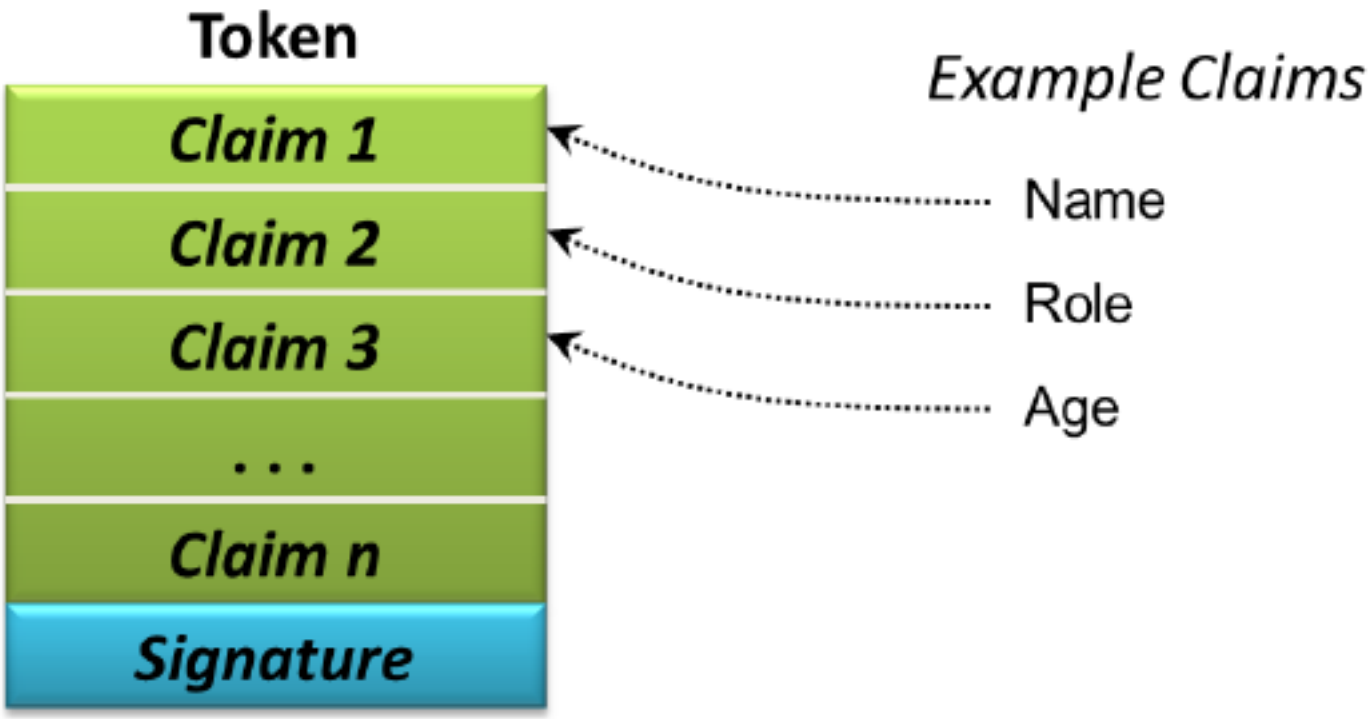
عنوان: **مروری بر Claim**
نویسنده: محمد باقر سیف اللهی
تاریخ: ۱۳۹۱/۰۶/۱۱ ۰:۴۹
آدرس: www.dotnettips.info
گروه‌ها: SharePoint, Authentication

تعریف :

در این پست قصد دارم در مورد claim که از آن به عنوان یک Abstraction برای شناسایی نام برده شده ، صحبت کنم و گریزی با ارتباط آن با شیرپوینت بزنم . مایکروسافت در جایی Claim را این گونه تعریف کرده بود : یک عبارت که یک شیئ ، آن را در باره خودش یا شیئ دیگری می‌سازد . Claim یک Abstraction برای شناسایی فراهم می‌کند . برای مثال میتوان گفت که یک عبارت که شامل نام ، شناسه ، کلید ، گروه بندی ، ظرفیت و ... باشد ، فراهم می‌کند .

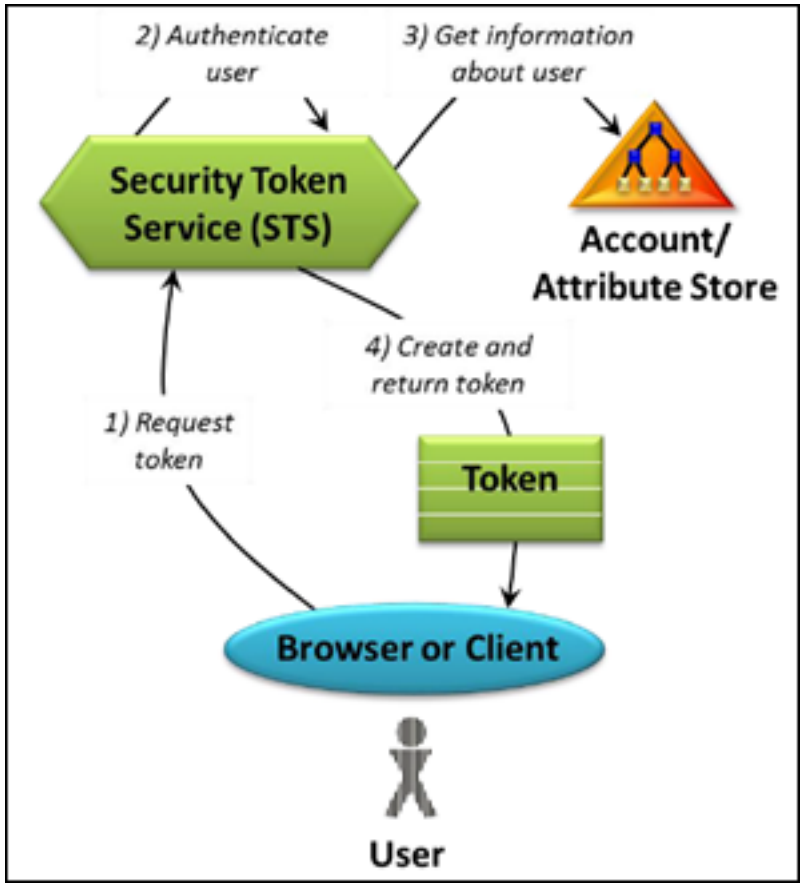
لازم است به تعریف Token هم اشاره ای شود . هنگامی که یک شناسه دیجیتالی در شبکه در حال گذر است ، فقط حاوی مجموعه ای از بایت‌ها است .(ارجاع به مجموعه ای از بایت‌ها که حاوی اطلاعات شناسایی به عنوان یک Token امنیتی با فقط یک Token باشد، امری عادی است) . در محیطی که بر مبنای Claim بنا شده است ، یک Token حاوی یک یا چند Claim است که هر یک می‌تواند برخی تکه‌های اطلاعاتی را برای شناسایی (بیشتر در مورد کاربران و افراد استفاده می‌شود) ، در خود جای دهد



Claimها تقریبا هر چیزی را در مورد یک کاربر می‌تواند ارائه دهد. . برای مثال در Token تصویر بالا ، 3 claim اول به اطلاعات نام و نقش و سن کاربر اشاره دارند .

