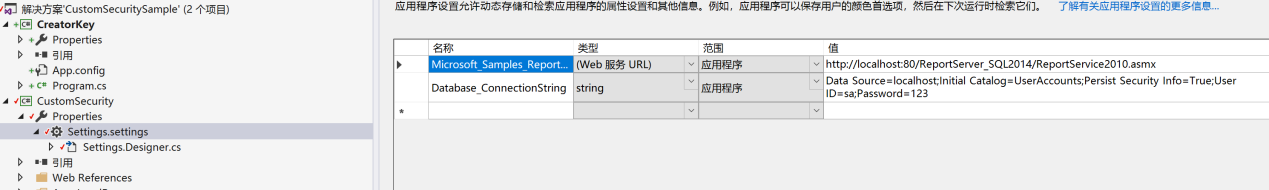
部署说明

**第1步：创建UserAccounts数据库**

数据库的链接串在“应用程序设置”里



## 步骤2：构建样本

您必须先编译并安装扩展。该过程假定您已将Reporting Services安装到默认位置：C：\ Program Files \ Microsoft Power BI Report Server \ PBIRS \ ReportServer \或C：\ Program Files \ Microsoft SQL Server Reporting Services \ SSRS \ ReportServer。这个位置将在本主题的其余部分被引用<install>。

如果您尚未创建强名称密钥文件，请使用以下说明生成密钥文件。

生成强大的名称密钥文件

* 打开Microsoft Visual Studio提示并指向.Net Framework 4.0。
* 使用更改目录命令（CD）将命令提示符窗口的当前目录更改为保存项目的文件夹。
* 在命令提示符处，运行以下命令以生成密钥文件：sn -k SampleKey.snk。

使用Visual Studio编译示例

* 在Microsoft Visual Studio中打开CustomSecuritySample.sln。
* 在解决方案资源管理器中，选择CustomSecuritySample项目。
* 看看CustomSecuritySample项目的参考。如果您没有看到Microsoft.ReportingServices.Interfaces.dll，请完成以下步骤：
* 在项目菜单上，单击添加引用。添加引用对话框打开。
* 单击.NET选项卡。
* 单击浏览，然后在本地驱动器上查找Microsoft.ReportingServices.Interfaces。默认情况下，程序集位于<install>\ReportServer\bin目录中。点击确定。选定的参考被添加到您的项目中。
* 在生成菜单上，单击生成解决方案。

调试

要调试扩展，您可能需要将调试器附加到ReportingServicesService.exe和RSPortal.exe。并为实现接口IAuthenticationExtension2的方法添加断点。

**第3步：部署和配置**

1、将<install>\ReportServer directory下的rsreportserver.config、rssrvpolicy.config、web.config复制改名备份；

2、将Logon.aspx页面复制到<install>\ReportServer directory。

3、将DM.Common.dll、DM.Common.pdb、DM.UBP.PBIRS.Security.dll、DM.UBP.PBIRS.Security.pdb、Newtonsoft.Json.dll（不需要复制）、Microsoft.IdentityModel.Tokens.dll、Microsoft.IdentityModel.Logging.dll、System.IdentityModel.Tokens.Jwt.dll八个文件赋值到<install>\ReportServer\bin目录；

4、将上述八个文件赋值到<install>\Portal目录；

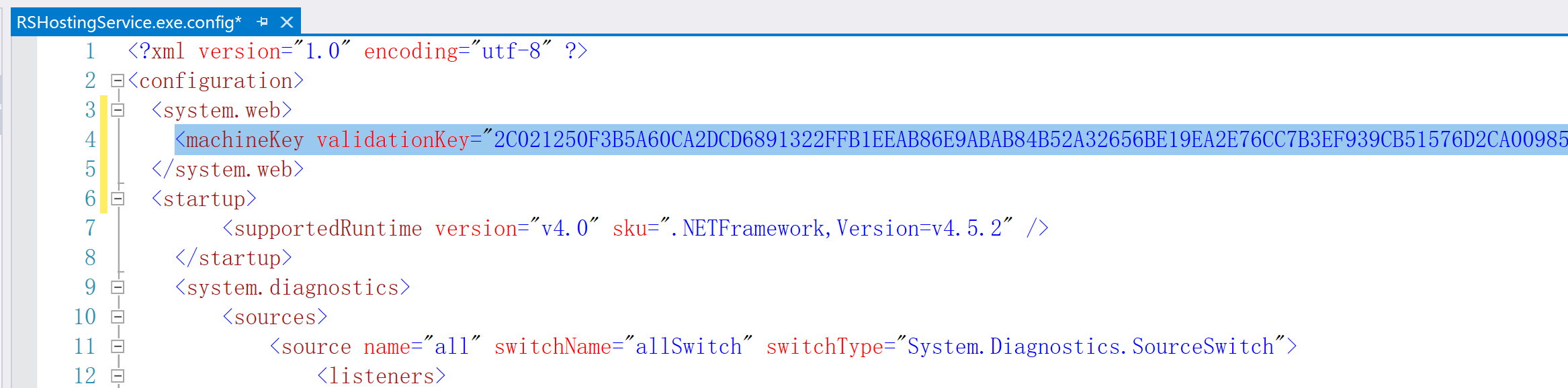
5、将上述八个文件赋值到<install>\PowerBI目录；

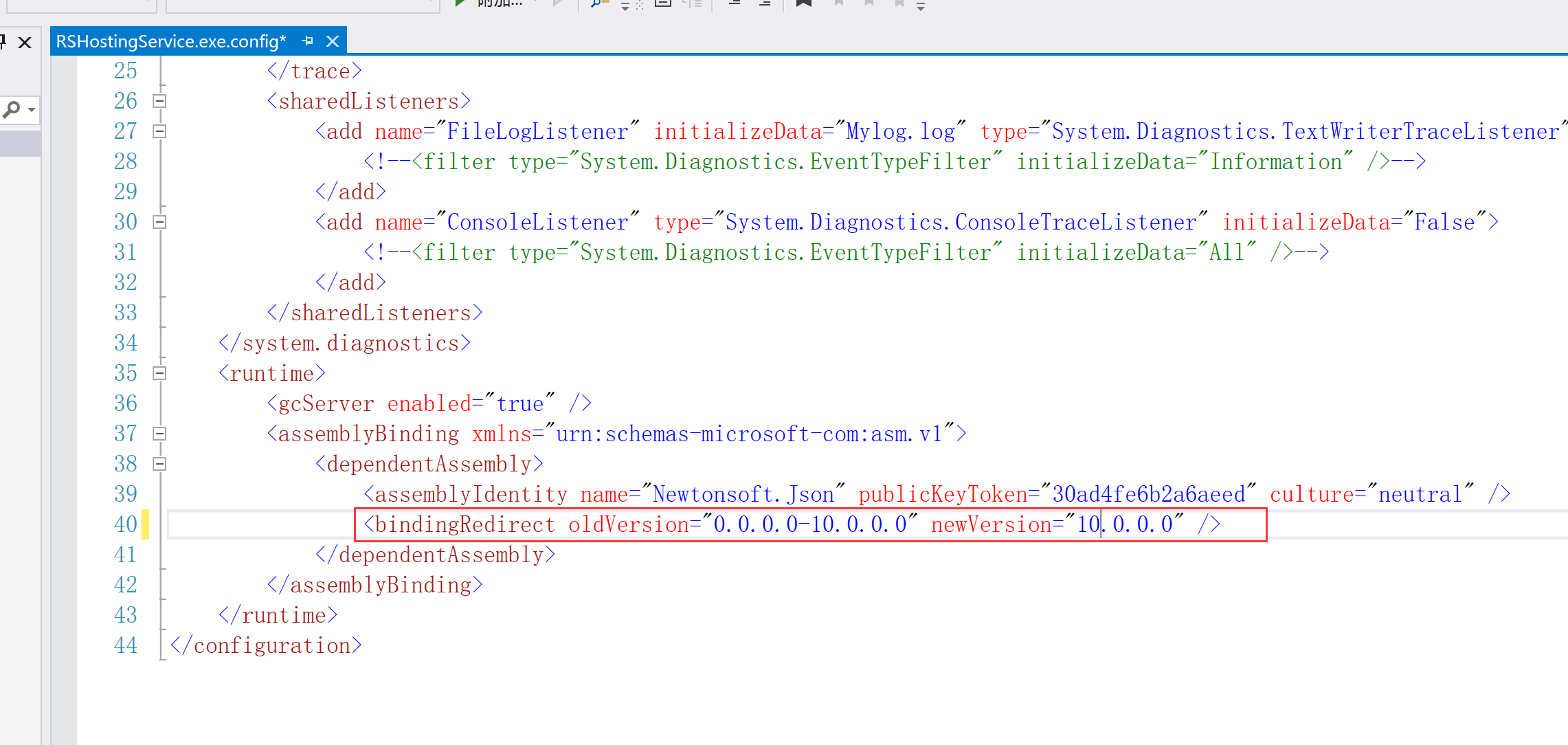
6、将Newtonsoft.Json.dll文件赋值到<install>\RSHostingService目录（可以不用做这步）；

