Inefficient Regular Expression Complexity in shescape

Low ericcornelissen published GHSA-gp75-h7j6-5pv3 on Aug 29

Package

shescape (npm)

Affected versions	Patched versions
>=1.5.1 <1.5.10	1.5.10

Description

Impact

This impacts users that use Shescape to escape arguments:

- for the Unix shells **Bash** and **Dash**, or any not-officially-supported Unix shell;
- using the `escape` or `escapeAll` functions with the `interpolation` option set to `true`.

An attacker can cause polynomial backtracking or quadratic runtime in terms of the input string length due to two Regular Expressions in Shescape that are vulnerable to Regular Expression Denial of Service (ReDoS). Example:

```
import * as shescape from "shescape";

/* 1. Prerequisites */
const options = {
  interpolation: true,
  // and
  shell: "/bin/bash",
  // or
  shell: "/bin/dash",
  // or
  shell: "some-not-officially-supported-shell",
  // or
  shell: undefined, // Only if the default shell is one of the affected shells.
};
```

```
/* 2. Attack */
let userInput = `foo${"{".repeat(150_000)}bar`; // quadratic runtime
// or
userInput = `=${":".repeat(150_000)}foobar`; // quadratic runtime
// or
userInput = `{${",".repeat(150_000)}`; // polynomial backtracking

/* 3. Usage */
shescape.escape(userInput, options);
// or
shescape.escapeAll([userInput], options);
```

Patches

This bug has been patched in v1.5.10 which you can upgrade to now. No further changes required.

For **Dash** only, this bug has been patched since v1.5.9 which you can upgrade to now. No further changes required.

Workarounds

Alternatively, a maximum length can be enforced on input strings to Shescape to reduce the impact of the vulnerability. It is not recommended to try and detect vulnerable input strings, as the logic for this may end up being vulnerable to ReDoS itself.

References

- Shescape Pull Request #373
- Shescape Release v1.5.10

For more information

- Comment on #373
- Open an issue at https://github.com/ericcornelissen/shescape/issues (*New issue* > *Question* > *Get started*)

Severity

Low

CVE ID

CVE-2022-36064

Weaknesses

CWE-1333