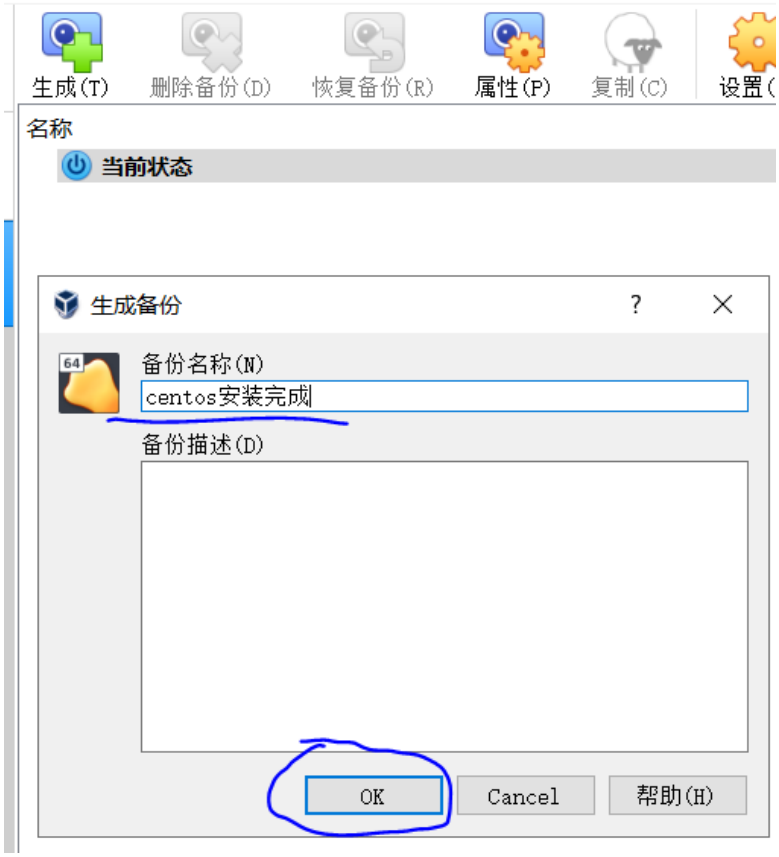


上图中可见，备份完成，后面 openGauss 的安装中如有问题，可以退回到 centos 刚安装成功的状态。

我们还可以回到明细页面，如下图所示点击即可。

附录 5：centOS7.6 上安装 openGauss2.1.0

从下面的路径中能下载 openGauss 最新版本以及相关工具

https://opengauss.org/zh/download.html

我们使用 openGauss_2.1.0 企业版进行安装

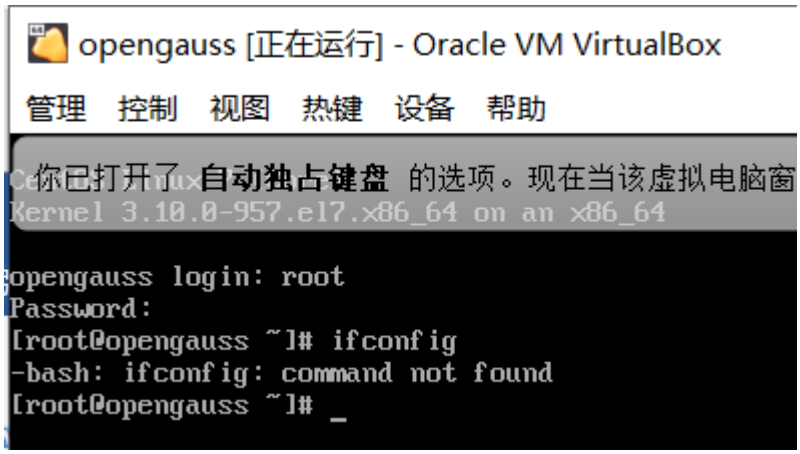
openGauss-2.1.0-CentOS-64bit-all.tar.gz。

