mm2 / **Little-CMS** Public

Code | Issues 8 | Pull requests 2 | Actions | Projects | Security and quality

Commit `6a68601`

mm2 committed on Feb 19 · 9 / 9

```
Fix for ParseCube integer overflow in LUT allocation

thanks to @zerojackyi for reporting
```

master · lcms2.19 ••• lcms2.19rc1

1 parent e8b6135 commit 6a68601

1 file changed +10 -1

Top

Filter files…

src

cmscgats.c

Search within code

src/cmscgats.c

```
@@ -3180,7 +3180,16 @@ cmsBool ParseCube(cmsIT8* cube, cmsStage** Shaper,
cmsStage** CLUT, char title[]
3180 3180
3181 3181                if (lut_size > 0) {
3182 3182
3183      -                  int nodes = lut_size * lut_size * lut_size;
     3183 +                  int nodes;
     3184 +
     3185 +                  /**
     3186 +                  * Professional LUT-generation tools (e.g., Nobe LutBake) list
65×65×65 as their highest supported size.
```

```
                3187 +                 */
                3188 +                 if (lut_size > 65)
                3189 +                     return SynError(cube, "LUT size '%d' is over maximum of
                     65", lut_size);
                3190 +
                3191 +                 nodes = lut_size * lut_size * lut_size;
                3192 +
3184            3193
3185            3194                   cmsFloat32Number* lut_table = (cmsFloat32Number*)
                     _cmsMalloc(cube->ContextID, nodes * 3 * sizeof(cmsFloat32Number));
3186            3195                   if (lut_table == NULL) return FALSE;
```

Comments 0

Please sign in to comment.