兵卒87

(踩坑)pyinstaller --key加密打包问题

在使用pyinstaller的--key参数加密python文件遇到了一系列问题.

执行pyinstaller.exe -key xxxx -F hellow.py,报错

报错提示需要pycrypto库,运行pip install pycrypto之后出现了报错

```
Collecting pycrypto
  Using cached https://files.pythonhosted.org/packages/60/db/645aa9af249f059cc3a368b118de33889219e0362141e75d4eaf6f
Installing collected packages: pycrypto
    Running setup.py install for pycrypto ... error
    ERROR: Command errored out with exit status 1:
     command: 'c:\users\xxxxxx\appdata\local\programs\python\python37\python.exe' -u -c 'import sys, setuptools, to
         cwd: C:\Users\xxxxxx\AppData\Local\Temp\pip-install-h6xa5sa0\pycrypto\
    Complete output (183 lines):
    running install
    running build
    running build_py
    creating build
    creating build\lib.win-amd64-3.7
    creating build\lib.win-amd64-3.7\Crypto
    copying lib\Crypto\pct_warnings.py -> build\lib.win-amd64-3.7\Crypto
    copying lib\Crypto\__init__.py -> build\lib.win-amd64-3.7\Crypto
    creating build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\hashalgo.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\HMAC.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\MD2.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\MD4.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\MD5.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\RIPEMD.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\SHA.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\SHA224.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\SHA256.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\SHA384.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\SHA512.py -> build\lib.win-amd64-3.7\Crypto\Hash
    copying lib\Crypto\Hash\__init__.py -> build\lib.win-amd64-3.7\Crypto\Hash
    creating build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\AES.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\ARC2.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\ARC4.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\blockalgo.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\Blowfish.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\CAST.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\DES.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\DES3.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\PKCS1_OAEP.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\PKCS1_v1_5.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\XOR.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    copying lib\Crypto\Cipher\__init__.py -> build\lib.win-amd64-3.7\Crypto\Cipher
    creating build\lib.win-amd64-3.7\Crypto\Util
    copying lib\Crypto\Util\asn1.py -> build\lib.win-amd64-3.7\Crypto\Util
```

公告

昵称： 兵卒87
园龄： 8个月
粉丝： 0
关注： 0
+加关注

< 2020年8

日	一	二	三
26	27	28	29
2	3	4	5
9	10	11	12
16	17	18	19
23	24	25	26
30	31	1	2

搜索

随笔档案

2019年11月(1)