7、通过IIS生成machineKey：

<machineKey validationKey="2C021250F3B5A60CA2DCD6891322FFB1EEAB86E9ABAB84B52A32656BE19EA2E76CC7B3EF939CB51576D2CA00985DC3C858A0A4015D40A9758FEEE1097157E67C" validation="AES" decryptionKey="566C58D2BF52F2A934FAFC89F92C389AE2896D27B2766E96" decryption="AES" />

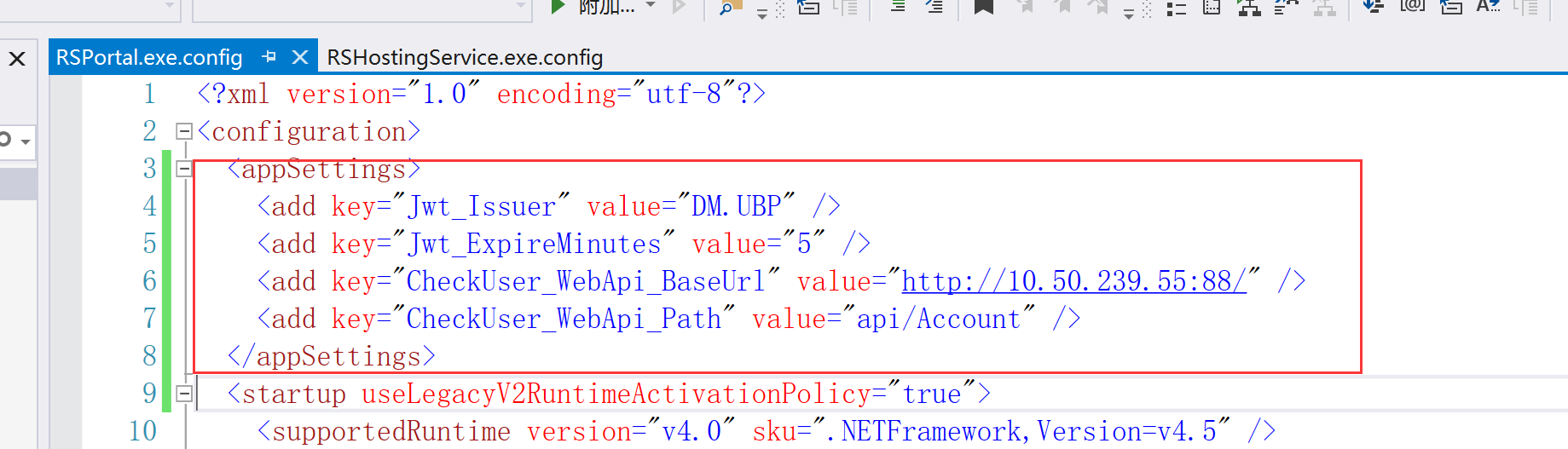
8、修改RSHostingService\RSHostingService.exe.config文件（这步可以不做）：



