



STATIC CODE ANALYSIS

Using HCL AppScan



NOVEMBER 29, 2020

SLICE

By Shreyas R Gurjar

		Number of Issues
Private data logged sink	15	
Insecure Use of setTimeout	16	
Insecure Use of Document.Write	7	
Insecure Configuration of Private Data Exposed via HTTP URL	2	
Insecure Use of Remote OS Command Execution	1	
Validation.Required Insecure Use of setInterval	1	

Critical
 High
 Medium
 Low
 Informational

WHY HCL AppScan ?

***HCL AppScan** help minimize web application attacks and expensive data breaches by automating testing of application security vulnerabilities. It allows you to test applications before deploying them and assess risk in production environments on an ongoing basis.*

Testing GitHub Repo

<https://github.com/winstonjs/winston>

Description: Private Data Logged Sink

The detected NodeJS code contains a console log or logger entry that uses a variable and/or string concatenation. This could allow an attacker to inject code to generate false log entries and/or log entries with sensitive data via Log Injection.

```
var msg = "127.0.0.1 - there's no place like home"; var loopIp = "127.0.0.1"; var thatPlace = "like home"; var count = 5; const name = 'Will Robinson'; console.info('count: %d', count); console.warn('Danger ${name}! Danger!'); console.error(new Error('Whoops, something bad happened')); logger.log('error', msg); logger.log('debug', "127.0.0.1 - there's no place" + thatPlace); logger.info(msg); logger.warn(loopIp + " - there's no place like home"); logger.error("127.0.0.1 - there's no place " + thatPlace);
```

Mitigation:

Always ensure any ExpressJS cookie session are configured with "httpOnly" set to "true".

```
console.assert(false, 'Whoops %s', 'didn\'t work'); console.log('count: 5'); console.trace('Show me'); logger.log('warn', "127.0.0.1 - there's no place like home"); logger.log('error', "127.0.0.1 - there's no place like home"); logger.info("127.0.0.1 - there's no place like home"); logger.warn("127.0.0.1 -
```

M

Medium - Private Data Logged Sink

Status: New

Issue No: 1

Severity: **Medium**

Status: New

Classification: Definitive

Location: logger.log('info', 'Hello created log files!', { 'foo': 'bar' });

Line: 26

Source File: \winston-master\winston-master\examples\create-file.js

Availability Impact: Partial

Confidentiality Impact: Partial

Integrity Impact: Partial

Date Created: Nov 29, 2020

Source: logger.log('info', 'Hello created log files!', { 'foo': 'bar' });

there's no place like home"); logger.error("127.0.0.1 - there's no place like home");

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	2
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.info('Hello, this is a logging event with a custom pretty print', { 'foo': 'bar' });
Line	17
Source File	\winston-master\winston-master\examples\custom-pretty-print.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.info('Hello, this is a logging event with a custom pretty print', { 'foo': 'bar' });

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	3
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.info('This will also print without { level }', { 'foo': 'bar' });
Line	22
Source File	\winston-master\winston-master\examples\delete-level.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.info('This will also print without { level }', { 'foo': 'bar' });

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	4
Severity:	Medium
Status	New
Classification	Definitive
Location	<pre>logger.warn(new Error('Error passed as info')); logger.log('error', new Error('Error passed as message')); logger.warn('Maybe important error: ', new Error('Error passed as meta')); logger.log('error', 'Important error: ', new Error('Error passed as meta')); logger.error(new Error('Error as info'));</pre>
Line	14,15,16,17,18,20
Source File	\\winston-master\\winston-master\\examples\\error.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	<pre>logger.warn(new Error('Error passed as info')); logger.log('error', new Error('Error passed as message')); logger.warn('Maybe important error: ', new Error('Error passed as meta')); logger.log('error', 'Important error: ', new Error('Error passed as meta')); logger.error(new Error('Error as info'));</pre>

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	5
Severity:	Medium
Status	New
Classification	Definitive
Location	<pre>logger.info(`test=\${'a'.repeat(245)}`); logger.info(`test=\${'b'.repeat(245)}`); logger.info(`test=\${'c'.repeat(245)}`); setTimeout(() => { logger.info(`test=\${'d'.repeat(245)}`); logger.info(`test=\${'e'.repeat(245)}`); logger.info(`test=\${'f'.repeat(245)}`); }, 2000);</pre>
Line	18,19,20,23,24,25
Source File	\\winston-master\\winston-master\\examples\\file-maxsize.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	<pre>logger.info(`test=\${'a'.repeat(245)}`); logger.info(`test=\${'b'.repeat(245)}`); logger.info(`test=\${'c'.repeat(245)}`); setTimeout(() => { logger.info(`test=\${'d'.repeat(245)}`); logger.info(`test=\${'e'.repeat(245)}`); logger.info(`test=\${'f'.repeat(245)}`); }, 2000);</pre>