阅读排行榜

1. (踩坑)pyinstaller --
(2833)

```
copying lib\Crypto\Util\Counter.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\number.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\py3compat.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\randpool.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\RFC1751.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\winrandom.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\_number_new.py -> build\lib.win-amd64-3.7\Crypto\Util
copying lib\Crypto\Util\__init__.py -> build\lib.win-amd64-3.7\Crypto\Util
creating build\lib.win-amd64-3.7\Crypto\Random
copying lib\Crypto\Random\random.py -> build\lib.win-amd64-3.7\Crypto\Random
copying lib\Crypto\Random\_UserFriendlyRNG.py -> build\lib.win-amd64-3.7\Crypto\Random
copying lib\Crypto\Random\__init__.py -> build\lib.win-amd64-3.7\Crypto\Random
creating build\lib.win-amd64-3.7\Crypto\Random\Fortuna
copying lib\Crypto\Random\Fortuna\FortunaAccumulator.py -> build\lib.win-amd64-3.7\Crypto\Random\Fortuna
copying lib\Crypto\Random\Fortuna\FortunaGenerator.py -> build\lib.win-amd64-3.7\Crypto\Random\Fortuna
copying lib\Crypto\Random\Fortuna\SHAd256.py -> build\lib.win-amd64-3.7\Crypto\Random\Fortuna
copying lib\Crypto\Random\Fortuna\__init__.py -> build\lib.win-amd64-3.7\Crypto\Random\Fortuna
creating build\lib.win-amd64-3.7\Crypto\Random\OSRNG
copying lib\Crypto\Random\OSRNG\fallback.py -> build\lib.win-amd64-3.7\Crypto\Random\OSRNG
copying lib\Crypto\Random\OSRNG\nt.py -> build\lib.win-amd64-3.7\Crypto\Random\OSRNG
copying lib\Crypto\Random\OSRNG\posix.py -> build\lib.win-amd64-3.7\Crypto\Random\OSRNG
copying lib\Crypto\Random\OSRNG\rng_base.py -> build\lib.win-amd64-3.7\Crypto\Random\OSRNG
copying lib\Crypto\Random\OSRNG\__init__.py -> build\lib.win-amd64-3.7\Crypto\Random\OSRNG
creating build\lib.win-amd64-3.7\Crypto\SelfTest
copying lib\Crypto\SelfTest\st_common.py -> build\lib.win-amd64-3.7\Crypto\SelfTest
copying lib\Crypto\SelfTest\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\common.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_AES.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_ARC2.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_ARC4.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_Blowfish.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_CAST.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_DES.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_DES3.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_pkcs1_15.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_pkcs1_oaep.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\test_XOR.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
copying lib\Crypto\SelfTest\Cipher\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Cipher
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\common.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_HMAC.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_MD2.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_MD4.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_MD5.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_RIPEMD.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_SHA.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_SHA224.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_SHA256.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_SHA384.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\test_SHA512.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
copying lib\Crypto\SelfTest\Hash\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Hash
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
copying lib\Crypto\SelfTest\Protocol\test_AllOrNothing.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
copying lib\Crypto\SelfTest\Protocol\test_chaffing.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
copying lib\Crypto\SelfTest\Protocol\test_KDF.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
copying lib\Crypto\SelfTest\Protocol\test_rfc1751.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
copying lib\Crypto\SelfTest\Protocol\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Protocol
creating build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
copying lib\Crypto\SelfTest\PublicKey\test_DSA.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
copying lib\Crypto\SelfTest\PublicKey\test_ElGamal.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
copying lib\Crypto\SelfTest\PublicKey\test_importKey.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
copying lib\Crypto\SelfTest\PublicKey\test_RSA.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
copying lib\Crypto\SelfTest\PublicKey\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\PublicKey
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Random
copying lib\Crypto\SelfTest\Random\test_random.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random
copying lib\Crypto\SelfTest\Random\test_rpoolcompat.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random
copying lib\Crypto\SelfTest\Random\test__UserFriendlyRNG.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random
copying lib\Crypto\SelfTest\Random\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Random\Fortuna
copying lib\Crypto\SelfTest\Random\Fortuna\test_FortunaAccumulator.py -> build\lib.win-amd64-3.7\Crypto\SelfTest
copying lib\Crypto\SelfTest\Random\Fortuna\test_FortunaGenerator.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\
copying lib\Crypto\SelfTest\Random\Fortuna\test_SHAd256.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\Fo
creating build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRNG
```

```
    copying lib\Crypto\SelfTest\Random\OSRNG\test_fallback.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRN
