huntr

Possible prototype pollution in Schema.path in automattic/mongoose

0

✓ Valid Reported on Jun 15th 2022

Description

Mongoose is a MongoDB object modeling tool designed to work in an asynchronous environment.
Affected versions of this package are vulnerable to Prototype Pollution. The `Schema.path()` function is vulnerable to prototype pollution when setting the schema object. This vulnerability allows modification of the Object prototype and could be manipulated into a Denial of Service (DoS) attack.

Proof of Concept

```
// poc.js
const mongoose = require('mongoose');
const schema = new mongoose.Schema();

malicious_payload = '__proto__.toString'

schema.path(malicious_payload, [String])

x = {}
console.log(x.toString()) // crashed (Denial of service (DoS) attack)
```

Impact

This vulnerability can be manipulated to exploit other types of attacks, such as Denial of service (DoS), Remote Code Execution, or Property Injection.

Chat with us

References

- Same issue
- See more about Prototype Pollution

CVE
CVE-2022-2564
(Published)

Vulnerability Type
CWE-1321: Prototype Pollution

Severity
High (7)

Registry
Npm

Affected Version
<=6.3.8

Visibility
Public

Status
Fixed

Found by

Khang Vo (doublevkay)
@vovikhangcdv
master

Fixed by

Valeri Karpov
@vkarpov15
maintainer

This report was seen 1,757 times.

We are processing your report and will contact the **automattic/mongoose** team within 24 hours. 5 months ago

Khang Vo (doublevkay) modified the report 5 months ago

Chat with us

Khang 5 months ago

Researcher

Hi, after putting a lot of effort to debug and bypass the fix commit of the issue the issue#10035 - Possible prototype pollution in mongoose.Schema. I found out a crucial issue here.

The root cause of the issue#10035 is not belongs to `Schema.add()` function as mentioned.

The root cause and the actual attack chain is `Schema() -> Schema.add() -> Schema.path` (called at (schema.js#L580) - which point out in my report. So the fix is incomplete and could be bypassed. The attack can reproceduce the attack scenario (using `Scheme()` function) with the following PoC.

```
// PoC.js - Bypass the issue #10035 fix
mongoose = require('mongoose');
var malicious_payload = '{"__proto__.toString": "Number"}';
console.log('Before:', {}.toString()); // [object Object]
mongoose.Schema(JSON.parse(malicious_payload));
console.log('After:', {}.toString()); // crashed
```

Note: Due to the interpreAsType check, we now no longer able to inject arbitrary value to properties. But, we are still able to choose which property could be polluted. It is potential to pollute `toString()` function as mentioned in the PoC and cause a process crash.

Khang Vo (doublevkay) modified the report 5 months ago

We have contacted a member of the **automattic/mongoose** team and are waiting to hear back 5 months ago

We have sent a follow up to the **automattic/mongoose** team. We will try again in 7 days. 5 months ago

We have sent a second follow up to the **automattic/mongoose** team. We will try again in 10 days. 5 months ago

We have sent a third and final follow up to the **automattic/mongoose** team. This report is now considered stale. 5 months ago

Khang 5 months ago

Chat with us

Hey @admin, could you help to manually contact maintainer or should do I?

Jamie Slome 5 months ago Admin

@vovikhangcdv - feel free to get in touch with the maintainers directly with this report URL.

If they do not want to sign-up or are happy for the report to go public, let me know as I can provide magic URL access to them, or make the report public.

Valeri Karpov 4 months ago Maintainer

Hi, I'm sorry for the delay, I missed the original email. We'll fix this issue. What is the best way to disclose once we've fixed it?

However, this issue is an unlikely edge case. You typically don't define schemas with untrusted user data - schemas are meant to configure Mongoose to validate untrusted user data.

Khang 4 months ago Researcher

Hey Valeri, nice to see your response.

I have seen your reply on the issue 10035 and I agree with it. Accepting user input into the `Schema` constructor is unreal. But If we concentrate on this reported case, the vulnerability is the 1st parameter of function `Schema.path()` - which is intended to allow a string input to modify the schema. And so far now, it becomes a real security problem when dependents trust `mongoose` to handle string input - which leads to unintended behavior crashing the application. The concept `Never Trust User Input` is a good practice to secure coding but it is not the insurance for security. I want to take an example of the case Inefficient Regular Expression Complexity in moment, both of us agree that users should not pass user-provided strings without sanity but it is good to eliminate the issue and awake the users.