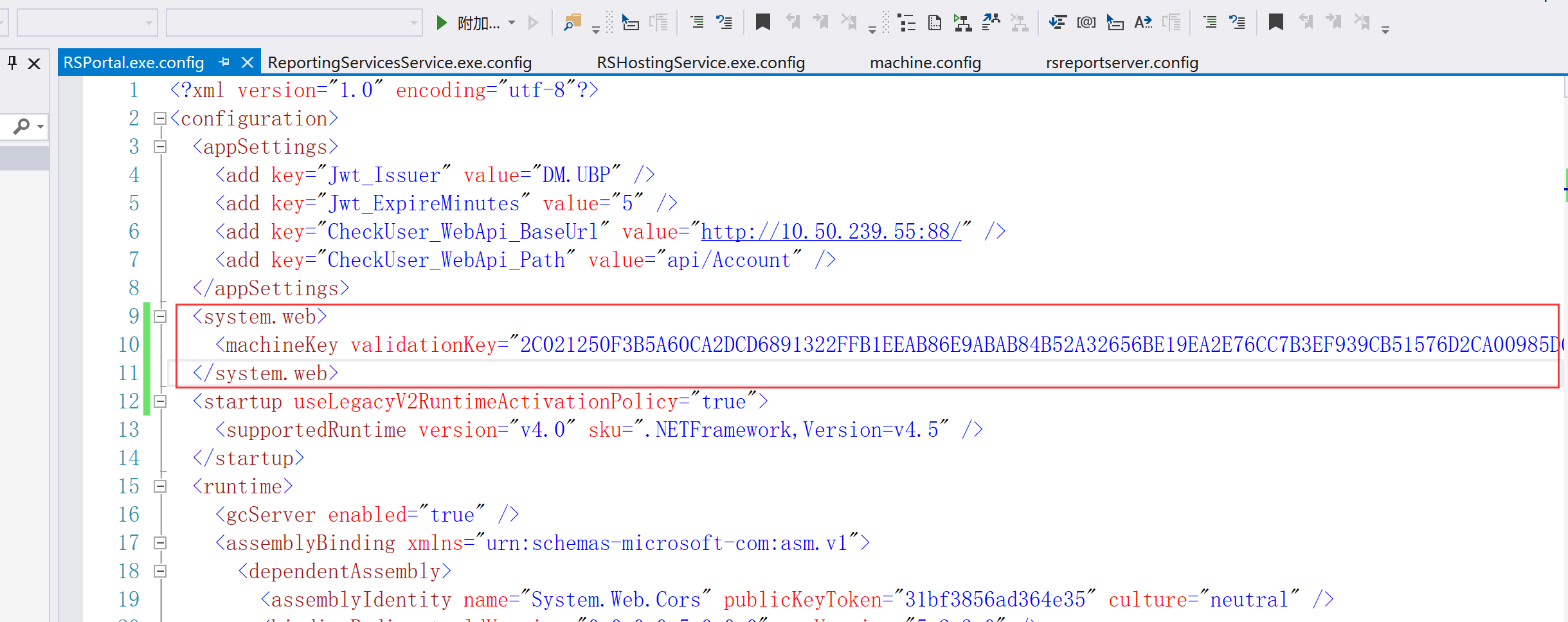


根据PBIRS不同的版本，会使用不同版本的Newtonsoft.Json，需要在这里修改newVersion值。

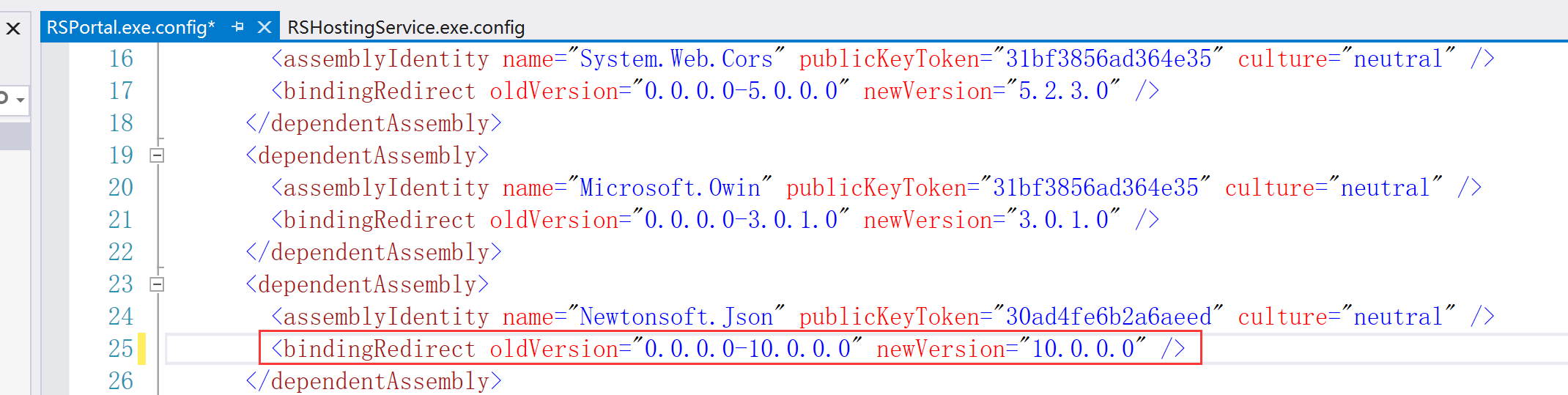
9、修改Portal\ RSPortal.exe.config文件：



（2020年01月27日版本，可以不做这步）



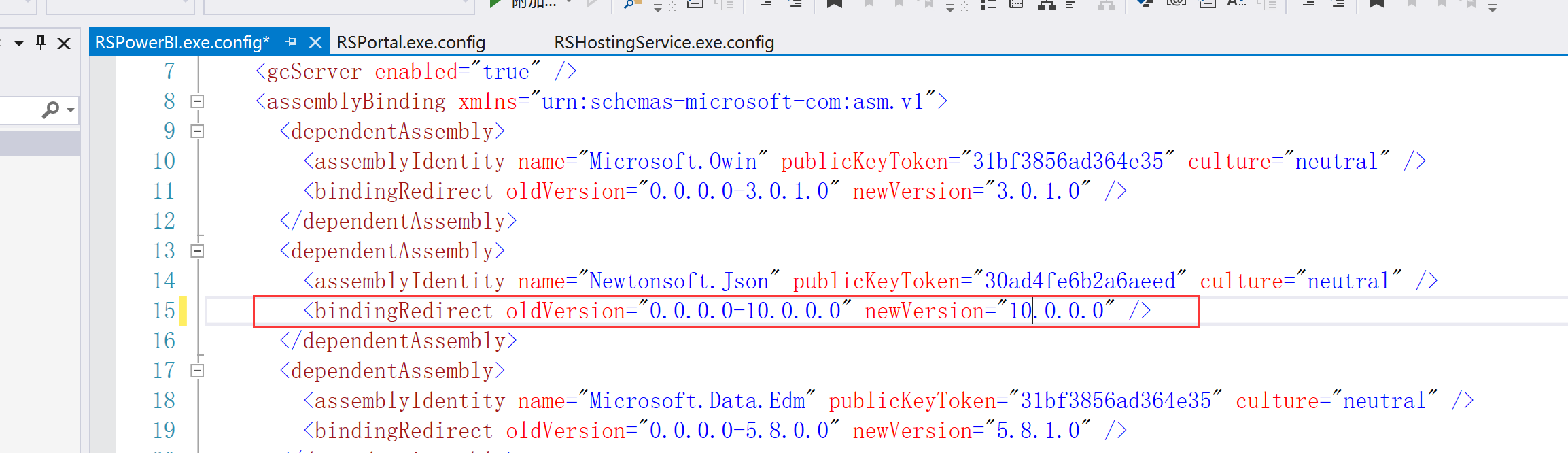
（2020年01月27日版本，可以不做这步，系统升级时已自动修改了newVersion值）



