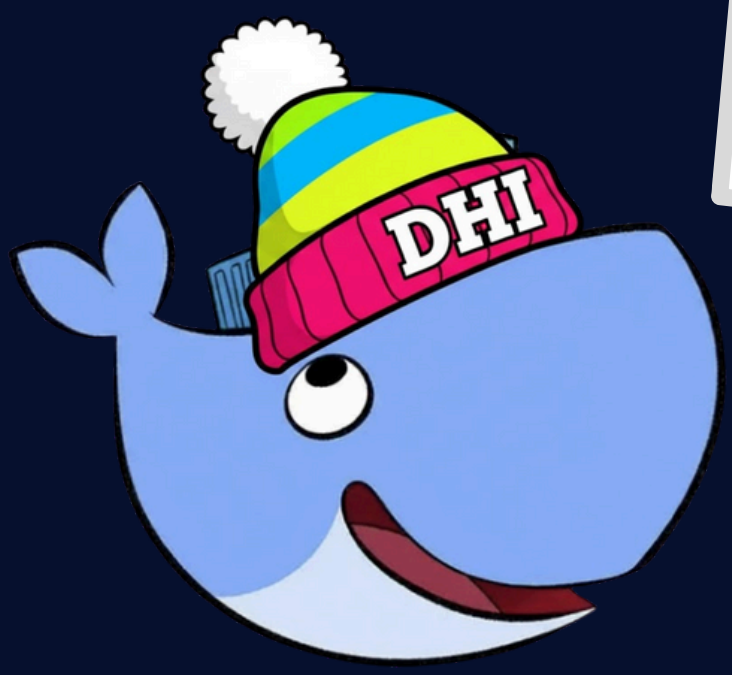
# DOCKER HARDENED IMAGES ARE OPEN SOURCE AND FREE!

**Docker** has made **Hardened Images (DHI)** fully open source under the Apache 2.0 license, giving every developer a secure, production-ready base image from the very first **FROM** line.

```
FROM dhi.io/node:24-dev AS build-stage
```

Secure base image in one line

- Security starts with the **base image**
- Most vulnerabilities come from **base images**, not your app
- **DHI** makes **secure-by-default** the easiest option




"EVERY DEVELOPER AND EVERY APPLICATION CAN (AND SHOULD!) USE DHI WITHOUT RESTRICTIONS."

OPEN SOURCE

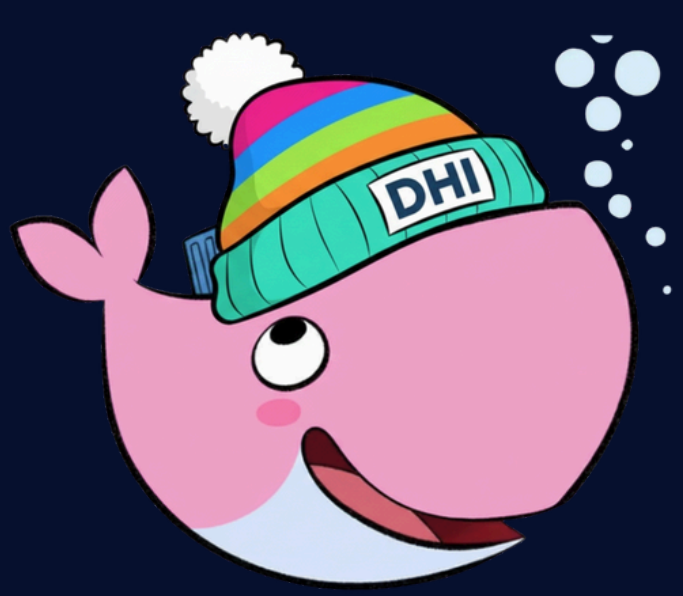
Edith Puclla

# WHAT ARE DOCKER HARDENED IMAGES (DHI)!

Docker Hardened Images (DHI) are minimal, production-ready base images maintained by Docker. They reduce the attack surface by including only what your application needs, resulting in near-zero CVEs by design and built-in supply-chain security.

