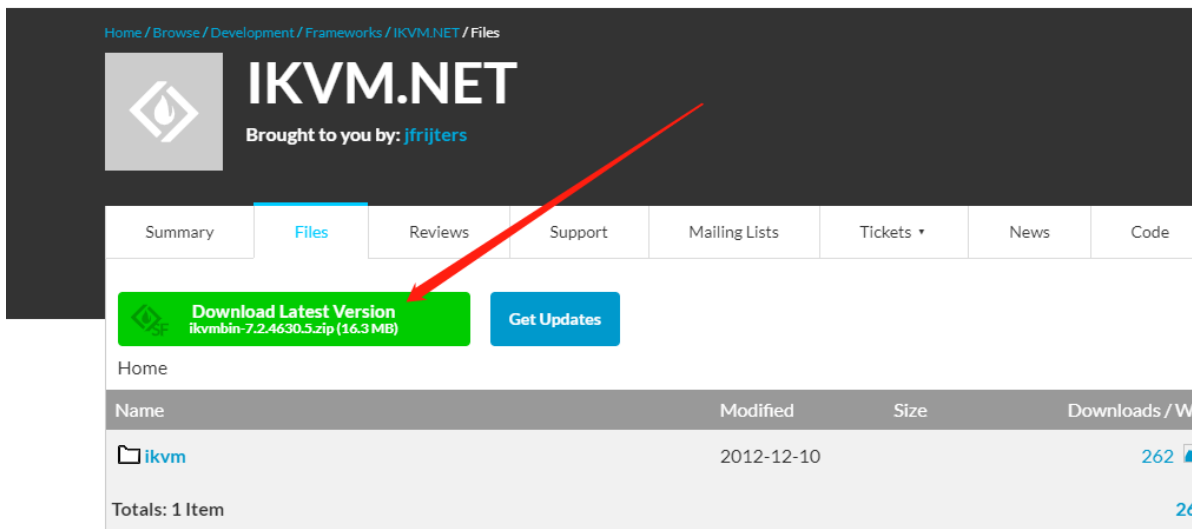
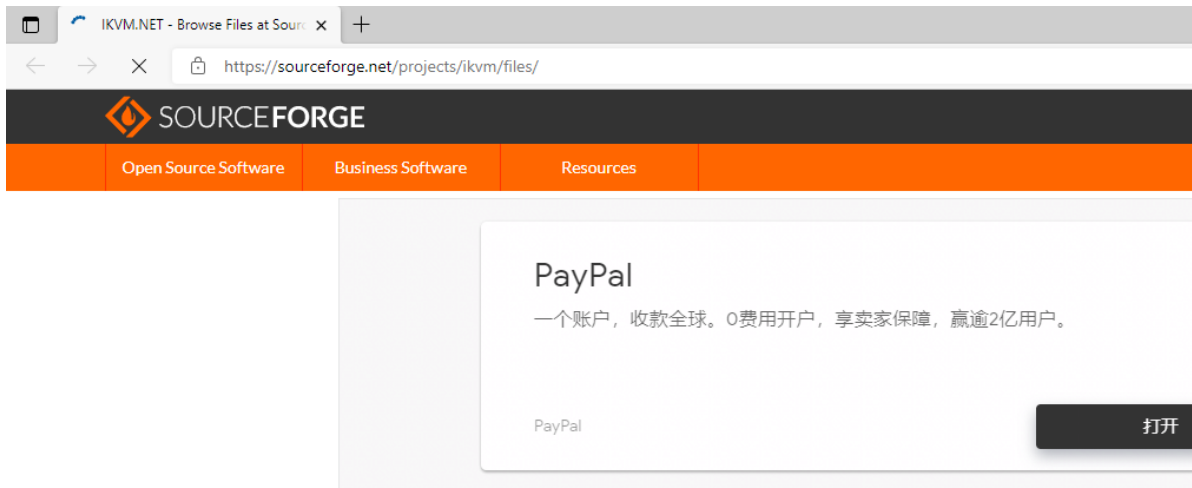


参考网址[在C#中调用Java生成的jar包文件的方法 - J.Y - 博客园 \(cnblogs.com\)](http://cnblogs.com/j-y/)与[Java将一个文件夹打成jar包 - cwone - 博客园 \(cnblogs.com\)](http://cnblogs.com/cwone/)

