

一. 准备工作:

1. 请确保服务器系统是 centos7.5
2. 检查服务器是否已经安装 g++ 和 gcc(一般来说, centos7.5 已经安装了)

查询是否有安装以及版本号, 如果没有请安装 g++ (对应的 gcc 版本号也要一致), 如果安装请直接跳过这一步

```
1 g++ --version
```

查看 gcc 版本号

```
1 gcc --version
```

正确的结果应该如下, 即显示了版本号, 而不是提示命令不存在

```
gcc version 4.4.7 20120313 (Red Hat 4.4.7-17) (GCC)
[root@instance-wf05me8n gcc]# gcc --version
gcc (GCC) 4.4.7 20120313 (Red Hat 4.4.7-17)
Copyright (C) 2010 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

[root@instance-wf05me8n gcc]# g++ --version
g++ (GCC) 4.4.7 20120313 (Red Hat 4.4.7-17)
Copyright (C) 2010 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

如果没有安装

安装 g++ 和 gcc 直接将所提供的 gcc for 7.5 包里的所有文件都 copy 到服务器上, 再执行指令即可

建立文件夹gcc 用于存放下载的rpm包

```
1 mkdir gcc
```

安装gcc(当执行下面的语句后, 系统会自动选出所需的依赖包进行安装, 不需要的就会自动清理)

```
1 rpm -Uvh *.rpm --nodeps --force
```

安装过程中出现的异常情况

注: rpm包版本号均用 xxx 代替, 使用时需将xxx替换成对应版本号

1. glibc和glibc-common相互依赖的问题

rpm -ivh glibc-common-xxx.rpm 报 dependence glibc = xxx , rpm -ivh glibc-xxx.rpm 报 dependence glibc-common= xxx, 将两个包一块安装即可

```
1 rpm -ivh glibc-xxx.rpm glibc-common-xxx.rpm
```

2. 出现 “rpm conflicts with file from package”

```
1 安装的时候增加-replacefiles参数, 例如:  
2 rpm -ivh xxx.rpm -replacefiles
```

二. 安装 python

1. 将 **python--Linux 安装** 文件夹下所有的包放置在linux的一个目录下, 例如 /home/python

2. 在终端进入包放置的目录下, (我以 /home/python 为例) :

终端输入

```
1 chmod 777 -R /home/python
```

回车,

```
1 cd /home/python
```

回车,

```
1 ./install.sh
```

回车, 等待完毕即可

如果出现错误

```
1 -bash: ./install.sh: /bin/sh^M: bad interpreter: No such file or director
y
```

1: 首先用vi命令打开当前的shell脚本文件

```
1 vi install.sh
```

2: 在vi命令模式中使用 :set ff 命令

```
cd ..
if [ -f /usr/bin/python ]
then
else
    mv -f /usr/bin/python /usr/bin/python_bak
    echo "/usr/bin/python is not exist."
fi
fileformat=dos
```

>- 仅将文本发送到当前选项卡

直接当下输入命令

```
1 :set ff=unix
```

退出 vim

重新执行等待安装完成即可

```
1 ./install.sh
```

3. 测试,

```
1 python
```

回车, 输入一下命令: (由于 6.8 默认已经自带 3.6 的 python, 所以可能会报错, 无视即可, 继续向下走)

```
1 import matplotlib.pyplot as plt
2 plt.plot([1,2,3],[5,7,4])
3 plt.savefig('/home/test.png')
```

执行完后在home目录下有 test.png 图片表示成功

4. 修改环境变量

在终端输入

```
1 vim /etc/profile
```

然后在 **profile** 文件中加上

```
1 export PATH=$PATH:/usr/local/ImageMagick/bin
```

注意： /usr/local/ImageMagick/bin 是 ImageMagick 的安装路径

在终端输入

```
1 source /etc/profile
```

使修改生效

5. 验证

在终端输入

```
1 convert --version
```

输出相关版本信息说明安装成功

```
[root@xgemp ImageMagick-7.0.8-8]# convert -version
Version: ImageMagick 7.0.8-8 Q16 x86_64 2018-08-02 https://www.imagemagick.org
Copyright: © 1999-2018 ImageMagick Studio LLC
License: https://www.imagemagick.org/script/license.php
Features: Cipher DPC HDRI OpenMP
Delegates (built-in): bzlib fontconfig freetype jng jpeg pangocairo png x xml zlib
```

执笔人: lianghuageng

日期: 2019-09-09