

How to crack protected obfuscated Spigot plugins. Please read through it carefully

(I'm not a java developer, so I do not 100% understand everything. I just figured this out by messing with things!)

[Introduction \(you may skip all the reading and just download the programs you need and follow the screenshots if you want\)](#)

Hi! If you're reading this, you probably want to crack spigot plugins. No problem! This for spigot plugins that have had their code obfuscated, meaning that most of the code was encrypted to look like a jumbled mess so anyone that decompiles it doesn't mess with the code. Also, if you do mess with the code, it will give you `org.bukkit.plugin.InvalidPluginException: java.lang.ClassFormatError: StackMapTable format error: wrong attribute size` in the console. It's basically an obfuscation error saying whatever was deleted was depending on something else and fucked up the formatting which we don't know how to fix since it's obfuscated. So what do we do? Just edit one thing! And we can't just deobfuscate it. You'd have to download some programs and do other unnecessary stuff when this is a better easier way for everyone. ***This may work with unobfuscated plugins (there aren't that many out there though) but I haven't tested it yet. It may be different, but continue to read to find out.***

[Getting ready by downloading programs](#)

I know, I know I hate downloading programs too but that's what we're gonna do.

First, we're gonna need DirtyJOE. This is an editing program but we're not going to use this to edit. We need this to view methods in the main .class file of the plugin. I'll explain more below. Download dirtyJOE here: <http://dirty-joe.com/>

