EcomLabs

STEP-BY-STEP GUIDE

Next.js 14 has no patch: your way out

What to do if you are still on Next.js 14 after the October 14, 2026 RCE advisories: move sharp to 0.35.5 while you migrate, upgrade to 15 with the official codemod, and verify versions before you deploy.

FOR	Next.js 14 apps, and 15 apps with an old lockfile
TESTED WITH	Next 14.2.35, 15.5.28 and 16.4.4 · npm 11.11.1 · pnpm 12.16.1 · Node 22.22.3 · Windows 11
UPDATED	November 2, 2026
DOCS	nextjs.org/docs/app/guides/upgrading/version-15
ARTICLE	ecomlabs.dev/g/rce-en
AUTHOR	Luis Morales · @luismoralesdev

Contents

Before you start	What affects you
Step 01	See what you have installed
Step 02	While you migrate: sharp 0.35.5 or later
Step 03	Upgrade from 14 to 15
Step 04	Verify, deploy and plan for 16
When you are done	Checklist

BEFORE YOU START

What affects you

On October 14, 2026, Next.js fixed two Critical RCEs and a DoS in v16.4.1 and v15.5.28. There is no patched 14.x: the support policy lists 14 as Unsupported, and the latest 14.x on npm is 14.2.35, from December 11, 2025.

LINE	RCE `USE CACHE`	RCE IMAGE (SHARP)	DOS SERVER ACTIONS	WHAT TO DO
16.x	up to 16.4.0	up to 16.4.0	up to 16.4.0	16.4.1 or later
15.x	no	up to 15.5.27	up to 15.5.27	15.5.28, then check sharp
14.x	no	all	all	sharp 0.35.5 now, then upgrade

The image RCE applies when Next.js runs on glibc-based Linux. The workaround in advisory GHSA-2464-fhr4-w9xw: make sure the `sharp` Next.js uses is 0.35.5 or later (it ships librsvg 2.63.2). The DoS only affects apps that use the App Router with at least one Server Action, and it has no workaround.

STEP 01

See what you have installed

From the project root, on the same machine or image where Next runs:

```
npm ls next
npm ls sharp
node -p "const s = require(require.resolve('sharp', { paths: [require.resolve('next')] })); s.versions.sharp +
' librsvg ' + s.versions.rsvg"
```

- `npm ls sharp` shows who brings sharp in. On 14 it may print `(empty)`: Next 14 does not declare sharp, so it is only there if your app installed it.
- The `node -p` line loads sharp the way the `next` package resolves it. With a fixed sharp it prints `0.35.5 librsvg 2.63.2` (we ran it with npm in Git Bash and PowerShell, and with pnpm).
- Skip `require('sharp/package.json')`: on 0.35.5 it throws `ERR_PACKAGE_PATH_NOT_EXPORTED`.

STEP 02

While you migrate: sharp 0.35.5 or later

If your app installed sharp

```
npm install sharp@^0.35.5
```

It is a direct dependency, so bump it like this. An override on it fails with `npm error code EOVERRIDE` ("Override for sharp@0.34.5 conflicts with direct dependency").

If sharp comes in through another package

npm, in the root `package.json`:

```
{
  "overrides": {
    "sharp": "^0.35.5"
  }
}
```

pnpm, in `pnpm-workspace.yaml` (current pnpm docs say the `pnpm` field in `package.json` is no longer read):

```
overrides:
  sharp: ^0.35.5
```

Yarn, in the root `package.json` (syntax from its docs; we did not test it):

```
"resolutions": {
  "sharp": "^0.35.5"
}
```

Then install and repeat Step 01. In our tests, npm 11.11.1 and pnpm 12.16.1 left `sharp@0.35.5` in the tree.

> **CAUTION**
> sharp 0.35.5 declares `node >=20.9.0`, while Next 14 accepts Node 18.17.0 and up. If you are still on Node 18, upgrade Node too. And this does not cover the DoS: it only buys you time to migrate.

STEP 03

Upgrade from 14 to 15

Work on a branch with the lockfile committed. Run the official codemod with 15 as the target; `upgrade latest` would take you to 16.4.4 today:

```
npx @next/codemod@canary upgrade 15
```

In our test it moved `next` to 15.5.28 and React from 18.3.1 to 19.3.1, and applied `next-request-geo-ip`, `next-async-request-api`, `app-dir-runtime-config-experimental-edge`, `next-experimental-turbo-to-turbopack` and `next-lint-to-eslint-cli`. Without an interactive terminal it accepts every default. Then, following the official version 15 guide:

- **React 19.** The minimum for `react` and `react-dom`; `useFormState` gives way to `useActionState`; bump `@types/react` and `@types/react-dom`. The Pages Router can stay on React 18.
- **Async request APIs.** Await `cookies`, `headers`, `draftMode`, `params` and `searchParams` (or unwrap them with `use` in synchronous components). Resolve every `@next-codemod-error` comment and `UnsafeUnwrapped` cast.
- **Caching.** `fetch` and `GET` Route Handlers are no longer cached by default (`cache: 'force-cache'`, `dynamic = 'force-static'`); the Client Router Cache stops reusing pages (`staleTimes`).

- **Renames.** `@next/font` to `next/font`, `serverComponentsExternalPackages` to `serverExternalPackages`, `bundlePagesExternals` to `bundlePagesRouterDependencies`, `experimental-edge` to `edge`. `geo` and `ip` are removed from `NextRequest`, along with automatic Speed Insights instrumentation.

This is what the codemod changed in a page that reads `params` and `cookies()`:

```
-export default function Page({ params }) {
-  const theme = cookies().get("theme")?.value ?? "light";
+export default async function Page(props) {
+  const params = await props.params;
+  const theme = (await cookies()).get("theme")?.value ?? "light";
```

STEP 04

Verify, deploy and plan for 16

```
npm ls next sharp
npx next build
npx next start
```

With the server up, request an image from `/_next/image` (for example `/_next/image?url=%2Ftest.png&w=640&q=75`) and open the pages that read `cookies()` or `params`. In our test the image came back as `200 image/webp` and the dynamic page rendered the cookie value. If you were already on 15, also run `npm update sharp`: going from 15.5.20 to 15.5.28, npm kept sharp 0.34.5 because Next's range (`^0.34.3 || ^0.35.5`) accepts it.

> **CAUTION**
>
> 15 gets security releases through January 31, 2027. Plan the move to 16 now (Node.js 20.9+, async-only request APIs, `middleware` becomes `proxy`). Read the `upgrade 16` diff: in our test it added `export const instant = false` and the build failed because the app does not use Cache Components; we removed those lines and the build passed.

WHEN YOU ARE DONE

Checklist

- [ ] `npm ls next sharp` shows next 15.5.28 (or 16.4.1 or later) and sharp 0.35.5 or later.
- [ ] The `node -p` line from Step 01 prints `librsvg 2.63.2`.
- [ ] No `@next-codemod-error` comments or `UnsafeUnwrapped` casts are left.
- [ ] `next build` and your tests pass; `/_next/image` returns 200.
- [ ] The new lockfile is committed and your CI installs from it.
- [ ] The move to 16 has a date, before January 31, 2027.