

ansible安装部署和配置、常用模块整理

原创

GeorgeKai

2018-02-01 14:10:29

评论(0)

491人阅读

今天心情不错~~~~第25个生日了，又遇昨晚百年难得一见的蓝月亮，所以昨晚连夜整理了文档，
会分为两部分发出去，**ansible**批量化部署在工作中是非常实用，建议呢 整理大量常用模块去练习

1.1.1 ansible软件介绍

1. ansible是一个基于python开发的自动化运维工具！（saltstack）
2. 其功能的实现是基础SSH远程连接服务的
3. ansible可以实现批量系统配置、批量软件部署、批量文件拷贝、批量运行命令等功能

1.1.2 ansible软件相关参考链接信息

http://docs.ansible.com/ansible/intro_installation.html
<http://www.ansible.com.cn/>
http://docs.ansible.com/modules_by_category.html
<http://www.ansible.cn/docs/>

1.1.3 ansible软件特点

1. 不需要单独安装客户端，基于ssh服务
2. 不需要安装服务端
3. 需要依靠大量的模块实现批量部署
4. 配置文件/etc/ansible/ansible.cfg

1.1.4 ansible软件安装部署和配置

PS: 为了安全，尽量不要配置外网

首先保证：可以免秘钥登录被管理端

1. 管理端安装软件：

```
yum install ansible -y
```

2. 被管理端安装软件：

```
yum install libselinux-python -y
```

PS: 如果关闭selinux，那么被管理端可以不安装（建议安装）

3. ansible软件管理配置文件

```
vim /etc/ansible/hosts  #定义ansible可管理的主机
[george]
172.16.1.7
172.16.1.31
172.16.1.41
```

4. 进行ansible软件批量管理：

```
ansible george -m command -a "uptime"
```

PS: 查看的是george主机组，模块为command，的主机负载信息

george #主机组

-m #指定模块参数（command为默认模块，不写也可以）

command #模块名称

-a #指定利用模块执行的动作参数，-a后面的是要执行的命令

uptime #批量执行的命令



1.1.5 ansible软件命令常用参数

- k: 以交互方式输入密码, 进行远程管理
- a: 指定应用模块的相应参数信息
- m: 指定应该什么模块 (默认为command模块)

1.1.6 ansible模块总结

参考ansible官方: http://docs.ansible.com/ansible/latest/modules_by_category.html
 进入all modules中, 使用google浏览器, ctrl+f根据查找模块名查找对应信息

ansible执行命令后输出信息中:

- 绿色——表示查询, 或者没有发生任何改变
- 红色——表示命令操作出现异常
- 屎×××——对远程主机做了相应改动
- 粉色——对操作提出建议或忠告

ansible系统命令帮助文档查看方法:

- ansible-doc -l —— 列出所有可用的模块信息
- ansible-doc -s cron —— 查看指定模块的参数信息

1. command模块实践: (默认模块)

```
ansible george -m command -a "ifconfig"
#批量显示远程主机的网卡信息
ansible george -m commadn -a "chair=/tmp touch kai.txt"
#批量切换到远程主机的/tmp目录下, 创建kai.txt这个文件
ansible george -m command -a "creates=/tmp/kai touch /tmp/kai"
#批量判断远程主机/tmp下有没有kai这个文件, 如果有就skip, 没有就执行后面的命令
ansible george -m command -a "removes=/tmp/kai.txt touch /tmp/123.txt"
#批量判断远程主机/tmp下有没有kai这个文件, 如果有就执行后面的命令, 没有就skip
ansible george -m command -a "ls -l"
#free_from (默认参数) 可以输入任何系统命令信息, 但是不包含一些特殊环境变量和特殊符号信息, 如: <, |, ;
```

2. ping模块实践:

```
[root@m01 scripts]# ansible george -m ping
172.16.1.31 | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
172.16.1.7 | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
```

#返回pong, 说明可以登录SSH连接, 这里ping但不是测试网络连通性的, 用于验证能否登录SSH连接, 在查看是否满足python的支持, 属于system模块

3. debug调试模块实践

```
[root@m01 scripts]# ansible george -m debug
172.16.1.31 | SUCCESS => {
  "changed": false,
  "msg": "Hello world!"
}
172.16.1.7 | SUCCESS => {
  "changed": false,
  "msg": "Hello world!"
}
```

#此模块在执行过程中打印语句，对于调试变量或表达式非常有用，而不一定会停止播放。与“When:”指令一起调试非常有用。

4. copy模块实践

1) copy参数src

例: `ansible george -m copy -a "src=/etc/hosts dest=/tmp/"`

#将本机/etc/hosts文件复制到远程主机的/tmp目录下（如果/tmp下已有同名hosts，那么会被覆盖）

例: `ansible george -m copy -a "src=/etc/hosts dest=/tmp/dir/"`

#如果远程主机没有dir目录，那么会创建dir目录

例: `ansible george -m copy -a "src=/etc/hosts dest=/tmp/1/2/3/4/"`

#但传输文件时，如果上级目录不存在，则不会创建，传输就无法成功

例: `ansible george -m copy -a "src=/tmp dest=/tmp/1/2/3/4"`

#传输目录时，如果远程主机目录不存在，传输时可以创建多层目录，

#如果传输的是目录本身及下面内容，后面不要加/

例: `ansible george -m copy -a "src=/tmp dest=/tmp/1/2/3/4"`

#如果传输的是目录下面的内容，后面必须加/

例: `ansible george -m copy -a "remote_src=true src=/etc/hosts dest=/tmp/1/2/"`

#批量操作远程主机，对他们本机上的文件进行本地操作

#设置为true时（默认为false），不支持递归复制

2) copy参数backup=yes

`ansible george -m copy -a "src=/etc/hosts dest=/etc/ backup=yes"`

#分发文件时，如果与远程主机下hosts文件内容不一致，那么会备份源文件为“hosts.5714...”以时间戳命名，在修改hosts源文件的内容

#不输入默认backup=no，就是不备份，会覆盖源文件

3) copy参数mode、owner、group

`ansible george -m copy -a "src=/etc/hosts dest=/tmp/ mode=0600 owner=george group=george"`

