

Title: Docker image build failed on Mac M1 chip

Post Body:

I'm trying to build a simple docker image, inside a maven project, adding the image build as part of the maven build process:

```
<build>
    <finalName>my-api</finalName>
    <plugins>
        <plugin>
            <groupId>org.springframework
FROM openjdk:8-jdk-alpine VOLUME /tmp EXPOSE 8080 ADD target/*.jar app.jar ENTRYPOINT [ "sh", "-c", "java -jar /app.jar" ]
```

But it fails, always get the same error trace, no matter which image I use, the error persists.

Error:

```
Caused by: com.spotify.docker.client.shaded.javax.ws.rs.ProcessingException: java.lang.UnsatisfiedLinkError: could not load FFI provider
jnr.ffi.provider.jffi.Provider
```

```
Caused by: java.lang.UnsatisfiedLinkError: java.lang.UnsatisfiedLinkError:
/private/var/folders/hz/rgppp8250rsdp86kf_tfjvqw0000gp/T/jffi8502916075702391528.dylib:
dlopen(/private/var/folders/hz/rgppp8250rsdp86kf_tfjvqw0000gp/T/jffi8502916075702391528.dylib, 0x0001): tried:
'/private/var/folders/hz/rgppp8250rsdp86kf_tfjvqw0000gp/T/jffi8502916075702391528.dylib' (fat file, but missing compatible architecture (have
'i386,x86_64', need 'arm64e')), '/usr/lib/jffi8502916075702391528.dylib' (no such file)
```

Other images I tried:

- openjdk:13-alpine3.9
- openjdk:8-jre-alpine3.9
- azul/zulu-openjdk-alpine:17.0.2-17.32.13-arm64

My java version: openjdk version "11.0.13" 2021-10-19 LTS

My Docker version: Docker version 20.10.11, build dea9396

Thanks in advance.

Accepted Answer:

It looks like the `dockerfile-maven-plugin` uses a runtime based on x86 architecture and won't run on Apple M1 (Arm).
The plugin is now inactive so you should try something else, for example the [fabric8-maven-plugin](#)

```
<plugin>
    <groupId>io.fabric8</groupId>
    <artifactId>docker-maven-plugin</artifactId>
```

Highest Rated Answer:

Both of the spotify docker maven plugins are no longer maintained. They need to upgrade their dependency to a version that supports aarch64.

In our case there was significant refactoring needed to move to fabric8's plugin or to use maven exec so we wanted to continue to use spotify plugin.

Fortunately, you can force the plugin to use a particular dependency by adding a `<dependencies>` section to your plugin section.

```
<plugin>
    <groupId>com.spotify</groupId>
    <artifactId>dockerfile-maven-p
```