

# cloudcmd

github.com/coderaiser/cloudcmd

Scanned 30 October 2026 at 09:26 · Report generated 30 October 2026 · JavaScript · 2,037 stars

OVERALL ASSESSMENT

Medium

Combined score 77 · 174 findings after filtering

## Summary

34 findings likely need attention, 140 look like noise. Most notable: This Dockerfile builds on a Node.js image that is affected by CVE-2026-111497 (Brotli d... (94% confidence).

Generated from finding counts. A plain-English summary was not available for this scan.

## Risk, by type

MalDet scores two different questions separately, because they call for different responses: code can be badly written without being hostile, and hostile code is not always badly written.

VULNERABILITY

Medium (74)

Insecure coding an attacker could exploit. Usually mistakes, and usually fixable in place.

MALICIOUS PATTERN

Low (3)

Signs the code may be trying to do harm: hidden payloads, credential access, persistence, exfiltration.

## What each detector found

| DETECTOR    | LOOKS FOR                                         | HIGH | MEDIUM | LOW | TOTAL |
|-------------|---------------------------------------------------|------|--------|-----|-------|
| dep_checker | Vulnerable, unpinned or typosquatted dependencies | 0    | 0      | 130 | 130   |
| semgrep     | Insecure coding patterns across many languages    | 2    | 33     | 0   | 35    |
| yara        | Signatures of malicious behaviour in source files | 8    | 0      | 1   | 9     |

34 of 174 findings were rated more likely genuine than noise by the classifier.

## Evidence

Identical findings are grouped, so a single rule firing many times takes one row. Ordered worst first. “Seen” is how many times that finding occurred.

| SEVERITY | DETECTOR | FINDING                                                                                                                                                                                                                                                                                                      | SEEN | EXAMPLE LOCATION                           |
|----------|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|--------------------------------------------|
| HIGH     | semgrep  | This Dockerfile builds on a Node.js image that is affected by CVE-2026-111497 (Brotli decoder out-of-bounds write, remote code execution) or whose patch level cannot be verified from the tag. Pin to a fixed release 22.23.4, 24.22.1, 26.12.1 or later on the same line. Node.js 20 and older, 23 and 25  | 1    | /docker/<br>Dockerfile:<br>1               |
| HIGH     | semgrep  | By not specifying a USER, a program in the container may run as 'root'. This is a security hazard. If an attacker can control a process running as root, they may have control over the container. Ensure that the last USER in a Dockerfile is a USER other than 'root'.                                    | 1    | /docker/<br>Dockerfile:<br>29              |
| HIGH     | yara     | YARA rule matched:<br>detect_download_and_execute — Downloads a remote resource AND executes it - the actually dangerous combination, not just downloading alone                                                                                                                                             | 5    | /docker/<br>Dockerfile.<br>io              |
| HIGH     | yara     | YARA rule matched:<br>detect_obfuscated_string_execution — Reconstructs a string character-by-character or from hex escapes, then immediately executes it - fromCharCode/chr() alone are common in ordinary text-processing code, so this requires both the reconstruction and an execution call together    | 1    | /client/<br>listeners/<br>index.js         |
| HIGH     | yara     | YARA rule matched:<br>detect_persistence_mechanism — Modifies a startup/scheduling location - legitimate application code essentially never does this                                                                                                                                                        | 2    | /docker/<br>Dockerfile.<br>io              |
| MEDIUM   | semgrep  | RegExp() called with a `prefix` function argument, this might allow an attacker to cause a Regular Expression Denial-of-Service (ReDoS) within your application as RegExp blocks the main thread. For this reason, it is recommended to use hardcoded regexes instead. If your regex is run on user-controll | 1    | /client/<br>dom/<br>current-<br>file.js:73 |

