

PluginVersionResolutionException

This error occurs when you employ a plugin without giving an explicit version for it and Maven fails to determine the latest release version of this plugin. In other words, a `<plugin>` or `<extension>` declaration in the POM without a `<version>` element or a command line like `mvn some-plugin:some-goal` can trigger this error. Possible causes for this error are:

1. You are referring to a non-existing plugin, e.g. by means of a typo in its group or artifact id.
2. You are using a third-party Maven plugin that is not deployed to the central Maven repository and your POM/settings is missing the required `<pluginRepository>` to download the plugin. Note that `<repository>` declarations are not considered when looking for the plugin version, only `<pluginRepositories>` are searched for plugins.
3. The plugin repository you have configured requires authentication and Maven failed to provide the correct credentials to the server. In this case, make sure your `${user.home}/.m2/settings.xml` contains a `<server>` declaration whose `<id>` matches the `<id>` of the plugin repository to use. See the [Maven Settings Reference](#) for more details.
4. The metadata listing the available versions of the plugin is broken. For each configured plugin repository, Maven will try to download a file named `maven-metadata.xml`. If those files are not properly maintained or get corrupted upon download, the version resolution can fail.

In case of any of the first three scenarios, check your POM and/or settings for the proper contents. If you configured the plugin repository in a profile of your `settings.xml`, also verify that this profile gets actually activated, e.g. via invoking `mvn help:active-profiles`.

In the special case that the problematic plugin originates from your POM, it is strongly recommended to explicitly specify the version you want. For commonly used plugins, you can do this rather easy via the `<pluginManagement>` section in a corporate parent POM. Not specifying the plugin version will render your build non-reproducible.

To check whether the metadata files are intact, browse your local repository which is usually located at `${user.home}/.m2/repository`. For a plugin with the coordinates `com.company:custom-maven-plugin`, the files to check are `com/company/custom-maven-plugin/maven-metadata-<repository-id>.xml` where `<repository-id>` indicates the repository from which the file was downloaded. If this file doesn't match the copy in the plugin repository, simply delete the local file and have Maven re-download it. If the `maven-metadata.xml` in the plugin repository itself appears to be broken, please report this to the maintainer of the plugin.