https://mp.weixin.qq.com/s/7135y3MkUlPIp-flfwscig

https://www.bilibili.com/video/av73194514?p=2

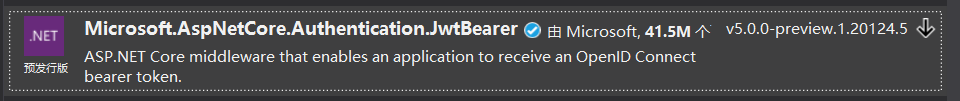
在要验证的方法前加上一个[Authorize]的Attribute，然后在ConfigureSeivices里AddAuthentication(); 这是认证方案，常用的是JWT，我们就需要配置一下JWT的认证方案。

Ids4和JWT的关系：ids4用到了jwt的一些内容。JWT是JSON Web Tokens，是Json web令牌。

[Authorize]是授权；

AddAuthentication();是认证；

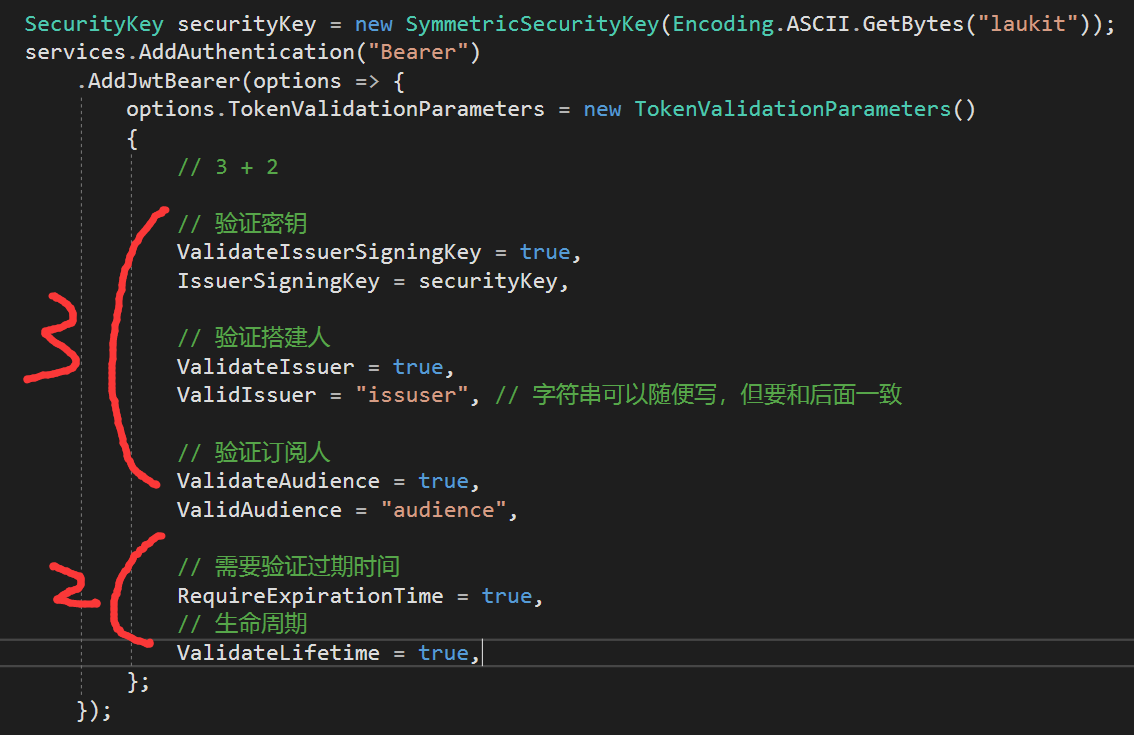
去Nuget搜JWT，安装下面这个包：



先new一个对称加密的密钥：其实一会才能用到，而且下面的密钥可能有bug。

