Learning GoLang By Doing

GWEST@REDHAT.COM

July 1, 2024

Programming is a Craft

- The best way to learn is by doing real projects
- There is a variety of use-cases that real project exposes you too
- Bigger projects also stress:
  - Good source control
  - Good organization of code
  - Exploration of libraries

Project 1

- PacMan
- Wrap javascript inside a golang program
- Serve the javascript code
- Run in minishift as a minimum size container
- Base javascript: https://github.com/glennswest/pacman

Assignment 1 Goals

- Learn:
  - Serving assets via http
  - Building minimum size containers for docker/ocp
  - Mixing js/nodejs inside of golang programs

Project 2

- Gash – Golang Bash/Busybox replacement
- Build a “bash” replacement, design to be the main system system for a mini linux.
- https://www.ibm.com/developerworks/linux/library/l-busybox/

Project 2 - Subcommands

Implement the most common linux commands

- First 10 commands
- Plus: Ability to execute external commands

https://www.hostinger.com/tutorials/linux-commands

Project 2 - Implementation

- Make it easy to add additional internal commands
- Make source organized as subdirectories for each command
- Extra points to make the main module discover additional internal commands

Project 3 - Smart Caching Proxy

- Build a semi-passive proxy designed to automatically cache containers to make repeated ocp installs fasters
- Implemented as a linux service
- Uses minio as storage
- Decodes quay/repo syntax
- Man-In-The-Middle style implementation

Project 3 – Things to learn

- REST API Serving/Decoding
- HTTP Request Pass THRU
- REST Client implementation
- Real-Time Streaming of data
- Working with openshift

Project 4 – Modifcation of existing project

- Ocp needs a front end load balancer that is implemented externally.
- Ploadb – implements a traffic manager like dns load balancer using api driven pdns
- Current ploadb uses ping to test nodes, add a proper health check using http, make the health check a config item using a embedded json field in pdns extra field.

Project 4 – Things to learn

- Health checks
- Json parsing
- PR and modification of existings project
- Async / event driven / mutli-threading

Project 4 - Assets

- https://github.com/glennswest/pdnsloadbalancer
- Master Check:
  - curl --insecure "https://192.168.1.202:6443/healthz"
- Worker Check
  - curl http://worker-2.gw.lo:1936/healthz/ready

Project 4 – Comment Structure


```
{
  "rrsets": [
   {
     "changetype": "replace",
     "comments": [],
     "name": "api-int.gw.lo.",
     "records": [
      {
        "content": "192.168.1.200",
        "disabled": false
      },
      {
        "content": "192.168.1.201",
        "disabled": false
      },
```

Project 5 – Interfacing nuodb to buffalo

- Buffalo is a implementation of rails in golang
- It supports multiple database
- Nuodb is supported in OCP, but buffalo does not yet support it
- https://gobuffalo.io/en/
- https://nuodb.com/