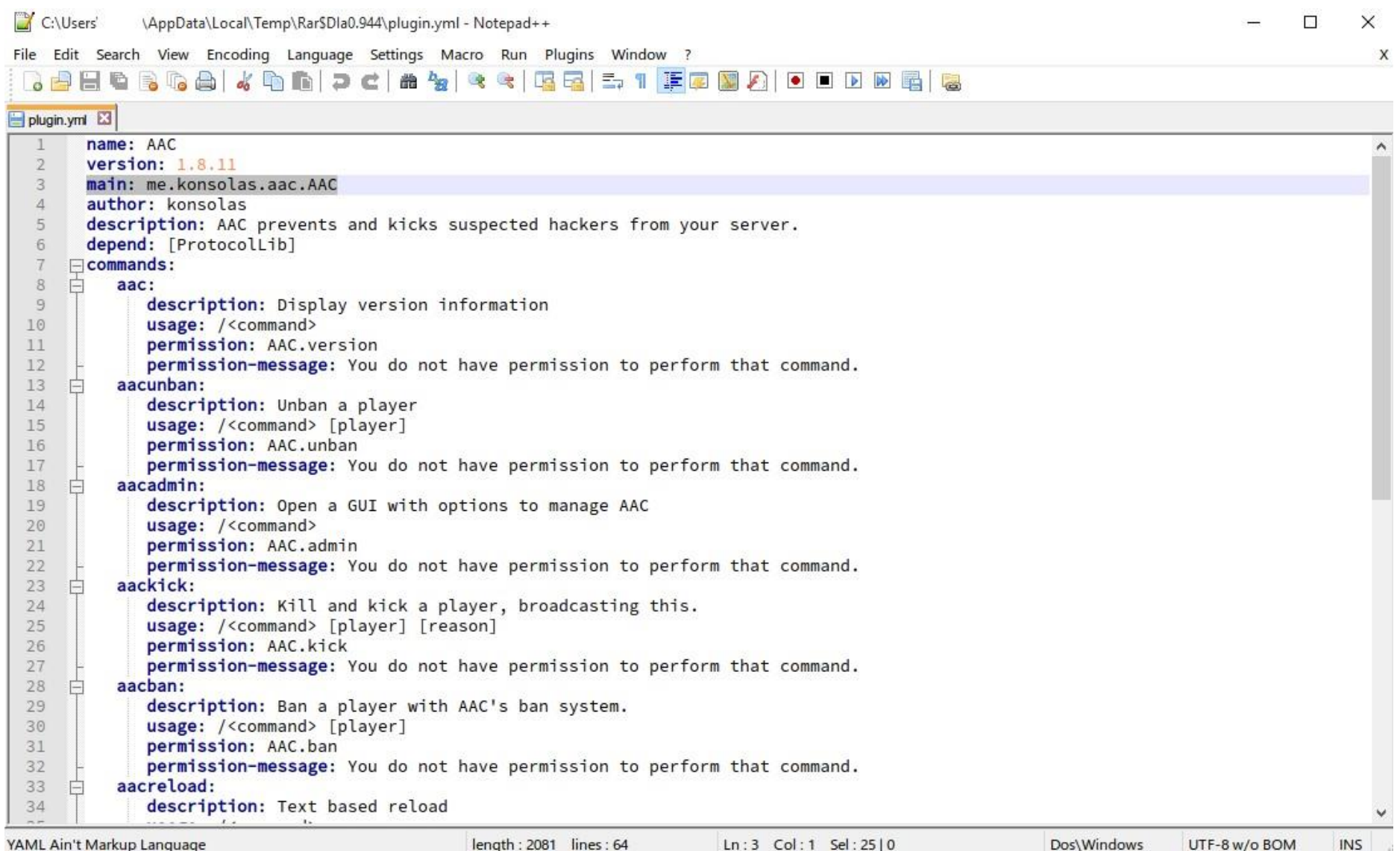
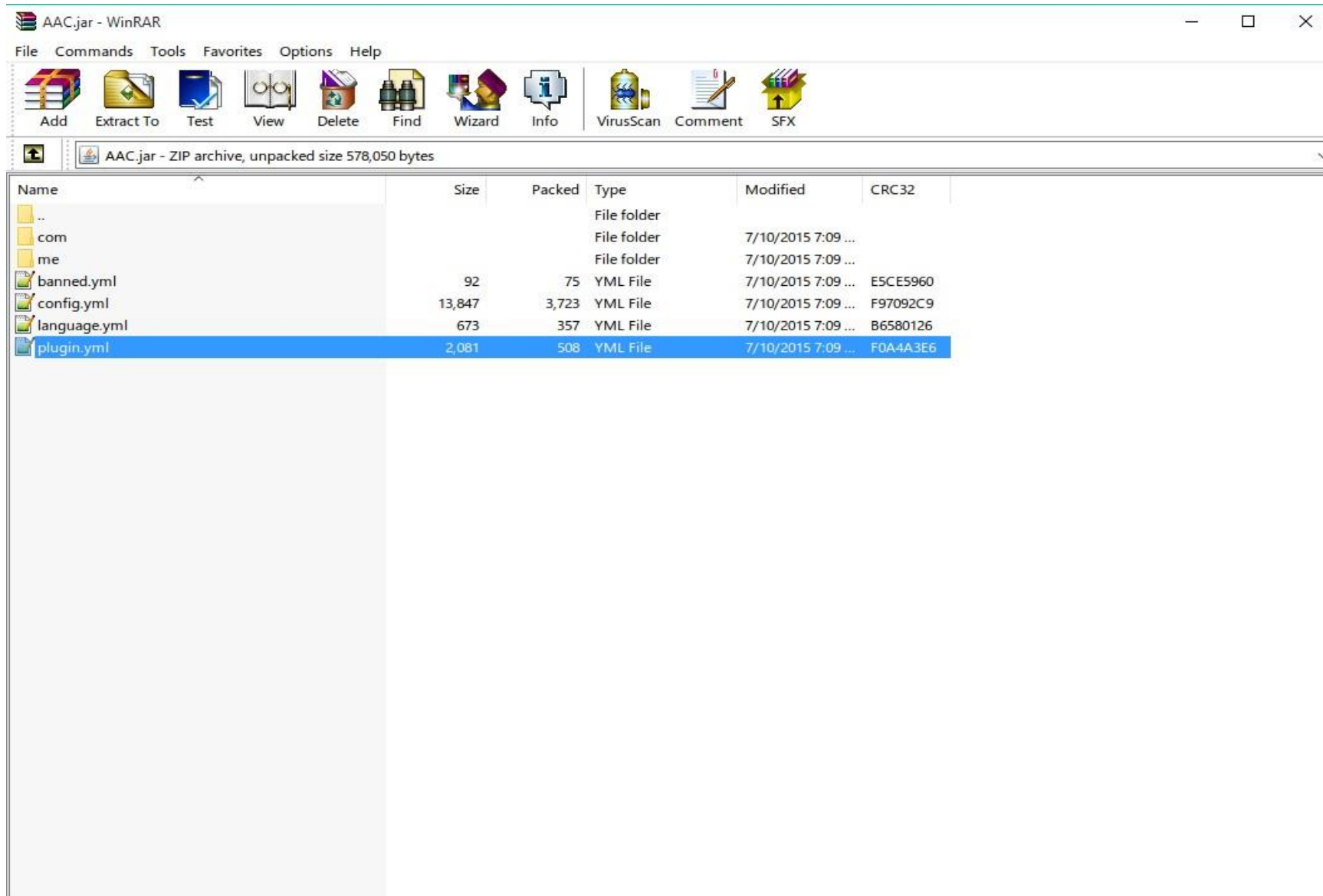
For this tutorial, our plugin will be AdvancedAntiCheat version 1.8.11 which was recently blocked due to piracy but I just cracked it. You can download it for yourself here: <http://adf.ly/1Mcr7U>