COMMON VULNERABILITIES AND EXPOSURES

## WHAT MAKES AN IMAGE "HARDENED"?




WOW!

```
# Before (traditional image)
FROM node:24-alpine

# After (Docker Hardened Image)
FROM dhi.io/node:24-dev
```

- It includes only what your application needs to run, with no extra tools or leftovers.
- Based on familiar distros like **Debian** and **Alpine**
- **Runtime** images contain only what's needed to run the app (no shell, no package manager)
- Fewer vulnerabilities compared to traditional community images

Edith Puclla

# HOW TO USE DOCKER HARDENED IMAGES !

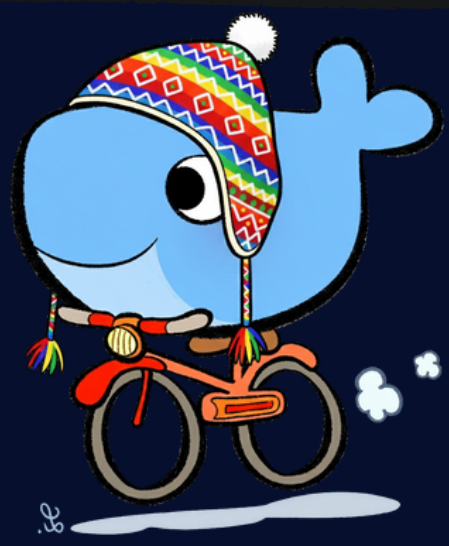
Using Docker Hardened Images requires you to only change the FROM line. Same **Dockerfile** but stronger security by default.

***-dev image**
→ used only to build (has compilers, tooling)
**runtime image**
→ minimal, production-only, hardened

DOCKERFILE

```
# Build stage
FROM dhi.io/node:24-dev AS build-stage

# Runtime stage
FROM dhi.io/node:24
```




## BUILD WITH SBOM AND PROVENANCE ENABLED

```
docker buildx build \
  -t edithturn/myapp:latest \
  --sbom=1 \
  --provenance=1 \
  --push .
```

Builds a hardened image, publishes it to Docker Hub, and attaches verifiable SBOM and provenance metadata by default.

## RESULT

- Your image is minimal and production-ready
- Security metadata (SBOM + provenance) is attached, not optional
- Nothing extra ships to production

Edith Puclla

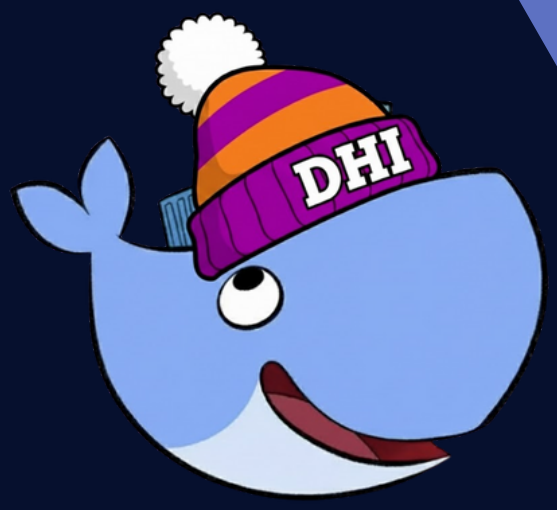
# INSPECT SBOM AND PROVENANCE

This SBOM shows exactly what's inside the image and where it comes from, helping teams track security and license risks.


**SBOM**
(SOFTWARE BILL OF MATERIALS)

```
docker buildx imagetools inspect edithturn/myapp
--format "{{json .SBOM.SPDX}}" | head -n 20
{
  "SPDXID": "SPDXRef-DOCUMENT",
  "creationInfo": {
    "created": "2026-01-03T18:10:31Z",
    "creators": [
      "Organization: Anchore, Inc",
      "Tool: syft-v1.29.0",
      "Tool: buildkit-v0.26.2"
    ],
    "licenseListVersion": "3.25"
```

Let's verify how the image was built, step by step, using SLSA provenance.




**SLSA**
SUPPLY-CHAIN LEVELS FOR SOFTWARE ARTIFACTS

```
docker buildx imagetools inspect edithturn/myapp \
  --format '{{json .Provenance.SLSA}}'
{
  "buildConfig": {
    "digestMapping": {
      "sha256:0b984064e871096b37f32": "step0",
      "sha256:e3f0527aa13d71cbe8ce0": "step1"
    },
    "llbDefinition": [
      {
        "id": "step0",
        "op": {
          "Op": {
            "source": {
              "attrs": {
                "image.resolvemode": "local"...
```

Docker Hardened Images make secure, verifiable containers accessible to everyone, and now it is Open source!

↳ Visit: docker.com/products/hardened-images/

Edith Puclla