How to fix a patch that fails to apply all hunks (due to CRLF EOLs)

TL;DR: Patch line endings must stay CRLF (^M). If you change it to LF and then back to CRLF, it won't work.

Context

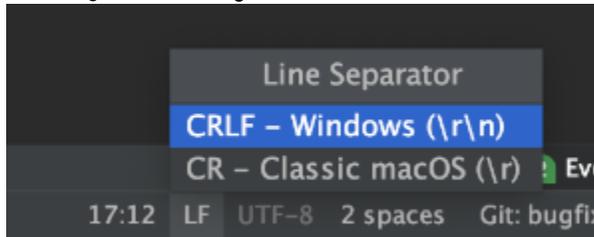This was written up when fighting with nonworking vckeditortextfield.txt patch, fixed in 540d79c2.

Patch may fail to apply for multiple reasons. Usually, you look at the .rej(ect) file, generated next to the patched file. You check rejected hunks and fix accordingly. But sometimes all hunks may be rejected, even though they appear correct.

This article is relevant when

- applying a patch
  - (i.e. using maven-patch-plugin)
- generated by diff
  - with GIT or
  - with IntelliJ IDEa
- to a file with CRLF endings
  - in my case this was a 3rd party library for ckeditor where files had CRLF EOLs. MGNLVA-6
- in a LF environment
  - anything non-windows
- that fails with .reject file containing all hunks of the patch as if nothing could be applied
- due to CRLF EOL issues (which you are not aware of).

Step-by-step guide

1. Generating patch
   a. Open the file you want to patch. This might be a 3rd party library file that maven downloads during build. IDEa will tell you if it is CRLF or LF at the bottom right. **Do not change this**.

      i.
   b. Make changes to the file so you end up with the desired version.
      i. If you are updating an old patch: Try to apply it with **apply patch** in IDEa. Some hunks won't apply. Fix merge conflicts manually, until you end up with desired result.
   c. To generate a patch **show local history,** select before-after versions, right click, **create patch.**
      i. When you select the two version, you should **NOT** see those `CRLF` and `LF` symbols at the top right corner. This would signify there was a change in EOLs. If you do, change the EOLs (see picture above) of the target version to match the original version (probably `CRLF` to `CRLF`)

      ii.
   d. Do **NOT** open the patch. Sounds paranoid, but it seems opening the patch with some editors changes EOLs.
2. If you need to make manual changes to the patch, open it with "vi" from command line.
   a. Notice ^M symbols at some of the EOLs.

```
--- /src/main/java/org/vaadin/openesignforms/ckeditor/widgetset/client/ui/VCKEditorTextField.java
+++ /src/main/java/org/vaadin/openesignforms/ckeditor/widgetset/client/ui/VCKEditorTextField.java
@@ -310,7 +310,7 @@
                setTabIndexAfterReady = false;^M
        }^M
        ^M
-       void loadEditor() {^M
+       protected void loadEditor() {^M
                if ( ckEditor == null && inPageConfig != null && ! ckEditorIsBeingLoaded ) {^M
                        ckEditorIsBeingLoaded = true;^M
                        //logState("loadEditor() calling CKEditorService.loadLibrary");^M
@@ -318,12 +318,7 @@

                                @Override^M
                                public void execute() {^M
:wq
```

b.

c. Notice only the hunk lines have the CR symbols (^M). The patch file itself does not show them (you cannot see it e.g. on the top three lines). You do not want to change these EOLs. If you make changes to the file, these need to stay there.

d. IDEa generated some more lines above the two file paths. I removed those, otherwise mvn-patch-plugin did not apply properly. These metadata changes should be the only reason to edit hte patch anyway, once you generated it with IDEa.

3. Try to apply patch if it works. If the maven-patch-plugin is configured to run your patch, run mvn verify.

> ⚠️ **Refuse GITs proactive EOLs change**
>
> GIT might offer to change the EOLs upon commit. Do not accept. Commit patch as is.

What happened

Editing the patch file in a different tool than vi erases the `^M` EOLs. Lines differ by one character and thus the lines around hunks do not match and fail to apply.

Even if you change the file to DOS file type via vi or CRLF format in IDEa, the ^Ms are not added back. Not all lines in the patch originally have the ^M symbol. It is present only within hunks. The diff takes the lines with their whitespace characters when generating patch, but the patch file itself has its own EOLs. Ignoring whitespace changes is turned on by default in maven-patch-plugin so do not look there for hope.

Related articles

Vaadin 8 Upgrade Effort