Next you need JBE which you can get here: <http://set.ee/jbe/>

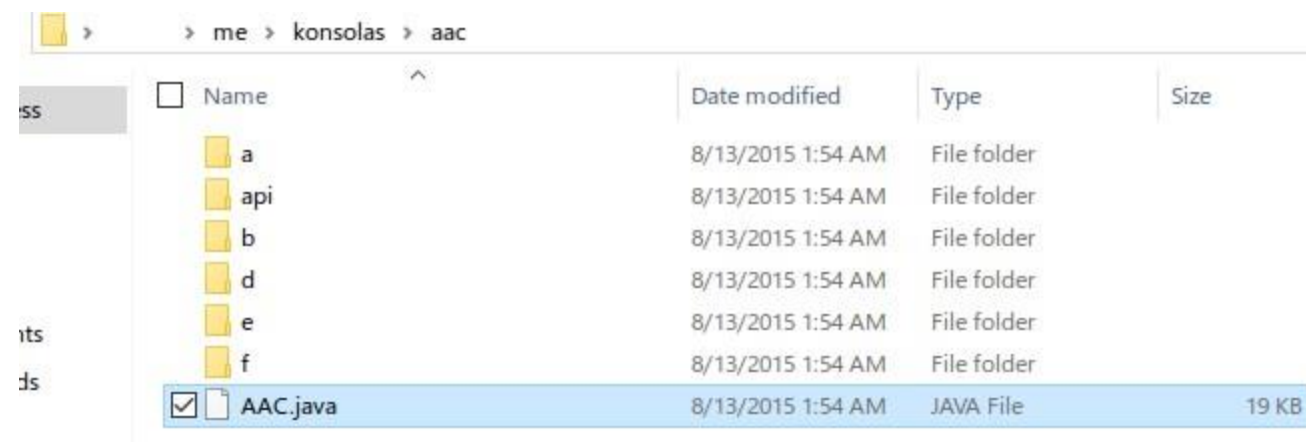
With those downloaded, extract all the rar/zip files and let's begin!

[Extracting the main .class file](#)

