Title: How to upgrade spring framework version in spring boot

Post Body:

I am using spring-boot 2.3.3.RELEASE with the according spring-boot-starter-parent in maven.

```
<parent>    <groupId>org.springframework.boot</groupId>    <artifactId>spring-boot-starter-parent</artifactId>    <version>2.3
```

Due to the spring4shell CVE I wanted to upgrade the spring-framework to 5.2.20.RELEASE instead of the already included 5.2.8.RELEASE. I tried overriding the `spring-framework.version` property from spring-boot-dependencies.

```
<spring-framework.version>5.2.20.RELEASE</spring-framework.version>
```

But it did not work. I also looked up the spring-boot-starter-web-2.3.3.RELEASE.pom and it has the spring-web dependency hardcoded to 5.2.8.RELEASE.

Are there any other ways of upgrading the spring-framework version in spring-boot besides adding all the new versions as dependencies to the `dependencyManagement` section? Thx

Full POM:

```
<?xml version="1.0"?> <project xmlns="http://maven.apache.org/POM/4.0.0"    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instan
```

EDIT: This is a part of mvn dependency:tree:

```
+- org.springframework.boot:spring-boot-starter-webflux:jar:2.3.3.RELEASE:compile [INFO] │  +- org.springframework.boot:spring
```

If you have a look at the `spring-boot-starter-webflux-2.3.3.RELEASE.pom` which includes the problematic spring-web 5.2.8.RELEASE you will find that the spring version is hardcoded to 5.2.8.RELEASE. So setting the `spring.framework` property in maven will have no effect.

```
<dependency>        <groupId>org.springframework</groupId>        <artifactId>spring-web</artifactId>        <version>5.2.8.R
```

Output of mvn help:effective-pom:

```
<dependency>        <groupId>org.springframework</groupId>  <!-- org.springframework:spring-framework-bom:5.2.8.RELEASE, lin
```

Edit after Solution by @Inthai2002: I have additonally an internal lib pom imported in my pom.xml

```
<dependencyManagement>        <dependencies>        <dependency>        <groupId>internal</groupId>
```

and this internal lib has the spring-boot-dependencies pom directly imported which leads to the fact that spring-framework.version property is ignored:

```
<dependency>        <groupId>org.springframework.boot</groupId>        <artifactId>spring-boot-dep
```

Accepted Answer:

I just tried your pom (with and without the **spring-framework.version** property) on a clean m2 repo. Without the property, spring-framework is 5.2.8, with the property, it is 5.2.20. Can you try on a clean repo?

The **spring-framework-bom** at version X is hardcoded to all the spring packages for version X (see https://repo1.maven.org/maven2/org/springframework/spring-framework-bom/5.2.8.RELEASE/spring-framework-bom-5.2.8.RELEASE.pom)

The **spring-framework.version** property is declared and used to pull the **spring-framework-bom** in **spring-boot-dependencies** and inherited by its descendants (see https://repo1.maven.org/maven2/org/springframework/boot/spring-boot-dependencies/2.3.3.RELEASE/spring-boot-dependencies-2.3.3.RELEASE.pom).

spring-boot-dependencies is parent of **spring-boot-starter-parent** (see https://repo1.maven.org/maven2/org/springframework/boot/spring-boot-starter-parent/2.3.3.RELEASE/spring-boot-starter-parent-2.3.3.RELEASE.pom).

Because the property is inherited by descendant, you can override its value at the pom of your application. By overriding it with 5.2.20, you are swapping out spring-framework-bom 5.2.8 for 5.2.20 which effectively pull most of the spring packages for 5.2.20

Highest Rated Answer:

As per the official doc : https://docs.spring.io/spring-boot/docs/current/reference/html/dependency-versions.html#appendix.dependency-versions.properties

```
<spring-framework.version>5.2.20.RELEASE</spring-framework.version>
```

will only override the versions for Group ID org.springframework and not to org.springframework.boot. I don't see a version property for spring boot, so might have to include the version in the dependency.