SECURITY OVERVIEW · DISTRIBUTIVE

DCP Worker for Red Hat Enterprise Linux 9

Package **dcp-worker-8.0.1-0.3.dev20261121.cpuonly.el9** Status as of **November 21, 2026** Tested on **RHEL 9.0, 9.2, 9.8**

> The DCP Worker runs compute jobs from the Distributive network on spare CPU capacity. Job code is treated as untrusted. It runs in a separate, locked-down process that cannot reach the network or the disk, on a JavaScript engine at the same patch level as current Chrome stable, and the whole worker runs as an unprivileged service account under strict systemd sandboxing. The package is signed. All 94 JavaScript-engine vulnerabilities disclosed in Chrome security releases from March to November 2026 are fixed.

What is in the package

COMPONENT	ROLE	VERSION
Worker service	Talks to the Distributive scheduler, fetches work, returns results. Runs only Distributive code.	`Node.js 26.13.1 LTS`
Evaluator	Runs job code (JavaScript and WebAssembly) in a sandboxed engine. CPU only; no GPU access.	`V8 15.8.62.61`
Libraries	npm packages for the worker service. Packages a server never loads (Windows event log, desktop session monitoring, build tools) are removed.	`pinned`
systemd units	`dcp-worker`, `dcp-evaluator`, `dcp`. Installed enabled; restart automatically.	–

Everything installs under `/opt/dcp`. Program files are owned by root and read-only to the service. The only writable location is `/opt/dcp/.dcp`, which holds the worker's identity key and is readable only by the service account. The only OS dependency is `libatomic`, installed by dnf from Red Hat repositories.

How job code is contained

A flaw in job handling would have to escape the JavaScript engine, then the evaluator process, then the service account. Each layer is independent.

- **Engine sandbox.** Each job slice runs in its own evaluator process with Node.js system APIs removed. Job code cannot open files, start processes or open sockets.
- **No network.** The operating system blocks the evaluator from connecting to anything except the local worker service.
- **No disk.** The file system is read-only to the evaluator apart from a private temporary directory. Home directories and the worker identity key are hidden from it.
- **No privileges.** Both services run as the `dcp` system account (no login shell), with an empty capability set, `NoNewPrivileges`, no device access and a system-call filter.
- **Kernel protection.** Kernel modules, tunables, logs, clock and control groups are protected; namespaces, real-time scheduling and set-uid are disallowed.
- **Measured.** `systemd-analyze security` rates both services 1.4 to 1.5, "OK", on a 0 to 10 scale where lower is better.

Network footprint

- Outbound HTTPS only, from the worker service to the Distributive scheduler, verified against the system CA store (corporate inspection proxies with an installed root CA work).
- No inbound ports. Two local ports listen on loopback only: `9000` (worker to evaluator) and `9080` (local status page). An optional firewall rule restricts both to the service account and root.
- The organisation's compute groups decide which jobs a machine accepts; the worker can be limited to the organisation's own groups.

Patch status

AREA	STATUS	DETAIL
JavaScript engine (V8)	Current	15.8.62.61 contains the exact V8 shipped in Chrome 158.0.8127.36, the stable release of November 17, 2026, plus 30 later security and stability merges from V8's 15.8 branch (to November 20).
Disclosed V8 CVEs	0 open	All 94 V8, Turbofan, Maglev and WebAssembly CVEs in Chrome security releases from March 24 to November 18, 2026 are fixed. 89 confirmed by locating the fix in the engine source; the other 5 have no public fix commit and are covered by containing Chrome 158's engine.
Node.js runtime	Current	Worker service on 26.13.1, the November 10, 2026 security release of the Node 26 LTS line.
npm libraries	Reduced	Unused packages and their advisories removed. Remaining advisories are in an interactive terminal dashboard the service does not use, plus one low-severity item.

No Node.js release yet carries V8 15.8, so the evaluator embeds it through the Node.js development line (28.0.0-pre). The evaluator uses only the engine and its platform layer; the worker service stays on the Node.js LTS release.

Testing

- Clean installs and upgrades on RHEL 9.0, 9.2 and 9.8: signed install, registration, real compute jobs completed, zero sandbox denials in the system journal, no service restarts.
- Node.js's own test suite against the evaluator's engine build: 5,242 tests passed, none failed. Distributive's evaluator test suite: 45 of 45 passed. V8's published regression tests for this year's engine CVEs that can run outside V8's own test shell all pass.
- Same job on identical hardware: median 482 s per slice, against 479 s on the previous engine; no measurable change.

Verifying the package

The RPM is signed with Distributive's package key. Install with signature checking enforced and compare the fingerprint:

```
24AD 9466 37A3 4EE6 220E 2D3F 5683 DE7D 5F4D 42AB
```

`sudo dnf install --setopt=localpkg_gpgcheck=1 <package>` refuses the package if the signature does not match. After installation, `rpm -V dcp-worker` confirms that no file has changed.

Keeping it current

- The evaluator's engine follows Chrome's stable V8 branch. Updated packages follow Chrome and Node.js security releases.
- A signed dnf repository is planned, so updates arrive through `dnf upgrade` with repository signature checks.