#改变文件的权限为0600，所有者和属组为george

4) copy参数force

`ansible george -m copy -a "src=/etc/hosts dest=/tmp/ force=yes"`

#默认为force=yes，如果和远程主机信息不一致，会覆盖

#如果force=no，那么远程主机同名文件不会做改变

5) copy参数content

`ansible george -m copy -a "content=123123 dest=/tmp/hosts"`

#写入信息到/tmp/hosts中会把源内容覆盖掉，谨慎操作，只能添加少量的信息

#添加多量的，可以用template模块

5. shell模块实践

`ansible george -m shell -a "hostname;hostname -i"`

#支持特殊符号，-a里面可接多个名，用分号分割

测试:用shell执行一个脚本很麻烦的，用script执行

1) 推送脚本过去，并授权

`ansible george -m copy -a "src=/tmp/test.sh dest=/tmp mode=+x"`

2) 运行脚本

```
ansible george -m shell -a "/tmp/test.sh"
```

6. script模块实践

```
ansible george -m script -a "/server/scripts/keepalived.sh"
#将本地脚本中的信息，在远程主机上执行
```

7. setup模块实践

```
ansible george -m setup
#显示远程主机的所有信息（后面加-v显示详细信息）
#提取IP、或架构信息等，X86来判断主机架构，安装合适软件
```

```
ansible george -m setup -v
#主要用于解决一些错误：如远程主机hang住了，ansible会输出少量信息（最多-vvvv）
```

8. yum模块实践

```
ansible george -m yum -a "name=iostat state=installed"
#批量使用yum安装软件iostat
```

9. service模块实践

```
ansible george -m service -a "name=cron state=stopped enable=no" #stopped是2个p
#临时停止cron服务，取消开机自启动（相反：state=started enable=yes）
```

10. file模块实践 #修改文件或目录属性信息，用于创建文件或目录，也可以用mode、owner、group定义文件或目录的权限信息

1) path参数

```
ansible george -m file "path=/tmp/ state=directory mode=0644"
#指定路径，是dest, name的别名，作用一样
```

2) state参数

```
ansible george -m file -a "dest=/opt/dir_01/ state=directory"
```

#创建目录为dir_01

```
ansible george -m file -a "dest=/opt/file_01 state=touch"
```

#创建文件为file_01

```
ansible george -m file -a "dest=/opt/file_01 state=absent"
```

#删除文件file_01

```
ansible george -m file -a "src=/opt/hosts dest=/opt/hosts_link state=link"
```

#创建符号链接，基于本机有源文件

11. cron模块实践 #定时任务模块

```

* * * * * /bin/sh /server/scripts/test.sh &>/dev/null
minute      # Minute when the job should run ( 0-59, *, */2, etc )
hour        # Hour when the job should run ( 0-23, *, */2, etc )
day         # Day of the month the job should run ( 1-31, *, */2, etc )
month       # Month of the year the job should run ( 1-12, *, */2, etc )
weekday     # Day of the week that the job should run ( 0-6 for Sunday-Saturday, *, etc )
job         # 定义定时任务与要做什么事
name        # 给定时任务加一个备注，避免创建出多个重复的定时任务(根据定时任务备份判断 是否生成一个新的定时任务)
stat        #若设置为present，表示创建定时任务，若设置为absent，表示删除指定定时任务
            #只能管理自己创建的定时任务，本来有的管理不了
disabled    #disable=yes注释掉定时任务（不生效），disable=no解除注释定时任务（生效）
```

```
ansible george -m cron -a "name='backup service' minute=*/5 job='/usr/sbin/ntpdate time.nist.gov >/dev/null 2>&1'"
```

#创建定时任务

```
ansible george -m cron -a "name='backup service' state=absent"
```

#删除定时任务

```
ansible george -m cron -a "name=dancy01 minute=*/5 job='/usr/sbin/ntpdate time.nist.gov >/dev/null 2>&1' disabled=yes"
```

#注释掉定时任务，反之取消注释

12.mount模块实践

```
ansible george -m mount -a "state=mounted src=172.16.1.31:/data path=/data fstype=nfs"
```

src #要被挂载的目录或文件

path #指定挂载点的路径

fstype #指定挂载时的文件系统类型

opts #在挂载时，指定挂载参数信息

state #state=mounted，在fstab文件中的备份将被激活挂载或适当配置，如果指定mounted的挂载挂载点不存在，会创建

#state=unmounted，设备将被卸载不会改变fstab文件中的信息

#state=absent和state=present，只处理fstab，但不影响目录的挂载

小伙伴们可以关注我的微信公众号：linux运维菜鸟之旅



关注“中国电信天津网厅”公众号，首次绑定可免费领2G流量，为你的学习提供流量！



版权声明：原创作品，如需转载，请注明出处。否则将追究法律责任