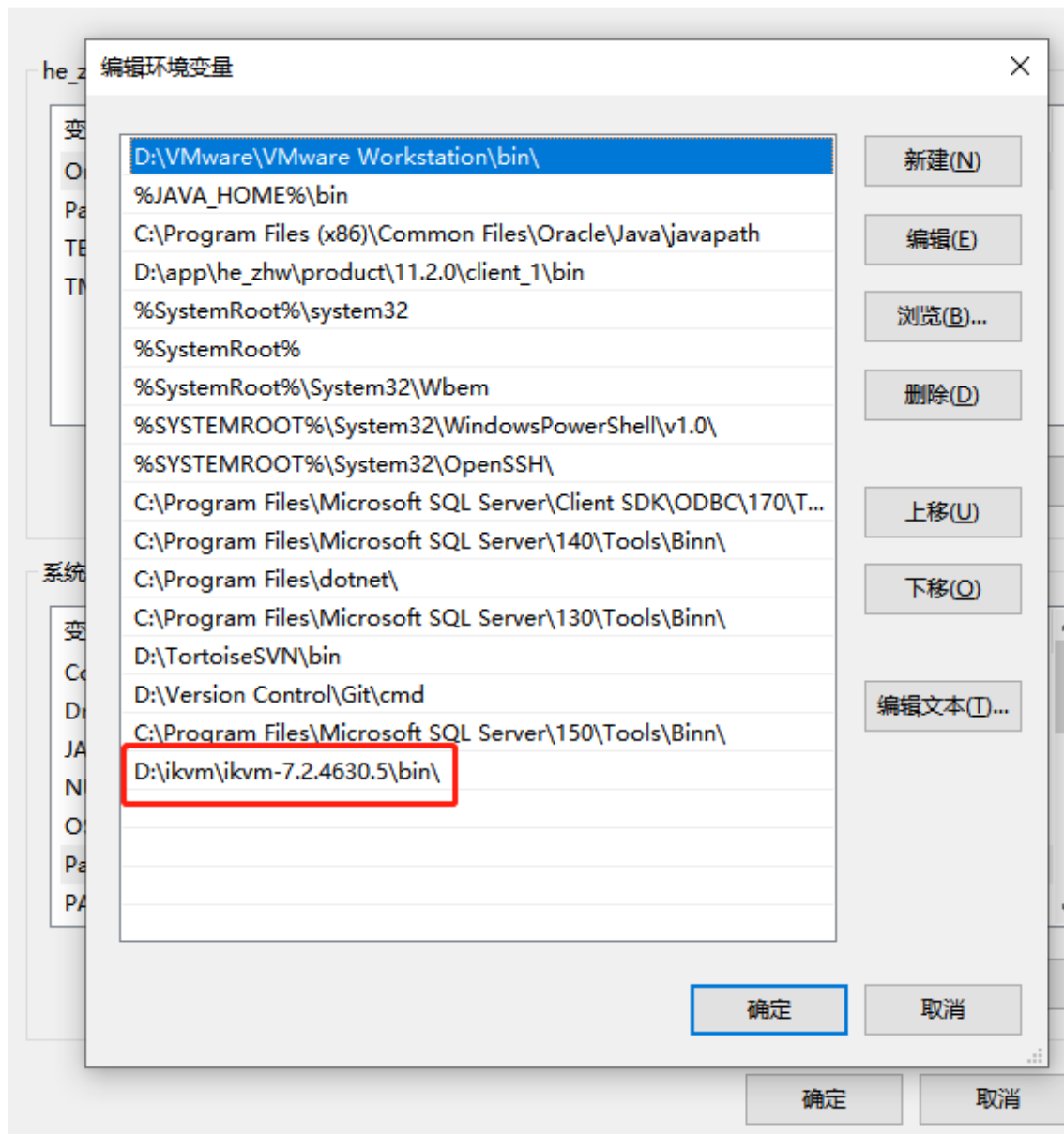
一、下载并安装IKVM.NET组件

下载网址<https://sourceforge.net/projects/ikvm/files/>

- 本例子下载的是(7.2.4630.5版本)