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	6
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.log({
Line	18,19,20,23,24,25
Source File	\winston-master\winston-master\examples\format-filter.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.log({

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	7
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.info('transaction ok', { creditCard: 123456789012345 });
Line	18,19,20,23,24,25
Source File	\winston-master\winston-master\examples\format-mutate.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.info('transaction ok', { creditCard: 123456789012345 });

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	8
Severity:	Medium
Status	New
Classification	Definitive
Location	<pre> logger.info('Found %s at %s', 'error', new Date()); logger.info('Found %s at %s', 'error', new Error('chill winston')); logger.info('Found %s at %s', 'error', /WUT/); logger.info('Found %s at %s', 'error', true); logger.info('Found %s at %s', 'error', 100.00); logger.info('Found %s at %s', 'error', ['1, 2, 3']); </pre>
Line	24,25,26,27,28,29
Source File	\\winston-master\\winston-master\\examples\\interpolation.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	<pre> logger.info('Found %s at %s', 'error', new Date()); logger.info('Found %s at %s', 'error', new Error('chill winston')); logger.info('Found %s at %s', 'error', /WUT/); logger.info('Found %s at %s', 'error', true); logger.info('Found %s at %s', 'error', 100.00); logger.info('Found %s at %s', 'error', ['1, 2, 3']); </pre>

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	9
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.log(
Line	17,18
Source File	\winston-master\winston-master\examples\splat.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.log(

Insecure Use of setTimeout:**Description:**

The detected NodeJS code contains a setTimeout function containing one or more arguments with a variable and/or string concatenation. This could be user input that may contain malicious code from an attacker.

```
string = 'boo'; setTimeout(function (str) { console.log(str) }, 100, string);
```

Mitigation:

Validate variables and avoid string concatenation.

```
setTimeout(function (str) { console.log(str) }, 100, 'boo')
```

H**High - Insecure Use of setTimeout**

Status:	New
Issue No:	10
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(() => {
Line	22
Source File	\winston-master\winston-master\examples\file-maxsize.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(() => {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	11
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(() => {
Line	21
Source File	\winston-master\winston-master\examples\simple-stream.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(() => {

Insecure Use of Document.Write:

Description:

Content being passed into the `document.write()` or the `document.writeln()` method could be carrying tainted data. Since this data is further used to display the HyperText Markup Language (HTML) inside a page, take steps to validate it.

The Document Object Model (DOM) as defined by the World Wide Web Consortium (W3C) provides two methods for controlling output being written to the document: `document.write()` and `document.writeln()`.

Content inputted to `document.write()` or the `document.writeln()` method should not be trusted without further validation. Using such content without validation can lead to a variety of Cross Site Scripting and HTML Injection attacks.

```
<script language="javascript"> ... var inText = <%request.getParameter("text_type")%>;
document.write("The text inputted is: " + inText); ... </script>
```

Mitigation:

In the cases where these methods carry input originating from an external location, further validation on the input is recommended.

Validation of such content when it is exposed to the user should be performed in the same manner as server-side input validation. Input validation should not be solely performed by client-side JavaScript components exposed to the user. Validation implemented in this manner can be bypassed by means of manipulating the browser and/or the submitted request. Use Server-Side validation or a combination of Client-Side and Server-Side validation.

Furthermore, any unnecessary content should not be passed to these methods.

```
<script language="javascript"> ... var inText = <%request.getEncodedParameter("text_type")%>;
document.write("The text inputted is: " + inText); ... </script>
```

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	12
Severity:	High
Status	New
Classification	Definitive
Location	console._stderr.write(`\${info[MESSAGE]}\${this.eol}`);
Line	51,65,79
Source File	\\winston-master\\winston-master\\lib\\winston\\transports\\console.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	console._stderr.write(`\${info[MESSAGE]}\${this.eol}`);

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	13
Severity:	High
Status	New
Classification	Definitive
Location	const written = this._stream.write(output, logged.bind(this));
Line	51,65,79
Source File	\\winston-master\\winston-master\\lib\\winston\\transports\\file.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	const written = this._stream.write(output, logged.bind(this));

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	14
Severity:	High
Status	New
Classification	Definitive
Location	this._stream.write(info); this._stream.write(`\${info[MESSAGE]}\${this.eol}`);
Line	50,57
Source File	\\winston-master\\winston-master\\lib\\winston\\transports\\stream.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	this._stream.write(info); this._stream.write(`\${info[MESSAGE]}\${this.eol}`);

