The Search for a Good Place to REST in Go

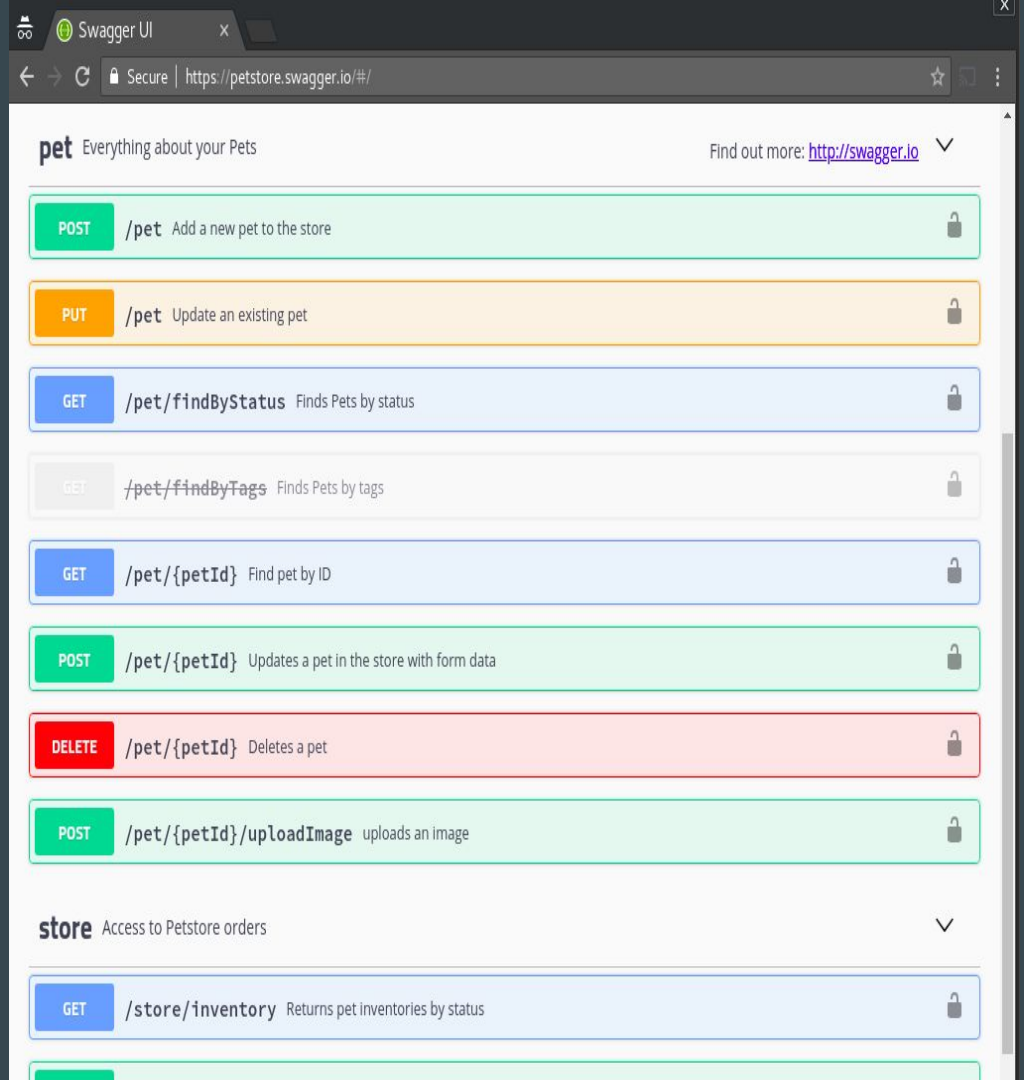
Eyal Posener
Software Developer @ Stratoscale

Agenda

- Intro to Open API/swagger
- Go-swagger Demo
- Go-swagger Pros - Cons
- Stratoscale/swagger

Open API / Swagger

- DSL for describing REST APIs. It can be written in either JSON or YAML.
- Docs and Tools: https://swagger.io/
- All you need for APIs:
  - Editors
  - Generate code client/server/mocks
  - UI
  - Auto CLI
- Develop in High Scale



Open API / Swagger

- Warnings:
  - It takes time
  - The specification is huge
- Alternatives
  - RAML
  - Protobuf (Not for REST)

Go-swagger

- Generate go code - server / client from swagger file
- Demo

Go-swagger Pros

- Does the job
  - Server/Client code
  - Automatic routing
  - Type checks, static typed code
- High pulse in Github
- Responsive to issues
- Used by some large companies/projects
  (Alibaba PouchAPI CoreOS DigitalOcean Kubernetes-Helm Kubernetes StratoScale VMWare)

Go-swagger Cons

- Painful configure_*.go file
- Required fields
- Hard to get http.Handler
- Hard to consume and customize generated client
- Not using standard time.Time
- Non-versioned go-openapi libraries

Stratoscale/swagger

- After some usage of go-swagger, we created our own version.
- Uses a swagger flag --template-dir for custom templates. (Only some of the files)
- Open source at https://github.com/Stratoscale/swagger
- Why?
  - Solves configure_*.go with a similar approach to gRPC + protobuf
  - Expose http.Handler
  - Usage of context.Context
  - Expose service interfaces
  - Expose client interfaces

Stratoscale/swagger Demo

Some More Examples

- main_test.go
- use client

Thanks!

- Read more: https://posener.github.io
- Use it:
  - https://github.com/Stratoscale/swagger
  - https://hub.docker.com/r/stratoscale/swagger
- Meetup demos: https://github.com/posener/meetups/tree/master/swagger
- Ask me: posener SHTRUDEL gmail.com