After you fix the issue, I could help to retest it. Then we can resolve and disclose this report. If you agree with my perspective, please assign CVE too. And the last one, I think we could create a Security Advisory. It will help the `Dependabot` alerts to users and help to secure `mongoose` dependents.

Jamie Slome 4 months ago Admin

@Valeri @Khang - we can take care of the entire disclosure process. Once you are both happy that the issue is a valid vulnerability and a fix has been established, we will automatically assign and publish a CVE for this specific report.

Chat with us

This will make this report page go public, and will be included in the CVE as a reference. This has the same effect as a Dependabot alert, as GitHub scrapes our CVEs and adds them to their

Security Advisory database :)

@Valeri - when you are ready with a fix, feel free to use the resolve button below, and you will need to provide us with some information about which commit SHA fixed the issue, and which release of the package will include the patch.

Let me know if either of you have any further questions 👍

Khang 4 months ago Researcher

I got it, thank you @Jamie!

Valeri Karpov 4 months ago Maintainer

Here's a fix we put in last night: https://github.com/Automattic/mongoose/commit/a45cfb6b0ce0067ae9794cfa80f7917e1fb3c6f8 . We did not put this fix in master yet, not sure whether that counts as disclosure. But if you're ok with it, we can ship a release with this fix within 24 hours.

Khang 4 months ago Researcher

I confirm the fix worked 👍

Jamie Slome 4 months ago Admin

@Valeri - sure, let us know once you have shipped the release with a fix.

Once you have done this, feel free to proceed with `Resolve` (`Valid and Fixed`) 👍

Valeri Karpov 4 months ago Maintainer

Hi, I just shipped v6.4.6 with the fix earlier today. I don't see an option to click "Resolve" anywhere in this UI though.