根据PBIRS不同的版本，会使用不同版本的Newtonsoft.Json，需要在这里修改newVersion值。

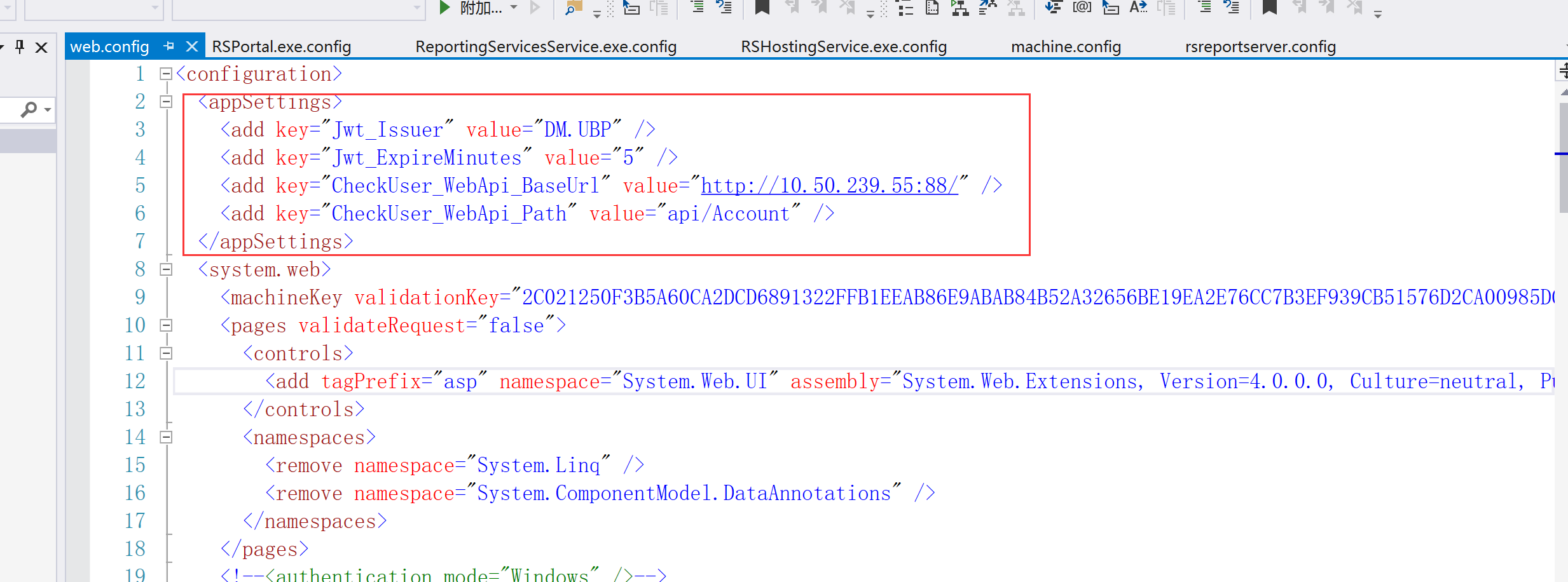
10、修改PowerBI\ RSPowerBI.exe.config文件：

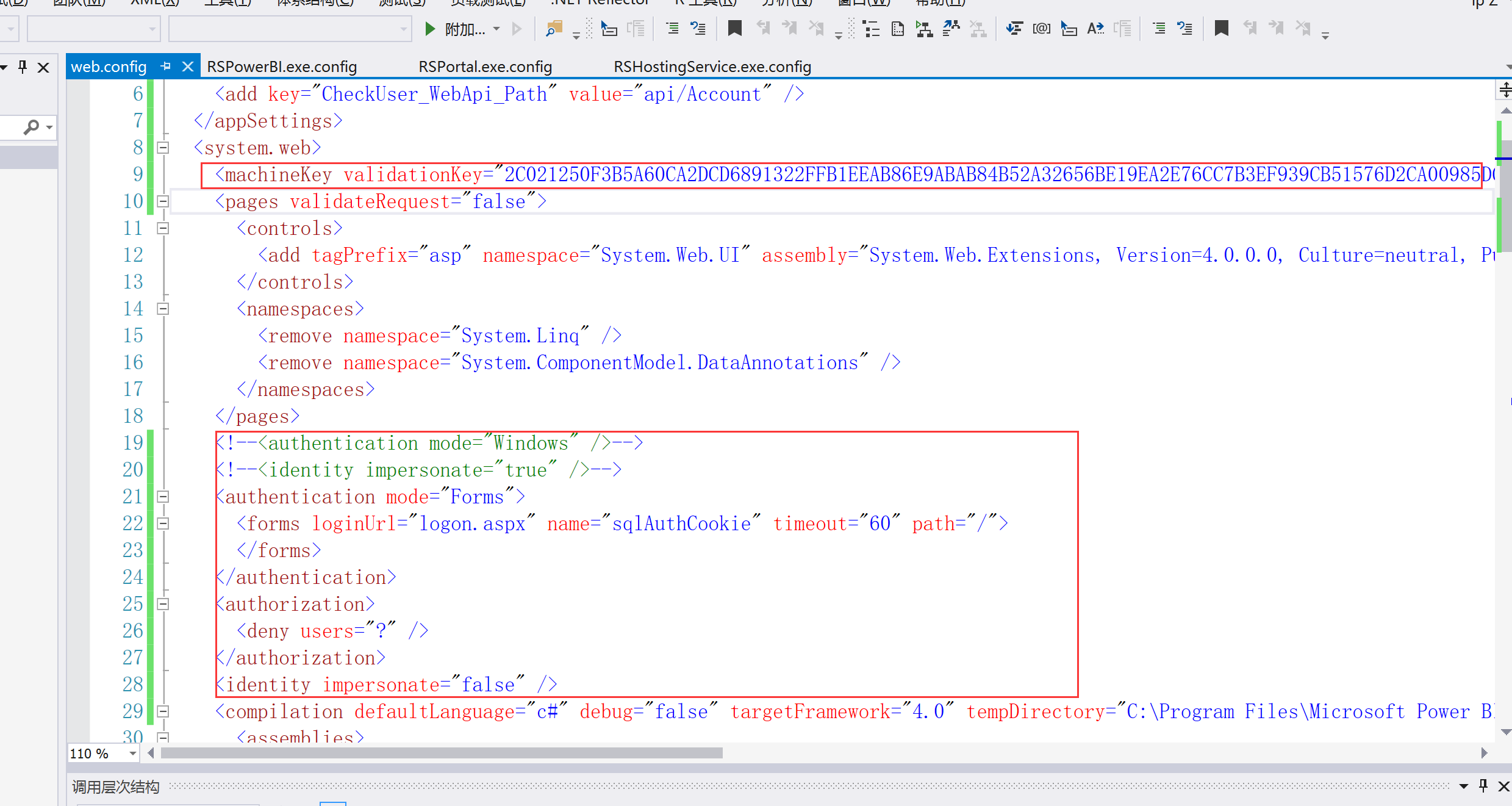
（2020年01月27日版本，可以不做这步，系统升级时已自动修改了newVersion值）

