**Start with Minimal Base Images**

Use distroless or minimal base images that contain only what's absolutely necessary:

* **Distroless images** (like Google's distroless) contain only your application and runtime dependencies
* **Alpine Linux** offers very small base images (typically 5MB)
* **Scratch** images for statically compiled binaries contain nothing but your application

**Multi-stage Builds**

Use multi-stage Dockerfiles to separate build dependencies from runtime:

dockerfile

*# Build stage*

FROM node:18-alpine AS builder

WORKDIR /app

COPY package\*.json ./

RUN npm ci --only=production

*# Production stage*

FROM node:18-alpine

WORKDIR /app

COPY --from=builder /app/node\_modules ./node\_modules

COPY . .

CMD ["node", "server.js"]

**Security Hardening**

* **Run as non-root user**: Create and use a dedicated user account
* **Remove package managers**: Uninstall apt, yum, apk after installing dependencies
* **Remove shells**: Delete /bin/sh, /bin/bash if not needed
* **Clear package caches**: Clean up after package installations
* **Remove documentation**: Delete /usr/share/doc, /usr/share/man directories

**Dependency Management**

* Install only production dependencies
* Pin specific versions to ensure reproducibility
* Regularly scan for vulnerabilities using tools like Trivy, Clair, or Snyk
* Remove build tools and development packages

**Layer Optimization**

* Combine RUN commands to reduce layers
* Place frequently changing instructions at the end
* Use .dockerignore to exclude unnecessary files
* Order instructions from least to most likely to change

**Automated Scanning and Updates**

Set up CI/CD pipelines that:

* Scan images for vulnerabilities before deployment
* Automatically rebuild images when base image updates are available
* Use image signing and verification
* Implement policy enforcement with tools like OPA Gatekeeper

These practices help create lightweight, secure, and maintainable container images that follow security best practices and reduce attack surface area.