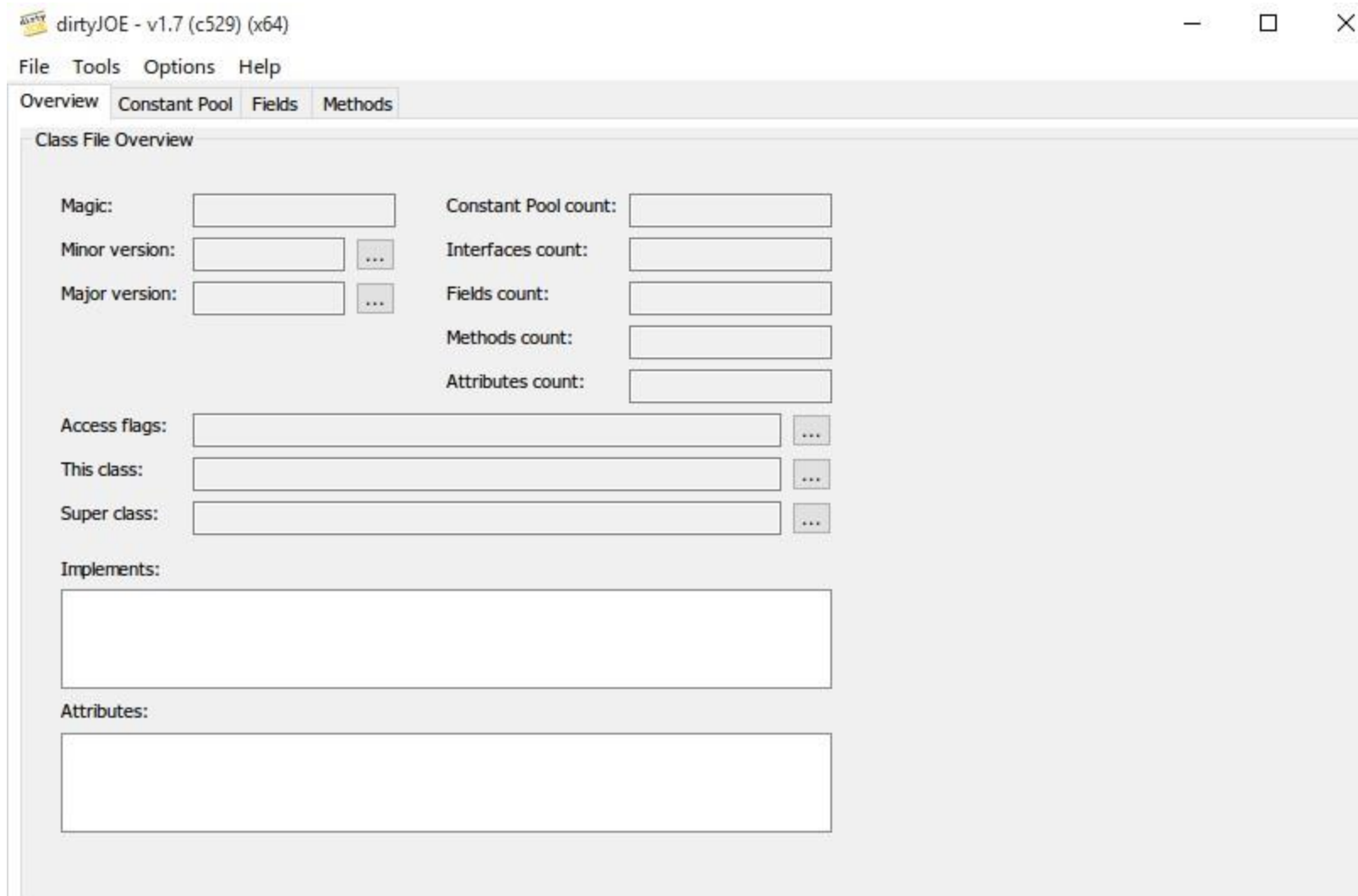
All java applications have a main class file. You can find out what it is by opening your .jar file with WinRAR and opening plugin .yml.



So as you can see, our main class file is named AAC judging by 'main: me.konsolas.aac.AAC' where me.konsolas.aac are the directories in the .jar file. For this plugin, our main class file will be located in me/konsolas/aac/ and our main class file is AAC.class



Grab this file and extract it to your desktop or wherever you want. Now let's open dirtyJOE.



Click File>Open and open your .class file. For this tutorial, it's AAC.class. Once it's opened, go to Methods.

dirtyJOE - v1.7 (c529) (x64) - C:\Users\ Desktop\AAC.class

File Tools Options Help

Overview Constant Pool Fields Methods

<init>
onEnable
onDisable
a
b
a
c
a
<clinit>
loadConfig0

Info

Name: <init> Attributes: Code

Descriptor: ()V

Access Flags: public

Demangled name: public void <init>()

max_stack: 5
max_locals: 1
code_length: 54
attributes_count: 1
exception_table_length: 0

00000000 : aload_0
00000001 : invokespecial void org.bukkit.plugin.java.JavaPlugin.<init>(<>)
00000004 : aload_0
00000005 : aload_0
00000006 : putfield me.konsolas.aac.AAC me.konsolas.aac.AAC.a
00000007 : aload_0
0000000A : new java.text.SimpleDateFormat
0000000D : dup

Now for many plugins that I've tested on, they want to call the antipiracy when the server is first started, so this method is *onEnable*

Since this plugin is obfuscated and the creator is smart, they'd probably want to confuse us by renaming the antipiracy call something tacky and unsuspecting. For this plugin, they're calling it config. The first thing onEnable is calling is the method loadConfig0