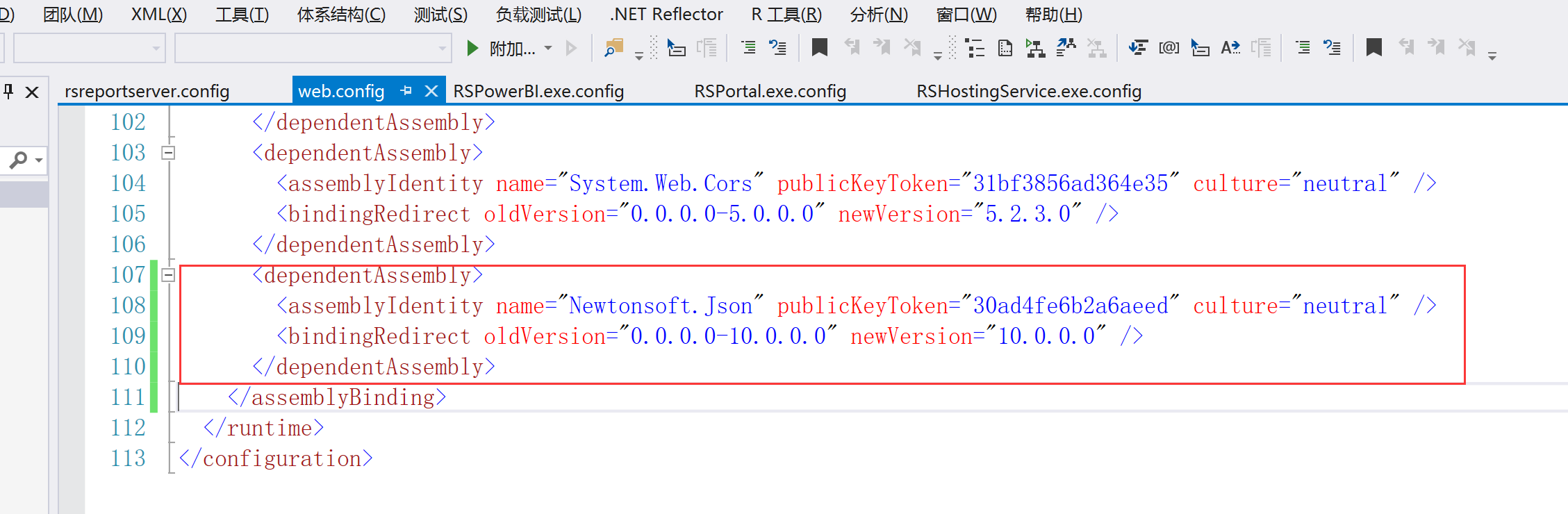


根据PBIRS不同的版本，会使用不同版本的Newtonsoft.Json，需要在这里修改newVersion值。

11、修改ReportServer\web.config文件：

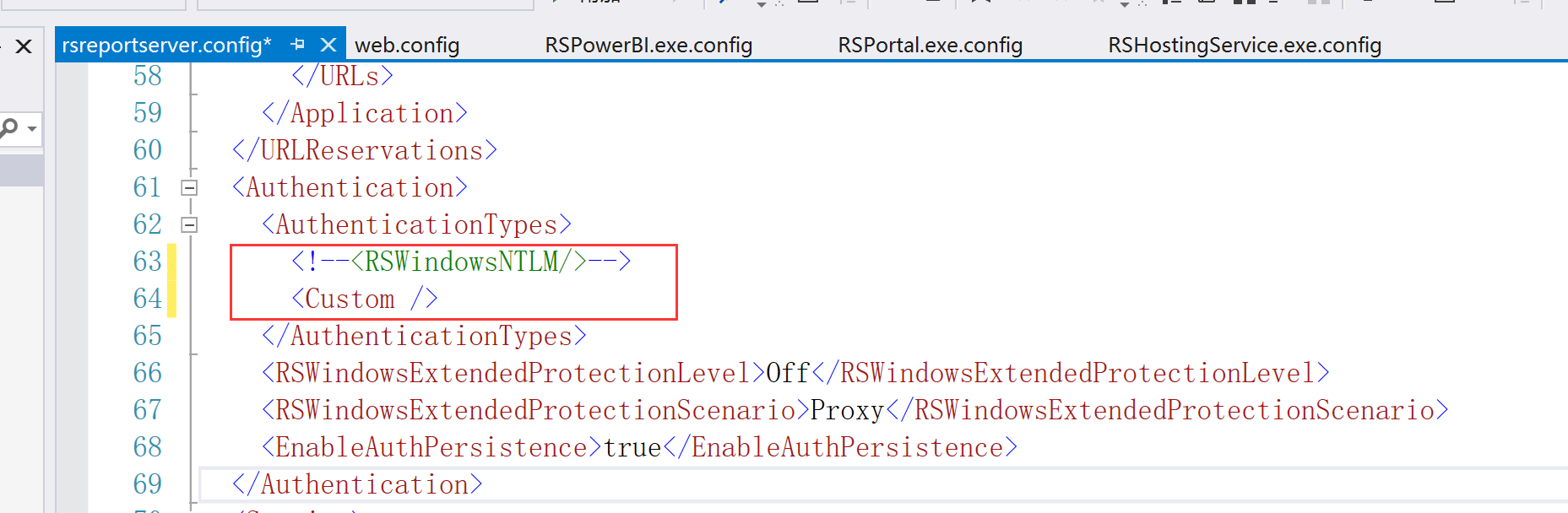


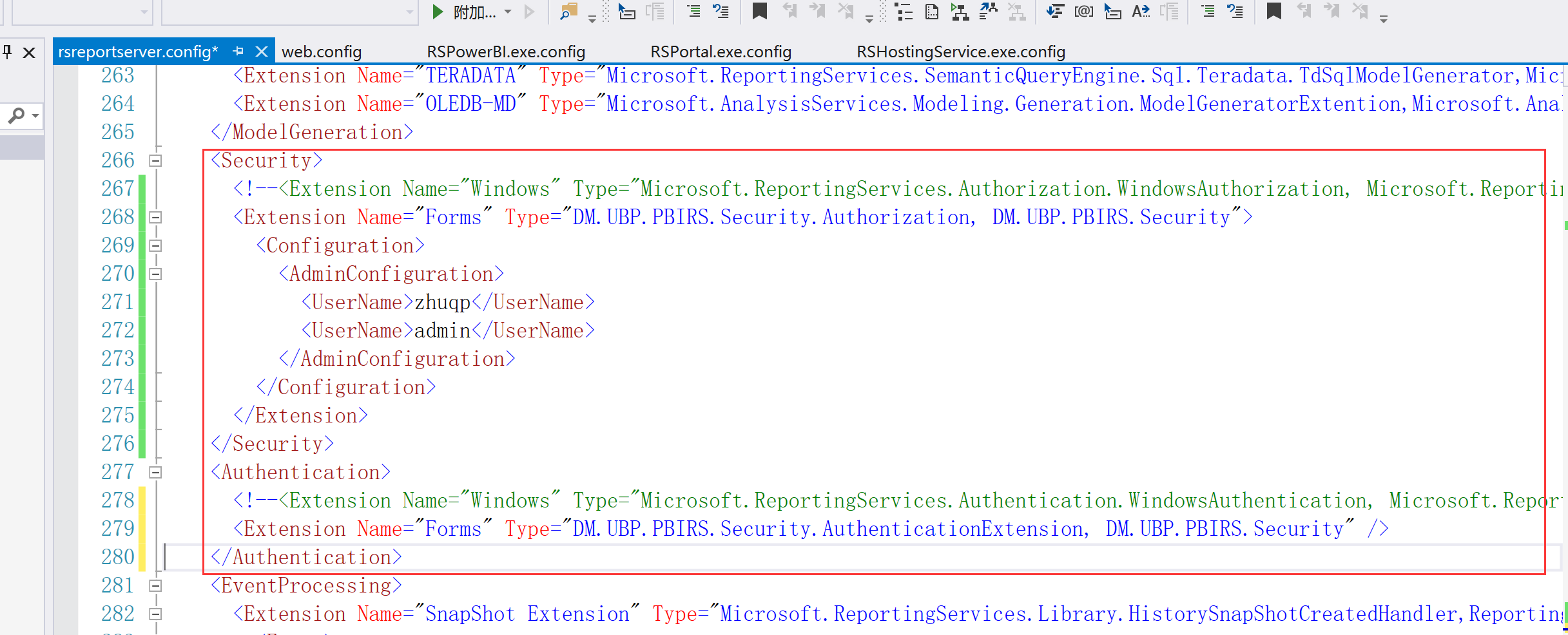


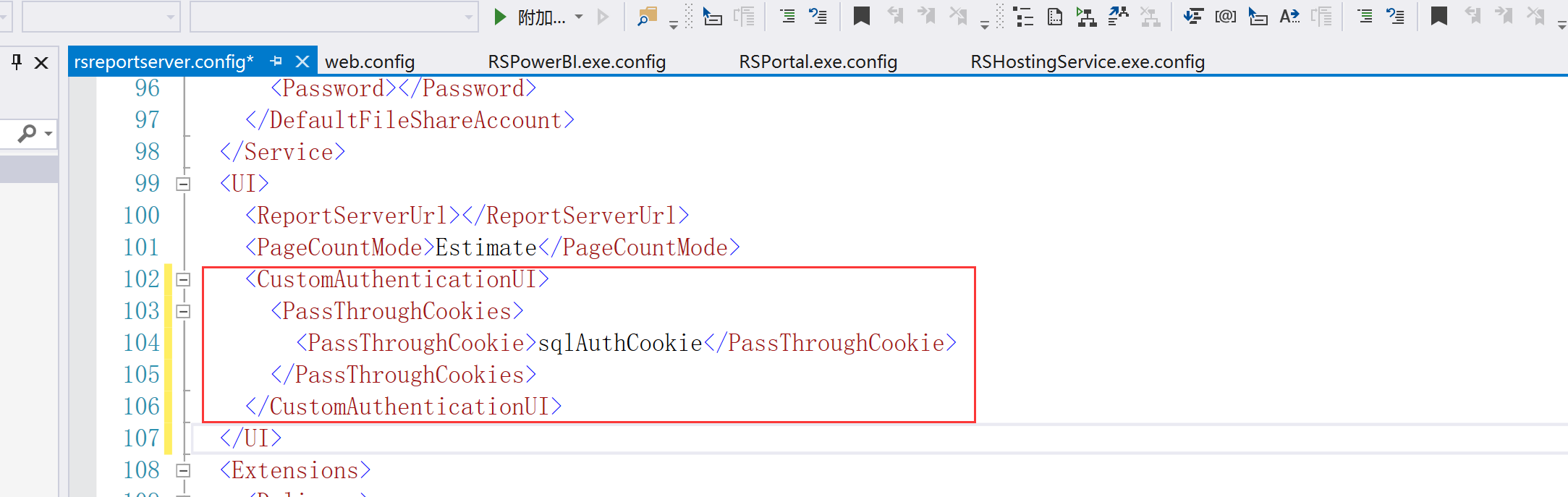


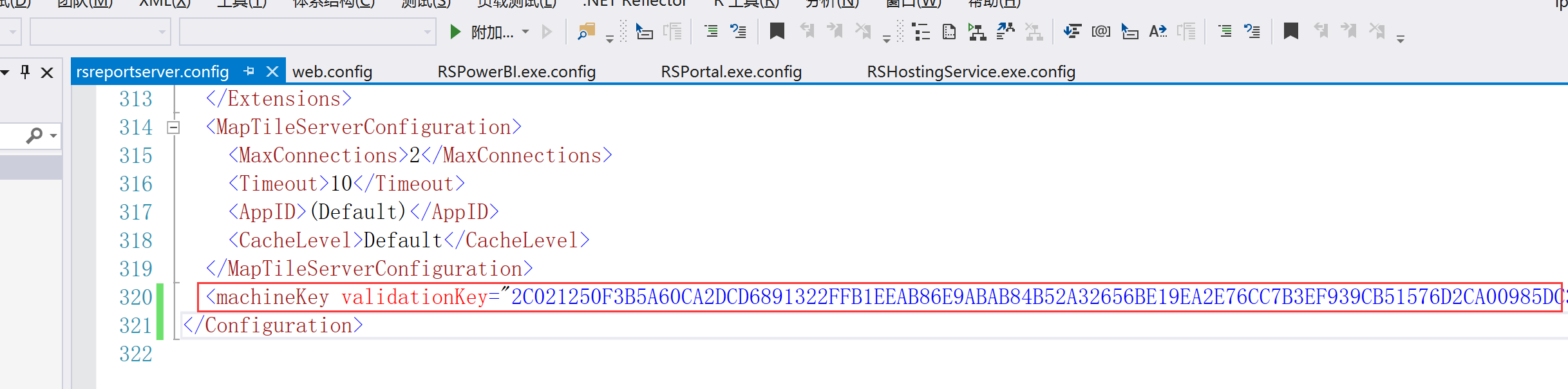
根据PBIRS不同的版本，会使用不同版本的Newtonsoft.Json，需要在这里修改newVersion值。