    copying lib\Crypto\SelfTest\Random\OSRNG\test_generic.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRNG
    copying lib\Crypto\SelfTest\Random\OSRNG\test_nt.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRNG
    copying lib\Crypto\SelfTest\Random\OSRNG\test_posix.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRNG
    copying lib\Crypto\SelfTest\Random\OSRNG\test_winrandom.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSR
    copying lib\Crypto\SelfTest\Random\OSRNG\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Random\OSRNG
    creating build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    copying lib\Crypto\SelfTest\Util\test_asn1.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    copying lib\Crypto\SelfTest\Util\test_Counter.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    copying lib\Crypto\SelfTest\Util\test_number.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    copying lib\Crypto\SelfTest\Util\test_winrandom.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    copying lib\Crypto\SelfTest\Util\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Util
    creating build\lib.win-amd64-3.7\Crypto\SelfTest\Signature
    copying lib\Crypto\SelfTest\Signature\test_pkcs1_15.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Signature
    copying lib\Crypto\SelfTest\Signature\test_pkcs1_pss.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Signature
    copying lib\Crypto\SelfTest\Signature\__init__.py -> build\lib.win-amd64-3.7\Crypto\SelfTest\Signature
    creating build\lib.win-amd64-3.7\Crypto\Protocol
    copying lib\Crypto\Protocol\AllOrNothing.py -> build\lib.win-amd64-3.7\Crypto\Protocol
    copying lib\Crypto\Protocol\Chaffing.py -> build\lib.win-amd64-3.7\Crypto\Protocol
    copying lib\Crypto\Protocol\KDF.py -> build\lib.win-amd64-3.7\Crypto\Protocol
    copying lib\Crypto\Protocol\__init__.py -> build\lib.win-amd64-3.7\Crypto\Protocol
    creating build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\DSA.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\ElGamal.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\pubkey.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\RSA.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\_DSA.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\_RSA.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\_slowmath.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    copying lib\Crypto\PublicKey\__init__.py -> build\lib.win-amd64-3.7\Crypto\PublicKey
    creating build\lib.win-amd64-3.7\Crypto\Signature
    copying lib\Crypto\Signature\PKCS1_PSS.py -> build\lib.win-amd64-3.7\Crypto\Signature
    copying lib\Crypto\Signature\PKCS1_v1_5.py -> build\lib.win-amd64-3.7\Crypto\Signature
    copying lib\Crypto\Signature\__init__.py -> build\lib.win-amd64-3.7\Crypto\Signature
    Skipping optional fixer: buffer
    Skipping optional fixer: idioms
    Skipping optional fixer: set_literal
    Skipping optional fixer: ws_comma
    running build_ext
    warning: GMP or MPIR library not found; Not building Crypto.PublicKey._fastmath.
    building 'Crypto.Random.OSRNG.winrandom' extension
    creating build\temp.win-amd64-3.7
    creating build\temp.win-amd64-3.7\Release
    creating build\temp.win-amd64-3.7\Release\src
    D:\Visual Studio\Microsoft Visual Studio\Professional\VC\Tools\MSVC\14.23.28105\bin\HostX86\x64\cl.exe /c /nolog
    winrand.c
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(27): error C2061: 语法错误: 标识符“intmax_t”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(28): error C2061: 语法错误: 标识符“rem”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(28): error C2059: 语法错误:“;”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(29): error C2059: 语法错误:“}”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(31): error C2061: 语法错误: 标识符“imaxdiv_t”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(31): error C2059: 语法错误:“;”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(41): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(42): error C2146: 语法错误: 缺少“)”(在标识符“_Number”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(42): error C2061: 语法错误: 标识符“_Number”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(42): error C2059: 语法错误:“;”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(43): error C2059: 语法错误:“)”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(46): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(47): error C2146: 语法错误: 缺少“)”(在标识符“_Numerator”的前
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(47): error C2061: 语法错误: 标识符“_Numerator”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(47): error C2059: 语法错误:“;”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(47): error C2059: 语法错误:“,”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(49): error C2059: 语法错误:“)”
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(51): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(57): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(64): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(70): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(77): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(83): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(90): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    D:\Windows Kits\10\include\10.0.18362.0\ucrt\inttypes.h(96): error C2143: 语法错误: 缺少“{”(在“__cdecl”的前面)
    error: command 'D:\\Visual Studio\\Microsoft Visual Studio\\Professional\\VC\\Tools\\MSVC\\14.23.28105\\bin\\Hos
    ----------------------------------------
ERROR: Command errored out with exit status 1: 'c:\users\lenovo\appdata\local\programs\python\python37\python.exe' -
```