dirtyJOE - v1.7 (c529) (x64) - C:\Users\ Desktop\AAC.class

File Tools Options Help

Overview Constant Pool Fields Methods

<init>
onEnable
onDisable
a
b
a
c
a
<clinit>
loadConfig0

Info

Name: onEnable Attributes: Code

Descriptor: ()V

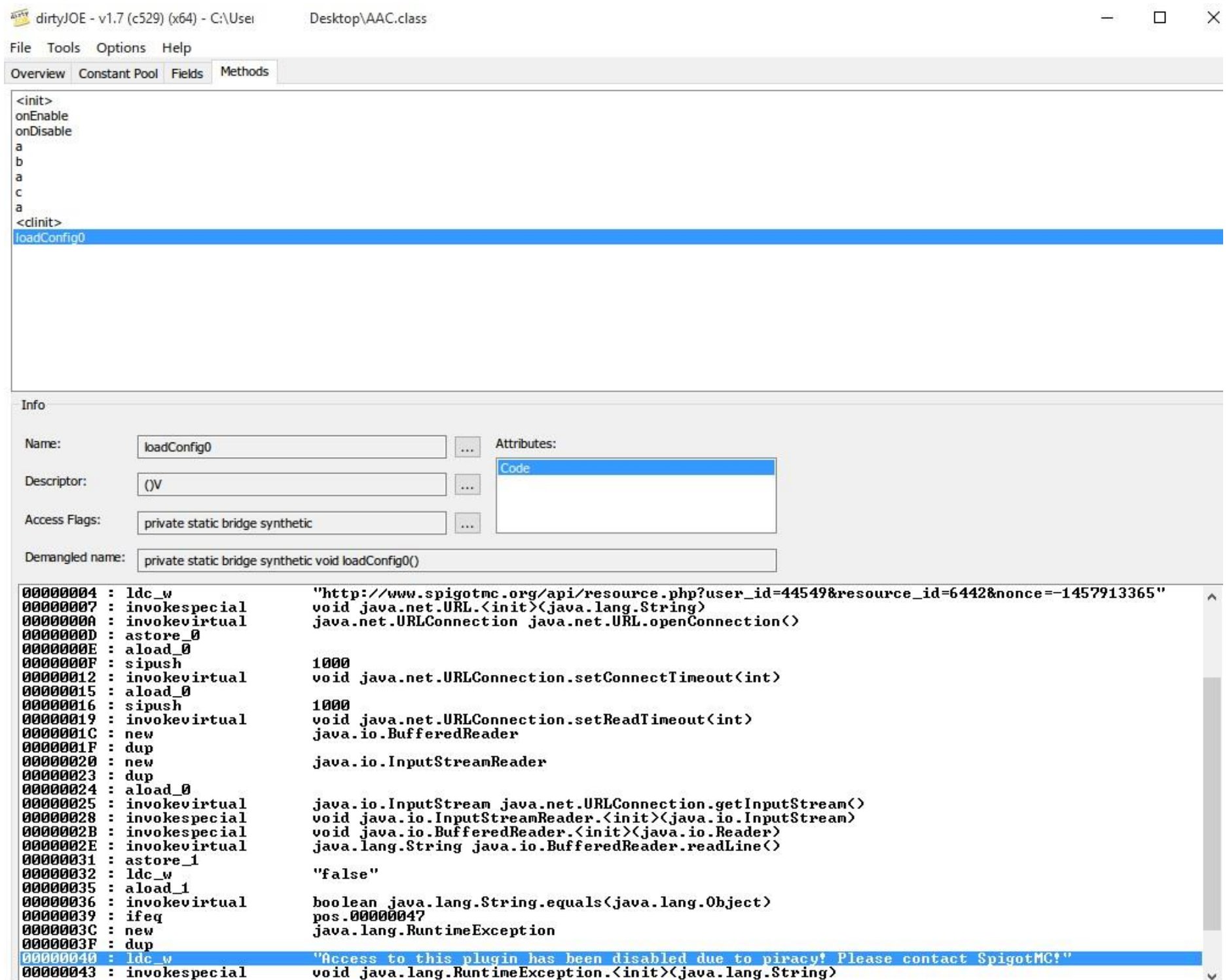
Access Flags: public

Demangled name: public void onEnable()

