

Title: How to solve could not find or load main class error with spring boot?

Post Body:

I created a Maven project for Spring Boot. I have a lot of Spring dependencies and one main class:

```
package com.vastserver; import org.springframework.boot.SpringApplication; import org.springframework.boot.autoconfigure.Spr
```

The folder structure of `src` directory is:

```
. ├── main ├── java ── ─── com ── ─── vastserver ── ─── MyArtifactApplication.java ───
```

In my `pom.xml` I use `maven-assembly-plugin` in order to build my project in a standalone `.jar` file. Even though I triple checked that the directory structure and main class file appear correctly in the `pom.xml` I keep getting the error: `Error: Could not find or load main class com.vastserver.MyArtifactApplication` when I run `mvn package` and then `java -cp target/vast-ad-server-artifactId-1.0-SNAPSHOT-jar-with-dependencies.jar com.vastserver.MyArtifactApplication` or `mvn exec:exec`. The main class does work if I run it from IntelliJ so I know the code is not the problem but rather Maven build settings. I lost at where my problem could be.

My `pom.xml` looks as follows:

```
<?xml version='1.0' encoding='UTF-8'?> <project xmlns='http://maven.apache.org/POM/4.0.0' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance' xsi:schemaLocation='http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd'>
```

Accepted Answer:

I think you should check the directory structure of the artifact that was built by maven. Usually, spring boot artifacts are prepared by a special spring boot plugin and not by a maven assembly plugin.

Although it shares the `.jar` suffix, it's not really a jar, it has special classloader to load classes from `BOOT-INF/lib` folder.

I've already provided a detailed answer on what happens exactly when the spring boot application starts [here](#) but bottom line if you use assembly plugin you'll have to prepare both manifest file and a fairly complicated structure of folders. Frankly, I think you should use spring boot plugin as a first resort to build spring boot applications.

Highest Rated Answer:

For me: Maven - Update Project worked.