1.引用DotNetCasClient.DLL 或者工程代码，在 E:\工作\DotNetCasClient 文件夹里

2.修改web.config

在configuration\ configSections节点下增加section节点

<configuration>

<configSections>

<!--单点登录配置,配置参数在casClientConfig节点-->

<section name="casClientConfig" type="DotNetCasClient.Configuration.CasClientConfiguration, DotNetCasClient"/>

</configSections>

</configuration>

在configuration节点下增加casClientConfig节点

<configuration>

<casClientConfig

casServerLoginUrl="http://www.jitar.com.cn:8080/usermgr3/login" casServerUrlPrefix="http://www.jitar.com.cn:8080/usermgr3/"

serverName=""

notAuthorizedUrl="~/NotAuthorized.aspx"

cookiesRequiredUrl="~/CookiesRequired.aspx"

redirectAfterValidation="true"

gateway="false"

renew="false"

singleSignOut="true"

ticketTimeTolerance="5000"

ticketValidatorName="Cas20"

proxyTicketManager="CacheProxyTicketManager" serviceTicketManager="CacheServiceTicketManager" gatewayStatusCookieName="CasGatewayStatus"/>

</configuration>

serverName可以不填写, DotNetCasClient代码已做特殊处理。填写的话就写asp.net站点的Url地址，那么访问asp.net站点时，只能按照这个来访问。

casServerLoginUrl 指向单点登录服务器/login

casServerUrlPrefix指向单点登录服务器

在configuration / system.web节点下增加authentication和 authorization节点

<configuration>

<system.web>

<authentication mode="Forms">

<forms name="casauth"

loginUrl="http://www.jitar.com.cn:8080/usermgr3/login"

timeout="300"

defaultUrl="~/Default.aspx"

cookieless="UseCookies"

slidingExpiration="true"

path="/"/>

</authentication>

<authorization>

<deny users="?"/>

</authorization>

</system.web>

</configuration>

path属性定义了对站点下那些页面进行登录验证。

/表示对整个站点都进行验证

/login表示访问login目录时进行验证

loginUrl属性表示登录的页面。可以是asp.net的自定义的登录页面。这里我们设置为CAS服务器上的登录页面，也就是说，如果没登录，将转向到CAS 的登录页面。

deny users="?" 必须的，这样才能对path下的页面验证用户是否登录。如果设置为 <allow users="\*"/>，则不验证。

在configuration / system.web节点下增加httpModules节点

<configuration>

<system.web>

<httpModules>

<!--单点配置-->

<add name="DotNetCasClient" type="DotNetCasClient.CasAuthenticationModule,DotNetCasClient"/>

</httpModules>

</system.web>

</configuration>

httpModules节点如果有多个，将DotNetCasClient放在最后一个，避免终止其他的httpModules.

在configuration / system.webServer节点下增加modules节点

<configuration>

<system.webServer>

<modules runAllManagedModulesForAllRequests="true">

<!--单点登录配置-->

<remove name="DotNetCasClient"/>

<add name="DotNetCasClient" type="DotNetCasClient.CasAuthenticationModule,DotNetCasClient"/>

</modules>

</system.webServer>

</configuration>

modules节点如果有多个，将DotNetCasClient放在最后一个.

注意下，先写remove节点 再写 add节点.

以上配置基本完成。如果在asp.net站点中有自己的注销退出功能，需要再增加下面的配置。

<configuration>

<!--退出的操作-->

<location path="CasLogOff.aspx">

<authorization>

<allow users="\*"/>

<allow users="?"/>

</authorization>

</system.web>

</location>

</configuration>

对文件CasLogOff.aspx不进行拦截验证。allow users="\*"是必须的，否则注销不了。

CasLogOff.aspx主要代码是：

Session.Clear();

Session.Abandon();//清除Session

if (DotNetCasClient.CasAuthentication.CasServerUrlPrefix != null)

{

if (!string.IsNullOrEmpty(DotNetCasClient.CasAuthentication.CasServerUrlPrefix))

{

DotNetCasClient.CasAuthentication.ClearAuthCookie();//清除本地票证cookie

DotNetCasClient.CasAuthentication.SingleSignOut();//去CAS服务器注销

}

}

然后再加上asp.net自己产品的一些注销处理。

以上方法是对全站点进行单点登录拦截验证。如果我们自己有登录页面，该如何进行设置？

3.自定义登录页面的设置：

对于自定义登录有2中情况。

第一：登录界面代码嵌入在其他页面中。例如：站点的首页是default.aspx,其中里面有自登录界面代码:<form action='default.aspx'>.......</form>

处理方式如下：

在Web.Config中的配置与上面的配置稍微有变化，就是将

<authorization>

<deny users="?"/>

</authorization>

删除，或者修改为

<authorization>

<allow users="\*"/>

</authorization>

意思是不拦截。

然后在站点建立子文件夹/login 如果已经存在，可以改为其他名称。

在该文件夹下，增加Web.Config文件，内容为：

<?xml version="1.0"?>

<configuration>

<system.web>

<authorization>

<deny users="?" />

</authorization>

</system.web>

</configuration>

在该文件夹下，增加default.aspx文件，对登录用户信息进行处理，设置Session Cookie之类asp.net系统需要的东西。

然后修改登录Form代码：

<form action=" http://www.jitar.com.cn:8080/usermgr3/redirectlogin" method="post">

<input type="text" name="username">

<input type="password" name="password">

<input type="hidden" name="service" value="<%=asp.net站点地址%>/login">

<input type="hidden" name="failpage" value="<%=asp.net站点地址%>">

</form>

service 中带着login,是让CAS登录后转向到login目录下的default.aspx来处理。

redirectlogin是在CAS服务器上我们自己增加的一个登录处理。

第二种情况：存在单独的一个login.aspx。

最好按照最前面的web.config来配置。这样就放弃了login.aspx,系统自动转向到CAS去登录。

如果想继续保留login.aspx进行如下处理：

将

<authorization>

<deny users="?"/>

</authorization>

删除，或者修改为

<authorization>

<allow users="\*"/>

</authorization>

login.aspx中的Form代码参照第1种情况来进行更改。