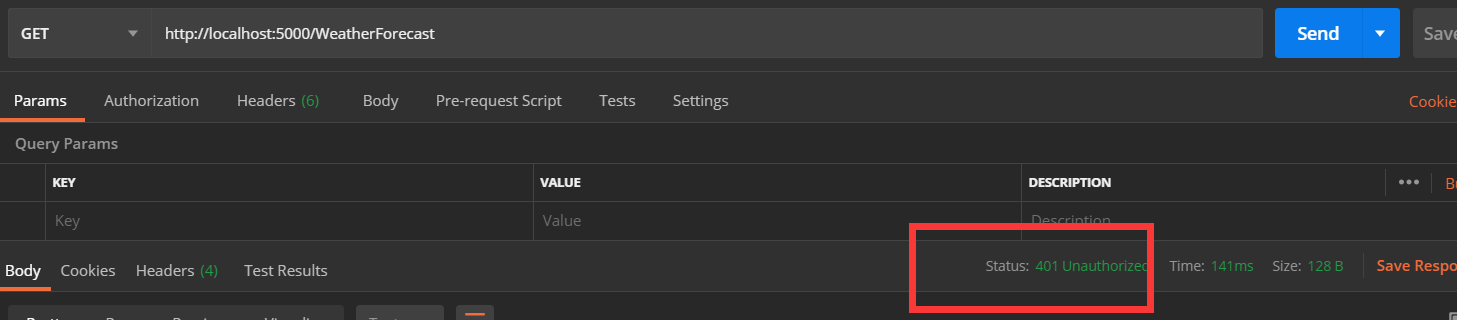


JWT封装需要的配置是3+2：



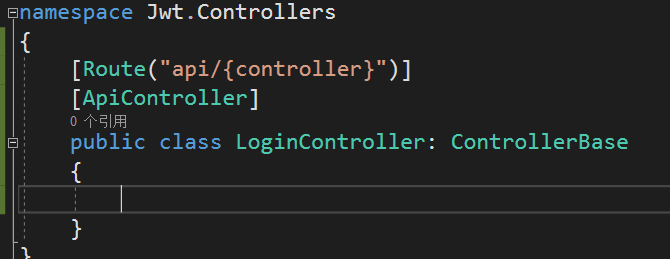
实际是Add一个Authentication服务，然后对JwtBearer进行配置。

测试一下，显示401，表示我们项目配置正确：

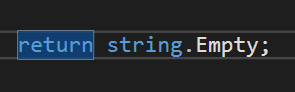


401无权限怎么解决呢，我们需要添加一个令牌去访问。先生成一个令牌：

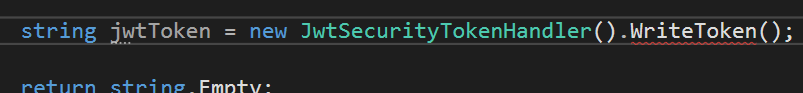
添加一个LoginController：



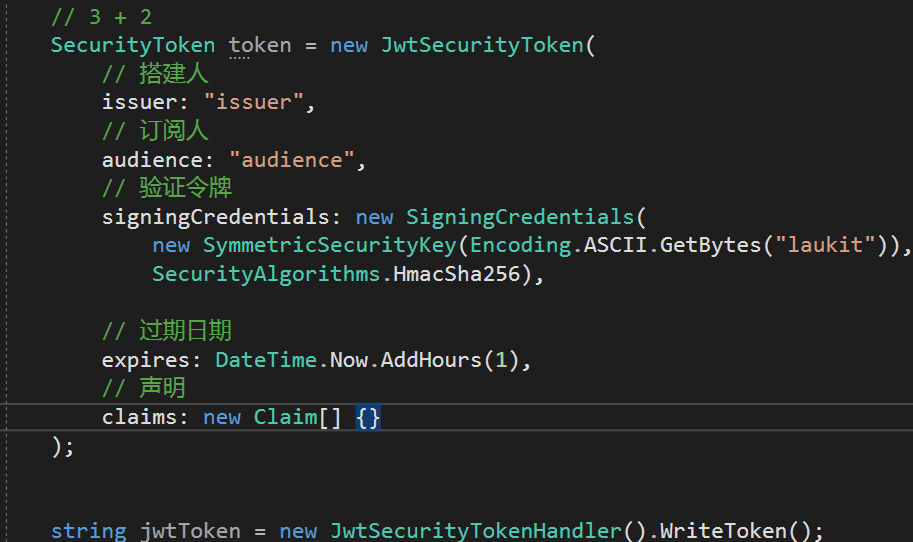
写一个GetToken方法，先倒着编程。因为方法需要返回一个Token，是string类型，我们先让它默认返回一个string.Empty：