فراهم کننده - توزیع کننده :

Claimها توسط یک فراهم کننده (Provider) توزیع می‌شوند (Issuer) و سپس به آنها یک یا چند مقدار ، اختصاص می‌یابد و در Security Token هایی که توسط یک توزیع کننده ، توزیع می‌شوند ، بسته بندی می‌شود و معمولا به عنوان Security Token Service یا STS شناخته می‌شوند . برای مشاهده تعریف اصطلاحات مرتبط به Claim به اینجا مراجعه کنید



STS ، می‌تواند توسط چند Identity Provider - IdP به مالکیت در بیاید . یک فراهم کننده شناسه در STS یا IP-STS ، یک سرویس است که درخواست‌ها را برای اطمینان از شناسایی Claimها مدیریت می‌کند . یک IP-STS از یک پایگاه داده که Identity Store نامیده می‌شود برای نگهداری و مدیریت شناسه‌ها و خصیصه‌های مرتبط با آنها استفاده می‌کند .Identity Store می‌تواند یک دیتا بیس معمولی مانند SQL Server باشد یا یک محیط پیچیده‌تر مانند Active Directory . (از قبیل Active Directory Lightweight Directory Service یا Domain Services) .

قلمرو - Realm

بیانگر مجموعه ای از برنامه‌ها ، URLها ، دامنه‌ها یا سایت هایی می‌باشد که برای Token ، معتبر باشد .معمولا یک Realm با استفاده از دامنه (microsoft.com) یا مسیری داخل دامنه (microsoft.com/practices/guides) تعریف می‌شود .بعضی وقت‌ها یک realm ، به عنوان Security Domain بیان می‌شود چرا که تمام برنامه‌های داخل یک مرز امنیتی ویژه ای را احاطه کرده است .

Identity Federation

Identity Federation در حقیقت دریافت کننده Token هایی است که در خارج از Realm شما ایجاد شده اند و در صورتی Token را می‌پذیرد که شما Issuer یا توزیع کننده را مورد اطمینان معرفی کرده باشد . این امر به کاربران اجازه می‌دهد تا بدون نیاز به ورود به realm تعریف شده خودشان ، از realm دیگری وارد برنامه شوند . کاربران با یک بار ورود به محیط برنامه ، به چندین realm دسترسی پیدا خواهند کرد .

Relying party application

هر برنامه سمت client که از Claim پشتیبانی کند

مزایای Claim

جدا سازی برنامه از جزییات شناسایی

انعطاف پذیری در احراز هویت

Single sign-on

عدم نیاز به VPN

متحد کردن مجموعه با دیگر شرکت ها

متحد کردن مجموعه با سرویس‌های غیر از AD

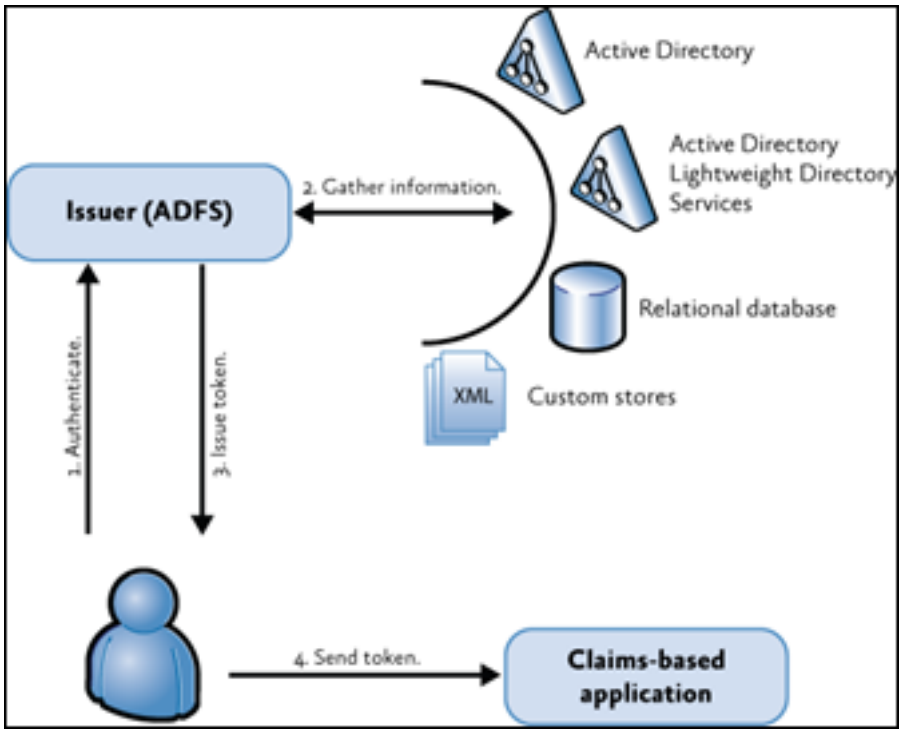
عناصر Claim

Claim شامل عناصر زیر می‌باشد :
Token

Claim

Provider/Issuer
Sharepoint STS
ADFS
ACS
OID
و غیره,

توزیع کننده‌ی ADFS



پرنکل‌ها و Tokenهای Claim

شاید این بخش ، یکی از سردرگم کننده‌ترین مفاهیم باشد . هنگامی که صحبت از Claim می‌شود ، عده ای دچار این عدم توجه صحیح می‌شوند که هر دو نوع مختلفی از Tokenها که با Claimها استفاده می‌شوند ، توسط تمام برنامه‌ها پشتیبانی نمی‌شوند . نکته قابل توجه نوع پروتکلی است که می‌خواهید از آن استفاده کنید و باید کامل از آن مطلع باشید .

Security Token هایی که در اینترنت رفت و آمد می‌کنند ، معمولا یکی از دو نوع زیر هستند :

- توکن‌های Security Assertion Markup Language یا SAML که ساختار XMLی دارند و encode شده اند و داخل ساختارهای دیگر از قبیل پیغام‌های HTTP و SOAP جای می‌گیرند

