wollardj / **simple-plist** Public

Code · Issues 4 · Pull requests 2 · Actions · Projects · Wiki · ...

New issue

Jump to bottom

Prototype Pollution using .parse() #60

Closed **keerok** opened this issue on Feb 23 · 14 comments · Fixed by #62

keerok commented on Feb 23

Hi, There's a prototype pollution vulnerability in .parse() related to the xml that are being parsed in it. In the following example the prototype pollution will affect the length parameter.

```
var plist = require('simple-plist');

var xml = `
<plist version="1.0">
    <key>metadata</key>
    <dict>
      <key>bundle-identifier</key>
      <string>com.company.app</string>
    </dict>
  </plist>`;

console.log(plist.parse(xml));
/**
 * * * * * * * * * * * * * * * * * * * * * * * * * *
 * * * * END OF THE NORMAL CODE EXAMPLE! * * * * * *
 * * * * * * * * * * * * * * * * * * * * * * * * * *
 **/


/**
 * * * * * * * * * * * * *
 * PROTOTYPE POLLUTION *
 * * * * * * * * * * * * *
 **/
var xmlPollution = `
<plist version="1.0">
  <dict>
    <key>__proto__</key>
    <dict>
      <key>length</key>
      <string>polluted</string>
    </dict>
  </dict>
```

```
</plist>`;
console.log(plist.parse(xmlPollution).length); // polluted
```

More information about the vulnerability: https://github.com/HoLyVieR/prototype-pollution-nsec18/blob/master/paper/JavaScript_prototype_pollution_attack_in_NodeJS.pdf

👍 3

SimenB commented on Mar 30

Since GHSA-gff7-g5r8-mg8m is a thing now, maybe a ping to **@wollardj** could get this fixed? 😀

👍 2

kathaypacific mentioned this issue on Mar 30

Update known vulnerabilities valora-inc/wallet#2270

Merged

Sujay-shetty commented on Mar 30

could you please update plist package to latest version (3.0.5) where this vulnerability is fully fixed?
TooTallNate/plist.js#114

csutorasr mentioned this issue on Mar 30

fix: update plist to 3.0.5 #62

Merged

mergify bot pushed a commit to valora-inc/wallet that referenced this issue on Mar 30

`Update known vulnerabilities (#2270)` …   ✕ `2e56a31`

wollardj closed this as completed in #62 on Mar 30

wollardj commented on Mar 30   Owner

I'm re-opening this for a bit. I want to write some tests to go along with this before I cut a new release, but I won't have time until later this evening.

wollardj reopened this on Mar 30

SimenB commented on Mar 30 • edited ▾

Seems weird the advisory points to this module if the bug is in a dependency...

EmilianoSanchez mentioned this issue on Mar 30

Vulnerability fixes and general polishing splitio/react-native-client#17

Merged

wollardj commented on Mar 30 Owner

I've just published v1.3.1 to npm (https://www.npmjs.com/package/simple-plist/v/1.3.1) and tagged it as `latest`. Let me know if anyone sees any issues or if any additional audits are failing.

🎉 7 ❤️ 2 🚀 2

Sujay-shetty mentioned this issue on Mar 31

Critical vulnerability in simple-plist package (Prototype Pollution using .parse()) apache/cordova-node-xcode#124

