**Boston Code Camp 22 Presentation**

 **Secure ASP.NET MVC5 Application with Asp.Net Identity**

 **Chagnde Wu**

**Outline**

1. **Basics of web security**

Purpose: Protect sensitive data

Strategies:

 Authentication -> Job for web dev

Authorization -> Job for web dev

 Safe Data transportation -> Job for both web dev and web server

 Prevent various attacks

 encrypt data even someone breaks in

1. **About Asp.net Identity**

Asp.net membership

 Simple membership

 Universal membership

 Identity

 Identity 2.0

1. **Get Started**
2. Visual Studio 2013 → Update 4 (MVC 5.2.2, Identity 2.1, Microsoft.Owin\* 3.0 )

 if Update 3, update the packages manually after create project

1. Create a MVC project with individual account - AspIdentityWeb1
2. Configure the project using

 The security work should as close to real deployment as possible, so

 use local IIS - project url http://www.private-yuandi.com

 use SQL server - run script to setting DB connection string

 configure Https binding

1. Build the project, make sure it runs.
2. **Get Around the new project**

 Project structure -- out-of-the-box controllers and views for account

 Try to login, DB tables are created even without create new user

 User can register and login

 try to create a user

 login in the user

 show authorize works

 [Show how authorize is used to access protected resource] –

 run script to populate the private area

1. **Use external Logins**

Enable external logins (Startup.Auth.cs)

 Register App on providers

Built-in support: Google, Facebook, Twitter, Microsoft

 Extended support: Yahoo, LinkedIn, Salesforce etc.

 [run script to set up external logins]

 About Yahoo: have to own a real web site to work, domain must be confirmed.

 Otherwise, get weird error.

 Use this web page as reference:

 [Add this dll from Nuget manager: Owin.Security.Providers]

 using Owin.Security.Providers.Yahoo

|  |  |  |
| --- | --- | --- |
| OAuth Providers | require https | call back url  |
| Googlehttps://console.developers.google.com/project | no | <your site url>/signin-google |
| Facebook <https://developers.facebook.com/apps> | no | <your site url> |
| Microsofthttps://account.live.com/developers/applications | nobut FQDNsupport Logo | <your site url>/signin-microsoft |
| Twitter | no | <your site url> |
| Yahoohttps://developer.yahoo.com/ | no, true domain | <your site url> |

 Local user can associate external logins; external logins can also create an associated local login.

 One step further:

 What about the logos for external logins?

 Recipe:

 Pretty social login buttons for ASP.NET MVC 5

 By Jerrie Pelser

 <http://www.beabigrockstar.com/blog/pretty-social-login-buttons-for-asp-net-mvc-5>

 Zosical: <http://zocial.smcllns.com/>

 Download Zocial from https://github.com/samcollins/css-social-buttons/
 Copy and paste the css and font file to your project content folder

 Register css with your bundle

 Customize \_ExternalLoginsListPartial.cshtml

 [run script Step2b\_zosial.bat]

 Cavet: Microsoft is left out

 Work around: edit the zocial.css, duplicate entries for windows, and rename as microsoft

 [run script Step2c\_zosial\_more.bat]

1. Verify user before allow login

 Asp.ent Identity is capable of verify user by email or smart phone before allow the registered user to login. This takes email as example.

 Step one: Setup your email service in IdentityConfig.cs

 Step two: Modify the AccountController.Regoister(model)

 Comment out the signin statement

 Uncomment out the Email confirmation section

 You can then add a message to ViewBag, and direct user to “Info” view

 Step three: Create a generic view “Info” that can show some quick display

 Step four: User follow the link from the email message to your view “ConfirmEmail”

 then sign from there.

 Step four: Disallow user login if the email is not verified

 Changes in Login action of AccountController

 Since you do the same email message 2 places (Register, Login), it makes sense to create a helper method such as SendEmailConfirmationTokenAsync to do so.

1. Two factor authentication with Email and SMS

 Reuirement: Setup and configure Email and SMS service routine ( IdentityConfig.cs)

 Comment out the add phone statements let user ass phone ( Manage/Index.cshtml)

 No other code is required for next steps

 User logged in, click the log-in, will goes to Manage view

 Register your phone, and get verified

 Enable the 2FA

 then logout, login

 You will see the requirement for 2FA.

 You can your second 2F in your browser so next time you will not be bothered.