然后定义一个string类型的Token：



查看WriteToken需要什么类型的参数，然后还是3+2：



这个3，无论在服务注册的时候还是生成令牌的时候，都是这3个，而且他们的内容要一模一样，不一样的话就验证不了。过期日期是令牌有效期，这里是1小时，声明是类似于身份证里的指纹之类的声明，可以在里面声明一些角色认证等。

最后改一下名字，填一下前面空出来的空：



也可以直接return上面的等号后的语句。

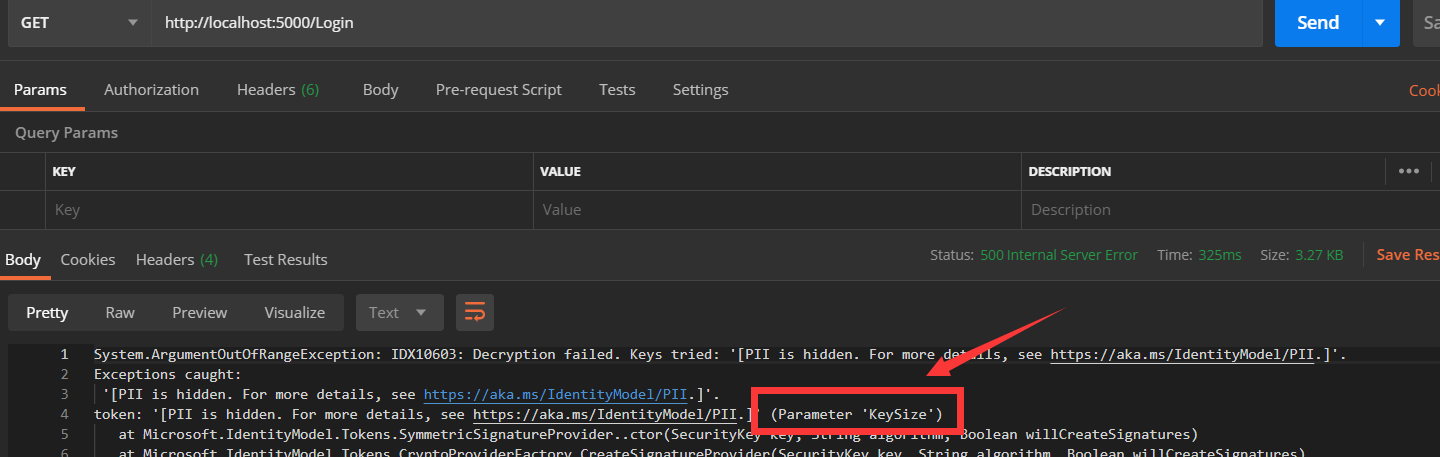
到现在为止，JWT已经完成了。它就有三块：授权（Attribute，[Authorize]）、认证方案（JwtBearer，把这个服务注入进去）、Token生成（LoginController）。

