

windows平台使用Microsoft Visual C++ Compiler for Python 2.7编译python扩展

在windows平台上安装python c extension的扩展包是件很痛苦的事情，一般通过安装vc/vs系列来编译C扩展，不过安装包都比较大。或者通过mingw编译，不过有时会在兼容性上出现点问题。

有个好消息就是微软为Python提供了专用的编译器Microsoft Visual C++ Compiler for Python 2.7（包含32位和64位） 下载地址: <http://aka.ms/vcpython27>

提示：在此感谢@ThunderEX的提醒，setuptools 6.0及后续版本可以自动识别Microsoft Visual C++ Compiler for Python 2.7了，赶紧升级吧。如果不能升级，请参考下面的操作步骤。

1.下载完成并安装。以本机为例，安装完成后的路径为：

```
1 | C:\Users\Administrator\AppData\Local\Programs\Common\Microsoft\Visual C++ for Python\9.0
```

2.修改python安装目录下Lib\distutils\msvc9compiler.py文件(如有必要可能msvccompiler.py文件也需要做相应更改，视系统而定)，找到get_build_version方法直接return 9.0

```
1 | def get_build_version():
2 |     """Return the version of MSVC
3 |     that was used to build Python.
4 |
5 |     For Python 2.3 and up, the
6 |     version number is included in
7 |     sys.version. For earlier
8 |     versions, assume the compiler is
9 |     MSVC 6.
10 |    """
11 |    return 9.0
12 |    prefix = "MSC v."
13 |    i = sys.version.find(prefix)
14 |    if i == -1:
15 |        return 6
16 |    i = i + len(prefix)
17 |    s,
18 |    rest = sys.version[i:].split("
19 |    ", 1)
20 |    majorVersion = int(s[:-2]) - 6
21 |    minorVersion = int(s[2:3]) / 10
22 |    .0
    # I don't think paths are
    affected by minor version in
    version 6
    if majorVersion == 6:
        minorVersion = 0
    if majorVersion >= 6:
```

```
        return majorVersion + minorVersion  
    # else we don't know what  
    # version of the compiler this is  
    return None
```

然后再找到find_vcvarsall方法直接返回vcvarsall.bat的路径（以自己机器安装后的路径为准）

```
1  def find_vcvarsall(version):  
2      """Find the vcvarsall.bat file  
3  
4      At first it tries to find the  
5      productdir of VS 2008 in the  
6      registry. If  
7      that fails it falls back to the  
8      VS90COMNTOOLS env var.  
9      """  
10     returnr'C:\Users\Administrator\  
11     AppData\Local\Programs\Common\Micro  
12     soft\Visual C++ for  
13     Python\9.0\vcvarsall.bat'  
14     vsbase = VS_BASE % version  
15     try:  
16         productdir =Reg.get_value(r  
17         "%s\Setup\VC" %vsbase,  
18  
19         "productdir")  
20     except KeyError:  
21         productdir = None  
22  
23     # trying Express edition  
24     if productdir is None:  
25         vsbase = VSEXRESS_BASE %ve  
26         rsion  
27         try:  
28             productdir =Reg.get_val  
29             ue(r"%s\Setup\VC" %vsbase,  
30  
31             "productdir")  
32         except KeyError:  
33             productdir = None  
34             log.debug("Unable to  
35             find productdir in registry")  
36  
37     if not productdir or notos.path  
38     .isdir(productdir):  
39         toolskey = "VS%0.f0COMNTOOL  
40         S"% version
```

```
41         toolsdir =os.environ.get(to
42         olskey, None)
43
44         if toolsdir andos.path.isdi
r(toolsdir):
            productdir =os.path.joi
n(toolsdir, os.pardir,
os.pardir, "VC")
            productdir =os.path.abs
path(productdir)
            if notos.path.isdir(pro
ductdir):
                log.debug("%s is
not a valid
directory" % productdir)
                return None
            else:
                log.debug("Env var %s
is not set or invalid" % toolskey)
                if not productdir:
                    log.debug("No productdir
found")
                    return None
                vcvarsall =os.path.join(product
dir, "vcvarsall.bat")
                if os.path.isfile(vcvarsall):
                    return vcvarsall
                log.debug("Unable to find
vcvarsall.bat")
                return None
```

3.上述完成之后就可以在windwos下正常编译python的C扩展。以pycrypto-2.6.1为例，执行如下命令

```
1 python setup.py install
```

当然也可以建立一个windows的二进制包

```
1 python setup.py bdist_wininst
```