使用到的工具有 putty（可以直接鼠标左键滑动选择复制，鼠标右键点击粘贴），WinSCP（用于上传安装包到服务器）

注意：安装过程中记得要做快照，成功完成几步后，记得做好备份快照。

启动虚拟机

登录（用户名 root，密码 openGauss），之后输入 ifconfig

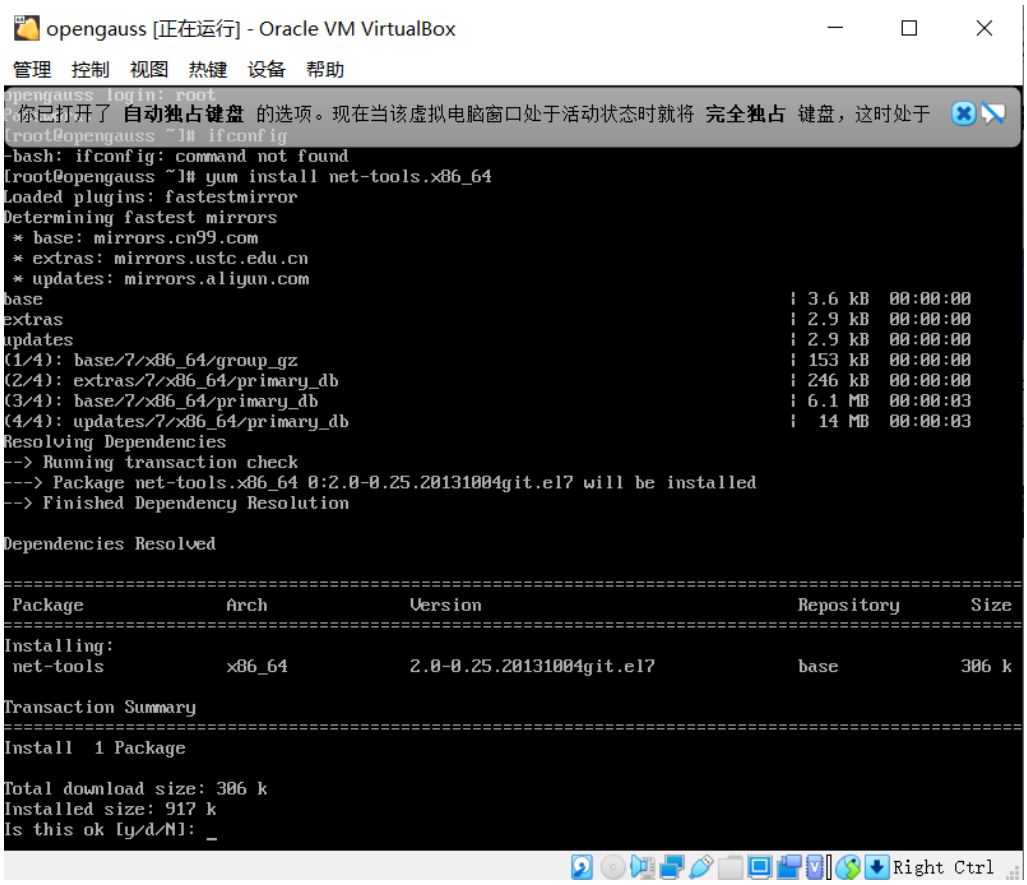


步骤 1 安装 ipconfig

centos7.6 没有安装 ipconfig

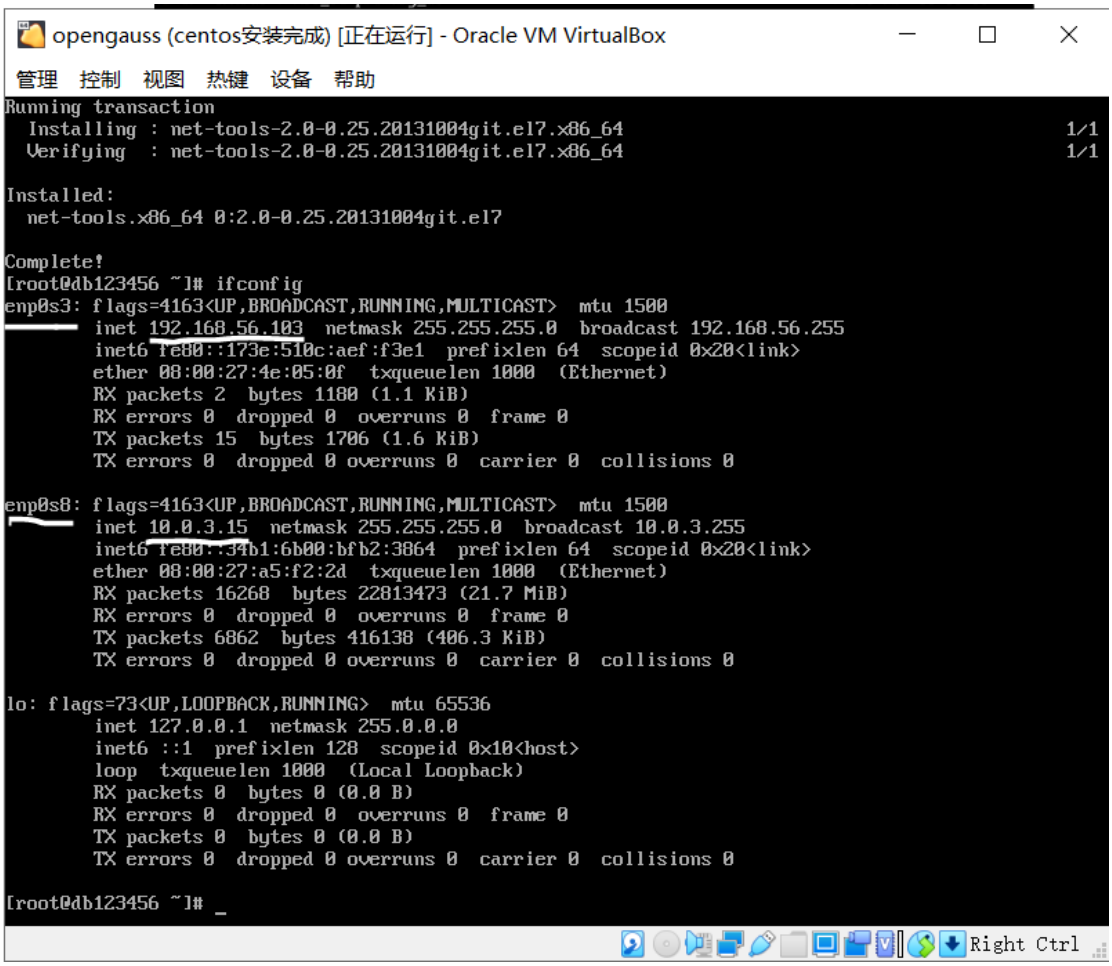
我们装一下。

自动下载安装包，安装。



全部选择 y

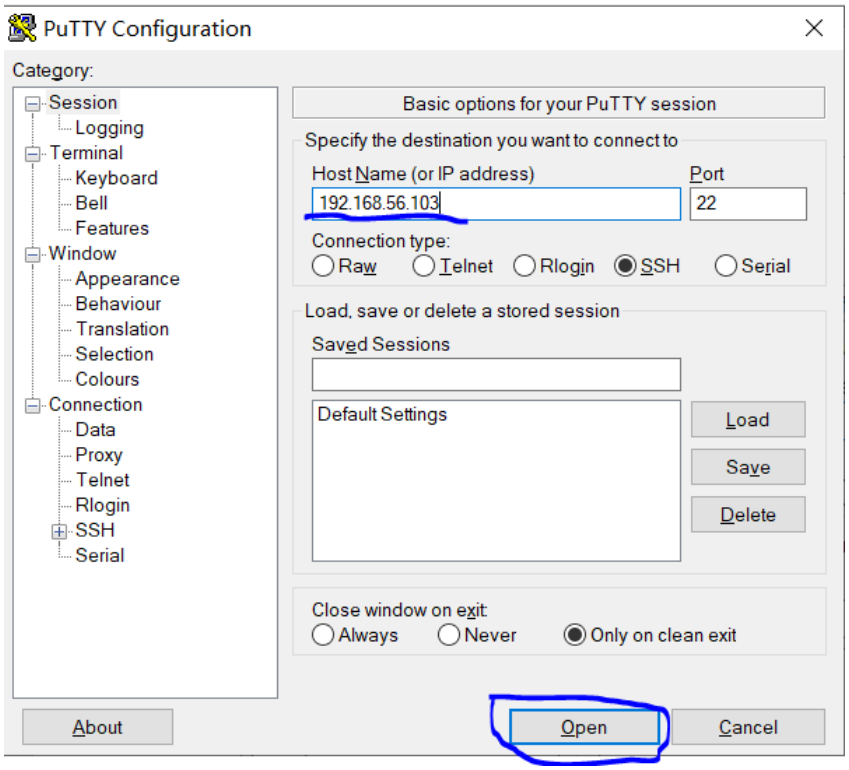
安装完成后，就可以用 ifconfig 了



从上图中可以看到 emp0s3 的 ip 为 192.168.56.103，注意必须是 192.168 开头的 ip，否则是有问题的。

而 emp0s8 的 ip 地址 10.0.3.15 在后面配置文件 clusterconfig.xml 中会使用。

获得 emp0s3 的 ip 后，我们就可以使用该 ip，用 putty 连接，进行后续 openGauss 的安装设置等工作。使用 putty 的好处就是可以直接复制粘贴命令了。



输入用户名为 root，密码为 openGauss@123，注意密码输入的时候，是看不到的

确认用户名和密码输入后，按回车键即可。

步骤 2　关闭防火墙

在 putty 中复制下面的命令（复制，然后在 putty 界面中点击鼠标右键，即可粘贴在其上）

```
systemctl stop firewalld.service
systemctl disable firewalld.service
```

具体如下

下面只用复制粘贴绿色的文字到 putty 上执行，后面也是如此，不再赘述。

```
[root@db123456 ~]# systemctl stop firewalld.service
[root@db123456 ~]# systemctl disable firewalld.service
Removed symlink /etc/systemd/system/multi-
user.target.wants/firewalld.service.
```

```
Removed symlink /etc/systemd/system/dbus-
org.fedoraproject.FirewallD1.service.
```

步骤 3 设置字符集及环境变量。

```
 [root@db123456 ~]# cat >>/etc/profile<<EOF
> export LANG=en_US.UTF-8
> EOF
[root@db123456 ~]# cat >>/etc/profile<<EOF
> export packagePath=/opt/software/openGauss
> EOF
[root@db123456 ~]#  source /etc/profile
[root@db123456 ~]# cat >>/etc/profile<<EOF
> export
LD_LIBRARY_PATH=$packagePath/script/gspylib/clib:$LD_LIBRARY_PATH
> EOF
[root@db123456 ~]# source /etc/profile
验证变量是否生效
[root@db123456 ~]#  echo $LD_LIBRARY_PATH
/opt/software/openGauss/script/gspylib/clib:
```

步骤 4 关闭 swap 交换内存。

```
[root@db123456 ~]# swapoff -a
```

步骤 5 准备 yum 环境。

备份原有的 yum 配置文件

```
[root@db123456 ~]# mv /etc/yum.repos.d/CentOS-Base.repo
/etc/yum.repos.d/CentOS-Base.repo.bak
```

下载可用源的 repo 文件

```
[root@db123456 ~]# curl -o /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.aliyun.com/repo/Centos-7.repo
  % Total    % Received % Xferd  Average Speed   Time    Time
Time  Current
                                 Dload  Upload   Total   Spent
Left  Speed
```

```
100  2523  100  2523    0     0  13867      0 --:--:-- --:--:-- --:--:-- 13939
```

查看 repo 文件内容是否正确

```
[root@db123456 ~]# cat /etc/yum.repos.d/CentOS-Base.repo
# CentOS-Base.repo
#
# The mirror system uses the connecting IP address of the client and the
# update status of each mirror to pick mirrors that are updated to and
# geographically close to the client.  You should use this for CentOS updates
# unless you are manually picking other mirrors.
#
# If the mirrorlist= does not work for you, as a fall back you can try the
# remarked out baseurl= line instead.
#
#

[base]