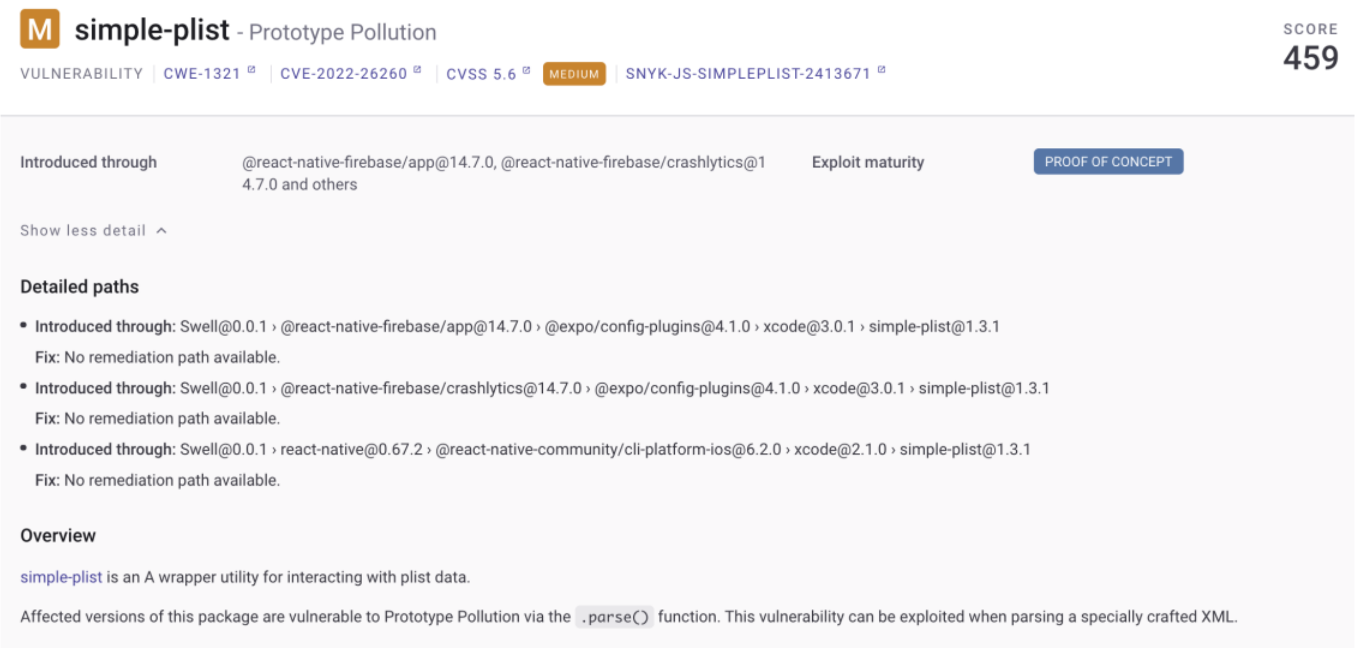
○ Open

srithar21 commented on Apr 1

Hi ,

I am getting the same issue in 1.3.1 version.

https://security.snyk.io/vuln/SNYK-JS-SIMPLEPLIST-2413671



M **simple-plist** - Prototype Pollution

VULNERABILITY | CWE-1321 | CVE-2022-26260 | CVSS 5.6 MEDIUM | SNYK-JS-SIMPLEPLIST-2413671

SCORE
459

Introduced through @react-native-firebase/app@14.7.0, @react-native-firebase/crashlytics@14.7.0 and others

Exploit maturity PROOF OF CONCEPT

Detailed paths

- **Introduced through:** Swell@0.0.1 › @react-native-firebase/app@14.7.0 › @expo/config-plugins@4.1.0 › xcode@3.0.1 › simple-plist@1.3.1
  Fix: No remediation path available.
- **Introduced through:** Swell@0.0.1 › @react-native-firebase/crashlytics@14.7.0 › @expo/config-plugins@4.1.0 › xcode@3.0.1 › simple-plist@1.3.1
  Fix: No remediation path available.
- **Introduced through:** Swell@0.0.1 › react-native@0.67.2 › @react-native-community/cli-platform-ios@6.2.0 › xcode@2.1.0 › simple-plist@1.3.1
  Fix: No remediation path available.

Overview

simple-plist is an A wrapper utility for interacting with plist data.

Affected versions of this package are vulnerable to Prototype Pollution via the `.parse()` function. This vulnerability can be exploited when parsing a specially crafted XML.

srithar21 mentioned this issue on Apr 1

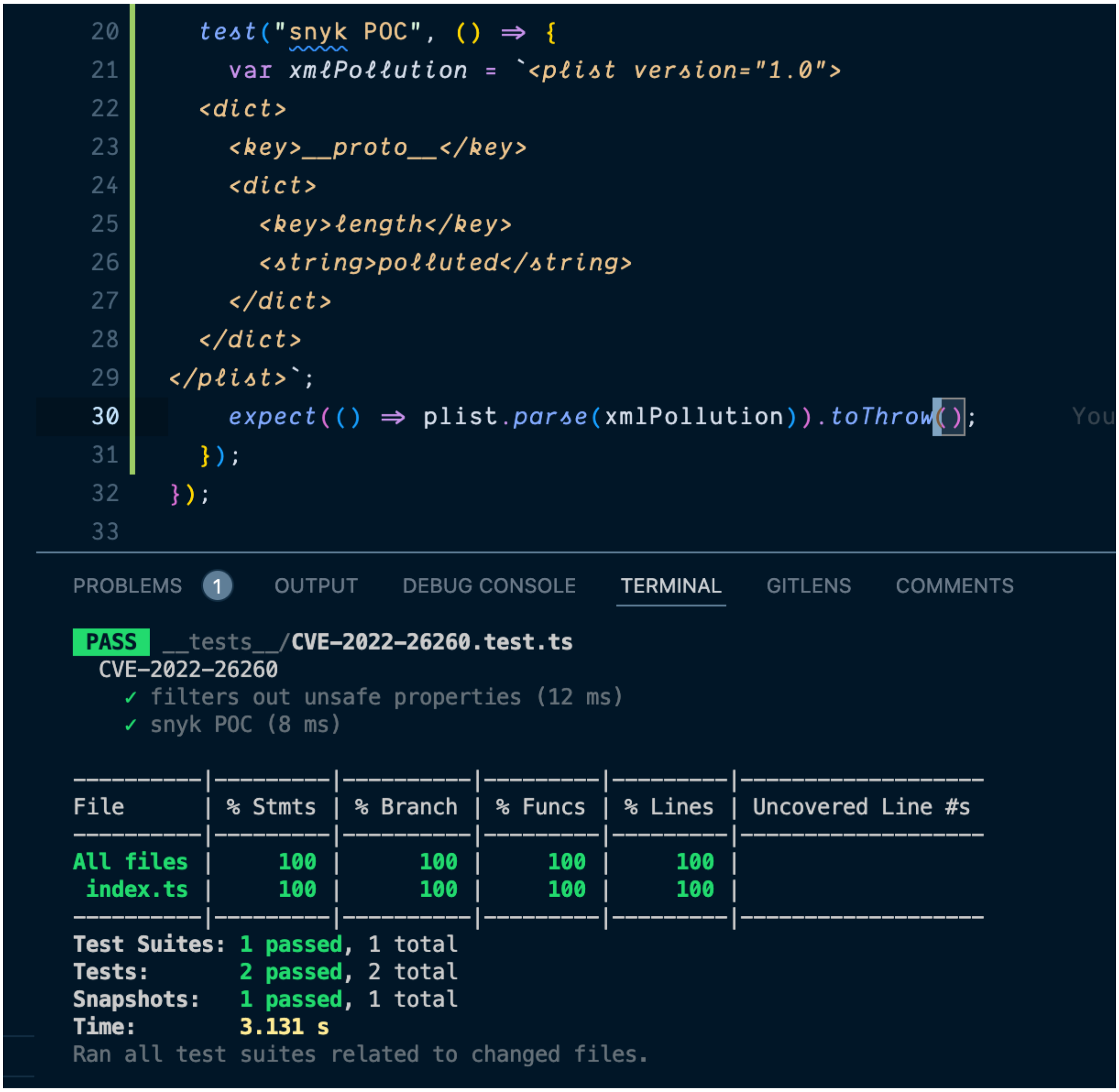
Prototype Pollution in simple-plist package - dependency of @react-native-firebase/app [solution: reinstall node_modules to get updated simple-plist] invertase/react-native-firebase#6172

