

[New issue](#)[Jump to bottom](#)

# jpeg-js DoS (infinite loop) #105

🔒 Closedsohomdatta1 opened this issue on Jun 2 · 1 comment · Fixed by [#106](#)

sohomdatta1 commented on Jun 2 • edited ▾

Contributor

The following input can create a infinite loop inside jpeg-js causing it to never return:

```
const jpeg = require('jpeg-js');

let buf = Buffer.from( 'ffd8ffc1f151d800ff51d800ffdaffde', 'hex' );
jpeg.decode( buf );
```

Based on some preliminary debugging it appears to be related to the following code:

[jpeg-js/lib/decoder.js](#)

Lines 579 to 589 in b58cc11

```
579     var maxH = 0, maxV = 0;
580     var component, componentId;
581     for (componentId in frame.components) {
582         if (frame.components.hasOwnProperty(componentId)) {
583             component = frame.components[componentId];
584             if (maxH < component.h) maxH = component.h;
585             if (maxV < component.v) maxV = component.v;
586         }
587     }
588     var mcusPerLine = Math.ceil(frame.samplesPerLine / 8 / maxH);
589     var mcusPerColumn = Math.ceil(frame.scanLines / 8 / maxV);
```

Here `maxH` and `maxV` are initialized to zero, but since there are no components, the values are never modified, leading to a divide by zero error in the last two line (which set `mcusPerLine` and `mcusPerColumn` to `Infinity`).

These values are later used inside the `decodeAsScan()` function, where the following loop condition never evaluates to false since `mcuExpected` is set to `frame.mcusPerLine * frame.mcusPerColumn` (i.e. `Infinity * Infinity`) at line 292 in `/lib/decoder.js`.

[jpeg-js/lib/decoder.js](#)

Line 297 in b58cc11

```
297      while (mcu < mcuExpected) {
```

found using [jsfuzz](#)



**sohomdatta1** mentioned this issue on Jun 2

**Add limits on sampling factors #106**

Merged

**sohomdatta1** commented on Jun 2

Contributor

Author

I've created a PR to fix this issue ([#106](#)) based on some digging around I did wrt to the JPEG specification.



**patrickhulce** closed this as completed in [#106](#) on Jun 3



**vince-fugnitto** mentioned this issue on Jun 16

**[BUG]Latest playwright depends on jpeg-js-0.4.3.tgz which cause the Whitesource Security alert microsoft/playwright#14816**

Closed



**TheKingTermux** mentioned this issue on Aug 15

**Infinite loop in jpeg-js TheKingTermux/alice#107**

Closed



4 tasks

Assignees

No one assigned

Labels

None yet

Projects

None yet

---


Milestone

No milestone

---

Development

Successfully merging a pull request may close this issue.

 [Add limits on sampling factors](#)

---

1 participant