| SEVERITY      | DETECTOR | FINDING                                                                                                                                                                                                                                                                                                      | SEEN | EXAMPLE LOCATION                      |
|---------------|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|---------------------------------------|
| <b>MEDIUM</b> | semgrep  | RegExp() called with a `ext` function argument, this might allow an attacker to cause a Regular Expression Denial-of-Service (ReDoS) within your application as RegExp blocks the main thread. For this reason, it is recommended to use hardcoded regexes instead. If your regex is run on user-controlled  | 2    | /client/modules/view/types.js:7       |
| <b>MEDIUM</b> | semgrep  | RegExp() called with a `wildcard` function argument, this might allow an attacker to cause a Regular Expression Denial-of-Service (ReDoS) within your application as RegExp blocks the main thread. For this reason, it is recommended to use hardcoded regexes instead. If your regex is run on user-contro | 1    | /common/util.js:24                    |
| <b>MEDIUM</b> | semgrep  | RegExp() called with a `char` function argument, this might allow an attacker to cause a Regular Expression Denial-of-Service (ReDoS) within your application as RegExp blocks the main thread. For this reason, it is recommended to use hardcoded regexes instead. If your regex is run on user-controlled | 1    | /client/key/set-current-by-char.js:12 |
| <b>MEDIUM</b> | semgrep  | Service 'web' allows for privilege escalation via setuid or setgid binaries. Add 'no-new-privileges:true' in 'security_opt' to prevent this.                                                                                                                                                                 | 1    | /docker-compose.yml:3                 |
| <b>MEDIUM</b> | semgrep  | Service 'web' is running with a writable root filesystem. This may allow malicious applications to download and run additional payloads, or modify container files. If an application inside a container has to save something temporarily consider using a tmpfs. Add 'read_only: true' to this service to  | 1    | /docker-compose.yml:3                 |
| <b>MEDIUM</b> | semgrep  | Checks for setting the environment variable NODE_TLS_REJECT_UNAUTHORIZED to 0, which disables TLS verification. This should only be used for debugging purposes. Setting the option rejectUnauthorized to false bypasses verification against the list of trusted CAs, which also leads to insecure transpor | 1    | /server/distribute/import.js:94       |

| SEVERITY      | DETECTOR    | FINDING                                                                                                                                                                                                                                                                                                      | SEEN | EXAMPLE LOCATION                 |
|---------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------|----------------------------------|
| <b>MEDIUM</b> | semgrep     | GitHub Actions step uses a mutable tag or branch reference. Tags and branch names can be silently repointed by the action owner, enabling supply-chain attacks — as seen in the trivy-action and kics-github-action compromises. Pin the reference to a full 40-character commit SHA instead, e.g. `uses: ac | 25   | /.github/workflows/nodejs.yml:36 |
| <b>LOW</b>    | dep_checker | Loose version constraint: '@cloudcmd/read-files-sync^2.0.0' — attacker could push malicious update                                                                                                                                                                                                           | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'http-auth^4.2.1' — attacker could push malicious update                                                                                                                                                                                                                           | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'url-loader^4.0.0' — attacker could push malicious update                                                                                                                                                                                                                          | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'package-json^10.0.0' — attacker could push malicious update                                                                                                                                                                                                                       | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'cssnano-preset-default^9.0.1' — attacker could push malicious update                                                                                                                                                                                                              | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: '@cloudcmd/create-element^2.0.0' — attacker could push malicious update                                                                                                                                                                                                            | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: '@cloudcmd/move-files^9.0.1' — attacker could push malicious update                                                                                                                                                                                                                | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'just-snake-case^3.2.0' — attacker could push malicious update                                                                                                                                                                                                                     | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'just-pascal-case^3.2.0' — attacker could push malicious update                                                                                                                                                                                                                    | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'just-capitalize^3.2.0' — attacker could push malicious update                                                                                                                                                                                                                     | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'html-webpack-plugin^5.6.3' — attacker could push malicious update                                                                                                                                                                                                                 | 1    | package.json                     |
| <b>LOW</b>    | dep_checker | Loose version constraint: 'request^2.76.0' — attacker could push malicious update                                                                                                                                                                                                                            | 1    | package.json                     |