Description: Private Data Logged Sink

The detected NodeJS code contains a console log or logger entry that uses a variable and/or string concatenation. This could allow an attacker to inject code to generate false log entries and/or log entries with sensitive data via Log Injection.

```
var msg = "127.0.0.1 - there's no place like home"; var loopIp = "127.0.0.1"; var thatPlace = "like home"; var count = 5; const name = 'Will Robinson'; console.info('count: %d', count); console.warn('Danger ${name}! Danger!'); console.error(new Error('Whoops, something bad happened')); logger.log('error', msg); logger.log('debug', "127.0.0.1 - there's no place" + thatPlace); logger.info(msg); logger.warn(loopIP + " - there's no place like home"); logger.error("127.0.0.1 - there's no place " + thatPlace);
```

Mitigation:

Always ensure any ExpressJS cookie session are configured with "httpOnly" set to "true".

```
console.assert(false, 'Whoops %s', 'didn\'t work'); console.log('count: 5'); console.trace('Show me'); logger.log('warn', "127.0.0.1 - there's no place like home"); logger.log('error', "127.0.0.1 - there's no place like home"); logger.info("127.0.0.1 - there's no place like home"); logger.warn("127.0.0.1 - there's no place like home"); logger.error("127.0.0.1 - there's no place like home");
```

H

High - Private Data Logged Sink

Status:	New
Issue No:	15
Severity:	High
Status	New
Classification	Definitive
Location	console.error(info[MESSAGE]); console.warn(info[MESSAGE]); console.warn(info[MESSAGE]);
Line	54,68,82
Source File	\winston-master\winston-master\lib\winston\transports\console.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	console.error(info[MESSAGE]); console.warn(info[MESSAGE]); console.warn(info[MESSAGE]);

Insecure Configuration of Private Data Exposed via HTTP URL:

Description:

The following NodeJS code is specifying a URL using HTTP, which is an insecure connection. Any data transmitted to the site will be sent unencrypted.

```
var http = require('http'); var http = require('http'); document.URL = "http://www.website.com"
```

Mitigation:

Always connect securely using an HTTPS protocol URL.

```
var http = require('https'); document.URL = https://www.website.com
```

L

Low - Insecure Configuration of Private Data Exposed via HTTP URL

Status:	New
Issue No:	16
Severity:	LOW
Status	New
Classification	Definitive
Location	const http = require('http');
Line	10
Source File	\winston-master\winston-master\lib\winston\transports\http.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	const http = require('http');

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	17
Severity:	High
Status	New
Classification	Definitive
Location	self.write(info);
Line	81
Source File	\winston-master\winston-master\lib\winston\create-logger.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	self.write(info);

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	18
Severity:	High
Status	New
Classification	Definitive
Location	this.write(Object.assign({}, this.defaultMeta, {
Line	251
Source File	\winston-master\winston-master\lib\winston\logger.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	19
Severity:	High
Status	New
Classification	Definitive
Location	let data = decode.write(buffer.slice(0, bytes));
Line	81
Source File	\\winston-master\\winston-master\\lib\\winston\\tail-file.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	let data = decode.write(buffer.slice(0, bytes));

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	20
Severity:	High
Status	New
Classification	Definitive
Location	timeout = setTimeout(gracefulExit, 3000);
Line	225
Source File	\\winston-master\\winston-master\\examples\\exception-handler.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	timeout = setTimeout(gracefulExit, 3000);

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	21
Severity:	High
Status	New
Classification	Definitive
Location	timeout = setTimeout(gracefulExit, 3000);
Line	231
Source File	\winston-master\winston-master\examples\rejection-handler.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	timeout = setTimeout(gracefulExit, 3000);

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	21
Severity:	High
Status	New
Classification	Definitive
Location	return setTimeout(read, 1000);
Line	87
Source File	\winston-master\winston-master\examples\tail-file.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	return setTimeout(read, 1000);

H**High - Private Data Logged Sink**

Status:	New
Issue No:	22
Severity:	High
Status	New
Classification	Definitive
Location	console.error('[winston] Unknown logger level: %s', info[LEVEL]); console.error(
Line	290,296
Source File	\\winston-master\\winston-master\\lib\\winston\\transports\\logger.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	console.error('[winston] Unknown logger level: %s', info[LEVEL]); console.error(

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	23
Severity:	Medium
Status	New
Classification	Definitive
Location	it('logger.log(level, error)', (done) => { logger.log('info', new Error('Errors lack .toJSON() lulz')); logger.log('info', err); logger.log('info', 'Caught error:', new Error('Errors lack .toJSON() lulz'));
Line	39,45,62,113
Source File	\\winston-master\\winston-master\\test\\formats\\errors-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	it('logger.log(level, error)', (done) => { logger.log('info', new Error('Errors lack .toJSON() lulz')); logger.log('info', err); logger.log('info', 'Caught error:', new Error('Errors lack .toJSON() lulz'));

H

High - Insecure Use of Document.Write

Status:	New
Issue No:	24
Severity:	High
Status	New
Classification	Definitive
Location	process.stdout.write(err.message);
Line	13
Source File	\winston-master\winston-master\test\helpers\scripts\exit-on-error.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	process.stdout.write(err.message);

