Class加密/解密（了解）

可以对自己的class文件 进行加密操作

然后自定义classloader的 加载器，加载到对应的字节，然后将这些字节通过对应的解密方式解密 然后再defineClass 到内容 并返回

```
public class T007_MSBClassLoaderWithEncription extends ClassLoader {
    public static int seed = 0B10110110;
    @Override
    protected Class<?> findClass(String name) throws ClassNotFoundException {
        File f = new File("c:/test/", name.replace('.', '/').concat(".msbclass"));

        try {
            FileInputStream fis = new FileInputStream(f);
            ByteArrayOutputStream baos = new ByteArrayOutputStream();
            int b = 0;

            while ((b=fis.read()) !=0) {
                baos.write(b ^ seed);
            }

            byte[] bytes = baos.toByteArray();
            baos.close();
            fis.close();//可以写的更加严谨

            return defineClass(name, bytes, 0, bytes.length);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return super.findClass(name); //throws ClassNotFoundException
    }

    public static void main(String[] args) throws Exception {

        encFile("com.mashibing.jvm.hello");

        ClassLoader l = new T007_MSBClassLoaderWithEncription();
        Class clazz = l.loadClass("com.mashibing.jvm.Hello");
        Hello h = (Hello)clazz.newInstance();
        h.m();

        System.out.println(l.getClass().getClassLoader());
        System.out.println(l.getParent());
    }

    private static void encFile(String name) throws Exception {
        File f = new File("c:/test/", name.replace('.', '/').concat(".class"));
        FileInputStream fis = new FileInputStream(f);
        FileOutputStream fos = new FileOutputStream(new File("c:/test/", name.replaceAll(".", "/").concat(".msbclass")));
        int b = 0;
        while((b = fis.read()) != -1) {
            fos.write(b ^ seed);  // 根据seed 做异或操作     解密的时候再异或
        }
        fis.close();
        fos.close();
    }
}
```