但是上面留了两个错误：

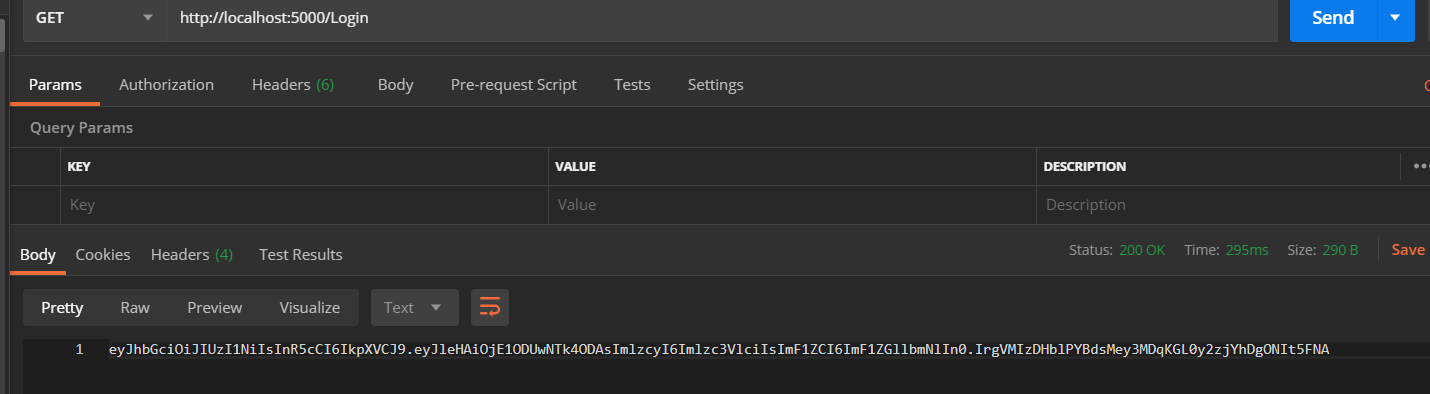
1. Kitlau作为密钥，它的长度不够，需要大于等于16位
2. 没有在中间件管道配置，验证没有开启，app.UseAuthentication ();加在UseRouting();后，app.UseAuthorization();前。

