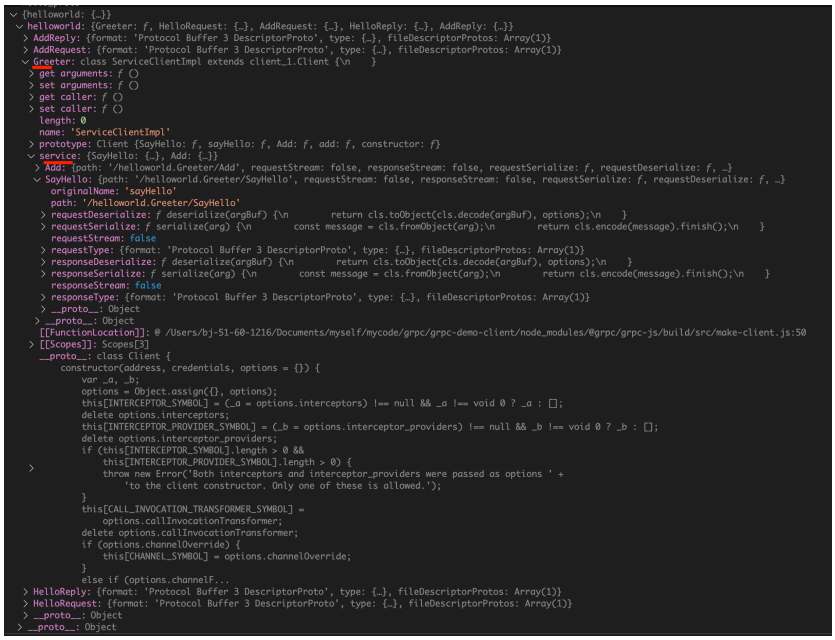
模块划分
http_proxy 代理 & 链接
load-balancer 负载均衡相关

一、loadPackageDefinition 转换grpc对象

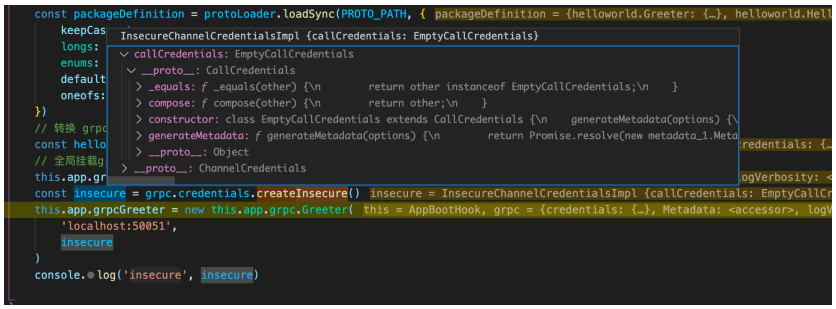
```
// 加载proto文件
const packageDefinition = protoLoader.loadSync(PROTO_PATH, {
    keepCase: true,
    longs: String,
    enums: String,
    defaults: true,
    oneofs: true,
})
// 将proto转换成 grpc对象
const hello_proto = grpc.loadPackageDefinition(packageDefinition)
// hello_proto 返回值如下
```

> 对比packageDefinition的返回值（在proto-loader文件），对象的层级结构发生了变化，将方法都放到了helloworld对象下，另外Greeter类下的方法都放到了service下



二、credentials.createInsecure

> 返回新的 无证书的 ChannelCredentials



三、Server类

1、addService

```
const server = new grpc.Server()
server.addService(hello_proto.Greeter.service, {
    sayHello,
    Add,
})
```

bindAsync