登录 | 注册

落魄影子

记录嵌入式技术生涯的点点滴滴！~~

目录视图   摘要视图   RSS 订阅

文章搜索

文章分类

3G模块 (1)
c/c++ learning (33)
Linux学习 (11)
linux设备驱动 (4)
MYSQL数据库 (4)
python (9)
u-boot (1)
生活 (1)
go语言 (1)
hadoop技术 (2)
数据结构 (8)
web框架django (2)
TI BLE (2)
Android系统 (3)

文章存档

展开

【免费公开课】Gulp前端自动化教程   【专家问答】陈绍英:大型IT系统性能测试实战   【博客活动】有奖征文--走进VR开发世界

linux环境下的python安装过程(含setuptools)

2013-03-18 20:14   45471人阅读   评论(1)   收藏   举报

分类: python(8)

版权声明:本文为博主原创文章,未经博主允许不得转载。

目录(?)   [+]

这里我不想采用诸如ubuntu下的apt-get install方式进行**Python**的安装,而是在linux下采用源码包的方式进行python的安装。

一、下载python源码包

打开ubuntu下的shell终端,通过wget命令下载python源码包,如下图所示:

```
zhm@master:/opt$ sudo wget http://www.python.org/ftp/python/2.7.3/Python-2.7.3.tgz
[sudo] password for zhm:
--2013-03-16 19:32:24--  http://www.python.org/ftp/python/2.7.3/Python-2.7.3.tgz
正在解析主机 www.python.org... 82.94.164.162, 2001:888:2000:d::a2
正在连接 www.python.org|82.94.164.162|:80... 已连接。
已发出 HTTP 请求,正在等待回应... 200 OK
长度: 14135620 (13M) [application/x-tar]
正在保存至: "Python-2.7.3.tgz"

100%[====================================>] 14,135,620   164K/s   in 88s

2013-03-16 19:33:52 (158 KB/s) - 已保存 "Python-2.7.3.tgz" [14135620/14135620])

zhm@master:/opt$
```

将python-2.7.3.tgz下载至/opt目录下。

二、**python**的解压

```
zhm@master:/opt$ ls
jdk1.7.0_09  Python-2.7.3.tgz
zhm@master:/opt$ sudo tar zxvf Python-2.7.3.tgz
```

三、**python**的**编译**与安装

在对python进行编译之前,必须对它进行配置。在unix/linux平台上的安装过程中,配置和编译过程全部已经自动化了,我们所需要做的就是输入几条命令然后回车即可。

通过执行脚本./configure开始编译,这一过程可以决定系统的配置并创建相应的Makefile文件。你也可以传递一些参数选项来配置这个脚本。当这一步完成后,就可以运行make了。操作如下图所示:

```
zhm@master:/opt/Python-2.7.3$ sudo ./configure --prefix=/usr/local/python2.7
[sudo] password for zhm:
checking for --enable-universalsdk... no
checking for --with-universal-archs... 32-bit
checking MACHDEP... linux2
checking EXTRAPLATDIR...
checking machine type as reported by uname -m... i686
checking for --without-gcc... no
checking for gcc... gcc
checking whether the C compiler works... yes
checking for C compiler default output file name... a.out
checking for suffix of executables...
checking whether we are cross compiling... no
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ISO C89... none needed
checking for --with-cxx-main=<compiler>... no
```

通过添加--prefix用于指定python的安装路径于/usr/local/python2.7,配置完成后,我们就可以执行make操作了.

```
zhm@master:/opt/Python-2.7.3$ make
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -Wstri
ct-prototypes  -I. -IInclude -I./Include   -DPy_BUILD_CORE -o Modules/python.o .
/Modules/python.c
gcc -pthread -c -fno-strict-aliasing -g -O2 -DNDEBUG -g -fwrapv -O3 -Wall -W
ct-prototypes  -I. -IInclude -I./Include   -DPy_BUILD_CORE -o Parser/acceler.o P
arser/acceler.c
```

上面make编译的时间可能会比较长,大概5-6分钟,完成之后我们就可以进行安排操作了,输入指令make
install(注意权限),如下:

```
zhm@master:/opt/Python-2.7.3$ sudo make install
```

下面我们查看一下安装后的python目录,指令如下:

```
zhm@master:/opt/Python-2.7.3$ ls /usr/local/python2.7/
bin  include  lib  share
zhm@master:/opt/Python-2.7.3$
```

上面在./configure的时候增加python的安装目录,所以我们用ls命令查看此目录的信息。其中bin目录下用于存
放python的相关执行程序。

但是,由于这是我们自己手动将python安装于/usr/local/python2.7目录下,此时$PATH环境变量无法找到此目
录下的python解释器,为此可以增加一个软链接,代码如下:

```
zhm@master:/opt/Python-2.7.3$ sudo ln -s /usr/local/python2.7/bin/python  /bin/p
ython2.7
zhm@master:/opt/Python-2.7.3$ python
python            python2.6          python2.7
python2           python2.6-config  python-config
zhm@master:/opt/Python-2.7.3$ python
```