仅仅在ConfigureServices注入服务，不在中间件管道开启，服务注入和不注入没有任何区别。

先改一下Login的路由，把api删掉，我们不想把路径中带上api。当然保留也可以。

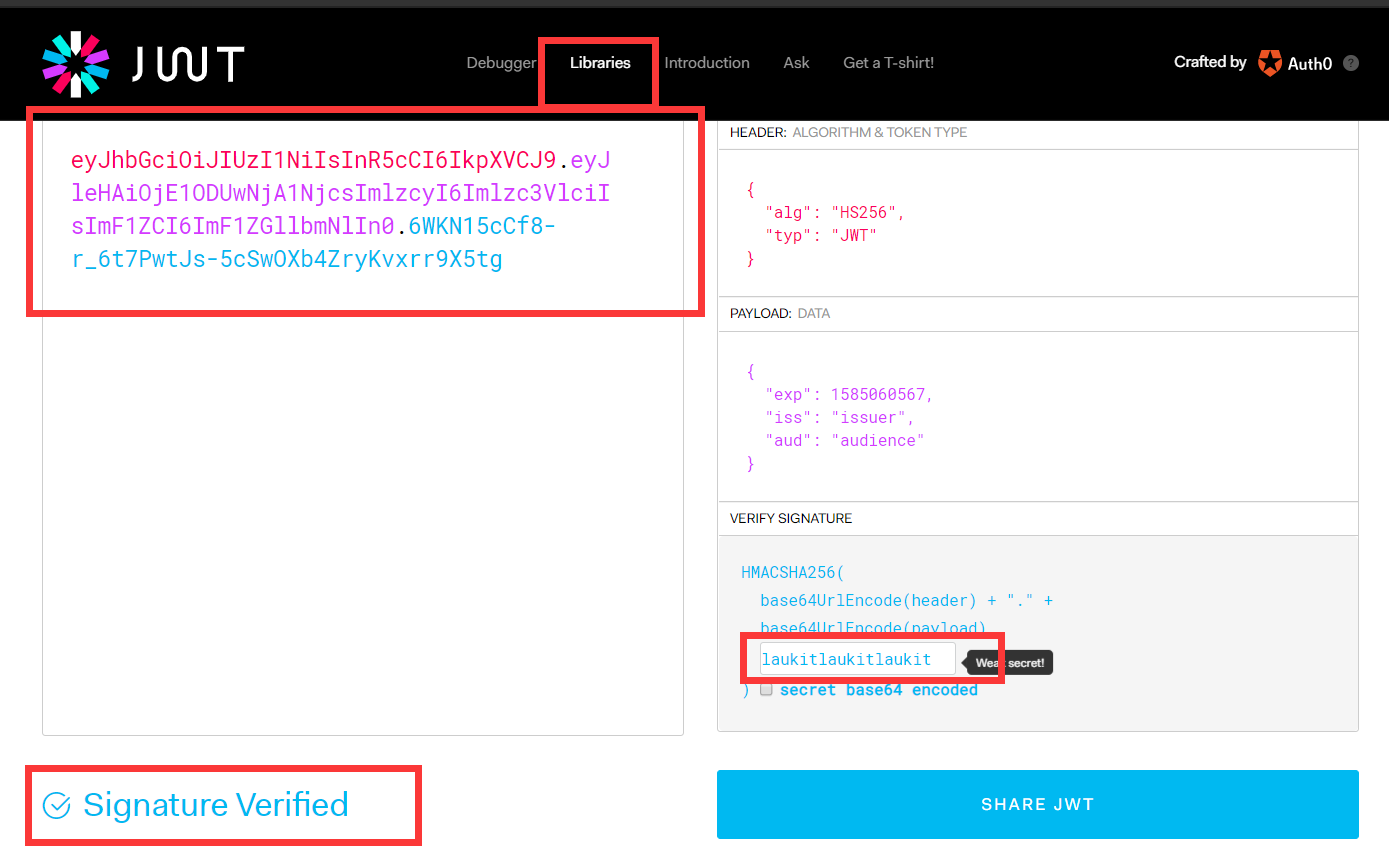


报错了，密钥长度不够。增加长度，注意保证注册服务和生成Token的密钥一致。



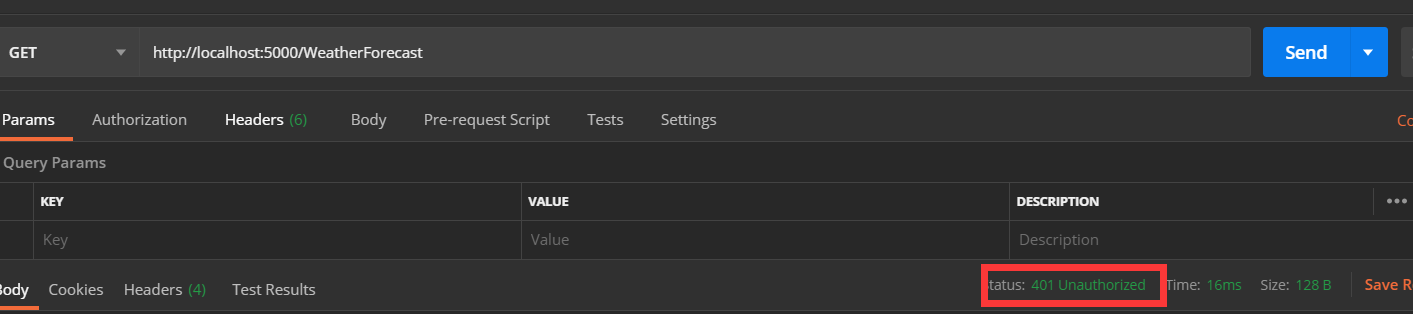
生成好了jwt这个Token。发现它很短，因为我们没有声明主体。

先去jwt的官网验证一下这个token：<https://jwt.io/> 看看里面都有什么信息。

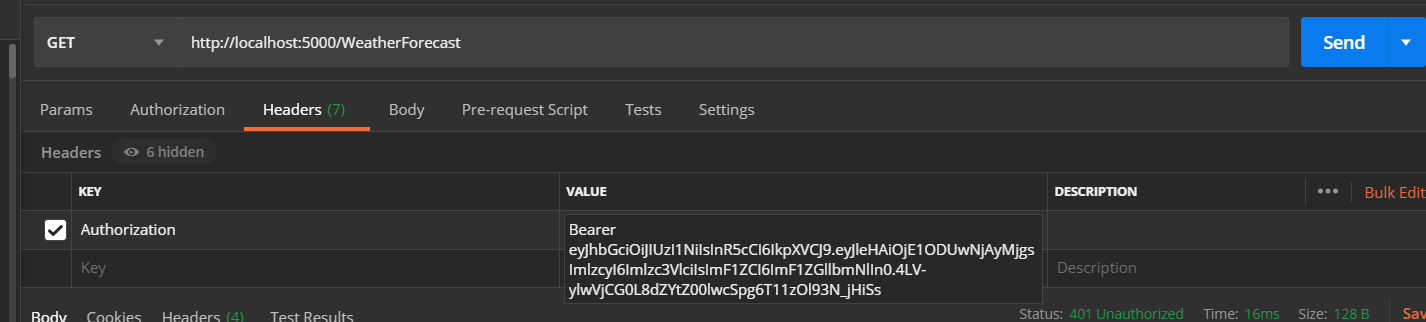


把token复制进去，签名复制到右下那里，发现左下角显示签名有效。