| SEVERITY | DETECTOR    | FINDING                                                                                  | SEEN | EXAMPLE LOCATION |
|----------|-------------|------------------------------------------------------------------------------------------|------|------------------|
| LOW      | dep_checker | Loose version constraint: 'files-io^4.0.0' — attacker could push malicious update        | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'for-each-key^2.0.0' — attacker could push malicious update    | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'minor^1.2.2' — attacker could push malicious update           | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'madrun^13.0.0' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'montag^2.0.1' — attacker could push malicious update          | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'morgan^1.6.1' — attacker could push malicious update          | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'multi-rename^3.0.0' — attacker could push malicious update    | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'nano-memoize^3.0.16' — attacker could push malicious update   | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'nodemon^3.0.1' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'nomine^5.0.2' — attacker could push malicious update          | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'once^1.4.0' — attacker could push malicious update            | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'onezip^7.0.0' — attacker could push malicious update          | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'open^11.0.0' — attacker could push malicious update           | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'path-browserify^1.0.1' — attacker could push malicious update | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'philip^3.0.0' — attacker could push malicious update          | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'pipe-io^4.0.1' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'place^1.1.4' — attacker could push malicious update           | 1    | package.js<br>on |

| SEVERITY | DETECTOR    | FINDING                                                                                | SEEN | EXAMPLE LOCATION |
|----------|-------------|----------------------------------------------------------------------------------------|------|------------------|
| LOW      | dep_checker | Loose version constraint: 'ponse^8.0.0' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'postcss^8.5.3' — attacker could push malicious update       | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'process^0.11.10' — attacker could push malicious update     | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'pullout^5.0.0' — attacker could push malicious update       | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'putout^42.0.5' — attacker could push malicious update       | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'qword^1.0.0' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'unionfs^4.0.0' — attacker could push malicious update       | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'superc8^12.6.0' — attacker could push malicious update      | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'supermenu^5.0.0' — attacker could push malicious update     | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'supertape^13.0.0' — attacker could push malicious update    | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'table^6.0.1' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'tar-stream^3.0.0' — attacker could push malicious update    | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'tryrequire^3.0.0' — attacker could push malicious update    | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'win32^8.0.0' — attacker could push malicious update         | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'yargs-parser^22.0.0' — attacker could push malicious update | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'writejson^3.0.0' — attacker could push malicious update     | 1    | package.js<br>on |
| LOW      | dep_checker | Loose version constraint: 'wraptile^3.0.0' — attacker could push malicious update      | 1    | package.js<br>on |

| SEVERITY | DETECTOR    | FINDING                                                                               | SEEN | EXAMPLE LOCATION |
|----------|-------------|---------------------------------------------------------------------------------------|------|------------------|
| LOW      | dep_checker | Loose version constraint: 'style-loader^4.0.0' — attacker could push malicious update | 1    | package.js<br>on |

Showing the 60 most significant of 144 distinct findings. The counts in the previous section cover all of them.

## How to read this report

- **This is a static scan.** The code was read, never executed. Behaviour that only appears at runtime is outside what any of these detectors can see.
- **Findings are not all equally real.** A classifier rates each finding on how likely it is to be genuine rather than noise, and that rating weights the scores. Measured against a hand-reviewed sample it reached precision 0.52 and recall 0.62 — useful for ranking, not a verdict on any single finding.
- **A high score is not proof of malice.** Security tools, penetration-testing frameworks and malware research repositories contain the same patterns as real malware by design. The two risk types above exist to keep that distinction visible.
- **Absence of findings is not proof of safety.** A detector that is not installed, or a language none of them parse, produces silence rather than a warning.
- **This scan is a snapshot.** The repository may have changed since 30 October 2026.