

第88期-学习gRPC指南与示例代码

2020年2月12日 12:17

Proto 协议文件应该怎么写？

本指南描述如何使用 ProtoBuf 语言规范来组织你的 .proto 文件，以及如何编译 .proto 文件来生成相应的操作类。它涵盖了proto3 所有语法。

<https://www.jianshu.com/p/f6ff6381a81a>

<https://www.cnblogs.com/sanshengshui/p/9739521.html>

其它平台 gRPC 技术栈

<https://github.com/grpc/grpc>

通道详细配置

```
// Create the gRPC channel
var channel = GrpcChannel.ForAddress(baseAddress, new GrpcChannelOptions
{
    HttpClient = new HttpClient(handler)
});
```

使用 Visual Studio 服务引用工具

×

添加新的 gRPC 服务引用

选择文件或 URL

☒ 文件(F)

浏览(B)...

☐ URL (U)

选择要生成的类的类型(S)

客户端

▼

确定(O)

取消(C)

安全注意事项

TLS加密传输配置

```
webBuilder.ConfigureKestrel(serverOptions =>
{
    serverOptions.ConfigureHttpsDefaults(listenOptions =>
    {
        listenOptions.SslProtocols = SslProtocols.Tls12;
    });
});
```

异常处理

异常消息通常被视为不应泄露给客户端的敏感数据。默认情况下，gRPC 不会将 gRPC 服务引发的异常的详细信息发送到客户端。相反，客户端将收到一条指示出错的一般消息。向客户端发送的异常消息可以通过 `EnableDetailedErrors` 重写（例如，在开发或测试中）。不应在生产应用程序中向客户端公开异常消息。

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddGrpc(options =>
    {
        options.EnableDetailedErrors = true;
    });
}
```

消息大小限制

传入消息到 gRPC 的客户端和服务将加载到内存中。消息大小限制是一种有助于防止 gRPC 消耗过多资源的机制。

gRPC 使用每个消息的大小限制来管理传入消息和传出消息。默认情况下，gRPC 限制传入消息的大小为 4 MB。传出消息没有限制。

在服务器上，可以使用 `AddGrpc` 以下内容为应用中的所有服务配置 gRPC 消息限制

```
public void ConfigureServices(IServiceCollection services)
{
    services.AddGrpc(options =>
    {
        options.MaxReceiveMessageSize = 1 * 1024 * 1024; // 1 MB
        options.MaxSendMessageSize = 1 * 1024 * 1024; // 1 MB
    });
}
```

```
    });  
}
```

gRPC 服务接口的版本控制

向后兼容性

非重大变化、二进制重大更改、协议重大更改和协议重大更改。

版本号服务

包名称与服务名称组合在一起，以标识服务地址。服务地址允许并行承载服务的多个版本：

greet.v1.Greeter

greet.v2.Greeter

```
syntax = "proto3";
```

```
package greet.v1;
```

```
service Greeter {  
    rpc SayHello (HelloRequest) returns (HelloReply);  
}
```

```
message HelloRequest {  
    string name = 1;  
}
```

```
message HelloReply {  
    string message = 1;  
}
```

gRPC 官方示例程序

<https://github.com/grpc/grpc-dotnet/tree/master/examples>