通过ln -s /usr/local/python2.7/bin/python /bin/python2.7命令可以在/bin路径下创建一软链接文件python2.7,
当访问此链接文件,就可以间接地访问/usr/local/python2.7下的python程序了。这种方式与windows下的快捷方
式类似。

下面可以通过运行python命令交互模式以查看刚刚安装的python版本情况:

```
zhm@master:/opt/Python-2.7.3$ python2.7
Python 2.7.3 (default, Mar 18 2013, 18:49:55)
[GCC 4.4.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
zhm@master:/opt/Python-2.7.3$
```

如上所示,通过直接执行python2.7可以访问/usr/local/python2.7下的python解释器,其版本为Python 2.7.3,这
个正是我们刚刚下载的python版本,或者也可以通过命令:

python2.7 -V来查看。至此Python安装结束。

四、Setuptools的安装过程

1、setuptools的下载

setuptools源码的下载与python类似,google上一搜索第一个便是,通过wget命令来下载,指令如下:

```
zhm@master:/opt$ sudo wget https://pypi.python.org/packages/source/s/setuptools/
setuptools-0.6c11.tar.gz#md5=7df2a529a074f613b509fb44feefe74e
[sudo] password for zhm:
--2013-03-18 19:12:01--  https://pypi.python.org/packages/source/s/setuptools/se
tuptools-0.6c11.tar.gz
正在解析主机 pypi.python.org... 140.211.10.69
正在连接 pypi.python.org|140.211.10.69|:443... 已连接。
已发出 HTTP 请求,正在等待回应... 200 OK
长度: 256862 (251K) [application/octet-stream]
正在保存至: "setuptools-0.6c11.tar.gz"

100%[===================================>] 256,862     153K/s   in 1.6s

2013-03-18 19:12:05 (153 KB/s) - 已保存 "setuptools-0.6c11.tar.gz" [256862/25686
2])

zhm@master:/opt$ ls
jdk1.7.0_09  Python-2.7.3  Python-2.7.3.tgz  setuptools-0.6c11.tar.gz
zhm@master:/opt$
```

2、setuptools的解压

```
zhm@master:/opt$ ls
jdk1.7.0_09  Python-2.7.3  Python-2.7.3.tgz  setuptools-0.6c11.tar.gz
zhm@master:/opt$ sudo tar zxvf setuptools-0.6c11.tar.gz
```

3、setuptools的编译及安装

编译如下,用python2.7版本的程序来执行setup.py脚本,此脚本带参数build,执行编译过程。

(注意:如果用其它版本的python解释器来执行时,如python2.6,将把setuptools安装至相应的版本目录下)

```
zhm@master:/opt/setuptools-0.6c11$ sudo python2.7 setup.py build
running build
running build_py
creating build
creating build/lib
copying pkg_resources.py -> build/lib
```

如果编译OK,就可以进行安装了,与上述指令类似,如下:

```
zhm@master:/opt/setuptools-0.6c11$ sudo python2.7 setup.py install
running install
running bdist_egg
running egg_info
writing setuptools.egg-info/PKG-INFO
writing top-level names to setuptools.egg-info/top_level.txt
```

如果不出问题就已经安装成功了,但是目前我安装不成功,显示如下错误:

```
    z = zipfile.ZipFile(zip_filename, mode, compression=compression)
  File "/usr/local/python2.7/lib/python2.7/zipfile.py", line 681, in __init__
    "Compression requires the (missing) zlib module"
RuntimeError: Compression requires the (missing) zlib module
zhm@master:/opt/setuptools-0.6c11$
```

根据错误提示,主要错误在于"Compression requires the (missing) zlib module",错误原因在于缺少zlib模块,但是最根本原因在于安装python2.7.3之前未进行依赖检查,最好通过命令sudo apt-get build-dep python进行检查。所以要解决zlib module缺少的问题,必须在安装python2.7.3之前先安装zlib1g-dev软件包,安装完成后再重新安装python2.7.3,然后再运行sudo python2.7 setup.py install,此时才可正常解决问题,如下所示:

```
running build_py
copying setuptools.egg-info/PKG-INFO -> build/bdist.linux-i686/egg/EGG-INFO
copying setuptools.egg-info/SOURCES.txt -> build/bdist.linux-i686/egg/EGG-INFO
copying setuptools.egg-info/dependency_links.txt -> build/bdist.linux-i686/egg/EGG-INFO
copying setuptools.egg-info/entry_points.txt -> build/bdist.linux-i686/egg/EGG-INFO
copying setuptools.egg-info/top_level.txt -> build/bdist.linux-i686/egg/EGG-INFO
creating 'dist/setuptools-0.6c11-py2.7.egg' and adding 'build/bdist.linux-i686/egg' to i
removing 'build/bdist.linux-i686/egg' (and everything under it)
Processing setuptools-0.6c11-py2.7.egg
Copying setuptools-0.6c11-py2.7.egg to /usr/local/python2.7/lib/python2.7/site-packages
Adding setuptools 0.6c11 to easy-install.pth file
Installing easy_install script to /usr/local/python2.7/bin
Installing easy_install-2.7 script to /usr/local/python2.7/bin

Installed /usr/local/python2.7/lib/python2.7/site-packages/setuptools-0.6c11-py2.7.egg
Processing dependencies for setuptools==0.6c11
Finished processing dependencies for setuptools==0.6c11
zhm@master:/opt/setuptools-0.6c11$
```