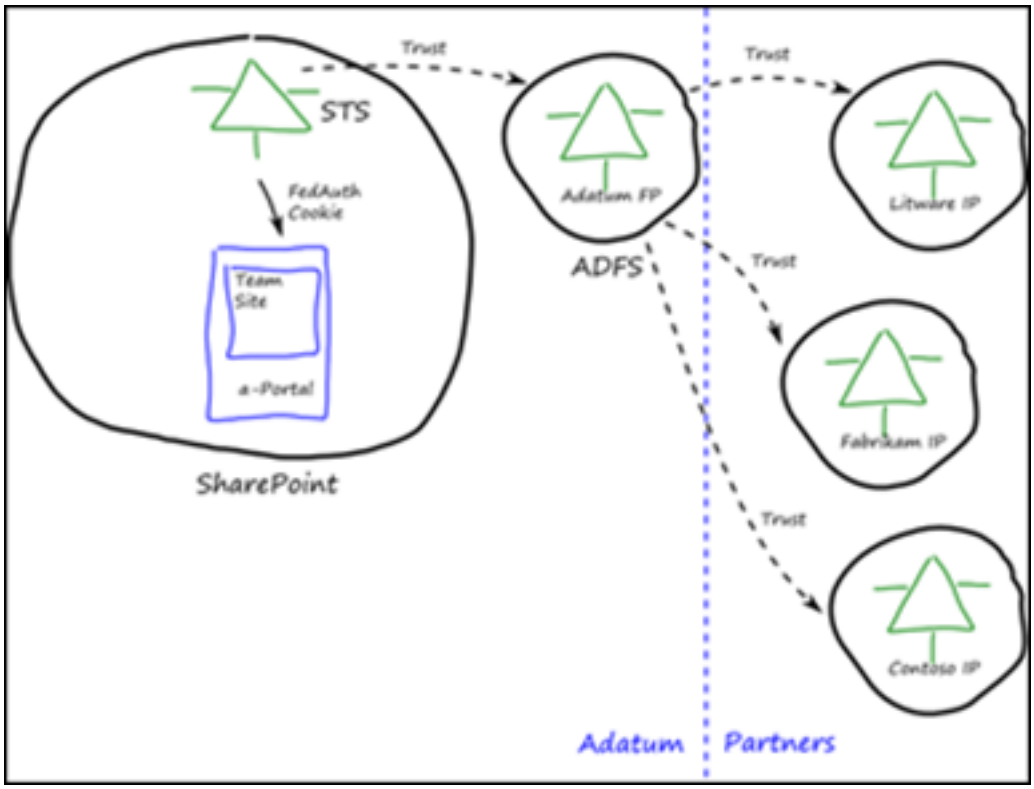
- Simple Web Token یا SWT که درون هدرهای درخواست یا پاسخ HTTP جای میگیرند .(WS-Federation)

نوع متفاوتی از Token که وابسته به مکانیسم احراز هویت است، ایجاد شده است . برای مثال اگر از Claim با Windows Sign-in استفاده می‌کنید ، شیرپوینت 2010 ، شیئ UserIdentity را به شیئ ClaimIdentity نبدیل می‌کند و claim را تقویت کرده و Token حاصله را مدیریت می‌کند . (این نوع Toaken جزء SAML نمی‌شود)

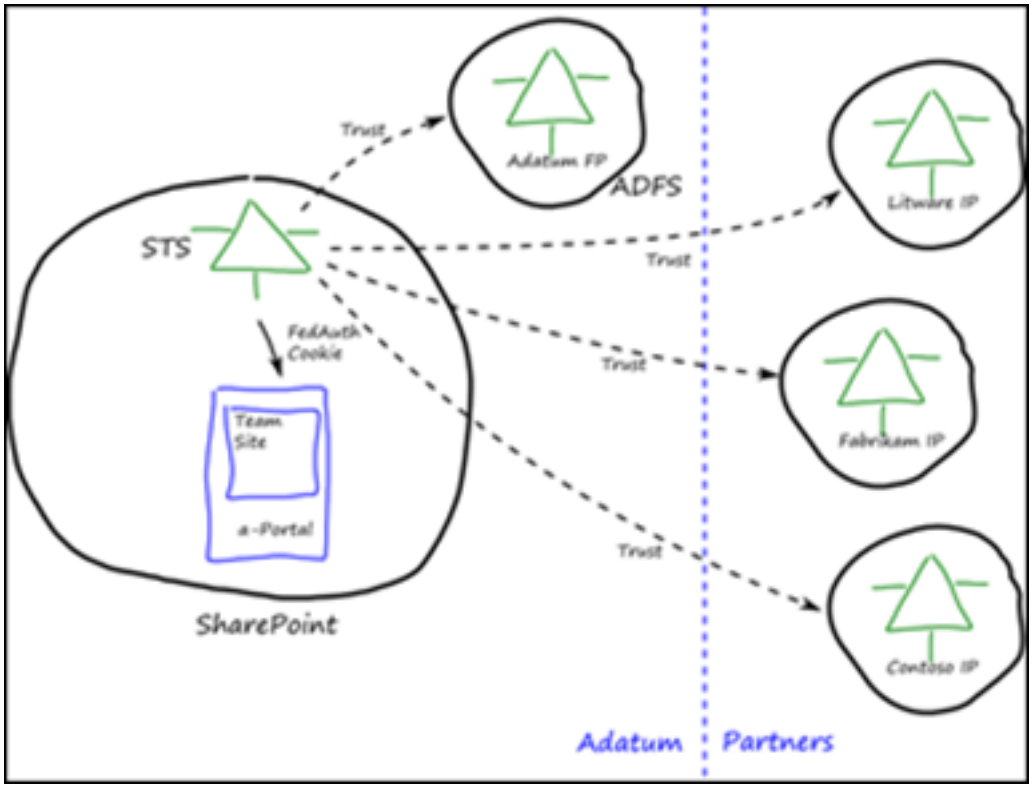
تنها راه به گرفتن توکن‌های SAML ، استفاده از یک Provider برای SAML است . مانند Windows Live ID یا ADFS . [+]

معماری برنامه‌های مبتنی بر Claim

نام مدل : Direct Hub Model



نام مدل : Direct Trust Model



مزایا :

- مدیریت راحت‌تر برای multiple trust relationships در ADFS نسبت به Sharepoint
- مدیریت ساده‌تر در single trust relationship در شیرپوینت و عدم نیاز به فراهم کننده‌های سفارشی سازی شده برای Claim
- قابلیت استفاده از ویژگی‌های ADFS برای پیگیری توزیع Token ها
- ADFS از هز دوی SAML و WS-Federation پشتیبانی می‌کند
- توزیع کننده ADFS اجازه می‌دهد تا خصیصه‌های LDAP را از AD استخراج کنید
- ADFS به شما اجازه استفاده از قواعد دستوری SQL را برای استخراج داده‌ها از دیگر پایگاه‌های داده می‌دهد
- کارایی و اجرای مناسب

معایب :

- کند بودن
- عدم پشتیبانی از SAML-P
- نیازمند تعریف کاربرها در AD یا نواحی مورد اطمینان

موفق باشید

نظرات خوانندگان

نویسنده: آزاده
تاریخ: ۱۳۹۲/۰۹/۱۰ ۱۵:۵۹

سلام . خسته نباشید.
سوالی داشتم ، اگر به عنوان مثال من دو کلاس شرکت و کارمند رو داشته باشم که رابطه یک به چند دارند. و هر کاربری که login کرد لازم باشه که فقط اطلاعات شرکت خودش رو ببینه ولی تنها role admin لازمه که اطلاعات همه شرکت رو پس از login ببیند. در اینصورت به نظر شما نیازی هست که از Claim base استفاده کنم با توجه به معایبی که گفته شده یا خیر؟
ممنونم

نویسنده: محمد باقر سیف اللهی
تاریخ: ۱۳۹۲/۰۹/۱۱ ۱۱:۳۶

سلام...
این مفهوم در لایه‌های زیر ساختی یک Application استفاده می‌شود (وابسته به Platform یا حتی پایین‌تر در infrastructure و در لایه‌های پیاده سازی برنامه با این مفهوم کاربرد ندارد). ضمنا بحث claim وابسته به مفهوم Authentication می‌باشد ولی مسئله شما با مفهوم Authorization سروکار دارد.