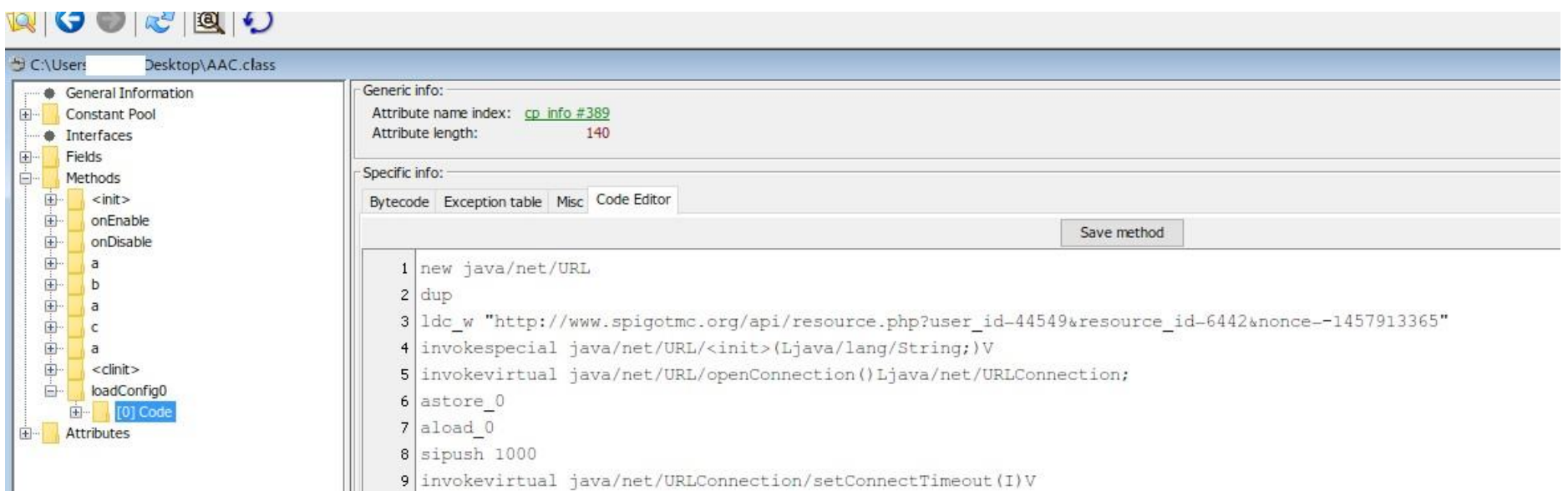
max_stack: 7
max_locals: 1
code_length: 171
attributes_count: 2
exception_table_length: 4

00000000 : invokestatic void me.konsolas.aac.AAC.loadConfig0(<>)
00000003 : invokestatic java.util.logging.Logger org.bukkit.Bukkit.getLogger(<>)
00000006 : getstatic java.lang.String[] me.konsolas.aac.AAC.q
00000009 : iconst_3
0000000A : aaload
0000000B : invokevirtual void java.util.logging.Logger.info(java.lang.String)
0000000E : aload_0
0000000F : new me.konsolas.aac.a.n

Let's have a look at loadConfig0



Well I'll be damned! They thought they would be sneaky huh? Well loadConfig is the entire public void code given to developers who want to add antipiracy to their premium plugins. Now we know where this is being called so now we need to open our .class file in JBE to edit this correctly. Open **jbe.sh** and open your .class file.



Now that loadConfig method is open, click the plus icon next to loadConfig and another folder named [0] Code appears like the screenshot. Click on it, then click Code Editor. This is where it gets even more fun!

We have to edit the spigotmc URL. Of course the resource ID will be different on every plugin but that doesn't matter! All we have to do is replace one thing. This is it for us

ldc_w http://www.spigotmc.org/api/resource.php?user_id=44549&resource_id=6442&nonce=-1457913365

So let's just go ahead and replace one letter. So what I did is I replaced user_id to just say user_i4. So now it should say

ldc_w http://www.spigotmc.org/api/resource.php?user_i4=44549&resource_id=6442&nonce=-1457913365

By going on the **original** URL, it just shows 'false' which tells the plugin the license is no longer valid and the antipiracy is turned on. But our edited URL is null and will only show a blank screen, which returns nothing, and the plugin enables like normal! Remember, you can edit anything in this url. But do not replace anything after spigotmc.org/api/ or you will get 404 errors in your console. Or you can replace the URL to anything (I haven't tried it yet) Nothing bad but it gets spammy. **Remember, you can edit anything, a letter or number and replace it!**

Now that you replaced your URL, save the method and exit JBE. Drag your main .class file to where you grabbed it. So for us, we're putting it back and replacing the original AAC.class file in me/konsolas/aac/ in our plugin.jar. Now once you did that, put your plugin into your plugins folder and it should start fine!

That's it!

If you are having trouble with this, or you cannot find the main .class file in your plugin, or you cannot find the method that calls the antipiracy public void or you have any other problems, drop me a *PM* on leakforums.net and I'll do it for you!

Link to my profile, teet1: <https://leakforums.net/user-525379>