E-2019-19507) in `jpv`

August 11, 2020 By **Sonatype Security Research Team**

3 minute read time

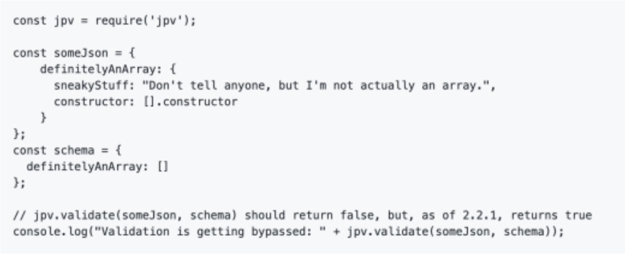
SHARE:

In addition to regular vulnerability data research, the Sonatype Security Research Team also contributes to the open-source community by going the extra mile when we discover flaws that were previously not reported. Recall, earlier this year when our team had discovered they could bypass a fix made to the **SheetJS** project. We took immediate steps to collaborate with the project developers, responsibly disclosing the details of the bypass, and working with them on rolling out a new fix. Consequently, we helped protect our customers by incorporating this newly discovered information into our data.

Fast-forward to recent presentations at DefCon which highlighted various vulnerabilities... Because DefCon is such a widely recognized event, the Security Research Team revisited the data of vulnerabilities mentioned there since they were likely to get renewed attention. One such vulnerability we examined was CVE-2019-19507 from "**Discovering Hidden Properties to Attack Node.js ecosystem**". Like with SheetJS, **we discovered that the vulnerability could still be exploited with the existing fix in place.**

Json Pattern Validator (JPV) is an open-source JSON schema validator which makes it easy to compare a given JSON object against a schema or particular pattern. A typical use-case for such a package would be to validate that incoming JSON is in an expected format. CVE-2019-19507 allows for an attacker to validate objects as arrays, by setting that object's `constructor.name` to be 'Array'. To fix this problem, JPV was updated to simply check to make sure that the constructors matched.

While updating our data for this CVE, Security Researcher Garrett Calpouzos discovered a way to iterate on this attack. By setting the nefarious object's constructor to be `[].constructor`, an attacker could once again successfully masquerade an object as an array and falsely get the JSON data validated by JPV.



```
const jpv = require('jpv');

const someJson = {
    definitelyAnArray: {
      sneakyStuff: "Don't tell anyone, but I'm not actually an array.",
      constructor: [].constructor
    }
};
const schema = {
  definitelyAnArray: []
};

// jpv.validate(someJson, schema) should return false, but, as of 2.2.1, returns true
console.log("Validation is getting bypassed: " + jpv.validate(someJson, schema));
```

PoC of Attack to Bypass Fix for CVE-2019-19507

Source: GitHub Issue

n the project via GitHub issue #10, he shared
a commendable job in their response, both
eceiving the PoC (two days after our initial

Over the weekend, the project developers added a new section of code to check specifically for arrays, using the native `Array.isArray` function along with other checks to ensure that data that claims to be an array is, in fact, an array. This new functionality to account for such an attack has been included in version 2.2.2. Consequently, users of `jpv` should consider upgrading to version **2.2.2** as present in npm.

While a new fix was underway over the weekend, Sonatype customers had already received the updated information on CVE-2019-19507, particularly the following Advisory Deviation:

> *While researching this vulnerability, the Sonatype Security Research team discovered a bypass for the fix the developer provided for this issue. As of 8/6/2020, we have reported the issue to the developer and are awaiting a response.*

To prevent confusion, we have assigned a new vulnerability identifier to this flaw: CVE-2020-17479. Customers using versions of `jpv` containing the fix for CVE-2019-19507 should be seeing this newly assigned identifier in our products (for `jpv` versions <2.2.2).

Cases like these illustrate how interaction with the open-source community helps keep components, customers, and their software supply chains secure.

DevOps-native organizations with the ability to continuously deploy software releases have an automation advantage that allows them to stay one step ahead of the hackers. **Sonatype Nexus customers were notified of the updates pertaining to this vulnerability within hours of the discovery. Their development teams automatically received instructions on how to remediate the risk.**

If you're not a Sonatype customer and want to find out if your code is vulnerable, you can use Sonatype's free Nexus Vulnerability Scanner to quickly find out.

Visit the Nexus Intelligence Insights page for a deep dive into other vulnerabilities like this one. Or subscribe to automatically receive Nexus Intelligence Insights hot off the press.
Tags: Nexus Lifecycle, vulnerabilities, Nexus Firewall, featured, Nexus Intelligence, Product

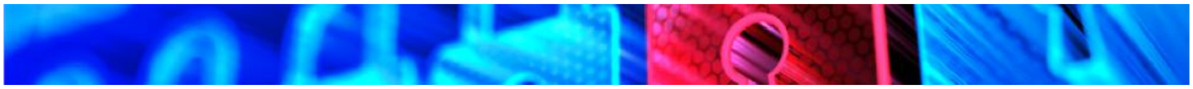
Written by Sonatype Security Research Team

Sonatype's Security Research Team is comprised 65 world class professionals with 500+ years of experience. The Team is focused on bringing real-time, in-depth intelligence and actionable information about open source and third party vulnerabilities to Sonatype customers.

AUTHOR POSTS TOPIC POSTS

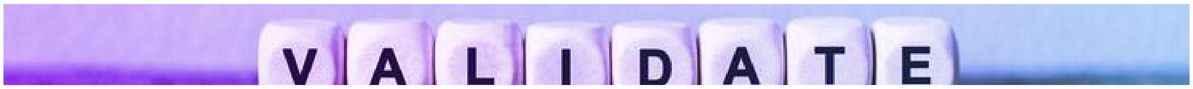
CursedGrabber strikes again: Sonatype spots new malware campaign against Software Supply Chains

Sonatype Security Research Team



Sonatype Stops Software Supply Chain Attack Aimed at the Java Developer Community

Sonatype Security Research Team



CVE-2020-17479: The return of Validation Bypass (CVE-2019-19507) in `jpv`

Sonatype Security Research Team

First Name*

Last Name*

Email*

Comment*

protected by reCAP

SUBMIT COMMENT

Products

Free Tools

Solutions

Resources

About

Pricing

Twitter

LinkedIn

Facebook

YouTube

GitHub

Sonatype Headquarters - 8161 Maple Lawn Blvd #250, Fulton, MD 20759
Tysons Office - 8281 Greensboro Drive – Suite 630, McLean, VA 22102
Australia Office - 60 Martin Place Level 1, Sydney, NSW 2000, Australia
London Office -168 Shoreditch High Street, E1 6HU London

Subscribe for all the latest software security news and events

Email

SUBSCRIBE

Terms of Service Privacy Policy Modern Slavery Statement Event Terms and Conditions Do Not Sell My Personal Information