این موارد برای مقیاس‌های بالا (مانند یک سازمان با کاربران زیاد و پیچیدگی‌های معماری بالا) نمود بیشتری پیدا می‌کند . بیان معایب دلیلی بر کاربردی نبودن آن نیست و با امکان سنجی می‌توان کاربردی و مفید بودن آن را سنجید.
موفق باشید

نویسنده: غلامرضا
تاریخ: ۱۳۹۳/۰۶/۰۶ ۱۵:۵۶

سلام . ببخشید به چه صورت میتونم تو asp.net webforms از claim base authorizatoin استفاده کنم . اگه منبع هم معرفی کنید ممنون میشم.

نویسنده: محمد باقر سیف اللهی
تاریخ: ۱۳۹۳/۰۶/۰۶ ۱۷:۴۶

سلام
این پیوندها را بررسی کنید codeproject
codeproject
msdn - Sample Code
msdn
موفق باشید

عنوان: **Best Practice ی برای تأیید اعتبار کردن کاربران در ASP.NET MVC 4**
نویسنده: محمد صاحب
تاریخ: ۱۳۹۲/۰۴/۱۰ ۱۶:۱۰
آدرس: www.dotnettips.info
گروه‌ها: Security, Authentication, ASP.Net MVC 4, MVC

به صورت پیش فرض دسترسی به تمامی اکشن‌ها مجاز است مگر اینکه آن اکشن به تگ Authorize مزین شود.
حال Best Practice این است که حتی اگه شما یک یا دو اکشنی دارید که نیاز است کاربرای خاصی به آن‌ها دسترسی داشته باشند **بهتر است** که دسترسی به تمام اکشن‌ها محدود شود و بعد آن اکشن‌هایی که نیاز است دسترسی عمومی داشته باشند، بهشون دسترسی داده بشه. در واقع هدف از این Best Practice جلوگیری از قلم افتادن یک اکشن به اشتباه، هنگام دادن تگ Authorize هست.

خوب، برای انجام اینکار از فیلترهای سفارشی سراسری (Global Filters) استفاده می‌کنیم. کافیه خط زیر رو به کلاس FilterConfig اضافه کنید.

```
filters.Add(new AuthorizeAttribute());
```

با این کار تمام اکشن‌های شما با تگ Authorize مزین می‌شوند و تمام دسترسی‌ها محدود.

نکته: این روش تا قبل از ASP.NET MVC 4 فقط برای سیستم‌هایی که هیچ اکشنی با دسترسی عمومی نداشتن جوابگو بوده و در صورت داشتن چنین اکشنی این روش جوابگوی شما نیست. ولی از ASP.NET MVC 4 به بعد با اضافه شدن تگ [AllowAnonymous] این مشکل حل شده.

حالا که تمامی اکشن‌ها محدود شدن، حالا نوبت به اکشن‌هایی می‌رسه که دسترسی عمومی به اونها آزاده.
برای این کار به راحتی از تگ [AllowAnonymous] استفاده می‌کنیم

```
[AllowAnonymous]
public ActionResult Index()
  {
       return View();
   }
```

فایل پروژه BestPracticeForAuthenticatingUsers.rar

نظرات خوانندگان

نویسنده: آیمو
تاریخ: ۱۳۹۲/۰۷/۰۳ ۹:۲۳

سلام . من برای پروژه ام همین کارو کردم . صفحه اول من از توی یک ناحیه(Area) لود میشه . به این صورت که درخواست میره به یک اکشن مثل Home از کنترلر ControlPanel که این اکشن Allow Anonymous هست . بعد از داخل اون Redirect میشه به همون Area و داخل اون هم یه کنترلر Home و یه اکشن Index هست .
اما موقع لود شدن redirect میشم به صفحه لاگین. لطفا راهنمایی کنید.

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۷/۰۳ ۹:۴۲

«بعد از داخل اون Redirect میشه» خودکار هست یا دستی؟ اگر خودکار هست، خوب طبیعی هست جایی که AllowAnonymous نداره (یعنی اکشن متد بعدی که به اون Redirect شده)، کاربر رو به صفحه لاگین هدایت کنه. Global Filters یعنی دقیقا همین. یعنی اعمال سراسری به همه جا، مگر اینکه با AllowAnonymous مستثنی شود.

نویسنده: آیمو
تاریخ: ۱۳۹۲/۰۷/۰۳ ۱۰:۲

اره خب حرفه شما درسته . اما هر دوتا اکشن من AllowAnonymous هست . در واقع از یک اکشن AllowAnonymous میخوام برم به یه اکشن AllowAnonymous داخل یه ناحیه . مشکل اینجاست که نمیره و میره تو صفحه لاگین .

نویسنده: محسن خان
تاریخ: ۱۳۹۲/۰۷/۰۳ ۱۰:۱۹

ممکن هست در اون صفحه دوم داری چیزی رو نمایش می‌دی که ارجاعی داره به یک اکشن متد محافظت شده. مثلا صفحه‌ات چند قسمتی هست. یک قسمت داره از یک اکشن متد غیر عمومی شده اطلاعات دریافت می‌کنه.

نویسنده: ایمان
تاریخ: ۱۳۹۳/۰۴/۰۳ ۱۴:۳۴

سوال من اینه که اگه بخواهیم این محدودیت رو محدودتر کنیم[مثلا فقط نقش abc] دسترسی داشته باشه، اونوقت باید [Authorize(Roles="abc")] رو روی اکشن بگذاریم؟

نویسنده: محمد صاحب
تاریخ: ۱۳۹۳/۰۴/۰۳ ۲۲:۶

اگه قراره روی فقط رو یه اکشن اعمال بشه درسته میتونی حتی میتونی روی کنترلر بزاری که تمام اکشن‌های اون کنترلر رو محدود کنه و اگه میخوای با روشی که تو این پست اومده عمل کنی باید خودت یک Authorize Attribute سفارشی بنویسی و به فیلترهای سراسری ت ادد کنی.

