Choosing & Building Better Images

Chet Burgess
Principal Engineer
3/8/2020

Disclaimer

I am not a lawyer

Compliance is about risk

Compliance Obligations

- Open Source Software Compliance
- Ship only the software you need
- Keep your software up-to—date

Open Source Software Compliance

- OSS exists within a legal framework that creates obligations for the developers and distributors
- Not all “open” licenses are as “open” or as “free” as you might think
  - Recent trend in ”Proprietary but open-ish” licenses (Elasticsearch)
- https://compliance.linuxfoundation.org

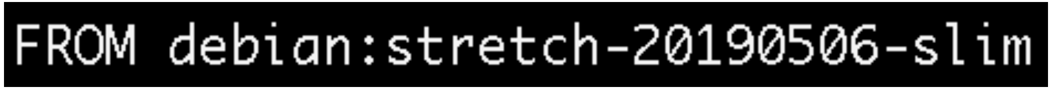
What’s wrong with containers?

- Unknown content
- Unnecessary software
- Out-of-date software

Common Problems & Fixes

“Dependency Rot”

- Use of fixed “dated” FROM



```
FROM debian:stretch-20190506-slim
```

- Slow release cycles/lack of updates
- No management/monitoring of dependencies

“Dependency Rot” Solutions

- Use a sliding tag (e.g. latest)
  - Be careful that “latest” actually means what you think it means
  - Things like “latest” and “stable” can change major versions
  - https://vsupalov.com/docker-latest-tag/
- Perform an update with your package manager
  - Example: `apt-get –y dist-upgrade`
  - Make sure you are using up-to-date package repositories
- Update your FROM line

“Dependency Rot” Solutions

- Do regular rebuilds, even if your software doesn’t change
  - dockerhub has a feature called “Repository Links” that can help
    - Does not work for “official” images (https://hub.docker.com/search?q=&type=image&image_filter=official)
    - Requires your FROM line to be a string (no variables) and the image must be hosted on dockerhub

“Dependency Rot” Solutions

- Monitor your dependencies
  - GitHub has built-in dependency monitoring
    - https://help.github.com/en/github/managing-security-vulnerabilities/managing-security-vulnerabilities-in-your-project
  - renovate GitHub App
    - Tool for tracking dependencies and automatically posting PRs to update them.
    - https://github.com/apps/renovate
    - https://github.com/marketplace/renovate
    - https://www.npmjs.com/package/renovate
  - Numerous other tools for monitoring your dependencies

Demo

“The Kitchen Sink”

- Very large container images
- Numerous layers
- Entire git tree left in the image
- Build tooling and assets left in the image
- Removing items from the filesystem in later layers without squashing

“The Kitchen Sink” Solutions

- Think carefully about what data is being ”left behind”
  - Remove your build deps
  - Cleanup after your package managers (apt, pip, yum, etc)
  - Limit the number of RUN commands
- Be careful about automatic installs
  - apt will install ”recommend” packages by default
  - use -–no-install-recommends
- Be very selective in the use of COPY
- Use build args for sensitive data

“The Kitchen Sink” Solutions

- Multi-stage Docker Builds
  - Allows a single Dockerfile to build multiple images in sequence and copy data from previous stages.
  - https://docs.docker.com/develop/develop-images/multistage-build/
- Squash your image at build
  - Experimental –-squash flag in docker 1.13
  - Use sparingly, this eliminate the benefits of layer caching
  - Not supported by dockerhub

Demo

“The Mystery”

- Images a that are a single file copied from a local build system
- curl | bash
- Installation of software from internal sources
- Bespoke/convoluted build process

“The Mystery” Solutions

- Build inside the image (multi-stage)
- Document your build process
- Make it easy to find your source and Dockerfile
  - https://github.com/opencontainers/image-spec/blob/master/annotations.md
  - http://label-schema.org/rc1/ (deprecated)
- Validate your downloads
  - Signature validation
  - Hash validation

Auditing & Updating

- How do I know the components and their licenses?
- How do I know if its up-to-date?

Auditing Tools

- https://anchore.com
- https://github.com/vmware/tern
- https://compliance.linuxfoundation.org/references/tools/
- https://github.com/quay/clair
- https://www.twistlock.com
- Many other OSS and commercial solutions

Auditing Tools - Anchore

- Opensource & Commercial offering for scanning images
- Able to inventory and monitor numerous types of packages for CVEs
  - Ubuntu, Debian, CentOS, Alpine, Python, Java, node/NPM, Ruby/Gems
- Rich policy engine that can enforce container best practices
- Numerous integrations available out of the box
  - Jenkins, CircleCI, GitHub Actions, etc
- https://anchore.com
- https://github.com/anchore/anchore-engine

Demo – Anchore Github Action

https://github.com/marketplace/actions/anchore-container-scan

Q & A

https://github.com/cburgess/docker-examples

Chet Burgess
Principal Engineer
3/8/2020

cisco