Closed

wollardj commented on Apr 2 • edited

Owner

@srithar21 - I'm not sure how they're able to do that. I attempted to verify their POC by adding a test for it and the underlying plist.js library throws when it detects a `__proto__` key.



```
    test("snyk POC", () => {
      var xmlPollution = `<plist version="1.0">
    <dict>
      <key>__proto__</key>
      <dict>
        <key>length</key>
        <string>polluted</string>
      </dict>
    </dict>
  </plist>`;
      expect(() => plist.parse(xmlPollution)).toThrow();        You
    });
  });
```

```
 PASS  __tests__/CVE-2022-26260.test.ts
  CVE-2022-26260
    ✓ filters out unsafe properties (12 ms)
    ✓ snyk POC (8 ms)

----------|---------|----------|---------|---------|-------------------
File      | % Stmts | % Branch | % Funcs | % Lines | Uncovered Line #s
----------|---------|----------|---------|---------|-------------------
All files |     100 |      100 |     100 |     100 |
 index.ts |     100 |      100 |     100 |     100 |
----------|---------|----------|---------|---------|-------------------
Test Suites: 1 passed, 1 total
Tests:       2 passed, 2 total
Snapshots:   1 passed, 1 total
Time:        3.131 s
Ran all test suites related to changed files.
```

Less important, but also a little suspicious, their POC wouldn't work anyway because the first byte is an unexpected ASCII sequence which would have made `simple-plist` attempt to use the binary parser. Since the file isn't binary, their example results in a different error with the message `"Unable to determine format for plist aStringOrBuffer"`.

It's entirely possible if not likely that their POC is suffering from a formatting problem, but it's still worth noting since the POC doesn't appear to be valid.

Happy to be wrong about this if someone can point out something that I might be missing.

😕 1

oanatinus commented on Apr 3

I wonder if you need to close this GitHub issue in order to trigger the various vulnerability databases to "recheck" the vulnerability...

elopezanaya commented on Apr 4

Hello, does someone know the status of this issue?, it seems like it was already solved , but this issue still is open, and in all CVE references is marked as not patched

MaeveOReilly commented on Apr 5

`npm audit` still listing it as open with no patch for me; I'm guessing **@oanatinus** is correct and this issue needs to be closed?

oanatinus commented on Apr 8

@wollardj what do you think, do you want to try closing this issue to see if that triggers the vulnerability databases to mark 1.3.1 as the fixed version?

wollardj commented on Apr 13 Owner

It's worth a shot

👍 2

wollardj closed this as completed on Apr 13

brice-noowu commented on Apr 20

Still an issue here, npm audit still listing it as open with no patch

maschad mentioned this issue on Apr 28

Security Vulnerability #63

Closed

TuurDutoit mentioned this issue on May 24

[GHSA-gff7-g5r8-mg8m] Prototype Pollution in simple-plist github/advisory-database#326

Merged

darakian commented on Jun 1

> It's worth a shot

In the future give us a shout over at https://github.com/github/advisory-database/ if we're out of date 😉

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

fix: update plist to 3.0.5
csutorasr/simple-plist

10 participants