عنوان: **متدهای احراز هویت در VS 2013**
نویسنده: آرمین ضیاء
تاریخ: ۱۳۹۲/۱۰/۱۵ ۲:۱۵
آدرس: www.dotnettips.info
گروه‌ها: ASP.Net, MVC, Authentication, ASP.NET Identity, Security

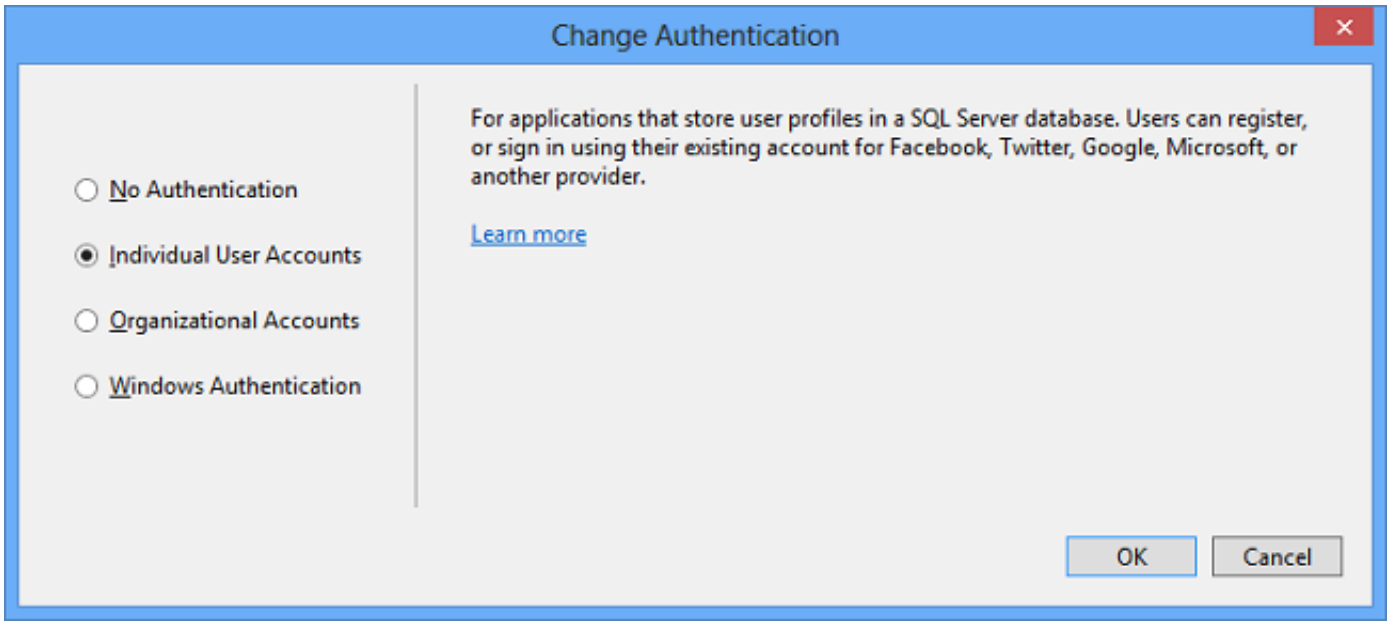
ویژوال استودیو 2013 چندین گزینه برای احراز هویت در قالب‌های پیش فرض پروژه‌های ASP.NET Web Forms, MVC, Web API ارائه می‌کند:

No Authentication

Individual User Accounts

Organizational Accounts

Windows Authentication



No Authentication

اگر گزینه **No Authentication** را انتخاب کنید، پروژه ایجاد شده صفحاتی را برای ورود به سایت نخواهد ساخت. همچنین رابط کاربری ای برای نمایش کاربر فعلی، کلاس‌های موجودیت‌ها برای یک دیتابیس عضویت و رشته‌های اتصالی نیز وجود نخواهند داشت.

Individual User Accounts

اگر گزینه‌ی **Individual User Accounts** را انتخاب کنید، اپلیکیشن شما برای استفاده از ASP.NET Identity (که پیش از این با نام ASP.NET Membership شناخته می‌شد) پیکربندی می‌شود. ASP.NET Identity کاربران را قادر می‌سازد تا با ساختن حساب کاربری جدیدی در سایت و یا با استفاده از تامین کننده‌های ثالثی مانند Facebook, Google و غیره به سایت وارد شوند. این فریم ورک برای ذخیره‌ی داده‌های پروفایل کاربران، بصورت پیش فرض از یک دیتابیس SQL Server LocalDB استفاده می‌کند که می‌توانید بعدا آنرا بر روی SQL Server یا Windows Azure SQL Database نیز منتشر کنید.

این قابلیت‌ها در Visual Studio 2013 در نسخه قبلی نیز وجود داشتند، اما کد سیستم عضویت آن مجددا بازنویسی شده‌است. این بازنویسی دارای مزایای زیر است:

سیستم عضویت جدید بجای استفاده از ماژول ASP.NET Forms Authentication بر پایه OWIN نوشته شده است. این بدین معنا است که از یک مکانیزم احراز هویت واحد می‌توانید در اپلیکیشن‌های ASP.NET Web Forms, MVC, Web API و SignalR استفاده کنید.
سیستم عضویت جدید توسط Entity Framework Code First مدیریت می‌شود و شامل تمامی کلاس‌هایی است که نماینده جداول و موجودیت‌ها هستند. این بدین معنا است که روی الگوی دیتابیس کنترل کامل دارید. سفارشی سازی و تغییر اطلاعات کاربران و پروفایل هایشان بسیار ساده‌تر است، تنها لازم است برای اعمال تغییرات از Code First Migrations استفاده کنید.

سیستم عضویت جدید بصورت خودکار در تمام قالب‌های پروژه پیش فرض، نصب و پیاده سازی می‌شود. این امکان برای تمام پروژه‌هایی که دات نت فریم ورک 4.5 را هدف قرار می‌دهند وجود دارد. ASP.NET Identity هنگام تولید وب سایت‌های اینترنتی که اکثر کاربرانشان خارجی (External) هستند گزینه خوبی است. اگر سازمان شما از Active Directory و یا Office 365 استفاده می‌کند و می‌خواهید پروژه‌تان قادر باشد تا احراز هویت کارمندان و شرکای تجاری تان را مدیریت کند، **Organizational Accounts** گزینه بهتری است.
برای اطلاعات بیشتر درباره‌ی Individual User Accounts به لینک‌های زیر مراجعه کنید:
asp.net/identity

Create an ASP.NET MVC 5 App with Facebook and Google OAuth2 and OpenID Sign-on

Web API - External Authentication Services

Adding External Logins to your ASP.NET application in Visual Studio 2013

Organizational Accounts
اگر گزینه **Organizational Accounts** را انتخاب کنید پروژه ایجاد شده برای استفاده از Windows Identity Foundation (WIF) پیکربندی خواهد شد. این فریم ورک برای احراز هویت کاربران از Windows Azure Active Directory (WAAD) استفاده می‌کند که شامل Office 365 نیز می‌شود.
Windows Authentication
اگر گزینه‌ی **Windows Authentication** را انتخاب کنید، پروژه ایجاد شده برای استفاده از Windows Authentication IIS Module پیکربندی خواهد شد. چنین اپلیکیشنی نام دامنه و نام کاربری را نمایش خواهد که یا از Active Directory می‌آید، یا از یک ماشین محلی (local machine). اما رابط کاربری ای برای ورود به سیستم وجود ندارد؛ چرا که اینگونه اپلیکیشن‌ها برای سایت‌های اینترانتی (Intranet) استفاده خواهند شد.
یک راه دیگر انتخاب گزینه On-Premises زیر شاخه Organizational Accounts است. این گزینه بجای استفاده از ماژول Windows Authentication از فریم ورک Windows Identity Foundation برای احراز هویت استفاده می‌کند. انجام چند مرحله دستی برای پیکربندی این گزینه لازم است، اما WIF امکاناتی را عرضه می‌کند که در ماژول احراز هویت ویندوز وجود ندارند. برای مثال، هنگام استفاده از WIF می‌توانید تنظیمات لازم را در Active Directory انجام دهید تا قادر به واکشی اطلاعات پوشه‌ها باشید (directory data querying).