name=CentOS-$releasever - Base - mirrors.aliyun.com
failovermethod=priority
baseurl=http://mirrors.aliyun.com/centos/$releasever/os/$basearch/
        http://mirrors.aliyuncs.com/centos/$releasever/os/$basearch/

http://mirrors.cloud.aliyuncs.com/centos/$releasever/os/$basearch/
gpgcheck=1
gpgkey=http://mirrors.aliyun.com/centos/RPM-GPG-KEY-CentOS-7

#released updates
[updates]
name=CentOS-$releasever - Updates - mirrors.aliyun.com
failovermethod=priority
baseurl=http://mirrors.aliyun.com/centos/$releasever/updates/$basearch/
```

```
http://mirrors.aliyuncs.com/centos/$releasever/updates/$basearch/

http://mirrors.cloud.aliyuncs.com/centos/$releasever/updates/$basearch/
gpgcheck=1
gpgkey=http://mirrors.aliyun.com/centos/RPM-GPG-KEY-CentOS-7

#additional packages that may be useful
[extras]
name=CentOS-$releasever - Extras - mirrors.aliyun.com
failovermethod=priority
baseurl=http://mirrors.aliyun.com/centos/$releasever/extras/$basearch/

http://mirrors.aliyuncs.com/centos/$releasever/extras/$basearch/

http://mirrors.cloud.aliyuncs.com/centos/$releasever/extras/$basearch/
gpgcheck=1
gpgkey=http://mirrors.aliyun.com/centos/RPM-GPG-KEY-CentOS-7

#additional packages that extend functionality of existing packages
[centosplus]
name=CentOS-$releasever - Plus - mirrors.aliyun.com
failovermethod=priority
baseurl=http://mirrors.aliyun.com/centos/$releasever/centosplus/$basearch/

http://mirrors.aliyuncs.com/centos/$releasever/centosplus/$basearch/

http://mirrors.cloud.aliyuncs.com/centos/$releasever/centosplus/$basearch/
gpgcheck=1
enabled=0
gpgkey=http://mirrors.aliyun.com/centos/RPM-GPG-KEY-CentOS-7

#contrib - packages by Centos Users
```

```
[contrib]
name=CentOS-$releasever - Contrib - mirrors.aliyun.com
failovermethod=priority
baseurl=http://mirrors.aliyun.com/centos/$releasever/contrib/$basearch/

http://mirrors.aliyuncs.com/centos/$releasever/contrib/$basearch/

http://mirrors.cloud.aliyuncs.com/centos/$releasever/contrib/$basearch/
gpgcheck=1
enabled=0
gpgkey=http://mirrors.aliyun.com/centos/RPM-GPG-KEY-CentOS-7
[root@db123456 ~]#
```

步骤 6 yum 安装相关包

执行以下命令，安装所需的包。

```
[root@db123456 ~]# yum install -y libaio-devel flex bison ncurses-devel glibc.devel patch lsb_release wget python3
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: mirrors.aliyun.com
 * extras: mirrors.aliyun.com
 * updates: mirrors.aliyun.com
。。。。。。。。。。
```

需要安装一段时间，安装成功后会显示 complete!