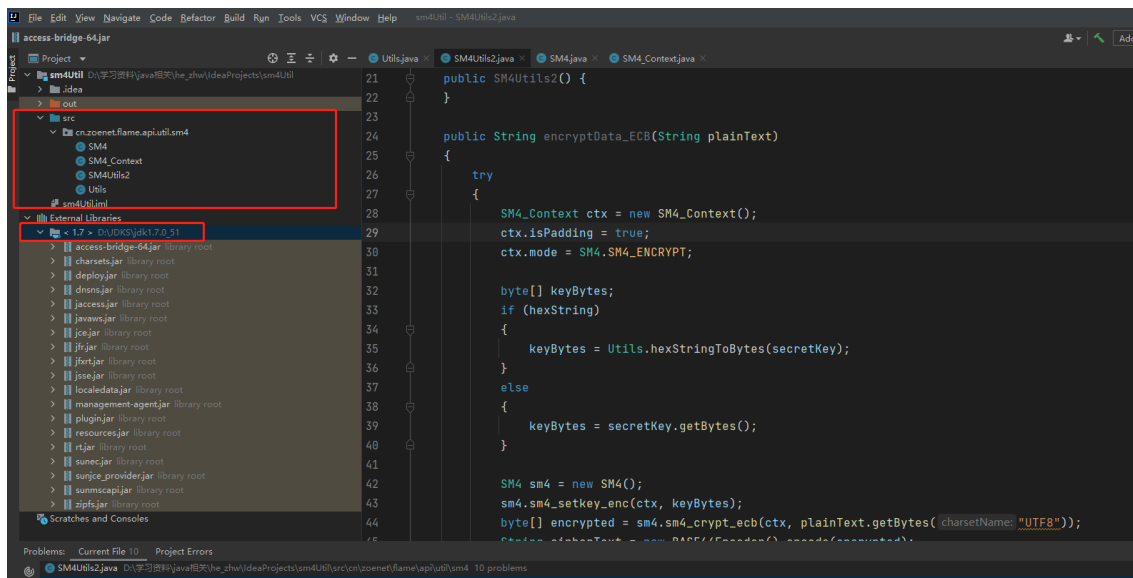


- 将下载好的压缩包解压,本例子解压到了 D:\ikvm 目录下
- 配置环境变量



二、根据jar包生成dll

- 特别注意:编译环境要在jdk1.7及以下, 亲测jdk1.8编译生成的dll引用后无法使用*(本例子使用的jdk版本为jdk1.7.0_51)



- 生成jar包(有安装idea或MyEclipse的可以直接用开发工具生成jar包)

1. 在桌面创建一个test文件夹
2. 在test文件夹下创建classes文件夹，并将编译好的class文件放进去

C:\Users\he_zhw\Desktop\test\classes\cn\zoenet\flame\api\util\sm4				
名称	修改日期	类型	大小	
SM4.class	2021/9/19 22:40	CLASS 文件	8 KB	
SM4_Context.class	2021/9/19 22:40	CLASS 文件	1 KB	
SM4Utils2.class	2021/9/19 22:40	CLASS 文件	3 KB	
Utils.class	2021/9/19 22:40	CLASS 文件	10 KB	

3. 在test文件夹下创建META-INF文件夹，并在里面创建MANIFEST.MF文件(空文件就行)

C:\Users\he_zhw\Desktop\test\META-INF				
名称	修改日期	类型	大小	
MANIFEST.MF	2021/9/19 20:16	MF 文件	0 KB	