گزینه‌های احراز هویت Organizational Accounts
دیالوگ **Configure Authentication** گزینه‌های متعددی برای احراز هویت توسط (Windows Azure Active Directory including Office 365) و Windows Server Active Directory در اختیارتان می‌گذارد:
Cloud - Single Organization

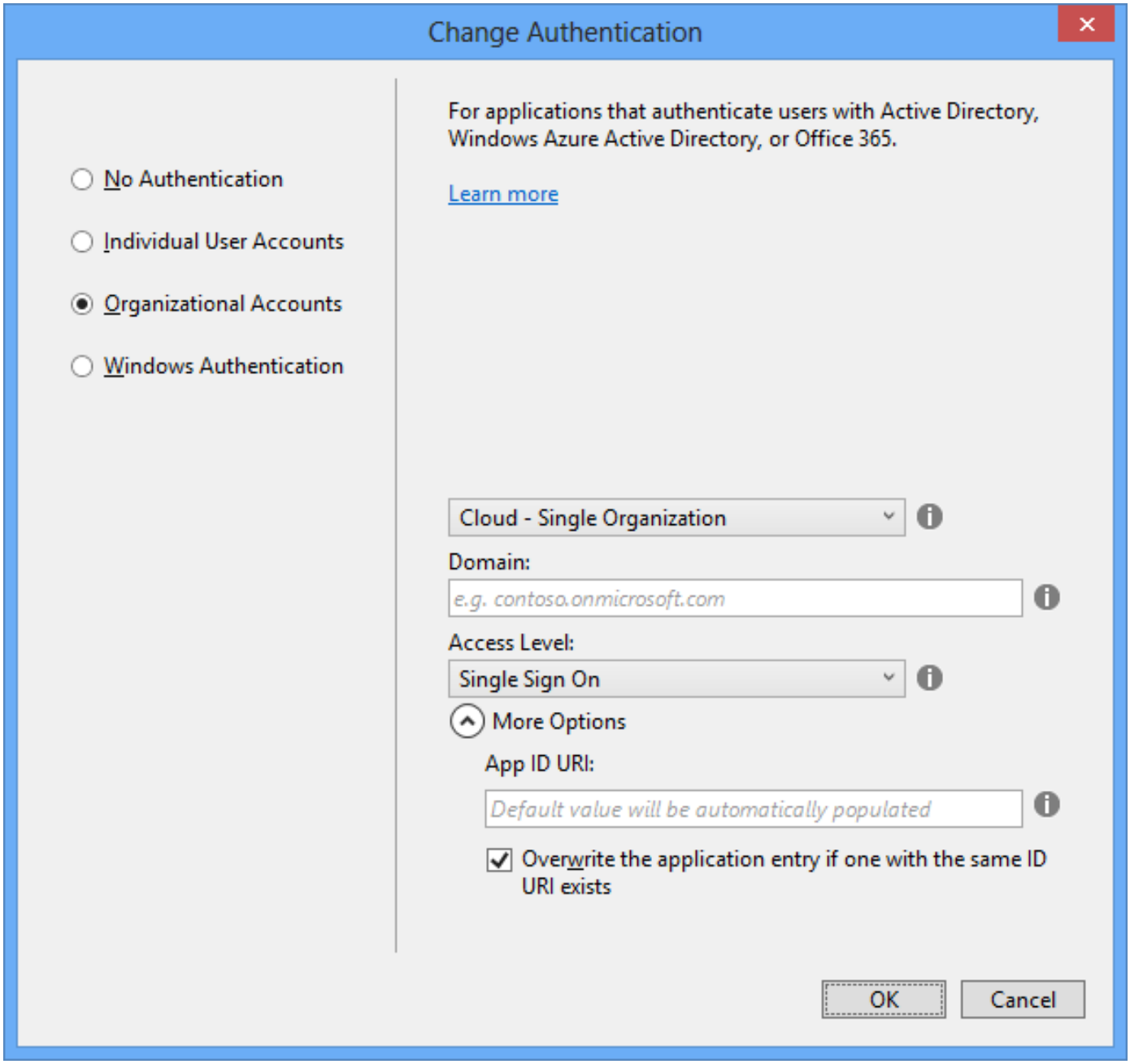
Cloud - Multi Organization

On-Premises

اگر می‌خواهید یکی از گزینه‌های WAAD را امتحان کنید اما حساب کاربری ای ندارید، روی این لینک کلیک کنید تا ثبت نام کنید.

نکته: اگر یکی از گزینه‌های WAAD را انتخاب کنید، باید اطلاعات هویتی (Credentials) یک مدیر کل را وارد کنید. برای نسخه نهایی Visual Studio 2013 برنامه هایی وجود دارد تا دیگر نیازی نباشد چنین مراحلی را تکمیل کنید. در این صورت ویژوال استودیو تنظیماتی را نمایش خواهد داد که یک مدیر می‌تواند بعدا از آنها استفاده کند تا اپلیکیشن را بصورت دستی در WAAD پیکربندی کند.

Cloud - Single Organization Authentication



از این گزینه برای احراز هویت کاربرانی استفاده کنید که در قالب یک OWIN Tenant تعریف می‌شوند. برای مثال سایتی با نام Company.com داریم که برای کارمندان این سازمان از طریق company.onmicrosoft.com قابل دسترسی خواهد بود. نمی‌توانید WAAD را طوری پیکربندی کنید که کاربران tenantهای دیگر نیز به اپلیکیشن شما دسترسی داشته باشند.

Domain

نام دامنه‌ای در WAAD که می‌خواهید اپلیکیشن را برای آن پیکربندی کنید، مثلا company.onmicrosoft.com. اگر از custom domain

ها استفاده می‌کنید مانند company.com بجای company.onmicrosoft.com می‌توانید این اطلاعات را اینجا وارد کنید.

سطح دسترسی

اگر اپلیکیشن نیاز به کوئری گرفتن یا بروز رسانی اطلاعات پوشه‌ها (directory information) را توسط Graph API دارد، از گزینه‌های **Single Sign-On, Read Directory Data** و یا **Single Sign-On, Read and Write Directory Data** استفاده کنید. در غیر اینصورت گزینه **Single Sign-On** را رها کنید. برای اطلاعات بیشتر به Application Access Levels و Using the Graph API to Query Windows Azure AD مراجعه کنید.

Application ID URI

بصورت پیش فرض، قالب پروژه یک شناسه application ID URI برای شما تولید می‌کند، که این کار با الحاق نام پروژه شما به نام دامنه WAAD صورت می‌گیرد. برای مثال، اگر نام پروژه Example باشد و نام دامنه contoso.onmicrosoft.com، شناسه خروجی https://contoso.onmicrosoft.com/Example می‌شود. اگر می‌خواهید بصورت دستی این فیلد را مقدار دهی کنید، گزینه **More Options** را انتخاب کنید. این شناسه باید با https:// شروع شود.

