

Golang Tutorial #5

go-grpc

- .proto to .pb
- client load-balancing

Protobuf & gRPC

- protobuf is a kind of structural encoding
- gRPC is a RPC defined in protobuf format
- support request-response, streaming interaction
- tools to generate cross-platform code
- currently, we use version: proto3

go gRPC started: .proto to .pb file

- [grpc intro](#)
- [go-grpc](#)
- `protoc --go_out=plugins=grpc:. *.proto`
- [go_grpc example](#)
- server and client struct implement interface
- `RegisgerXXXServiceServer` `NewXXXServiceClient`
- example in the example3/

gRPC client load-balancing

- golang grpc lib help handle connection pool
- will connect to given address
- dns addr: `dns:///im-broker-headless:9199`
- client dial options to tune performance

```
var opts []grpc.DialOption
opts = append(opts, grpc.WithInsecure())
opts = append(opts, grpc.WithBalancerName(roundrobin.Name))
opts = append(opts, grpc.WithKeepaliveParams(keepalive.ClientParameters{Timeout: time.Second * 15}))

c, err := grpc.Dial(addr, opts...)
```