步骤 7 设置默认 Python 版本为 3.x。

```
[root@db123456 ~]# cd /usr/bin
[root@db123456 bin]# mv python python.bak
[root@db123456 bin]# ln -s python3 /usr/bin/python
[root@db123456 bin]#  python -V
Python 3.6.8
```

用 yum --help 命令来验证 yum 是否能使用

```
[root@db123456 bin]#  yum --help
  File "/usr/bin/yum", line 30
    except KeyboardInterrupt, e:
                            ^
SyntaxError: invalid syntax
```

发现不可用

不能使用,则需要修改/usr/bin/yum 文件，把文件中第一行#!/usr/bin/python 修改为#!/usr/bin/python2.7（或者对应的 python 2.x 的版本）。通过 vi 打开文档后，输入＂i＂切换到编辑模式，通过键盘上下左右键移动光标到修改位置修改内容，修改完成后按下 ESC 键退出编辑模式，最后输入＂:wq＂退出并进行保存。如果不修改，则输入“:q!”。注意:必须是英文字符。

```
[root@db123456 bin]#  vi /usr/bin/yum
```

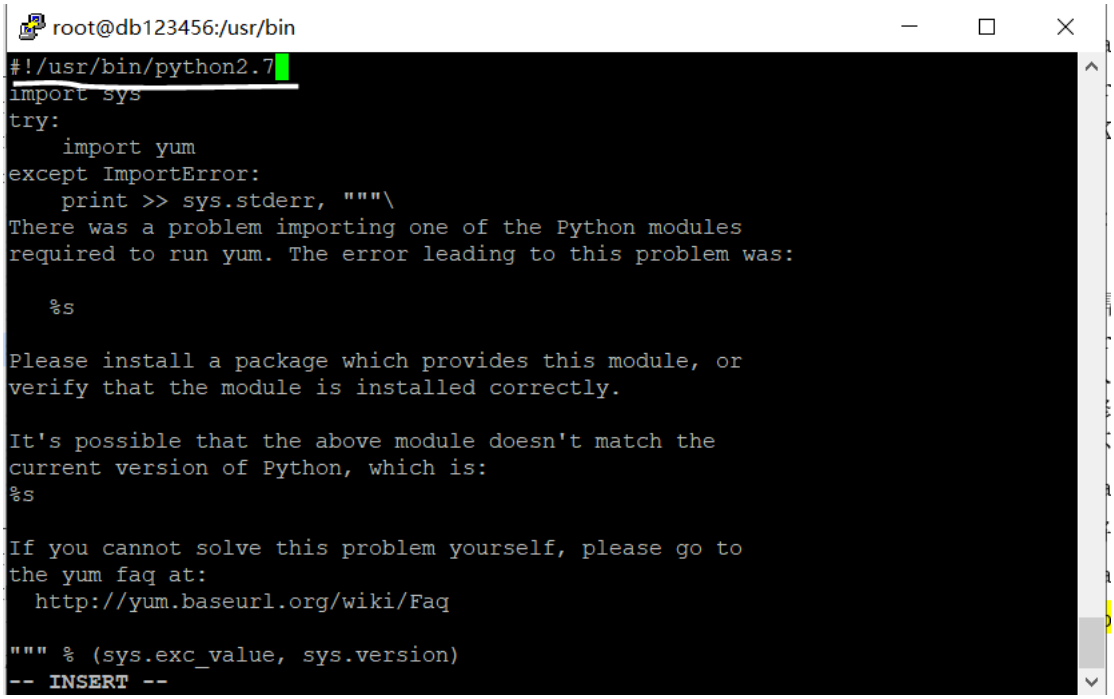
打开文档，将下面用黄色标亮的文字修改一下

```
[root@db123456 bin]# vi /usr/bin/yum
#!/usr/bin/python
import sys
try:
    import yum
```

按键盘上的 i，就可以修改，修改为

```
#!/usr/bin/python2.7
```

修改后，保存退出，先按键盘上的 esc 键退出编辑模式，然后输入:wq!进行保存



再次用 yum --help 命令来验证 yum 是否能使用

```
[root@db123456 bin]# yum --help
Loaded plugins: fastestmirror
Usage: yum [options] COMMAND

List of Commands:

。。。。。。。。。。。。。。。。。。
```

经过测试，这次 ok

步骤 8 创建存放数据库的安装目录。

```
[root@db123456 bin]# mkdir -p /opt/software/openGauss
[root@db123456 bin]# cd /opt/software/openGauss
[root@db123456 openGauss]#
```

步骤 9 创建 XML 配置文件，用于数据库安装。

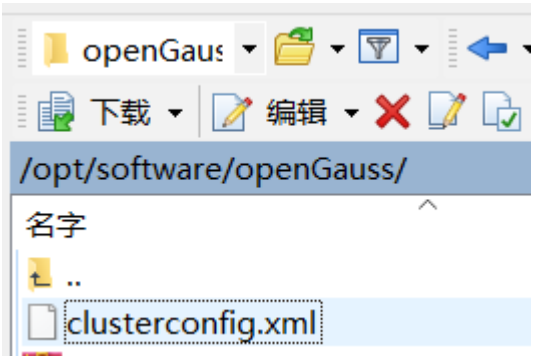
```
[root@db123456 openGauss]# vi clusterconfig.xml
```

将以下内容添加进 clusterconfig.xml 文件中。输入 " i " 切换到编辑模式，复制内容粘贴到文档中，然后按下 ESC 键退出编辑模式，然后输入 " :wq " 退出并进行保存。