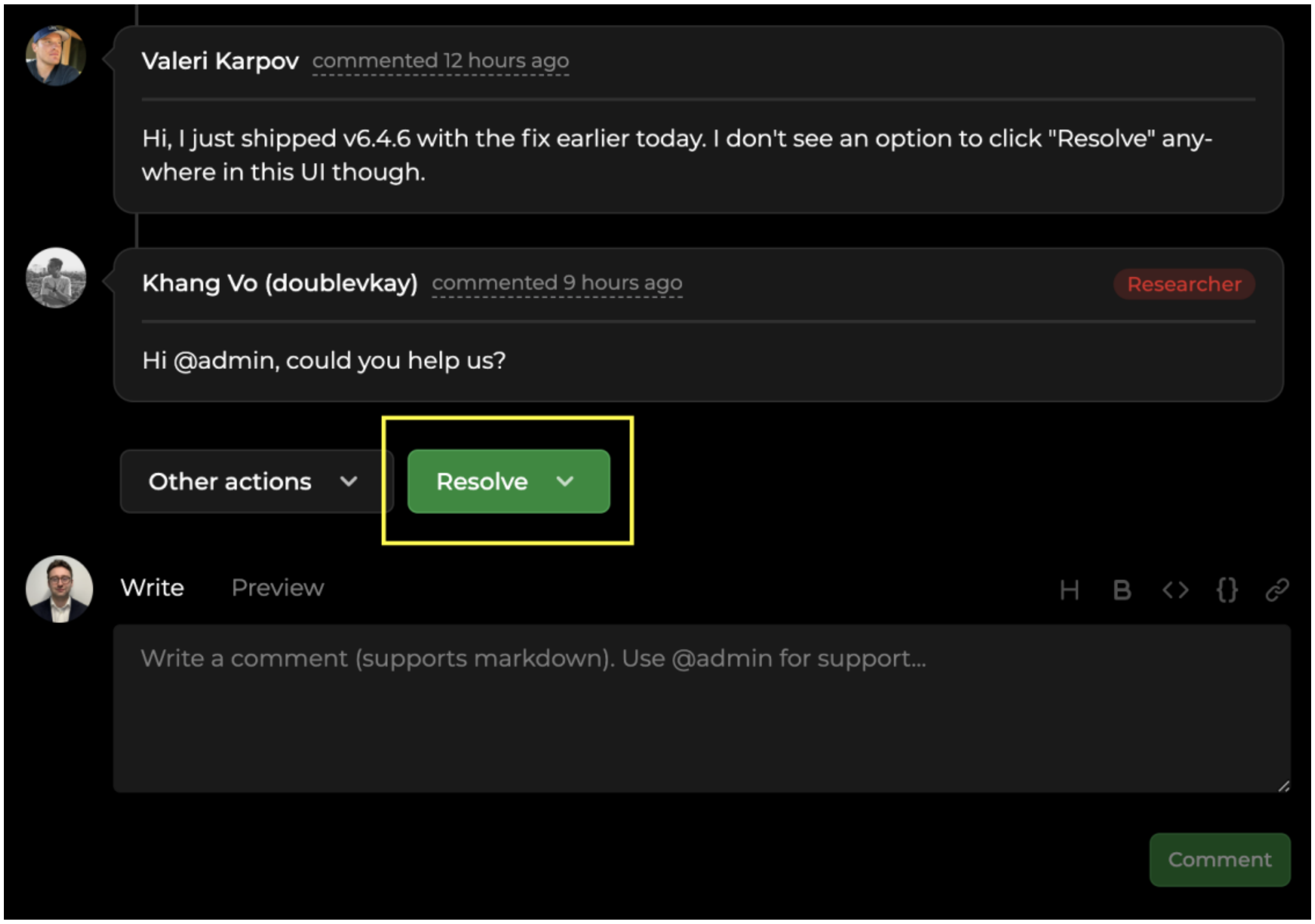
Khang 4 months ago Researcher

Hi @admin, could you help us?

Chat with us

Jamie Slome 4 months ago Admin

@Valeri - are you not able to see the following:



Valeri Karpov commented 12 hours ago

Hi, I just shipped v6.4.6 with the fix earlier today. I don't see an option to click "Resolve" anywhere in this UI though.

Khang Vo (doublevkay) commented 9 hours ago Researcher

Hi @admin, could you help us?

Other actions
Resolve
Write
Preview
Write a comment (supports markdown). Use @admin for support...
Comment

Khang 4 months ago Researcher

Hi there, any update?

