ShwethaKalyanaraman/atom:apm/package.json

Snapshot taken 19 hours ago.

Retest now

Vulnerabilities	6 via 45 paths
Dependencies	509
Source	GitHub
Taken by	Recurring
Tested with	package-lock.json,package.json
Repository	atom
Branch	master
Manifest	apm/package.json

HIGH SEVERITY

Arbitrary Command Injection

Vulnerable module: open

Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › open@0.0.5

Remediation: No remediation path available.

Overview

open Open a file or url in the user's preferred application.

Affected versions of this package are vulnerable to Arbitrary Command Injection. Urls are not properly escaped before concatenating them into the command that is opened using `exec()` .

MEDIUM SEVERITY

Denial of Service (DoS)

Vulnerable module: mem

Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libnpx@10.2.0 › yargs@11.0.0 › os-locale@2.1.0 › mem@1.1.0

Remediation: No remediation path available.

Overview

mem is an optimization technique used to speed up consecutive function calls by caching the result of calls with identical input.

Affected versions of this package are vulnerable to Denial of Service (DoS) attacks. Old results are not deleted from the cache and could cause a memory leak.

MEDIUM SEVERITY

Insecure Randomness

Vulnerable module: cryptiles
Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › hawk@3.1.3 › cryptiles@2.0.5

Remediation: Your dependencies are out of date, otherwise you would be using a newer cryptiles than cryptiles@2.0.5. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › npm-registry-client@8.5.1 › request@2.81.0 › hawk@3.1.3 › cryptiles@2.0.5

Remediation: Your dependencies are out of date, otherwise you would be using a newer cryptiles than cryptiles@2.0.5. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › hawk@3.1.3 › cryptiles@2.0.5

Remediation: Your dependencies are out of date, otherwise you would be using a newer cryptiles than cryptiles@2.0.5. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

…and 2 more

Overview

cryptiles is a package for general crypto utilities.

Affected versions of this package are vulnerable to Insecure Randomness. The `randomDigits()` method is supposed to return a cryptographically strong pseudo-random data string, but it was biased to certain digits. An attacker could be able to guess the created digits.

MEDIUM SEVERITY

Time of Check Time of Use (TOCTOU)

Vulnerable module: chownr
Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libnpmhook@4.0.1 › npm-registry-fetch@3.1.1 › make-fetch-happen@4.0.1 › cacache@11.0.2 › chownr@1.0.1

registry-fetch@3.1.1 › make-fetch-happen@4.0.1 › cacache@11.0.2 › chownr@1.0.1

Remediation: No remediation path available.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › pacote@8.1.6 › make-fetch-happen@4.0.1 › cacache@11.0.2 › chownr@1.0.1

Remediation: No remediation path available.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › npm-registry-fetch@1.1.0 › make-fetch-happen@3.0.0 › cacache@10.0.4 › chownr@1.0.1

Remediation: No remediation path available.

…and 10 more

Overview

Affected versions of chownr are vulnerable to Time of Check Time of Use (TOCTOU). It does not dereference symbolic links and changes the owner of the link.

LOW SEVERITY

Prototype Pollution

Vulnerable module: hoek
Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › hawk@3.1.3 › boom@2.10.1 › hoek@2.16.3

Remediation: Your dependencies are out of date, otherwise you would be using a newer hoek than hoek@2.16.3. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › hawk@3.1.3 › cryptiles@2.0.5 › boom@2.10.1 › hoek@2.16.3

Remediation: Your dependencies are out of date, otherwise you would be using a newer hoek than hoek@2.16.3. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › hawk@3.1.3 › sntp@1.0.9 › hoek@2.16.3

Remediation: Your dependencies are out of date, otherwise you would be using a newer hoek than hoek@2.16.3. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

…and 17 more

Overview

hoek is a Utility methods for the hapi ecosystem.

Affected versions of this package are vulnerable to Prototype Pollution. The utilities function allow modification of the `Object` prototype. If an attacker can control part of the structure passed to this function, they could add or modify an existing property.

PoC by Olivier Arteau (HoLyVieR)

```
var Hoek = require('hoek');
var malicious_payload = '{"__proto__":{"oops":"It works !"}}' ;

var a = {};
console.log("Before : " + a.oops);

Hoek.merge({}, JSON.parse(malicious_payload));
console.log("After : " + a.oops);
```

LOW SEVERITY

Prototype Pollution

Vulnerable module: extend

Introduced through: atom-package-manager@2.1.2

Detailed paths and remediation

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › libcipm@2.0.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › extend@3.0.1

Remediation: Your dependencies are out of date, otherwise you would be using a newer extend than extend@3.0.1. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › npm-lifecycle@2.0.3 › node-gyp@3.7.0 › request@2.81.0 › extend@3.0.1

Remediation: Your dependencies are out of date, otherwise you would be using a newer extend than extend@3.0.1. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

Introduced through: atom-bundled-apm@* › atom-package-manager@2.1.2 › npm@6.2.0 › npm-registry-client@8.5.1 › request@2.81.0 › extend@3.0.1

Remediation: Your dependencies are out of date, otherwise you would be using a newer extend than extend@3.0.1. Try reinstalling your dependencies. If the problem persists, one of your dependencies may be bundling outdated modules.

…and 2 more

Overview

extend is a port of the classic extend() method from jQuery.

Affected versions of this package are vulnerable to Prototype Pollution. Utilities function can be tricked into modifying the prototype of "Object" when the attacker control part of the structure passed to these function. This can let an attacker add or modify existing property that will exist on all object.

API Status Vulnerability DB Blog Documentation