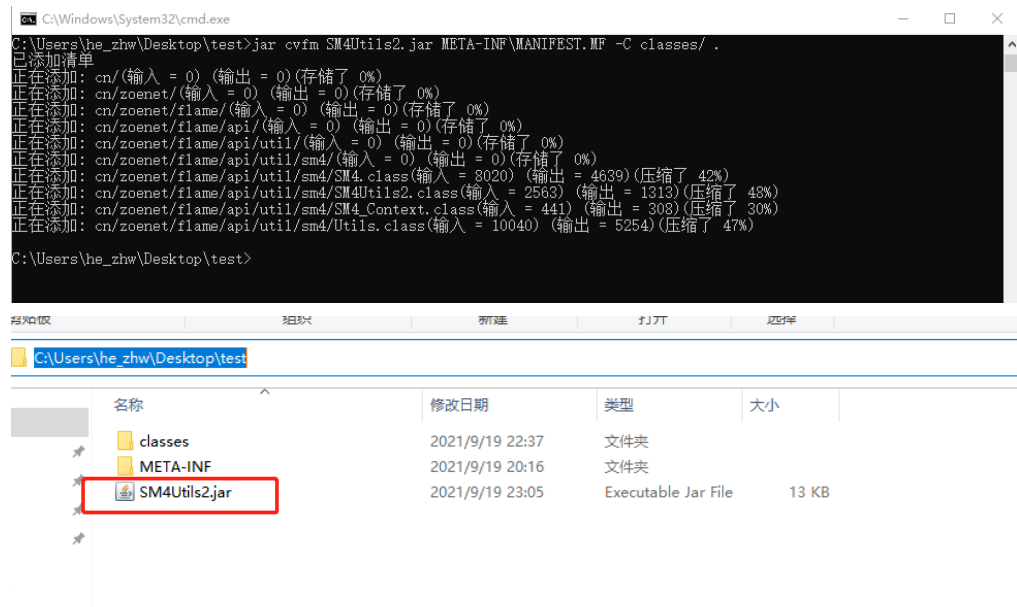
4. 生成jar包

- 在test目录下输入cmd命令

cmd				
cmd				
搜索 "cmd"				
classes	2021/9/19 22:37	文件夹		
META-INF	2021/9/19 20:16	文件夹		

- 执行命令生成jar包

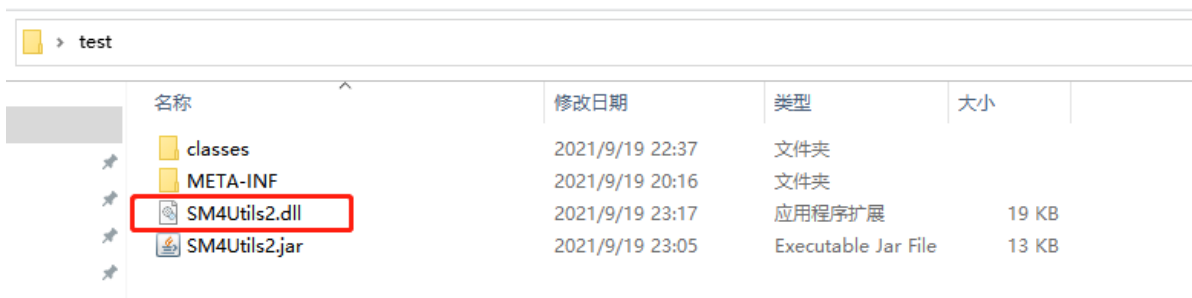
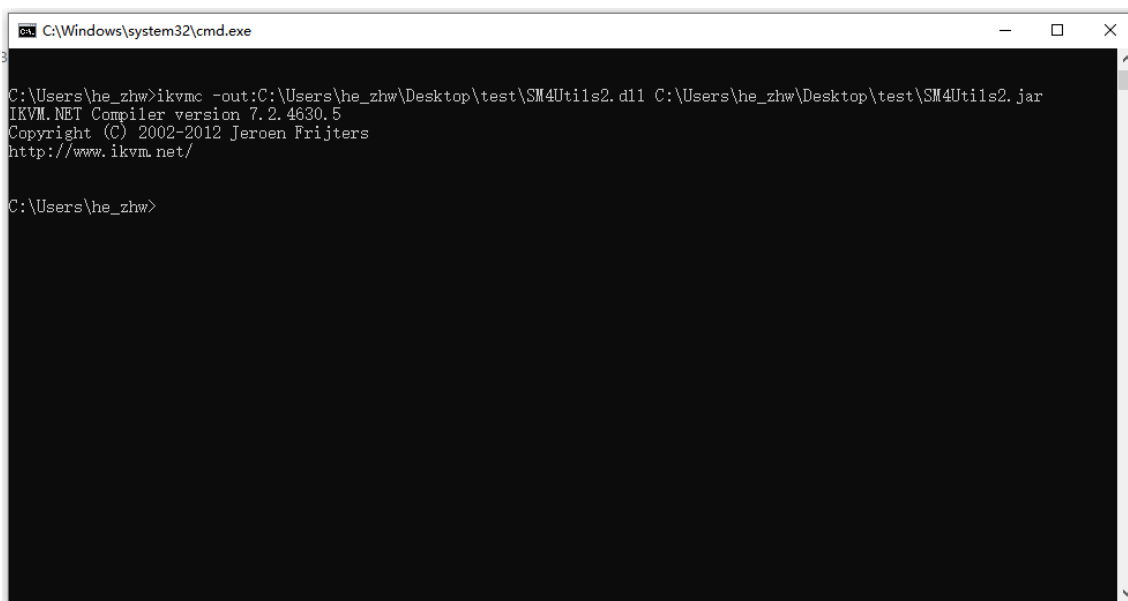
```
jar cvfm SM4Utils2.jar META-INF\MANIFEST.MF -C classes/ .
```