12、修改ReportServer\RSReportServer.config文件：

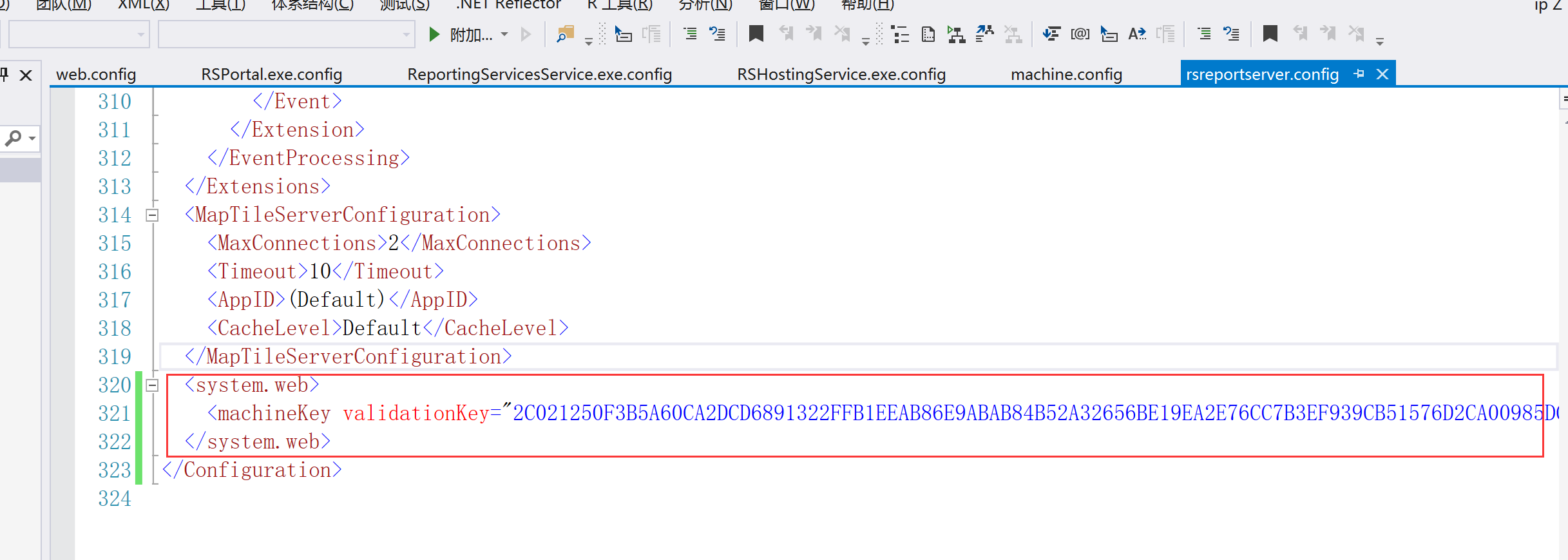




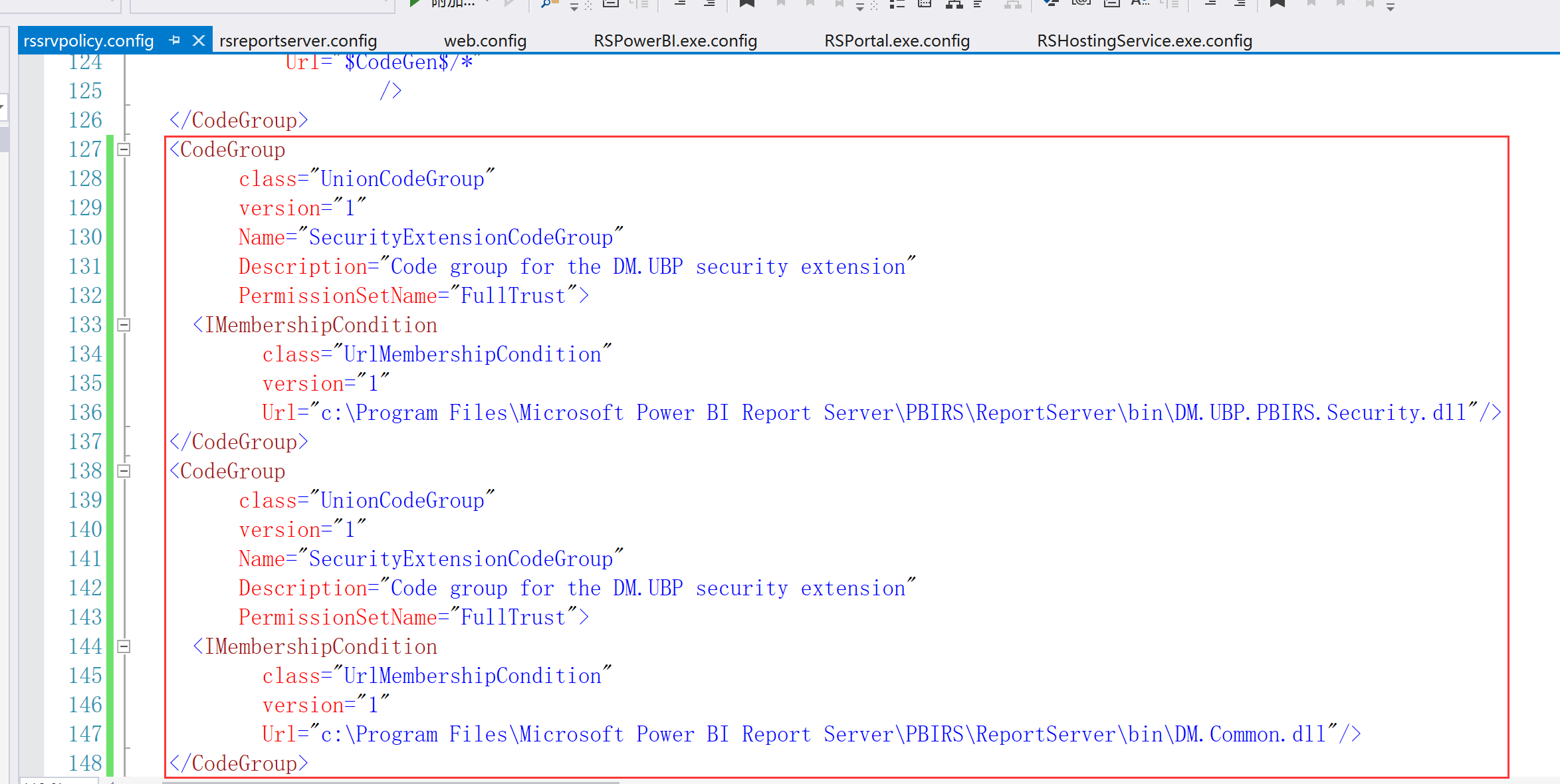




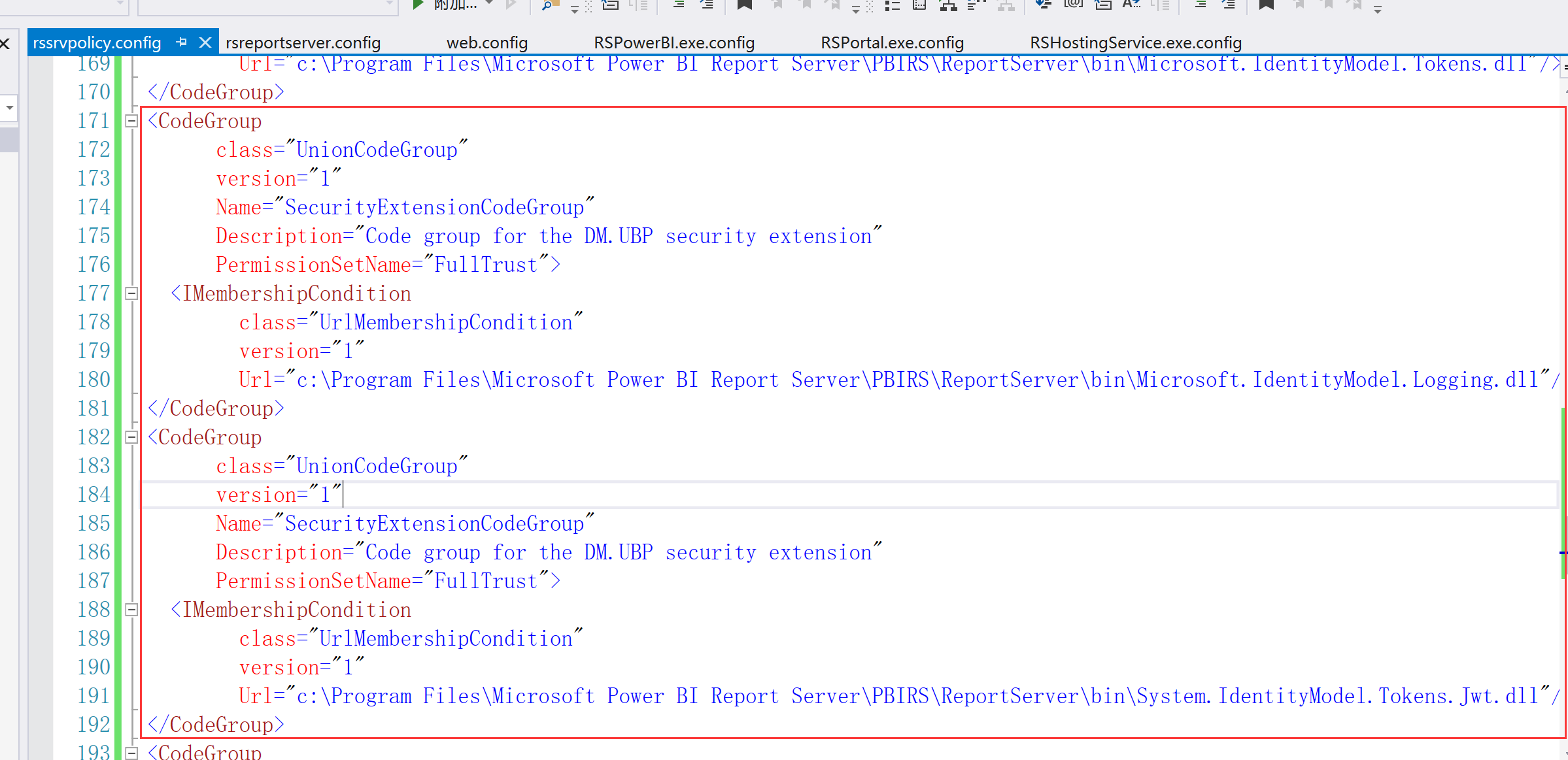
或者

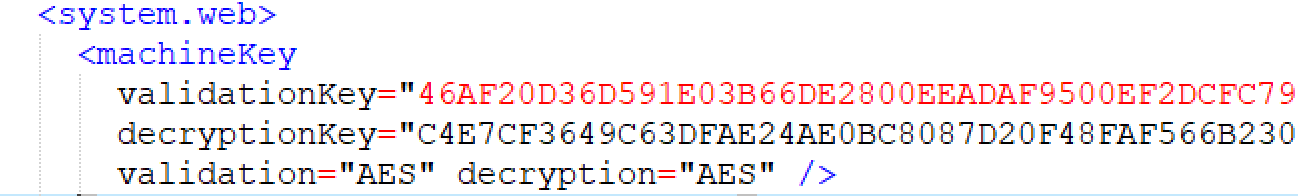


13、修改ReportServer\rssrvpolicy.config文件：







14、修改%windir%\Microsoft.NET\Framework64[version]\config\machine.config：

15、将Oracle的dbhome和client目录增加“NT SERVICE\PowerBIReportServer”的执行和读取权限，尤其是sqlnet.ora文件的权限；

16、将“key.public.json”文件赋值到ReportServer目录下；

定制安全扩展所需的基本配置与以前的版本相同。对于存在于ReportServer文件夹中的web.config和rsreportserver.config，需要进行以下更改。报表管理器不再有单独的web.config，门户网站将继承与reportserver端点相同的设置。

部署样本

* 将Logon.aspx页面复制到<install>\ReportServer directory。
* 将Microsoft.Samples.ReportingServices.CustomSecurity.dll和Microsoft.Samples.ReportingServices.CustomSecurity.pdb复制到该<install>\ReportServer\bin目录。
* 将Microsoft.Samples.ReportingServices.CustomSecurity.dll和Microsoft.Samples.ReportingServices.CustomSecurity.pdb复制到该<install>\Portal目录。
* 将Microsoft.Samples.ReportingServices.CustomSecurity.dll和Microsoft.Samples.ReportingServices.CustomSecurity.pdb复制到该<install>\PowerBI目录。（这只需要为Power BI Report Server完成。）

如果一个PDB文件不存在，它不是由上面提供的构建步骤创建的。确保将Debug / Build的Project Properties设置为生成PDB文件。

修改ReportServer文件夹中的文件

* 修改RSReportServer.config文件。
* 使用Visual Studio或简单的文本编辑器（如记事本）打开RSReportServer.config文件。RSReportServer.config位于该<install>\ReportServer目录中。
* 找到<AuthenticationTypes>元素并修改设置，如下所示：
* < Authentication >
* < AuthenticationTypes >
* < Custom />
* </ AuthenticationTypes >
* < RSWindowsExtendedProtectionLevel > Off </ RSWindowsExtendedProtectionLevel >
* < RSWindowsExtendedProtectionScenario > Proxy </ RSWindowsExtendedProtectionScenario >
* < EnableAuthPersistence > true </ EnableAuthPersistence >

