



# How the heck do I debug distroless containers?

Eric Smalling

Staff Solutions Architect, [Chainguard.dev](https://chainguard.dev)

# Agenda

- Distroless: What / Why / How
- Runtime debugging
- Extras (time permitting)
  - Build time techniques

# Distroless Images

**What** are they?

**Why** would I use them?

**How** do you use them?

# Distroless Images

**What** are they?

**Why** would I use them?

**How** do you use them?



# NodeJS 22 Images

<u>Image</u>	<u>Size (MB)</u>	<u>Base Distro</u>	<u>Packages</u>	<u>Vulns</u>	<u>Critical</u>	<u>High</u>	<u>Medium</u>	<u>Low</u>	<u>Negligible</u>	<u>Unknown</u>	<u>Ignored</u>
node:22.9.0-bookworm	1590	Debian 12	618	586	10	115	262	36	439	257	533
node:22.9.0-bookworm-slim	333	Debian 12	292	67	1	1	13	3	46	3	0

data gathered 10/5/24 using gype v0.81.0

# NodeJS 22 Images

<u>Image</u>	<u>Size (MB)</u>	<u>Base Distro</u>	<u>Packages</u>	<u>Vulns</u>	<u>Critical</u>	<u>High</u>	<u>Medium</u>	<u>Low</u>	<u>Negligible</u>	<u>Unknown</u>	<u>Ignored</u>
node:22.9.0-bookworm	1590	Debian 12	618	586	10	115	262	36	439	257	533
node:22.9.0-bookworm-slim	333	Debian 12	292	67	1	1	13	3	46	3	0
gcr.io/distroless/nodejs22-debian12:debug	212	Debian 12	10	18	1	0	7	0	10	0	0
gcr.io/distroless/nodejs22-debian12:latest	210	Debian 12	9	14	1	0	3	0	10	0	0

data gathered 10/5/24 using gype v0.81.0

# NodeJS 22 Images

<u>Image</u>	<u>Size (MB)</u>	<u>Base Distro</u>	<u>Packages</u>	<u>Vulns</u>	<u>Critical</u>	<u>High</u>	<u>Medium</u>	<u>Low</u>	<u>Negligible</u>	<u>Unknown</u>	<u>Ignored</u>
node:22.9.0-bookworm	1590	Debian 12	618	586	10	115	262	36	439	257	533
node:22.9.0-bookworm-slim	333	Debian 12	292	67	1	1	13	3	46	3	0
gcr.io/distroless/nodejs22-debian12:debug	212	Debian 12	10	18	1	0	7	0	10	0	0
gcr.io/distroless/nodejs22-debian12:latest	210	Debian 12	9	14	1	0	3	0	10	0	0
cgr.dev/chainguard/node:latest-dev	943	Wolfi	319	0	0	0	0	0	0	0	0
cgr.dev/chainguard/node:latest	198	Wolfi	270	0	0	0	0	0	0	0	0

data gathered 10/5/24 using gype v0.81.0



# NodeJS 22 Images

<u>Image</u>	<u>Size (MB)</u>	<u>Base Distro</u>	<u>Packages</u>	<u>Vulns</u>	<u>Critical</u>	<u>High</u>	<u>Medium</u>	<u>Low</u>	<u>Negligible</u>	<u>Unknown</u>	<u>Ignored</u>
node:22.9.0-bookworm	1590	Debian 12	618	586	10	115	262	36	439	257	533
node:22.9.0-bookworm-slim	333	Debian 12	292	67	1	1	13	3	46	3	0
gcr.io/distroless/nodejs22-debian12:debug	212	Debian 12	10	18	1	0	7	0	10	0	0
gcr.io/distroless/nodejs22-debian12:latest	210	Debian 12	9	14	1	0	3	0	10	0	0
cgr.dev/chainguard/node:latest-dev	943	Wolfi	319	0	0	0	0	0	0	0	0
cgr.dev/chainguard/node:latest	198	Wolfi	270	0	0	0	0	0	0	0	0
node:22.9.0-alpine	213	Alpine 3.20.3	220	0	0	0	0	0	0	0	0

data gathered 10/5/24 using gype v0.81.0

# Distroless Images

**What** are they?

**Why** would I use them?

**How** do you use them?

# Distroless Images

**Runtime** Debugging Demonstration

# Thank you!

Eric Smalling  
Staff Solutions Architect  
[about.me/ericsmalling](https://about.me/ericsmalling)

[chainguard.dev](https://chainguard.dev)

## Resources:

- Tools used/mentioned:

- [Google Distroless Images](#)
- [Docker Debug](#)
- [cdebug](#)
- [dive](#)
- [Grype](#)

- Related Content:

- [Images as Code: The pursuit of declarative image builds](#)
- [Software Engineering Daily: Container Security with Matt Moore](#)
- [Debugging Distroless Containers with Docker Debug](#)
- [Debugging Distroless Images with Kubectl Debug and CDebug](#)



<https://github.com/ericsmalling/presentations/tree/main/debugging-distroless>