下面的文档中黄色标出的地方，需要根据 centos 安装时的主机名和 ip 地址进行设置(主机名可以见命令提示符中的名字)。下面这个文件直接复制粘贴后，记得一定要核对一下，特别是前面两行，是否有问题。如有问题直接根据下面的样子进行编辑输入。

clusterconfig.xml 配置文件的生成中，除了上面给出的直接复制文档的方案外，也可以自己新建记事本文件，将下面的内容复制过去修改，然后保存再使用 winSCP 上传到下面所示正确的位置。

注意：clusterconfig.xml 如果是直接复制的话，请一定核对没有问题才保存。一般这样直接复制，可能会少一部分内容。。自己补全就好。



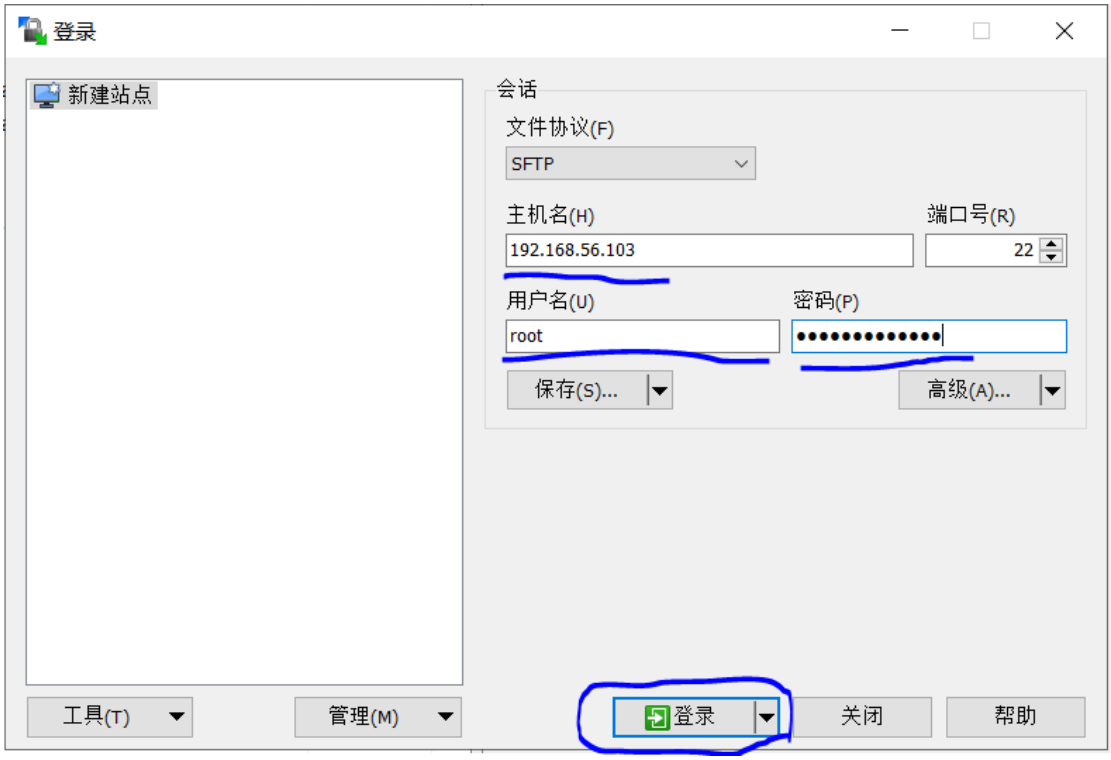

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<ROOT>
    <!-- openGauss 整体信息 -->
    <CLUSTER>
        <PARAM name="clusterName" value="dbCluster" />
        <PARAM name="nodeNames" value="db123456" />
        <PARAM name="backIp1s" value="10.0.3.15"/>
        <PARAM name="gaussdbAppPath" value="/opt/gaussdb/app" />
        <PARAM name="gaussdbLogPath" value="/var/log/gaussdb" />
        <PARAM name="gaussdbToolPath" value="/opt/huawei/wisequery"
/>
        <PARAM name="corePath" value="/opt/opengauss/corefile"/>
        <PARAM name="clusterType" value="single-inst"/>
    </CLUSTER>
    <!-- 每台服务器上的节点部署信息 -->
    <DEVICELIST>
        <!-- node1 上的节点部署信息 -->
        <DEVICE sn="1000001">
