WorkshopPLUS – Security: Modern Authentication and Authorization

WorkshopPLUS

Delivery Options: 3 days [Remote/Onsite]
Difficulty Level: 300 - Advanced

Description

Building applications that operate in the internet environment requires understanding the options available for performing authentication and authorization. These options include, both a variety of protocols such as OAuth2 and WS-Federation, and tools and toolkits, such as Azure AD, ASP.NET identity handling and MSAL.

This offering is designed to provide you with the knowledge you need to create applications that require cloud-appropriate authentication and authorization technology and to help you understand the new approach based on standard protocols such as OAuth2, OpenID Connect, JWT and SAML. It covers both common architectural patterns, industry standard protocols and tools used to implement them. The tools and infrastructure aspects of the course are focused on Microsoft technology.

Objectives

- Understand how access control, authentication and authorization changes when applications and/or users use the internet.
- Learn how to use Microsoft infrastructure, Azure AD (corporate or B2C), and development tools to secure your applications using industry protocols such as OAuth2 or SAML.

Outcomes

- Ability to use AAD and other Azure services to authenticate users and authorize API applications

Methodology

Learn by doing

You will work directly with a Microsoft engineer to understand modern authentication protocols, how they are supported by Microsoft cloud infrastructure and how to use them in applications.

Hands-on labs

- Using OpenIdConnect/OAuth2
- Registering applications in AAD
- Using MS Graph
- Using Azure B2C
- Using Web App EasyAuth
- Authentication/authorization in Web Apps and APIs

Scope

Modern authentication protocols, their use in Azure cloud and application code.

Agenda

Day 1

Architecture: Architectural patterns, practices, and protocols used for handling authentication and authorization in zero-trust environments

Day 2

Infrastructure: Use Azure AD and Azure B2C as token issuers; Use Azure Services (App Service, APIM and AAD Proxy) to secure web applications.

Day 3

Coding: Develop applications using modern authentication protocols, using toolkits like MSAL and Microsoft.Identity.Web package

This data sheet is for informational purposes only.
MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY

Microsoft

Delivery Outline

Requirements

Participants Application architects and developers **Skill requirements** ▪ Experience with Visual Studio ▪ Basic knowledge of C# to understand the source code shown on demos and to complete the labs **Time commitment** ▪ Three full-day engagement with relevant roles	**Delivery requirements** The workshop provides a lab environment accessible through a browser. It also includes a free Azure subscription. Attendees may choose to use own workstations with: ▪ Windows 10 or later ▪ Visual Studio 2022 (Free Community edition or higher) ▪ Fiddler with https decryption enabled (optional) ▪ .NET Core 7.x SDK ▪ Internet access with at least 1 Mbps bandwidth per student

Knowledge Transfer

Day 1	Architecture, OAuth2 and OpenID Connect	▪ Overview of authentication and authorization issues in internet-based applications, purpose of various protocols ▪ Details of protocols; reviews the various flows defined by OAuth2, how they apply to common application topologies, their security threat models
Day 2	Azure AD, Microsoft Graph, Azure AD PowerShell, Azure AD B2C, Web App Service "Easy Auth", APIM, AppProxy	▪ Purpose and features of the Azure AD ▪ Introduction to programmatic access to Azure AD and other MS cloud services ▪ Introduction to the Azure AD tenant type specifically designed for consumer and citizen identities, support for social and custom identities ▪ Azure services providing no-code authentication/authorization support ('Easy Auth', APIM and AAD Proxy)
Day 3	Developing Applications, ASP .NET identify handing, MSAL toolkits	▪ Hands-on implementation of a set of related applications using OAuth2 protocols, Microsoft Graph APII and various other features of Azure AD ▪ Review of toolkits used to initiate passive protocols in web applications and handle (validate/augment) received security tokens ▪ Review of APIs used to obtain OAuth2 and OIDC tokens from Azure AD, ADFS and/or federated providers (Gmail, FB, etc.)

For more information: Please contact your Microsoft Representative for more details.

This data sheet is for informational purposes only.
MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY

Microsoft