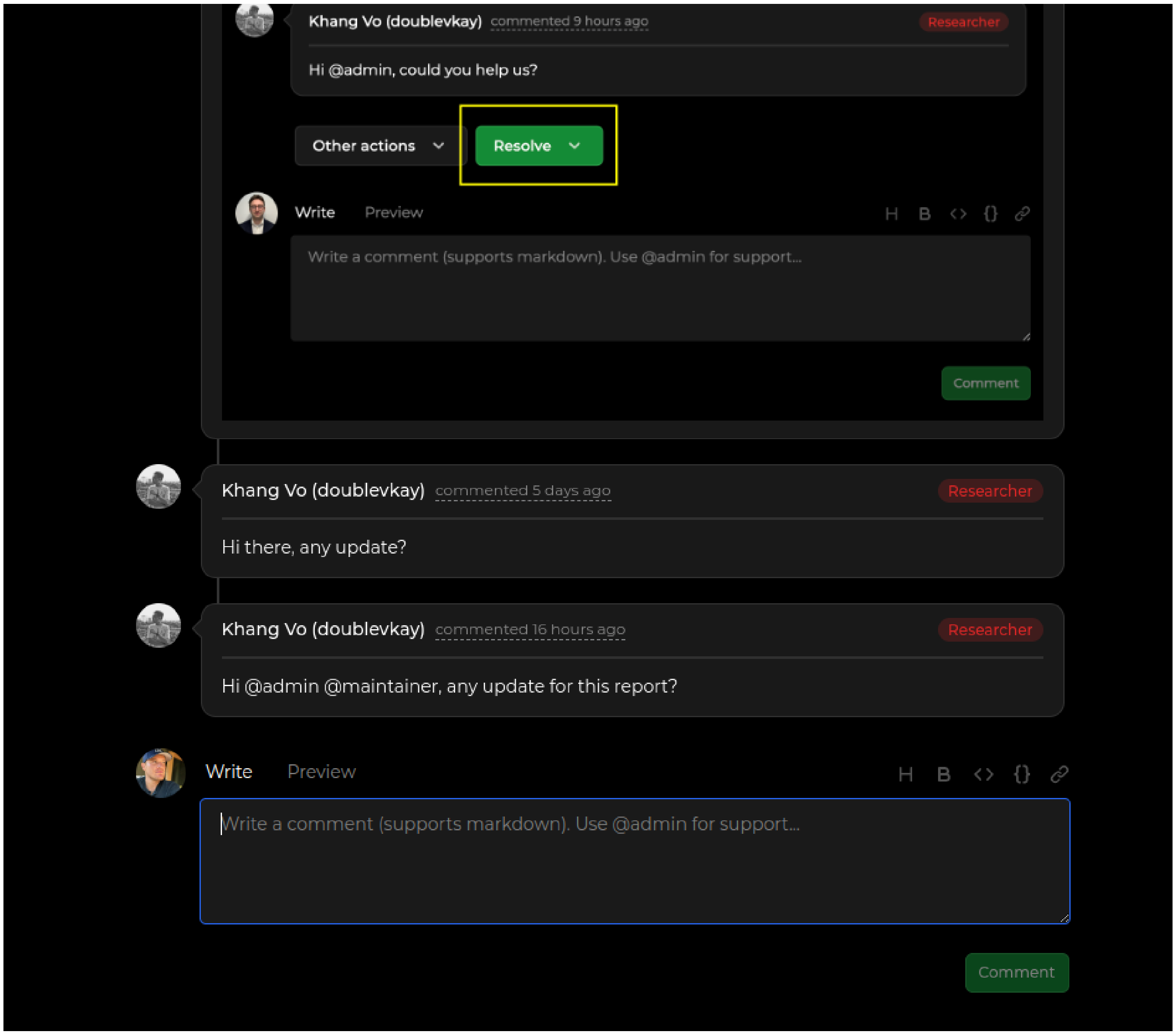
Khang 4 months ago Researcher

Hi @admin @maintainer, any update for this report?

Valeri Karpov 4 months ago Maintainer

Nope, below is what I see:

Chat with us


Khang Vo (doublevkay) commented 9 hours ago Researcher

Hi @admin, could you help us?

Other actions
Resolve
Write
Preview
Write a comment (supports markdown). Use @admin for support...
Comment

Khang Vo (doublevkay) commented 5 days ago Researcher

Hi there, any update?

Khang Vo (doublevkay) commented 16 hours ago Researcher

Hi @admin @maintainer, any update for this report?

Write
Preview
Write a comment (supports markdown). Use @admin for support...
Comment

. Doing a ctrl+f for "Resolve" returns no results outside of comments.

Khang 4 months ago Researcher

Hi @admin, could you help?

Jamie Slome 4 months ago Admin

Spotted the problem 🎉 For some reason @vkarpov15, you were not listed as
this project. I have now gone ahead and done this, and you should be able to

Chat with us

Please ensure you log out and log back into the platform, and that you have no magic tokens in the URL.

Valeri Karpov modified the Severity from High to Low 4 months ago

The researcher has received a minor penalty to their credibility for miscalculating the severity: -1

Valeri Karpov validated this vulnerability 4 months ago

Khang Vo (doublevkay) has been awarded the disclosure bounty ✓

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Valeri Karpov marked this as fixed in **6.4.6** with commit **a45cfb** 4 months ago

Valeri Karpov has been awarded the fix bounty ✓

This vulnerability will not receive a CVE ✗

Valeri Karpov 4 months ago Maintainer

I'm sorry for updating the severity, that was a mistake on my part. I read this doc more closely and "high" is more appropriate given that we're measuring "severity, not risk". However, changing the severity back required selecting some toggles and I didn't want to make another mistake - I'm not very familiar with this UI.

Khang 4 months ago Researcher

Can we assign cve for this issue? Of course with the new severity changed.

Jamie Slome modified the Severity from Low to High (7) 4 months ago

Jamie Slome 4 months ago Admin

Both are now sorted 👍

Chat with us

Timothee 3 months ago

good job @vovikhangcdv! I have submitted a vulnerability report and fix for the exact same bug to security@tidelift.com 6 months ago but the vulnerability has never been pached... CVE-2022-

24304 status still showing "Reserved" as of today. I guess I should have tried my luck on huntr.dev platform in the first place.

Khang 3 months ago Researcher

I had submitted it to security@tidelift.com but got no response too. I lastly tried to connect via Github issue, and glad that Valeri was noticed. About the "Reserved" status of CVE, actually, I have no idea too.
Anyway, thank you, and wish u all luck, Timothee.

Sign in to join this conversation