- 生成dll

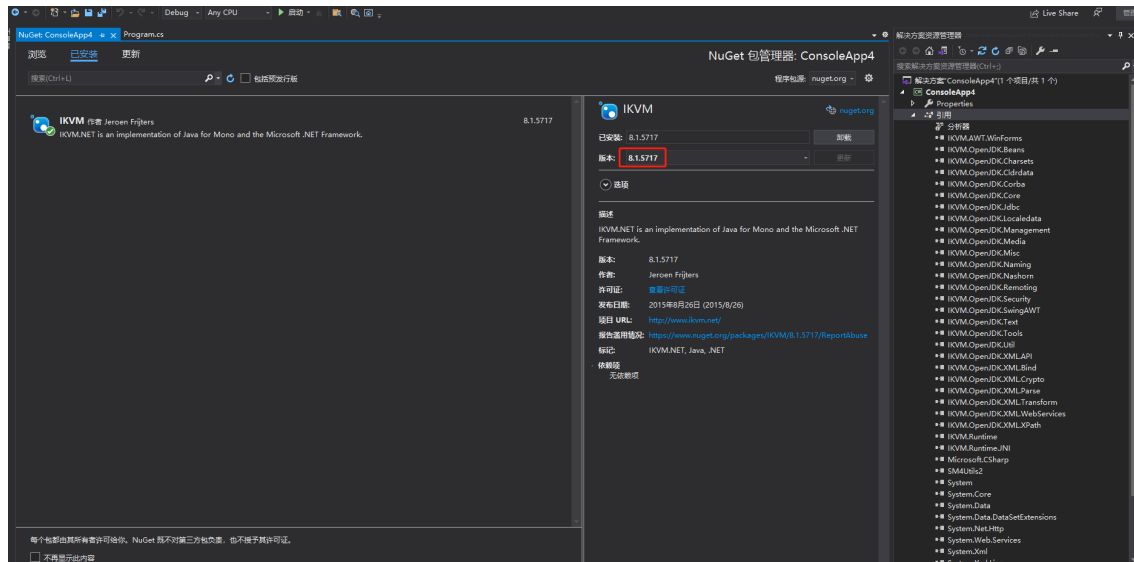
cmd执行以下命令(前提是之前已经配置号环境变量, 否则要切换到ikvmc的bin目录下执行)

```
ikvmc -out:C:\Users\he_zhw\Desktop\test\SM4Utils2.dll
C:\Users\he_zhw\Desktop\test\SM4Utils2.jar
```



三、引用dll并测试

- Nugget引用ikvm的dll包(本例使用的是8.1.5717版本,引用过低的版本会报错)



- 引用生成的SM4Utils2.dll

测试代码

```
using cn.zoenet.flame.api.util.sm4;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Web.Services.Description;

namespace ConsoleApp4
{
    class Program
    {
        static void Main(string[] args)
        {
            string plainText = File.ReadAllText("plainText.txt");
            SM4Utils2 sm4Utils2 = new SM4Utils2();
            sm4Utils2.secretKey = "1234567890123456";
            sm4Utils2.hexString = false;
            string cipherText = sm4Utils2.encryptData_ECB(plainText);
            string v = sm4Utils2.decryptData_ECB(cipherText);
            Console.WriteLine($"密文{cipherText}");
            Console.WriteLine($"解密后的原文{v}");
            Console.ReadKey();
        }
    }
}
```