听从网上建议,用pycryptodome代替pycrypto,使用pip安装成功

然后再使用pyinstaller.exe -key xxxx -F hellow.py

加密打包成功,无报错,但在运行hellow.exe时,出现了错误,如下

```
Traceback (most recent call last):
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\loader\pyimod03_importers
    return self._pyz_archive.extract(fullname)[1]
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\loader\pyimod02_archive.p
    obj = zlib.decompress(obj)
zlib.error: Error -3 while decompressing data: incorrect header check

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "site-packages\PyInstaller\loader\pyiboot01_bootstrap.py", line 25, in <module>
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\loader\pyimod03_importers
    bytecode = self.get_code(spec.loader_state)
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\loader\pyimod03_importers
    raise ImportError('Loader FrozenImporter cannot handle module ' + fullname)
ImportError: Loader FrozenImporter cannot handle module os
[13708] Failed to execute script pyiboot01_bootstrap
```

初看很像是requests库数据压缩问题,后经研究发现并不是requests库问题,问题是出在了pycryptodome库上.通过一番寻找,发现了一个解决这个问题的博客

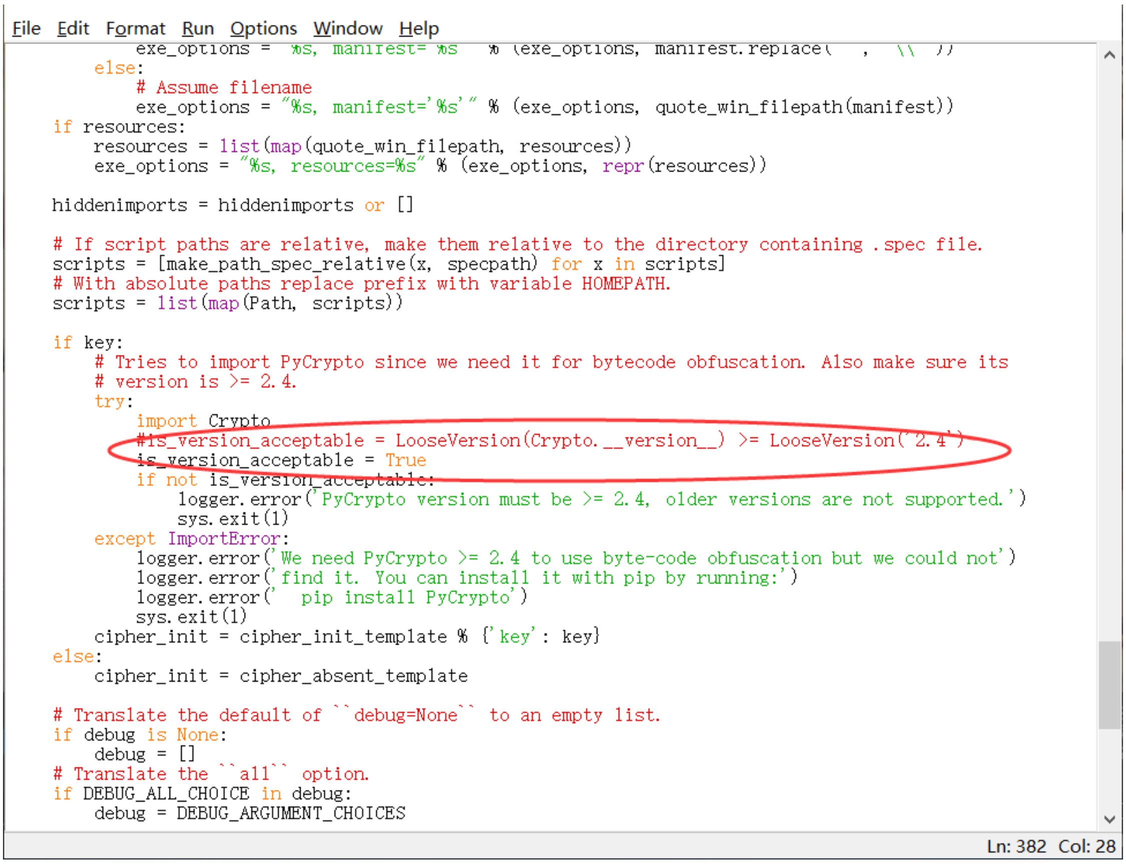
(参考博客:https://blog.csdn.net/qq_16166591/article/details/100624505)

pip uninstall pycryptodome,卸载pycryptodome按照该博客安装了pycrypto,安装pycrypto成功后执行pyinstaller.exe -key xxxx -F hellow.py

