

## 13 OS Command Injection on Jison [all-parser-ports]

Share:     

### TIMELINE



48piraj submitted a report to [Node.js third-party modules](#).

Sep 7th (3 years ago)

I would like to report OS Command Injection vulnerability on [Jison](#) in parser ports. (*CSharp, PHP*)

It allows arbitrary OS shell command execution through a crafted command-line argument.

### Basic Information

Module: *jison*

Version: `0.4.18`

NPM Project Page: <https://www.npmjs.com/package/jison>

### Module Description

An API for creating parsers in JavaScript

Jison generates bottom-up parsers in JavaScript. Its API is similar to Bison's, hence the name. It supports many of Bison's major features, plus some of its own. If you are new to parser generators such as Bison, and Context-free Grammars in general, a good introduction is found in the Bison manual. If you already know Bison, Jison should be easy to pickup.

Briefly, Jison takes a JSON encoded grammar or Bison style grammar and outputs a JavaScript file capable of parsing the language described by that grammar. You can then use the generated script to parse inputs and accept, reject, or perform actions based on the input.

### Module Stats

Downloads in the last week: (<https://api.npmjs.org/downloads/point/last-week/jison>)

Code 70 Bytes	<a href="#">Wrap lines</a> <a href="#">Copy</a> <a href="#">Download</a>
1 downloads : 138857	
2 start : 2019-08-31	
3 end : 2019-09-06	
4 package : jison	

Downloads in the last month: (<https://api.npmjs.org/downloads/point/last-month/jison>)

Code 70 Bytes	<a href="#">Wrap lines</a> <a href="#">Copy</a> <a href="#">Download</a>
1 downloads : 678197	
2 start : 2019-08-08	
3 end : 2019-09-06	
4 package : jison	

Stats by npm-stat: <https://npm-stat.com/charts.html?package=jison>

### Vulnerability

Jison has parsing/lexing template to php, C# which don't sanitize `process.argv` (command line arguments), before passing it to `child_process.exec()`, hence allowing arbitrary shell command injection.

The vulnerable code is in `/ports/csharp/Jison/Jison/csharp.js` at [csharp.js#L19](#)

Code 176 Bytes	<a href="#">Wrap lines</a> <a href="#">Copy</a> <a href="#">Download</a>
1 console.log("Executing: " + "jison " + process.argv[2]);	
2	
3 exec("jison " + process.argv[2], function (error) {	
4     if (error) {	
5         console.log(error);	
6         return;	
7     }	

### Steps To Reproduce:

1. Installing Jison command-line tool via `npm install jison -g`
2. Obtaining Jison parsing templates: `git clone https://github.com/zaach/jison`
3. `cd jison/ports/csharp/Jison/Jison/`
4. Payload: `node csharp.js "echo">pwncd"`
5. Check if the attack was successful or not (dummy payload was executed or not): `ls -la`

Similarly, `/ports/php/php.js` is vulnerable too as it contains the same blob ([php.js#L19](#)). `""` was added just to isolate the payload.

### Patch

Sanitizing the input. Using `execFile` (this method signatures force developers to separate the command and its arguments)

### Supporting Material/References:

- Windows 10 1803 (OS Build 17134.950)
- NodeJS Version: v10.16.3
- NPM Version: 6.9.0

### Wrap up

- I contacted the maintainers to let them know M

Arbitrary OS command execution.


1 attachment:  
F576968: json-osi.png

 l\_analyst\_layla HackerOne triage changed the status to Triaged.  
Hello @0x48piraj,


Sep 7th (3 years ago)

Thank you for your submission! We were able to validate your report, and have submitted it to the appropriate remediation team for review. They will let us know the final ruling on this report, and when/if a fix will be implemented. Please note that the status and severity are subject to change.


Regards,  
@bassguitar

 0x48piraj posted a comment.  
That sounds great, thanks for the response!!


Sep 14th (3 years ago)

 0x48piraj posted a comment.  
Any updates?

Dec 5th (3 years ago)


 0x48piraj posted a comment.  
6 months, updates?

Feb 29th (3 years ago)

 nasr0x01 posted a comment.  
Sorry about the late @0x48piraj, I now pinged the internal team again and will get back to you as soon as there's more to share.

Mar 6th (3 years ago)


Regards,  
@nasr0x01

 marcinhoppe Node.js third-party modules staff closed the report and changed the status to Resolved.


Apr 28th (3 years ago)

 marcinhoppe Node.js third-party modules staff requested to disclose this report.

Apr 28th (3 years ago)

 This report has been disclosed.

May 28th (3 years ago)

 marcinhoppe Node.js third-party modules staff changed the scope from **None** to **json**.

Jun 18th (3 years ago)