            <PARAM name="name" value="db123456"/>
            <PARAM name="azName" value="AZ1"/>
            <PARAM name="azPriority" value="1"/>
            <!-- 如果服务器只有一个网卡可用，将 backIP1 和 sshIP1 配置
成同一个 IP -->
            <PARAM name="backIp1" value="10.0.3.15"/>
            <PARAM name="sshIp1" value="10.0.3.15"/>
        <!--dbnode-->
        <PARAM name="dataNum" value="1"/>
        <PARAM name="dataPortBase" value="26000"/>
        <PARAM name="dataNode1" value="/gaussdb/data/db123456"/>
        </DEVICE>
    </DEVICELIST>
</ROOT>
```

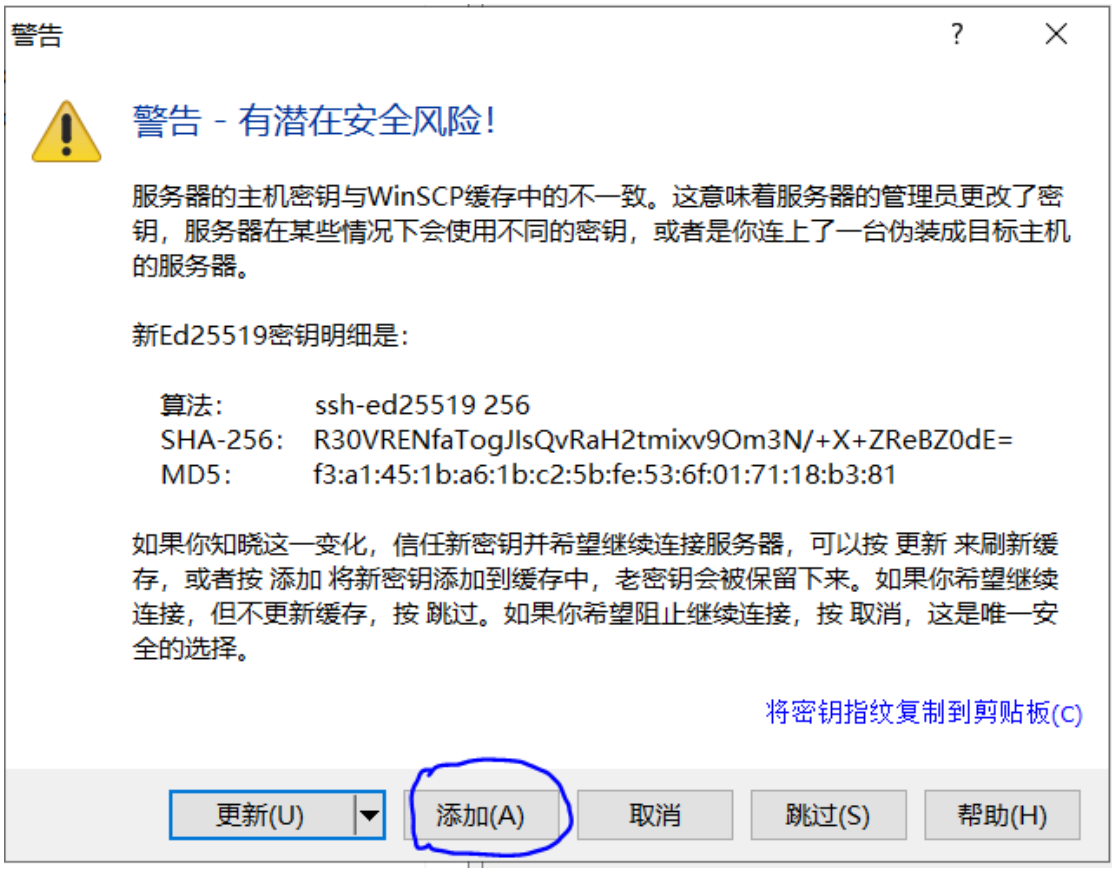
步骤 10　将安装包解压至存放目录

使用 openGauss-2.1.0-CentOS-64bit-all.tar.gz，已经提前下载

使用 winSCP 连接服务器，上传该安装包

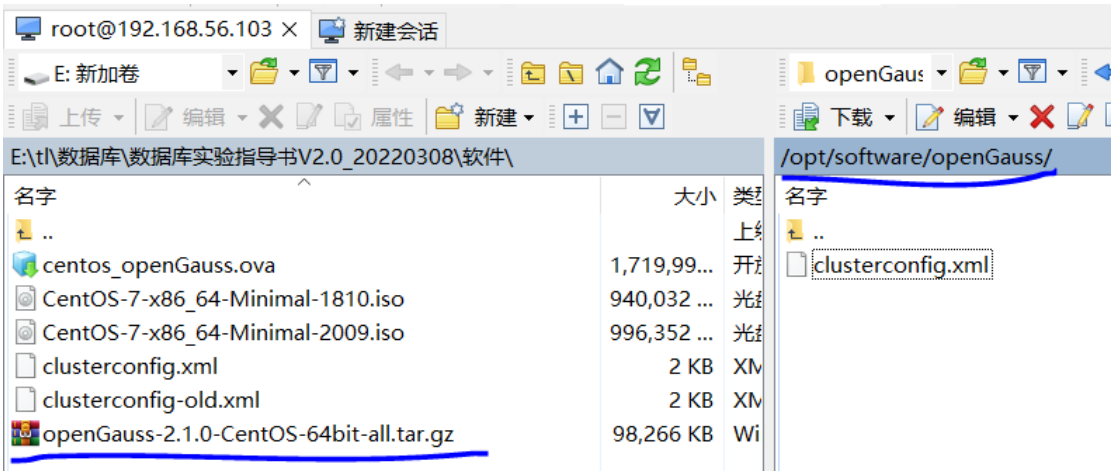


这里主机名是通过 ifconfig 获得的。用户名和密码是安装 centOS 时设置的，但是设置的用户名为 root，密码是 openGauss@123，输入后，点击登录

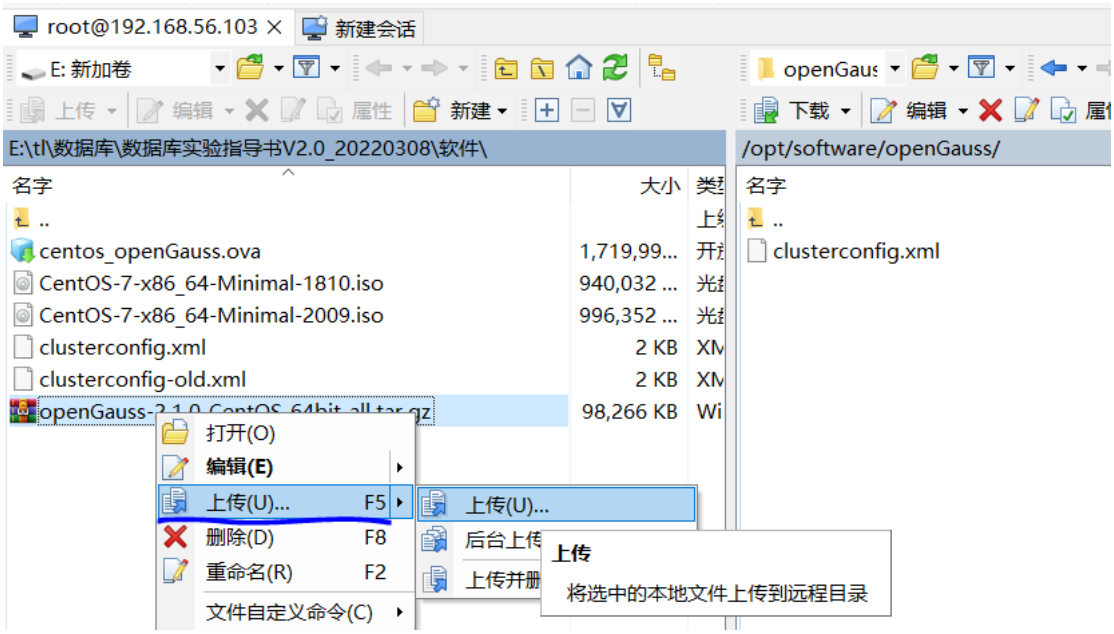


登录后，在左边选择自己电脑中存放安装包的位置

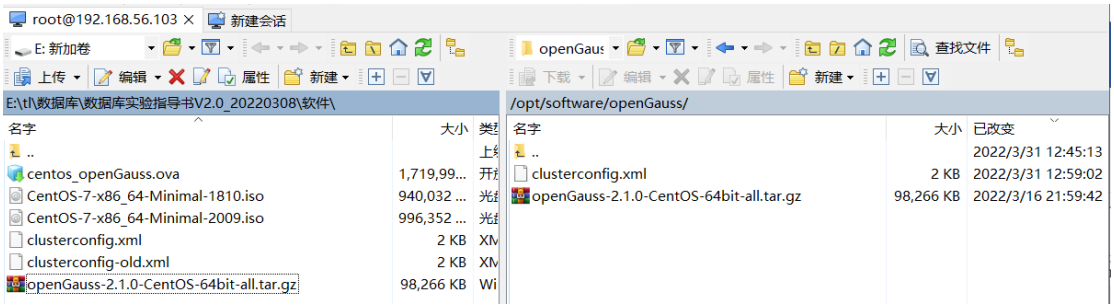
右边就是服务器中，要上传安装包的位置



在左侧的压缩包中点击鼠标右键，在出现的菜单中选择上传



上传成功



步骤 11　将下载好的安装包解压至存放目录。

先解压 openGauss-2.1.0-CentOS-64bit-all.tar.gz 包。