发现仍然有报错:

```
78 INFO: PyInstaller: 3.4
78 INFO: Python: 3.7.4
79 INFO: Platform: Windows-10-10.0.17763-SP0
Traceback (most recent call last):
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\runpy.py", line 193, in _run_module_as_main
    "__main__", mod_spec)
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\runpy.py", line 85, in _run_code
    exec(code, run_globals)
  File "C:\Users\xxxxxx\AppData\Local\Programs\Python\Python37\Scripts\pyinstaller.exe\__main__.py", line 7, in <modul
  File "c:\users\lenovo\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\__main__.py", line 109, i
    spec_file = run_makespec(**vars(args))
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\__main__.py", line 56, ir
    spec_file = PyInstaller.building.makespec.main(filenames, **opts)
  File "c:\users\xxxxxx\appdata\local\programs\python\python37\lib\site-packages\PyInstaller\building\makespec.py", li
    is_version_acceptable = LooseVersion(Crypto.__version__) >= LooseVersion('2.4')
AttributeError: module 'Crypto' has no attribute '__version__'
```

报错信息提示大意是没有发现Crypto的版本号,直接采用一个暴力一点的办法:按照错误提示,在python37\lib\site-packages\PyInstaller\building\目录中找到makespec.py文件,修改源码,将is_version_acceptable = LooseVersion(Crypto.__version__) >= LooseVersion('2.4')直接改为is_version_acceptable = True,如图所示



```
            exe_options = "%s, manifest=%s" % (exe_options, manifest.replace("'", "\\'"))
        else:
            # Assume filename
            exe_options = "%s, manifest='%s'" % (exe_options, quote_win_filepath(manifest))
    if resources:
        resources = list(map(quote_win_filepath, resources))
        exe_options = "%s, resources=%s" % (exe_options, repr(resources))

    hiddenimports = hiddenimports or []

    # If script paths are relative, make them relative to the directory containing .spec file.
    scripts = [make_path_spec_relative(x, specpath) for x in scripts]
    # With absolute paths replace prefix with variable HOMEPATH.
    scripts = list(map(Path, scripts))

    if key:
        # Tries to import PyCrypto since we need it for bytecode obfuscation. Also make sure its
        # version is >= 2.4.
        try:
            import Crypto
            #is_version_acceptable = LooseVersion(Crypto.__version__) >= LooseVersion('2.4')
            is_version_acceptable = True
            if not is_version_acceptable:
                logger.error('PyCrypto version must be >= 2.4, older versions are not supported.')
                sys.exit(1)
        except ImportError:
            logger.error('We need PyCrypto >= 2.4 to use byte-code obfuscation but we could not')
            logger.error('find it. You can install it with pip by running:')
            logger.error('  pip install PyCrypto')
            sys.exit(1)
        cipher_init = cipher_init_template % {'key': key}
    else:
        cipher_init = cipher_absent_template

    # Translate the default of ``debug=None`` to an empty list.
    if debug is None:
        debug = []
    # Translate the ``all`` option.
    if DEBUG_ALL_CHOICE in debug:
        debug = DEBUG_ARGUMENT_CHOICES
```

保存退出后再次执行pyinstaller.exe -key xxxx -F hellow.py,没有报错.运行hellow.exe,exe文件正常运行.

至此,这个坑总算是解决了.

好文要顶 关注我 收藏该文

兵卒87
关注 - 0
粉丝 - 0

+加关注

0 0

posted @ 2019-11-14 16:32 兵卒87 阅读(2834) 评论(0) 编辑 收藏

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册 ， 访问 网站首页。

相关博文：

· Python打包方法——PyinstallerCentOS下踩坑记录

· pyinstaller打包pyqt文件

· pyinstaller打包exe遇到的坑

· Pyqt5python2pyinstaller打包爬坑

· Python打包方法——Pyinstaller

» 更多推荐...