1. Password recovery, Account lockout

 Password recovery is supported out-of-the-box by Asp.Net Identity

 Step 1: Uncomment out Account/Login.cshtml ForgotPassword action link

 Step 2: Go AccountController ForgotPassword action, uncomment out the code

 Step 3: You can configure the options in dataProtectionProvider IdentityConfig.cs

 Account lockout

 IdentityConfig.cs Create() -- UserLockoutEnabledByDefault is for 2-FA only
 SignInManager.PasswordSignInAsync -- shouldLockout: true for user

1. **Customize User**

 You can customize the application user, for example to display friendly name instead of Email

 Step 1: Add a property to ApplicationUser in IdentityModels.cs

 Step 2: Update the view model for user to enter DisplayName when register

 in (AccountViewModels.cs RegisterViewModel)

 Step 3: Update Register.cshtml to let user enter DisplayName

 Step 4: Display to friendly name for logged in user

 Update \_LoginPartial.cshtml to take advantage of display name

 Use extension method to do so (Facility/IdentityExtensions.cs)

 You can do the same for External user login

 You need to update the DB schema

 One way is using EF Migration, use the following command

 Enable-Migrations -ContextTypeName AspIdentityWeb1.Models.ApplicationDbContext

 Add-Migration AddUserDisplayName

 Update-Database

1. **Work with users and role administration**

 Code intensive

 The sample project has this feature, but removed in release version

 Think about using Thinktecture tools

 Do your own

 Step 1: Create ApplicationRoleManager just like ApplicationUserManager

 Step 2: Create ApplicationRoleDbContext in IdentityModels.cs

 Step 3: Register ApplicationRoleManager and ApplicationRoleDbContext for Owin Context

 (in Startup.Auth.cs)

 Step 4: Add Controller and View for creating Role, manager users

Resources

**Introduction to Asp.net Identity**

**Create a secure ASP.NET MVC 5 web app with log in, email confirmation and password reset (C#)**

By Rick Anderson

<http://www.asp.net/mvc/overview/security/create-an-aspnet-mvc-5-web-app-with-email-confirmation-and-password-reset>

**Introduction to Asp.net Identity**

By [Pranav Rastogi](http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity#author-37788), [Rick Anderson](http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity#author-20106), [Tom Dykstra](http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity#author-36250) and [Jon Galloway](http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity#author-5506)

<http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity>

**Account Management (External Logins, 2-FA, SMS)**

**Asp.net OAuth external login developer help center**

By Jerrie Pelser

<http://www.beabigrockstar.com/guides/aspnet-oauth>

**Pretty social login buttons for ASP.NET MVC 5**

By Jerrie Pelser

<http://www.beabigrockstar.com/blog/pretty-social-login-buttons-for-asp-net-mvc-5>

**Zosical:**

<http://zocial.smcllns.com/>

**Account Confirmation and Password Recovery with ASP.NET Identity (C#)**

By Rick Anderson, Pranav Rastogi, Suhas Joshi and Hao Kung

<http://www.asp.net/identity/overview/features-api/account-confirmation-and-password-recovery-with-aspnet-identity>

**ASP.NET MVC 5 app with SMS and email Two-Factor Authentication**

By Rick Anderson

<http://www.asp.net/mvc/overview/security/aspnet-mvc-5-app-with-sms-and-email-two-factor-authentication>

**Code! MVC 5 App with Facebook, Twitter, LinkedIn and Google OAuth2 Sign-on (C#)**

By Rick Anderson

<http://www.asp.net/mvc/overview/security/create-an-aspnet-mvc-5-app-with-facebook-and-google-oauth2-and-openid-sign-on>

**External Authentication Services with ASP.NET Web API (C#)**

by Robert McMurray

http://www.asp.net/web-api/overview/security/external-authentication-services

**Two-factor authentication using SMS and email with ASP.NET Identity**

By Rick Anderson, Pranav Rastogi, Suhas Joshi and Hao Kung

<http://www.asp.net/identity/overview/features-api/two-factor-authentication-using-sms-and-email-with-aspnet-identity>

Yahoo Login

<http://www.beabigrockstar.com/guides/aspnet-oauth/aspnet-mvc-yahoo-login>

**Create Roles, Customize User and Manager Users**

**Working with Roles in ASP.NET Identity for MVC**

Sheo Narayan

<http://www.dotnetfunda.com/articles/show/2898/working-with-roles-in-aspnet-identity-for-mvc>

**ASP.NET Identity 2.0: Customizing Users and Roles**

John Atten

<http://typecastexception.com/post/2014/06/22/ASPNET-Identity-20-Customizing-Users-and-Roles.aspx>

**Overview of Custom Storage Providers for ASP.NET Identity**

Tom FitzMacken

<http://www.asp.net/identity/overview/extensibility/overview-of-custom-storage-providers-for-aspnet-identity>

**Implementing a Custom MySQL ASP.NET Identity Storage Provider**

By Raquel Soares De Almeida, Suhas Joshi and Tom FitzMacken

<http://www.asp.net/identity/overview/extensibility/implementing-a-custom-mysql-aspnet-identity-storage-provider>