在此时修改主机名
for i in lb-nginx1 lb-nginx2 web-nginx1 web-nginx2 tomcat1
tomcat2 redis-cluster1 redis-cluster2 redis-cluster3 mha-mysql-
manager mha-mysql-master mha-mysql-master-bak mha-mysql-slave; do
ssh $i hostnamectl set-hostname $i;done
```

剧本初始化

```
# vim server-init.yaml

---
- hosts: all
  remote_user: root
  # 不收集变量信息，加快速度
  gather_facts: no
```

```yaml
  tasks:
    - name: "1．关闭Selinux"
      shell: sed -i "s/SELINUX=enforcing/SELINUX=disabled/g"
/etc/selinux/config
    - name: SELINUX=disabled
      selinux: state=disabled

    - name: "2．关闭防火墙"
      service: name=firewalld state=stopped enabled=no

    - name: "3．设置时区"
      file: src=/usr/share/zoneinfo/Asia/Shanghai
dest=/etc/localtime state=link owner=root group=root force=yes

    - name: "4．同步时间"
      shell: (echo "* 1 * * * ntpdate time.windows.com >/dev/null
2>&1";crontab -l) |crontab

    - name: "5．历史命令显示操作时间"
      shell: echo 'export HISTTIMEFORMAT="%F %T `whoami` "' >>
/etc/bashrc

    - name: "6．设置SSH超时时间"
      shell: echo "export TMOUT=600" >> /etc/profile

    - name: "7．配置阿里yum源"
      copy: src=/etc/yum.repos.d/CentOS-Base.repo
dest=/etc/yum.repos.d/
    - name: "配置阿里拓展源"
      copy: src=/etc/yum.repos.d/epel.repo dest=/etc/yum.repos.d/
    - name: "建立yum缓存"
      shell: yum makecache fast

    - name: "8．安装常用工具"
      shell: echo install...
    - name: "install vim"
      yum: name=vim
    - name: "install lrzsz"
      yum: name=lrzsz
    - name: "install mlocate"
      yum: name=mlocate
    - name: "install net-tools"
      yum: name=net-tools
```

```
      - name: "install wget"
        yum: name=wget
      - name: "install bash-completion"
        yum: name=bash-completion
```

```
# 执行剧本
ansible-playbook server_init.yaml

# 拷贝hosts文件
ansible all -m copy -a 'src=/etc/hosts dest=/etc'

# 更新系统
- name: upgrade all packages
  yum: name=* state=latest
```