出现以上图就表明setuptools工具已经安装成功。并且可以看到此setuptools安装的路径在/usr/local/python/python2.7目录下。

我们可以查看一下:

```
zhm@master:/opt/setuptools-0.6c11$ ls /usr/local/python2.7/bin
2to3            easy_install-2.7  pydoc     python2     python2.7-config  python-config
easy_install    idle              python    python2.7   python2-config    smtpd.py
zhm@master:/opt/setuptools-0.6c11$
```

我们可以看到在刚安装的python目录下存在一些easy_install程序,这些就是setuptools工具的一些程序,方便

我们后期安装第三方模块使用,在此我们继续作个软链接:

```
zhm@master:/opt/setuptools-0.6c11$ sudo ln -s /usr/local/python2.7/bin/easy_install /bin/easy_in
stall
zhm@master:/opt/setuptools-0.6c11$ easy_install
error: can't create or remove files in install directory
```

这样,当我们直接输入easy_install就可以访问此程序了。

4、用setuptools自动安装django及numpy

安装django命令如下:

sudo easy_install django

但是有时会发生"unknow url type: htpps"错误,如下图(本人就发生啦,很不幸!)

```
zhm@master:/opt/setuptools-0.6c11$ sudo easy_install-2.7.3 django
[sudo] password for zhm:
Searching for django
Reading http://pypi.python.org/simple/django/
Reading http://www.djangoproject.com/
Download error: unknown url type: https -- Some packages may not be found!
Reading http://www.djangoproject.com/m/bad-installer.txt
Best match: Django 1.5
Downloading https://www.djangoproject.com/m/releases/1.5/Django-1.5.tar.gz
error: Download error for https://www.djangoproject.com/m/releases/1.
5.tar.gz: unknown url type: https
zhm@master:/opt/setuptools-0.6c11$
```

出现以上问题的解决办法只有一个,即在安装python2.7.3之前未安装libssl-dev库,因此需要安装此库,然后再

重新按上述方式安装一遍python和setuptools,最后再用easy_install安装django,此时才能成功安装。安装

libssl-dev库命令如下:

sudo apt-get install libssl-dev

安装numpy命令如下:

sudo easy_install numpy

就可以自动安装了,这种方式和sudo apt-get install 类似

<div align="center">

顶　　　踩

3　　　　0

</div>

上一篇　关于python模块自动安装(setuptools)

下一篇　pyton内置容器类型练手

我的同**类**文章

python（8）

参考知**识库**

Python知识库

8737 关注 | 801 收录

猜你在找

Python自动化开发实战视频课程-全新基础篇   Linux环境下MySQL-python安装过程
最牛JavaScript课程   Linux环境下Python的安装过程
Python项目实训   Linux环境下Python的安装过程 内有python各版本的下
Laravel的Linux容器Vagrant+Homestead+Composer+Ya   linux环境python安装setuptools使用easy_install的
深入浅出MySQL入门必备   Centos环境下MySQL-python安装过程

linux运维培训　python教程　php视频教程　　python安　**python中文手册**　红帽linux认证

python环境　linux培训机构　python开发工具　python视频　python学习手册　炒股票新手入门

查看评论

1楼 lingangluckyy 2015-07-16 18:59发表

为啥不在安装之前就提醒错误呢?- -||

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点,不代表CSDN网站的观点或立场

核心技**术类**目

全部主题　　Hadoop　　AWS　　移动游戏　　Java　　Android　　iOS　　Swift　　智能硬件　　Docker　　OpenStack
VPN　　Spark　　ERP　　IE10　　Eclipse　　CRM　　JavaScript　　数据库　　Ubuntu　　NFC　　WAP　　jQuery
BI　　HTML5　　Spring　　Apache　　.NET　　API　　HTML　　SDK　　IIS　　Fedora　　XML　　LBS　　Unity
Splashtop　　UML　　components　　Windows Mobile　　Rails　　QEMU　　KDE　　Cassandra　　CloudStack
FTC　　coremail　　OPhone　　CouchBase　　云计算　　iOS6　　Rackspace　　Web App　　SpringSide　　Maemo
Compuware　　大数据　　aptech　　Perl　　Tornado　　Ruby　　Hibernate　　ThinkPHP　　HBase　　Pure　　Solr
Angular　　Cloud Foundry　　Redis　　Scala　　Django　　Bootstrap