بصورت پیش فرض، اگر اپلیکیشنی که در WAAD تهیه و تدارک دیده شده است، شناسه‌ای یکسان با شناسه موجود در پروژه Visual Studio داشته باشد، پروژه شما به اپلیکیشن موجود در WAAD متصل خواهد شد. اگر می‌خواهید تدارکات جدیدی ببینید تیک گزینه **Overwrite the application entry if one with the same ID already exists** را حذف کنید.

اگر تیک این گزینه حذف شده باشد، و ویژوال استودیو اپلیکیشنی با شناسه‌ای یکسان را پیدا کند، عددی به آخر URI اضافه خواهد شد. مثلا فرض کنید نام پروژه Example است و اپلیکیشنی نیز با شناسه https://contoso.onmicrosoft.com/Example در WAAD وجود دارد. در این صورت اپلیکیشن جدیدی با شناسه ای مانند https://contoso.onmicrosoft.com/Example_ 20130619330903 ایجاد می‌شود.

تهیه و تدارک اپلیکیشن در WAAD

برای آنکه یک اپلیکیشن WAAD ایجاد کنید و یا پروژه را به یک اپلیکیشن موجود متصل کنید، ویژوال استودیو به اطلاعات ورود یک مدیر کل برای دامنه مورد نظر، نیاز دارد. هنگامی که در دیالوگ **Configure Authentication** روی **OK** کلیک می‌کنید، اطلاعات ورود یک مدیر کل از شما درخواست می‌شود و نهایتا هنگامیکه روی **Create Project** کلیک می‌کنید، ویژوال استودیو اپلیکیشن شما را در WAAD پیکربندی می‌کند.

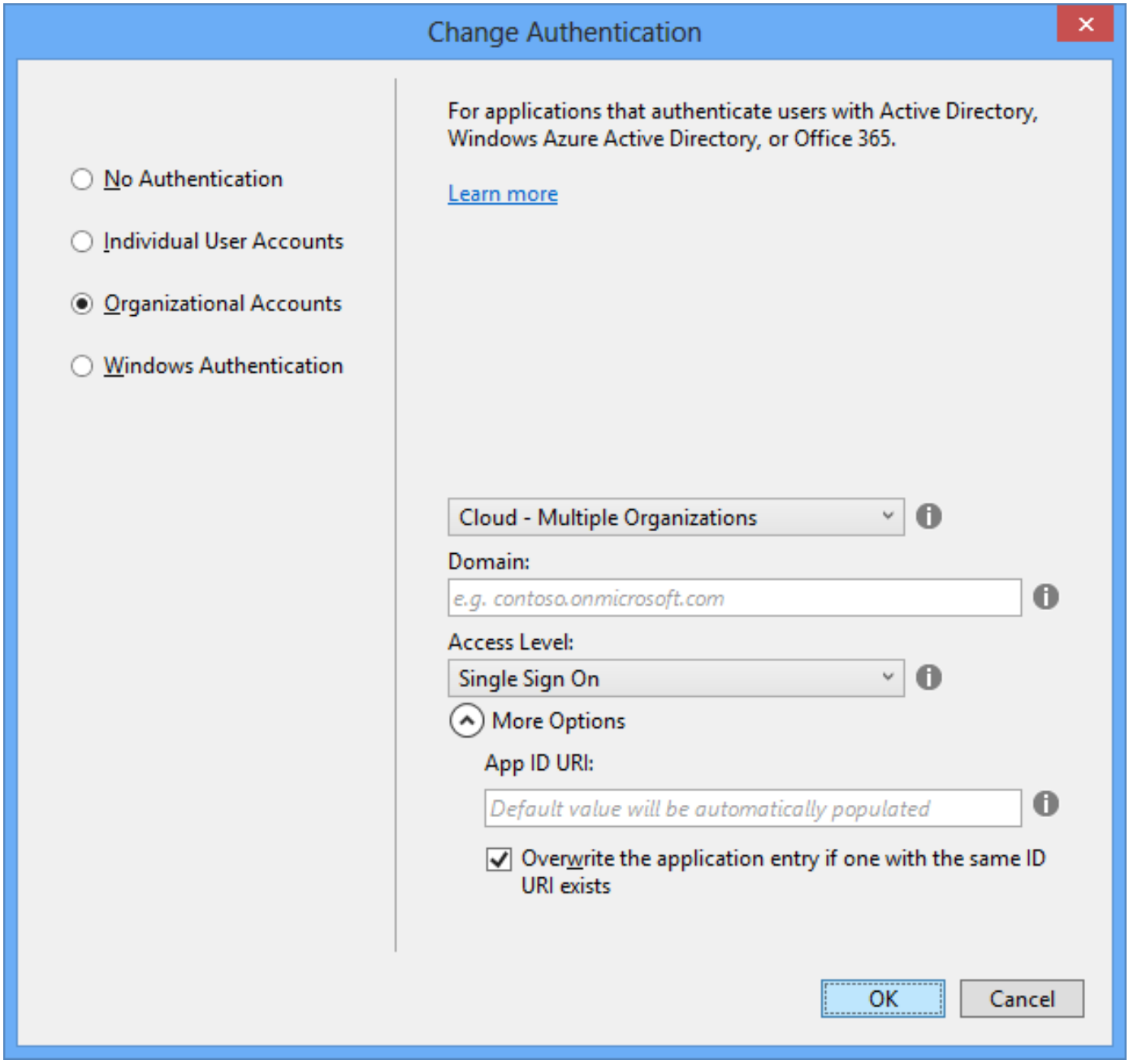
برای اطلاعات بیشتر درباره نحوه استفاده از مدل احراز هویت **Cloud - Single Organization** به لینک‌های زیر مراجعه فرمایید:
Windows Azure Authentication

Adding Sign-On to Your Web Application Using Windows Azure AD

Developing ASP.NET Apps with Windows Azure Active Directory

مقالات مذکور برای ویژوال استودیو 2013 بروز رسانی نشده اند. برخی از مراحلی که در این مقالات بصورت دستی باید انجام شوند در Visual Studio 2013 مکانیزه شده است.

