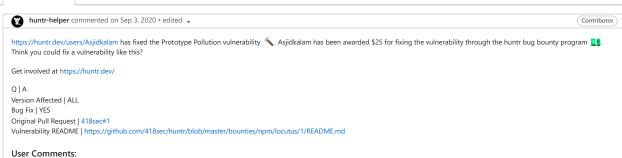New issue

Security Fix for Prototype Pollution - huntr.dev #418

⟲ **Merged**   **kvz** merged 2 commits into `locutusjs:master` from `418sec:master` ⬚ on Sep 3, 2020

Conversation 1   Commits 2   Checks 0   Files changed 1

🐮   ⊙ **huntr-helper** commented on Sep 3, 2020 • edited ▾   Contributor

https://huntr.dev/users/Asjidkalam has fixed the Prototype Pollution vulnerability 🔧. Asjidkalam has been awarded $25 for fixing the vulnerability through the huntr bug bounty program 🎖️.
Think you could fix a vulnerability like this?

Get involved at https://huntr.dev/

Q | A
Version Affected | ALL
Bug Fix | YES
Original Pull Request | 418sec#1
Vulnerability README | https://github.com/418sec/huntr/blob/master/bounties/npm/locutus/1/README.md

User Comments:

📊 Metadata *

Prototype Pollution bug

Bounty URL: https://www.huntr.dev/bounties/1-npm-phpjs

⚙️ Description *

phpjs is a community built PHP binding in JavaScript. This package is vulnerable to Prototype Pollution via `parse_str`.
(Changed to `locutus` - https://github.com/kvz/locutus)

Prototype Pollution refers to the ability to inject properties into existing JavaScript language construct prototypes, such as objects. JavaScript allows all Object attributes to be altered, including their magical attributes such as `_proto_`, `constructor` and `prototype`. An attacker manipulates these attributes to overwrite, or pollute, a JavaScript application object prototype of the base object by injecting other values. Properties on the Object.prototype are then inherited by all the JavaScript objects through the prototype chain. When that happens, this leads to either denial of service by triggering JavaScript exceptions, or it tampers with the application source code to force the code path that the attacker injects, thereby leading to remote code execution.

🖥️ Technical Description *

The bug is fixed by validating the input `strArray` to check for prototypes. It is implemented by a simple validation to check for prototype keywords (`__proto__`, `constructor and prototype`), where if it exists, the function returns the object without modifying it, thus fixing the Prototype Pollution Vulnerability.

🐛 Proof of Concept (PoC) *

Clone the project, install the required dependencies and on running the below snippet of code, it triggers prototype pollution and logs `true`.

```
const phpjs = require('phpjs');
phpjs.parse_str("__proto__[polluted]=true",{});
console.log(polluted);
```


🔥 Proof of Fix (PoF) *

After the fix is applied, it returns an error since the polluted referred in the PoC is no more accessible(which is intended). Hence fixing the issue.

```
value = (tmp.length < 2) ? '' : fixStr(tmp[1]);

//sanitize key
if (key.includes('__proto__') || key.includes('constructor') || key.includes('prototype')) {
  break;
}

while (key.charAt(0) === ' ') {
  key = key.slice(1);
}
if (key.indexOf('\x
  key = key.slice(0,

if (key && key.char
  keys = [];
  postLeftBracketPos
  for (j = 0; j < ke
    if (key.charAt(
      postLeftBracke
    } else if (key.c
      if (postLeftB
        if (!keys.le
          keys.push(
        }
        keys.push(ke
        postLeftBrac
        if (key.cha
          break;
        }
```

Terminal output:
```
→ phpjs-pp node index.js
true
→ phpjs-pp node index.js
/mnt/d/Bug/Huntr/phpjs-pp/index.js:3
console.log(polluted);
            ^

ReferenceError: polluted is not defined
    at Object.<anonymous> (/mnt/d/Bug/Huntr/phpjs-pp/index.js:3:13)
    at Module._compile (module.js:652:30)
    at Object.Module._extensions..js (module.js:663:10)
    at Module.load (module.js:565:32)
    at tryModuleLoad (module.js:505:12)
    at Function.Module._load (module.js:497:3)
    at Function.Module.runMain (module.js:693:10)
    at startup (bootstrap_node.js:188:16)
    at bootstrap_node.js:609:3
→ phpjs-pp |
```

👍 **User Acceptance Testing (UAT)**

Just prevented some keywords as `key` and no breaking changes are introduced. :)

Asjidkalam and others added 2 commits 2 years ago

-o- 🧑 `fixed prototype pollution`   0eb16d8

-o- 🧑 Merge pull request #1 from Asjidkalam/master ···   0884b03

🧑 **kvz** merged commit `f2868ae` into `locutusjs:master` on Sep 3, 2020

🧑 **kvz** commented on Sep 3, 2020   [Collaborator]

Thank you! Published as 2.0.12 👌

📇 🧑 **adam-nygate** mentioned this pull request on Nov 3, 2020

Potential stagnation of open issues on h1 bounty program nodejs/security-wg#654

⊘ Closed

Reviewers

No reviews

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close these issues.

None yet

---

4 participants

🧑 🧑 🧑 🧑