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	25
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	28
Source File	\winston-master\winston-master\examples\default-rejection.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	25
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	24
Source File	\\winston-master\\winston-master\\examples\\exit-on-error.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	26
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	23
Source File	\\winston-master\\winston-master\\examples\\log-exception.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	27
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	30
Source File	\winston-master\winston-master\examples\log-rejection.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	28
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	25
Source File	\winston-master\winston-master\examples\unhandle-exceptions.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

Insecure Use of Remote OS Command Execution:

Description:

The detected NodeJS code contains a `child_process "exec"` method, used to execute a command under an external shell, using a variable and/or string concatenation. The command provided could be user input that may contain malicious code from an attacker.

```
child_process.exec(userCode);
```

Mitigation:

Validate any variables and avoid any string concatenation.

```
child_process.exec('echo "The %$HOME variable is $HOME"');
```

C

Critical- Remote OS Command Execution

Status:	New
Issue No:	29
Severity:	High
Status	New
Classification	Definitive
Location	exec('rm -rf ' + path.join(__dirname, '..', 'fixtures', 'logs', 'testmaxfiles*'), this.callback);
Line	37
Source File	\\winston-master\\winston-master\\test\\transports\\file-maxfiles-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	exec('rm -rf ' + path.join(__dirname, '..', 'fixtures', 'logs', 'testmaxfiles*'), this.callback);

Insecure Use of setInterval:

Description:

The detected NodeJS code contains an setInterval function containing an argument(s) with a variable and/or string concatenation. This could be user input that may contain malicious code from an attacker.

```
var counter = 0; var short = 1; setInterval(function() { counter ++; }, short);
```

Mitigation:

Validate any variables and avoid any string concatenation.

```
setInterval(function() { console.log("hello"); }, "short");
```

H

High - Insecure Use of setInterval

Status:	New
Issue No:	30
Severity:	High
Status	New
Classification	Definitive
Location	const interval = setInterval(function () {
Line	43,82
Source File	\\winston-master\\winston-master\\test\\transports\\00-file-stress-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	const interval = setInterval(function () {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	31
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function() {
Line	47,90,132,
Source File	\winston-master\winston-master\test\transports\00-file-stress-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function() {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	32
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(() => logKbytes(4), 100);
Line	46
Source File	\winston-master\winston-master\test\transports\01-file-maxsize-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(() => logKbytes(4), 100);

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	33
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(() => logKbytes(1, created), 1);
Line	70
Source File	\\winston-master\\winston-master\\test\\transports\\file-archive-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(() => logKbytes(1, created), 1);

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	34
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(() => logKbytes(1, created), 1);
Line	62
Source File	\\winston-master\\winston-master\\test\\transports\\file-tailrolling-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(() => logKbytes(1, created), 1);

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	35
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.info(++counters.write); logger.info(msg); msgs.forEach(msg => logger.info(msg));
Line	44,87,130
Source File	\winston-master\winston-master\test\transports\00-file-stress-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	logger.info(++counters.write); logger.info(msg); msgs.forEach(msg => logger.info(msg));

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	36
Severity:	Medium
Status	New
Classification	Definitive
Location	logger.info('log twice 2'); // this raises the `mainError` for the transport logger.info('log four times 4'); // this raises the `otherError` for the transport
Line	65,104
Source File	\winston-master\winston-master\test\transports\error-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020

L

Insecure Configuration of Private Data Exposed via HTTP URL

Status:	New
Issue No:	37
Severity:	LOW
Status	New
Classification	Definitive
Location	const http = require('http');
Line	8
Source File	\winston-master\winston-master\lib\winston\transports\http-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	const http = require('http');

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	38
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(function () {
Line	65
Source File	\winston-master\winston-master\test\log-exception-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(function () {

H

High - Insecure Use of setTimeout

Status:	New
Issue No:	39
Severity:	High
Status	New
Classification	Definitive
Location	setTimeout(done.bind(null, null, counters), 100);
Line	65
Source File	\winston-master\winston-master\test\tail-file-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Date Created	Nov 29, 2020
Source:	setTimeout(done.bind(null, null, counters), 100);

M

Medium - Private Data Logged Sink

Status:	New
Issue No:	40
Severity:	Medium
Status	New
Classification	Definitive
Location	extendedLogger.log('info', { test: 1 }); extendedLogger.log('warn', { test: 2 });
Line	99,100
Source File	\winston-master\winston-master\test\logger-test.js
Availability Impact	Partial
Confidentiality Impact	Partial
Integrity Impact	Partial
Date Created	Nov 29, 2020
Source:	extendedLogger.log('info', { test: 1 }); extendedLogger.log('warn', { test: 2 });