GoGuardian®

Iteratively Migrating an HTTP Service to Go and gRPC Using `grpc-gateway` and `httputil.ReverseProxy`

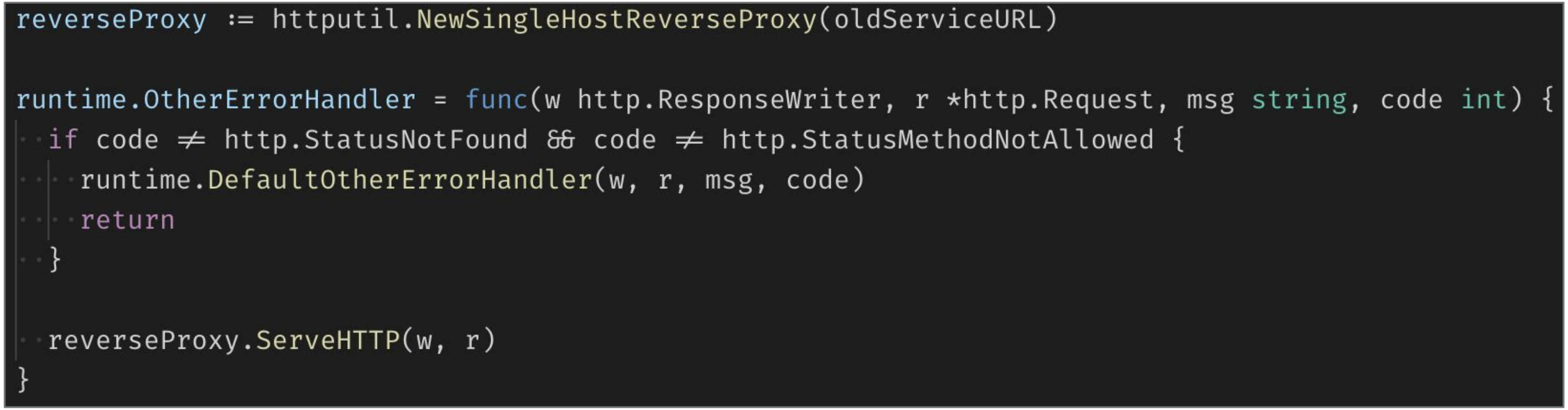
Igor Dubinskiy
GitHub: idubinskiy
Twitter: idubins

NOT SURE IF HTTP SERVICE
OR GRPC BEHIND GRPC-GATEWAY
imgflip.com

Options to migrate a service from HTTP to gRPC

- Everything but the browser
  a. Modify clients to use gRPC
  b. Use `grpc-gateway`; see below
- Browser (using `grpc-gateway`)
  a. Modify clients to call new service per route as each required RPC is implemented
  b. Modify clients to call new service for all routes once all required RPCs are implemented
  c. New service proxies unimplemented routes to existing service

Proxy using `grpc-gateway` and `httputil.ReverseProxy`



```go
reverseProxy := httputil.NewSingleHostReverseProxy(oldServiceURL)

runtime.OtherErrorHandler = func(w http.ResponseWriter, r *http.Request, msg string, code int) {
  if code != http.StatusNotFound && code != http.StatusMethodNotAllowed {
    runtime.DefaultOtherErrorHandler(w, r, msg, code)
    return
  }

  reverseProxy.ServeHTTP(w, r)
}
```

https://github.com/idubinskiy/http-to-grpc-gateway

Demo

Thank you!