```
[root@db123456 openGauss]# tar -zxvf  openGauss-2.1.0-CentOS-64bit-all.tar.gz
openGauss-2.1.0-CentOS-64bit-om.tar.gz
openGauss-2.1.0-CentOS-64bit.tar.bz2
openGauss-2.1.0-CentOS-64bit-om.sha256
openGauss-2.1.0-CentOS-64bit.sha256
upgrade_sql.tar.gz
upgrade_sql.sha256
```

再解压 openGauss-2.1.0-CentOS-64bit-om.tar.gz

```
[root@db123456 openGauss]# tar -zxvf openGauss-2.1.0-CentOS-64bit-om.tar.gz
./lib/
./lib/pyasn1/
./lib/pyasn1/codec/
./lib/pyasn1/codec/cer/
。。。。。。。。。。。。
```

解压后如下，用 ls 命令查看如下：

```
[root@ db123456 openGauss]# ls
clusterconfig.xml                         openGauss-2.1.0-CentOS-64bit-om.tar.gz  simpleInstall
lib                                       openGauss-2.1.0-CentOS-64bit.sha256     upgrade_sql.sha256
openGauss-2.1.0-CentOS-64bit-all.tar.gz   openGauss-2.1.0-CentOS-64bit.tar.bz2    upgrade_sql.tar.gz
openGauss-2.1.0-CentOS-64bit-om.sha256    script                                  version.cfg
```

安装包解压后，会在/opt/software/openGauss 路径下自动生成 script 子目录，并且在 script 目录下生成 gs_preinstall 等各种 OM 工具脚本。

这些是我们后续安装需要使用的。

更改权限。

```
[root@db123456 openGauss]#  chmod 755 -R /opt/software
[root@db123456 openGauss]# cd /opt/software/openGauss/script
```

步骤 12 执行初始化脚本。

```
[root@db123456 script]#  python gs_preinstall -U omm -G dbgrp -X
/opt/software/openGauss/clusterconfig.xml
[GAUSS-50217] : Failed to decompress version.cfg.The cmd is cd
/opt/software/openGauss/script/../ && tar -xpf `head -1 version.
cfg`*.tar.bz2 ./version.cfg. The output is tar (child): lbzip2:
Cannot exec: No such file or directory
tar (child): Error is not recoverable: exiting now
tar: Child returned status 2
tar: Error is not recoverable: exiting now.
```

出现错误，因为少解压软件，安装。。

```
[root@db123456 script]# yum  install bzip2
```

报错，无法安装

```
Downloading packages:
  File "/usr/libexec/urlgrabber-ext-down", line 28
    except OSError, e:
                  ^
SyntaxError: invalid syntax
```

从网上查找解决方案

```
[root@db123456 script]# vi /usr/libexec/urlgrabber-ext-down
```

将第一行"#!/usr/bin/python" 改为 "#!/usr/bin/python2.7"即可。

修改后，重新安装

```
[root@db123456 script]# yum  install bzip2
```

这次安装成功！

继续执行初始化脚本

```
[root@db123456 script]# python gs_preinstall -U omm -G dbgrp -X
/opt/software/openGauss/clusterconfig.xml
Parsing the configuration file.
Successfully parsed the configuration file.
Installing the tools on the local node.
```

