

# Docker Appliance GoLang

Small, Fast, Single Purposed

# Concept

- Tiny Image
- Quick to Stand Up
- Micro Service

# Platform

- Docker Container
- Container Management?

# Go

- Compiled Binaries
- Garbage Collection
- Concurrency?

# Typical Container

- FROM golang:onbuild
- EXPOSE 8080

# Minimal Container

- FROM scratch
- ADD HttpServerWithFirewall /
- CMD ["/HttpServerWithFirewall"]

# Go Build Script for Static Linking

- `CGO_ENABLED=0 GOOS=linux go build -a -installsuffix cgo -o HttpServerWithFirewall .`
- `docker build -t theminimalserver -f MinimalGoDockerfile .`

# Environment Variables

- CGO\_ENABLED - allow C compiling
- GOOS - target operating system



# Build flags

- -a - force rebuilding of packages that are up to date
- -installsuffix - a suffix to use in the name of the package installation directory, in order to keep output separate from default builds.
- -o - output

# docker images

•	REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
•	theminimalserver	latest	f909ef1a423a	About a minute ago	6.676 MB
•	thetypicalserver	latest	1c3d28ae8e2b	33 minutes ago	774.5 MB

# References

- Code - `git@github.com:pcdwr/godfw.git`
- Minimal Docker Container - <https://blog.codeship.com/building-minimal-docker-containers-for-go-applications/>