

```
lis, _ := net.Listen("tcp", addrGRPC)
grpcServer := grpc.NewServer(
    grpc_middleware.WithUnaryServerChain(
        grpc_recovery.UnaryServerInterceptor(),
    ),
)
gw.RegisterServer(grpcServer, srv)
go grpcServer.Serve(lis)
```

```
gwmux := runtime.NewServeMux(  
    runtime.WithMarshalerOption(runtime.MIMEWildcard,  
        &runtime.JSONPb{OrigName: true, EmitDefaults: true}),  
)  
  
gw.RegisterHandlerFromEndpoint(ctx, gwmux, addrGRPC, opts)  
  
runtime.SetHTTPBodyMarshaler(gwmux)  
runtime.GlobalHTTPErrorHandler = httpError  
  
go http.ListenAndServe(addrApi, wsproxy.WebsocketProxy(gwmux))
```