再访问WeatherForecast发现401：

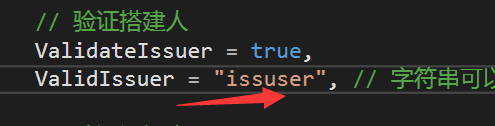


复制刚才拿到的token，再在request的header里添加：其中value是Bearer+空格+Token

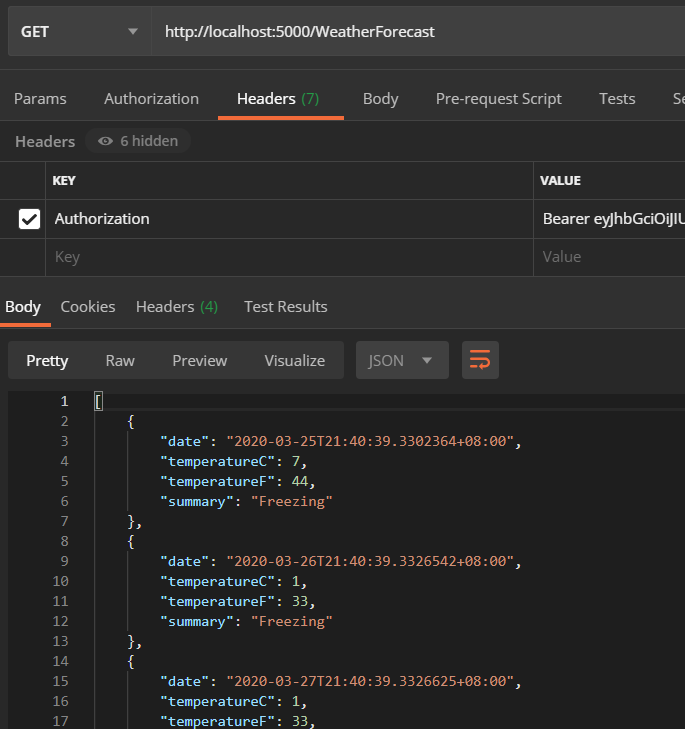


发送之后依然401，然后我们去Startup的中间件管道中设置中间件，开启认证

发现还是不行，回头找bug：



这里应该是issuer，打错了。如果是大项目，这要找到天涯海角。所以以后这种东西一定要复制：



JwtBearer认证体系要求我们的request header里的Authorization这个Key的Value必须以Bearer空格开头。