```
Successfully installed the tools on the local node.
Setting pssh path
Successfully set core path.
Are you sure you want to create the user[omm] and create trust for it (yes/no)? yes
Please enter password for cluster user.
Password:这里输入密码为 openGauss@123，但是输入的时候是看不到的
Please enter password for cluster user again.
Password: 这里输入密码为 openGauss@123，但是输入的时候是看不到的
Successfully created [omm] user on all nodes.
Preparing SSH service.
Successfully prepared SSH service.
Checking OS software.
Successfully check os software.
Checking OS version.
Successfully checked OS version.
Creating cluster's path.
Successfully created cluster's path.
Set and check OS parameter.
Setting OS parameters.
Successfully set OS parameters.
Warning: Installation environment contains some warning messages.
Please get more details by "/opt/software/openGauss/script/gs_checkos -i A -h opengauss --detail".
Set and check OS parameter completed.
Preparing CRON service.
Successfully prepared CRON service.
Setting user environmental variables.
Successfully set user environmental variables.
Setting the dynamic link library.
Successfully set the dynamic link library.
Setting Core file
Successfully set core path.
Setting pssh path
Successfully set pssh path.
Setting Cgroup.
```

Successfully set Cgroup.

Set ARM Optimization.

No need to set ARM Optimization.

Fixing server package owner.

Setting finish flag.

Successfully set finish flag.

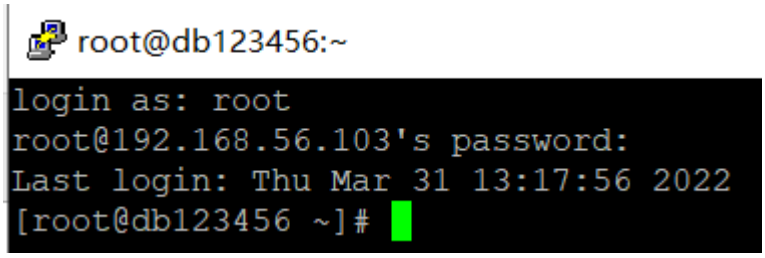
Preinstallation succeeded.

初始化安装成功!

步骤 13 初始化数据库

用 init 6 重启下虚拟机（主要是为了释放一些内存资源）。

[root@db123456 script]# init 6



等虚拟机重启后，再重新打开 putty 连接

输入用户名 root，密码 openGauss@123

先更新下权限。

[root@db123456 ~]# chmod 755 -R /opt/software

使用 omm 用户进行数据库初始化。

[root@db123456 ~]# su - omm

Last login: Thu Mar 31 13:16:05 CST 2022 on pts/0

[omm@db123456 ~]$

[omm@db123456 ~]$ cd /opt/software/openGauss/script

为防止意外发生，做下面这一步前，最好做个快照。

[omm@db123456 script]$ gs_install -X /opt/software/openGauss/clusterconfig.xml --gsinit-parameter="--encoding=UTF8" --dn-guc="max_process_memory=4GB" --dn-guc="shared_buffers=128MB" --dn-guc="bulk_write_ring_size=128MB" --dn-guc="cstore_buffers=16MB"

Parsing the configuration file.

Check preinstall on every node.

Successfully checked preinstall on every node.

Creating the backup directory.

Successfully created the backup directory.

begin deploy..

Installing the cluster.

begin prepare Install Cluster..

Checking the installation environment on all nodes.

begin install Cluster..

Installing applications on all nodes.

Successfully installed APP.

begin init Instance..

encrypt cipher and rand files for database.

Please enter password for database: 这里输入密码为 openGauss@123，但输入时看不到

Please repeat for database: 这里输入密码为 openGauss@123，但是输入的时候是看不到的

begin to create CA cert files

The sslcert will be generated in /opt/gaussdb/app/share/sslcert/om

Cluster installation is completed.

Configuring.

Deleting instances from all nodes.

Successfully deleted instances from all nodes.

Checking node configuration on all nodes.

Initializing instances on all nodes.

Updating instance configuration on all nodes.

Check consistence of memCheck and coresCheck on database nodes.

Configuring pg_hba on all nodes.

Configuration is completed.

Successfully started cluster.

Successfully installed application.

end deploy..

安装成功!

步骤 14 测试!!

[omm@db123456 script]$ su - omm

```
Password: 这里输入密码为 openGauss@123
[omm@db123456 ~]$ gs_om -t start
Starting cluster.
=========================================
[SUCCESS] db123456
2022-03-31 16:07:45.993 62456151.1 [unknown] 140641864811456
[unknown] 0 dn_6001 01000  0 [BACKEND] WARNING:  could not create any
HA TCP/IP sockets
=========================================
Successfully started.
[omm@db123456 ~]$ gsql -d postgres -p 26000 -r
gsql ((openGauss 2.1.0 build 590b0f8e) compiled at 2021-09-30
14:29:04 commit 0 last mr  )
Non-SSL connection (SSL connection is recommended when requiring
high-security)
Type "help" for help.

openGauss=#
```

步骤 15 清理软件安装包（不清理也可以）

```
[omm@db123456 ~]$ exit
logout
[root@db123456 ~]#  cd /opt/software/openGauss/
[root@db123456 openGauss]#  ll
total 296180
-rwxr-xr-x.  1 omm  dbgrp      1364 Mar 31 12:59 clusterconfig.xml
drwxr-xr-x. 15 root root       4096 Mar 31 13:15 lib
drwxr-xr-x.  9 root root        118 Mar 31 13:16 libcgroup
-rwxr-xr-x.  1 root root  100623501 Mar 16 21:59 openGauss-2.1.0-
CentOS-64bit-all.tar.gz
-rwxr-xr-x.  1 root root         65 Sep 30 14:40 openGauss-2.1.0-
CentOS-64bit-om.sha256
-rwxr-xr-x.  1 root root   13672565 Sep 30 14:40 openGauss-2.1.0-
CentOS-64bit-om.tar.gz
```

```
-rwxr-xr-x.  1 root root         65 Sep 30 14:41 openGauss-2.1.0-
CentOS-64bit.sha256
-rwxr-xr-x.  1 root root   87531625 Sep 30 14:41 openGauss-2.1.0-
CentOS-64bit.tar.bz2
-rwxr-xr-x.  1 root root  101189534 Mar 31 13:15 openGauss-Package-
bak_compiled.tar.gz
drwxr-xr-x.  6 root root       4096 Sep 30 14:40 script
drwxr-xr-x.  2 root root       4096 Sep 30 14:40 simpleInstall
-rwxr-xr-x.  1 root root         65 Sep 30 14:39 upgrade_sql.sha256
-rwxr-xr-x.  1 root root     224994 Sep 30 14:39 upgrade_sql.tar.gz
-rwxr-xr-x.  1 root root         32 Sep 30 14:39 version.cfg
[root@db123456 openGauss]#  rm -rf openGauss-2.1.0-CentOS-64bit-
all.tar.gz
[root@db123456 openGauss]# rm -rf openGauss-2.1.0-CentOS-64bit-
om.tar.gz
```

附录 6：建库建表 SQL 示例（表名、字段名最好改为英文）

下面仅仅为示例，数据库名称，数据表名称以及表中字段名称都最好使用英文。

```
CREATE DATABASE dingdan_db OWNER tang;

--创建名为客户的数据表--
CREATE TABLE 客户
(
客户编号 char(4) ,
姓名 varchar(10) ,
地址 varchar(20),
邮政编码 varchar(6),
收支差额 money,
贷款限额 money,
代理商编号 char(4),
PRIMARY KEY(客户编号),--在表级定义实体完整性
);
--插入数据到名为客户的数据表中--
```