Cloud - Multi Organization Authentication



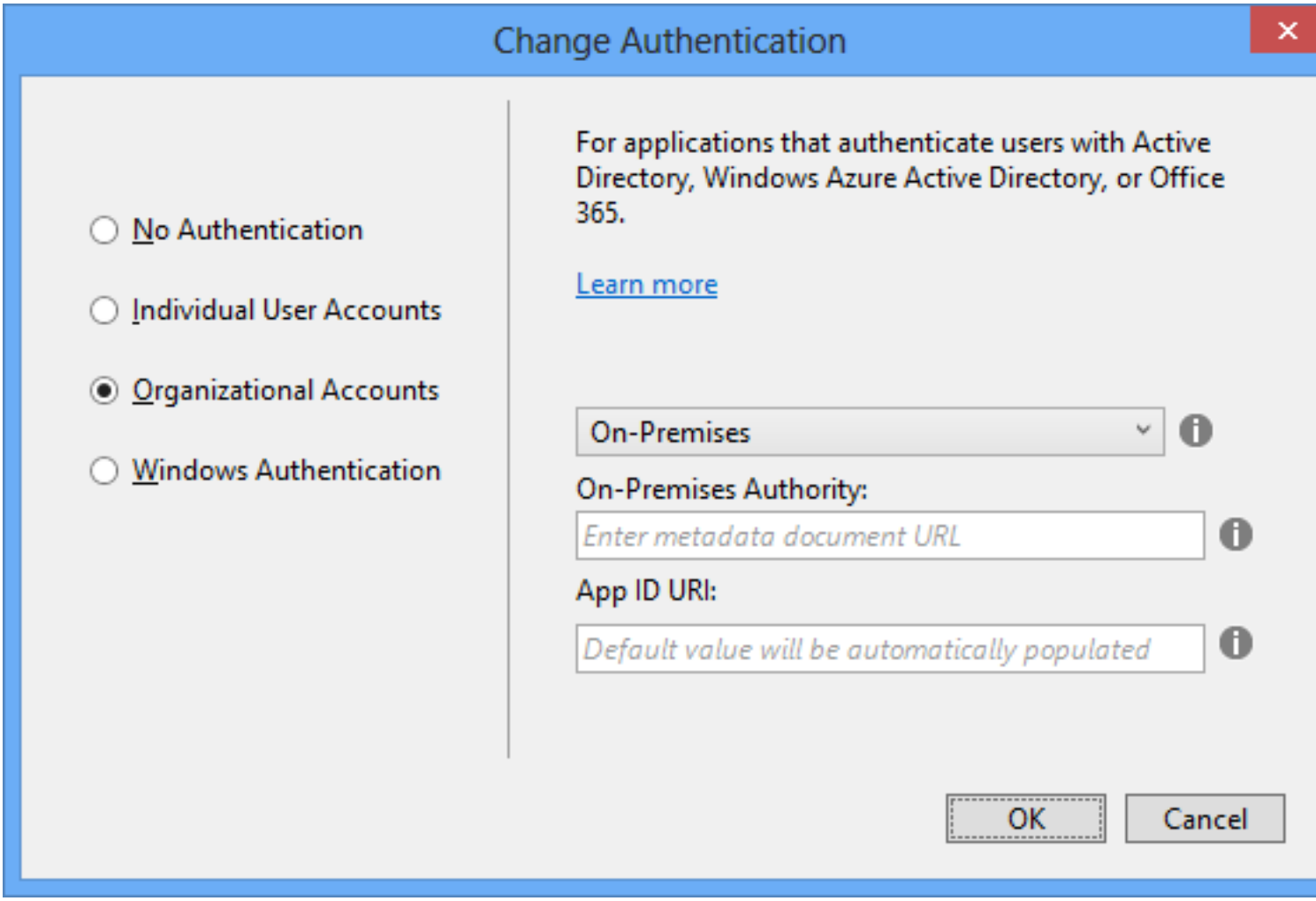
از این گزینه برای احراز هویت کاربرانی استفاده کنید که در WAADهای متعددی تعریف شده‌اند. برای مثال، نام سایت contoso.com است و برای کارمندان دو سازمان از طریق آدرس‌های contoso.onmicrosoft.com و fabrikam.onmicrosoft.com قابل دسترسی خواهد بود. نحوه پیکربندی این مدل نیز مانند قسمت قبلی است.

برای اطلاعات بیشتر درباره احراز هویت **Cloud - Multi Organization** به لینک‌های زیر مراجعه کنید:

Easy Web App Integration with Windows Azure Active Directory, ASP.NET & Visual Studio

Developing Multi-Tenant Web Applications with Windows Azure AD

On-Premises Organizational Accounts



این گزینه را هنگامی انتخاب کنید که کاربران در Windows Server Active Directory (AD) تعریف شده اند و نمی‌خواهید از WAAD استفاده کنید. از این مدل برای ایجاد وب سایت‌های اینترنت و اینترانت می‌توانید استفاده کنید. برای یک وب سایت اینترنتی از Active Directory Federation Services (ADFS) استفاده کنید.

برای یک وب سایت اینترانتی، می‌توانید کلا این گزینه را رها کنید و از Windows Authentication استفاده کنید. در صورت استفاده از گزینه Windows Authentication لازم نیست تا آدرس سند متادیتا (metadata document URL) را فراهم کنید، همچنین توجه داشته باشید که Windows Authentication امکان کوئری گرفتن از پوشه‌ها و کنترل سطوح دسترسی در Active Directory را ندارد.

On-Premises Authority

آدرس سند متادیتا. این سند اطلاعاتی درباره مختصات Authority دارد که اپلیکیشن از آنها برای به پیش بردن روند احراز هویت و ورود به سایت استفاده می‌کند.

Application ID URI

یک شناسه منحصر به فرد که AD از آن برای شناسایی اپلیکیشن استفاده می‌کند. می‌توانید این فیلد را خالی رها کنید تا ویژوال استودیو بصورت خودکار اپلیکیشنی بدین منظور بسازد.

قدم‌های بعدی

در این مقاله با مدل‌های مختلف احراز هویت در اپلیکیشن‌های Visual Studio 2013 آشنا شدید و برخی تغییرات و امکانات جدید نیز بررسی شدند. برای اطلاعات تکمیلی به ASP.NET and Web Tools for Visual Studio 2013 Release Notes مراجعه کنید.

نظرات خوانندگان

نویسنده: هومن
تاریخ: ۱۳۹۳/۰۳/۱۹ ۱۹:۵۴

سلام
- بابت مطلب خوبتون ممنون.من یه سوال داشتم ، در پورتال‌های سازمانی و در حالتی که نیاز نداریم پالیسی رو از اکتیو بخونیم ، بهتره از حالت آخر یعنی Windows Authentication استفاده کنیم درسته؟
من از این حالت استفاده کردم، و یک مشکل و یک سوال برام پیش اومده...
- مشکلم اینه که در سیستمی که با یوزر اکتیو بالا اومده خوب کار میکنه و یوزر رو تشخیص میده، اما روی لپتاپم چون لوکال هست و طبیعیه اکتیو دایرکتوری موجود نیست نمیتونه یوزر رو تشخیص بده(پنجره لاگین مرورگر را باز میکند) و من نمیتونم برناممو گسترش بدم و برای تست نیاز به اجرای برنامه دارم ، این مشکلو چجوری برطرف کنم؟
- سوالم اینه که ما هیچ کجا اسم دامین یا آی پی سرور رو به برنامه نمیدیم ، پس سیستم احراز هویت چجوری دامین رو تشخیص میده و باهاش کار میکنه ؟

نویسنده: محسن خان
تاریخ: ۱۳۹۳/۰۳/۲۰ ۹:۱۸

اگر از IE استفاده کنید، مشکلی نباید باشه. چون IE با سیستم اعتبارسنجی مبتنی بر ویندوز یکپارچه هست. اگر با IE صفحه لاگین مرورگر باز میشه، به تنظیمات امنیتی اون مراجعه کنید و سایت رو در قسمت trusted sites اضافه کنید:
http://support.microsoft.com/kb/258063

سمت سرور هم باید در تنظیمات IIS، گزینه‌ی اعتبارسنجی مبتنی بر ویندوز فعال باشه:

http://www.asp.net/mvc/tutorials/older-versions/security/authenticating-users-with-windows-authentication-cs