</ Authentication >

* 定位<Security>和<Authentication>元素，所述内<Extensions>元件，并修改设置如下：
* < Security >
* < Extension > Name = “ Forms ” Type = “ Microsoft.Samples.ReportingServices.CustomSecurity.Authorization，Microsoft.Samples.ReportingServices.CustomSecurity ” >
* < Configuration >
* < AdminConfiguration >
* < UserName > [username] </ UserName >
* </ AdminConfiguration >
* </ Configuration >
* </ Extension >

</ Security >

< Authentication >

< Extension Name = “ Forms ” Type = “ Microsoft.Samples.ReportingServices.CustomSecurity.AuthenticationExtension，Microsoft.Samples.ReportingServices.CustomSecurity ” />

</ Authentication >

注意：1、如果您在没有安装安全套接字层（SSL）证书的开发环境中运行示例安全扩展，则必须<UseSSL>在前一个配置条目中将该元素的值更改为False。我们建议您在将Reporting Services与Forms Authentication结合使用时始终使用SSL。

2、如果不加上[username]，则会显示权限不够，能够登陆但无法访问具体信息。

修改RSSrvPolicy.config文件

* 您需要为您的自定义安全扩展添加代码组，以便为​​您的扩展授予FullTrust权限。您可以通过将代码组添加到RSSrvPolicy.config文件来完成此操作。
* 打开位于<install>\ReportServer目录中的RSSrvPolicy.config文件。
* 将以下<CodeGroup>元素添加到具有$ CodeGen的URL成员资格的安全策略文件中的现有代码组（如下所示），然后将如下内容添加到RSSrvPolicy.config中。请确保根据您的ReportServer安装目录更改以下路径：
* < CodeGroup
* class = “ UnionCodeGroup ”
* version = “ 1 ”
* Name = “ SecurityExtensionCodeGroup ”
* Description = “示例安全扩展的代码组”
* PermissionSetName = “ FullTrust ” >
* < IMembershipCondition
* class = “ UrlMembershipCondition ”
* version = “ 1 ”
* Url = “ C ：\ Program Files \ Microsoft Power BI Report Server \ PBIRS \ ReportServer \ bin \ Microsoft.Samples。ReportingServices.CustomSecurity.dll “/>

</ CodeGroup >

修改完后应该如下：

* <CodeGroup
* class="UnionCodeGroup"
* version="1"<span style="font-size: 9pt;">
* </span>PermissionSetName="FullTrust">
* <IMembershipCondition
* class="UrlMembershipCondition"
* version="1"
* Url="$CodeGen$/\*"/>
* </CodeGroup>
* <CodeGroup
* class="UnionCodeGroup"
* version="1"
* Name="SecurityExtensionCodeGroup"
* Description="Code group for the sample security extension"
* PermissionSetName="FullTrust">
* <IMembershipCondition
* class="UrlMembershipCondition"version="1"
* Url="C:\Program Files\Microsoft SQL Server\MSRS11.MSSQLSERVER\Reporting Services\ReportServer\bin\Microsoft.Samples.ReportingServices.CustomSecurity.dll"/></CodeGroup>

注意：为简单起见，表单身份验证示例是弱命名的，并且需要安全策略文件中的简单URL成员资格条目。在您的生产安全扩展实现中，您应该创建强名称的程序集，并在为程序集添加安全策略时使用强名称成员条件。有关强名称程序集的更多信息，请参见MSDN上的创建和使用强名称程序集主题。

修改Report Server的Web.config文件

* 在文本编辑器中打开Web.config文件。默认情况下，该文件位于<install>\ReportServer目录中。
* 找到<identity>元素并将“模拟”属性设置为false。

< identity impersonate = “ false ” />

* 找到该<authentication>元素并将模式属性更改为窗体。另外，将以下<forms>元素添加为元素的子<authentication>元素，并按如下方式设置loginUrl，name，timeout和path属性：
* < authentication mode = “ Forms ” >
* < forms loginUrl = “ logon.aspx ” name = “ sqlAuthCookie ” timeout = “ 60 ” path = “ / ” > </ forms >

</ authentication >

* 在<authorization>元素后面直接添加以下<authentication>元素。
* < 授权 >
* < 拒绝 用户 = “？” />

</ 授权 >

这将拒绝未经身份验证的用户访问报表服务器的权利。<authentication>元素的先前建立的loginUrl属性将未经身份验证的请求重定向到Logon.aspx页面。

## 第4步：生成机器密钥

使用表单身份验证要求所有报表服务器进程都可以访问身份验证Cookie。这涉及到配置机器密钥和解密算法 - 这是以前设置SSRS以在扩展环境中工作的人员熟悉的步骤。

生成并添加< machineKey >下<Configuration>在RSReportServer.config文件。

<machineKey

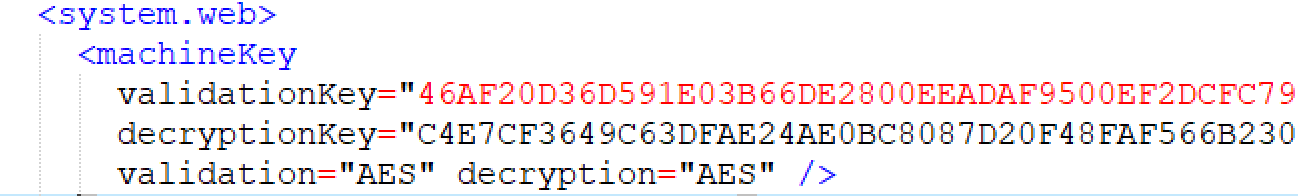
validationKey="46AF20D36D591E03B66DE2800EEADAF9500EF2DCFC79F25994363AB31BC99E977BEC75697F613707D062983D30B057AEBC102A5690662812CAFB197EE070E232"

decryptionKey="C4E7CF3649C63DFAE24AE0BC8087D20F48FAF566B2305DE9"

validation="AES" decryption="AES" />

注意大小写

您应该使用特定于您的部署的验证密钥，有几种工具可以生成密钥，例如Internet Information Services Manager（IIS）

还需要修改%windir%\Microsoft.NET\Framework64[version]\config\machine.config：

## 第5步：配置Passthrough cookie

新门户和报表服务器使用内部soap API进行一些操作。当需要将其他Cookie从门户网站传递到服务器时，PassThroughCookies属性仍然可用。更多详细信息：[https](https://msdn.microsoft.com/en-us/library/ms345241.aspx) : //msdn.microsoft.com/en-us/library/ms345241.aspx在rsreportserver.config文件中添加以下内容<UI>

< UI >

< CustomAuthenticationUI >

< PassThroughCookies >

< PassThroughCookie > sqlAuthCookie </ PassThroughCookie >

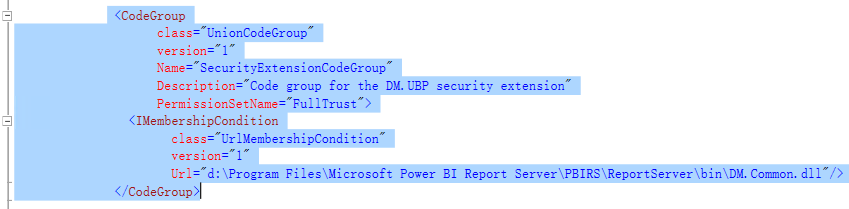
</ PassThroughCookies >

</ CustomAuthenticationUI >

</ UI >

## 第6：关于安全调用DLL异常处理

在DM.UBP.PBIRS.Security.dl中调用其它DLL函数如出现安全/权限提示。在需要在rssrvpolicy.config中增加相应DLL配置，并且重启服务。如下



2019年05月20日版升级：

由于使用的Newtonsoft.Json.dll版本升级到了1102，会导致JWT验证Token失败，需要将ReportServer\bin下的Newtonsoft.Json.dll替换成1003版本。其它目录下Newtonsoft.Json.dll是否需要替换待定。