



Azure Study Group

AZ-301 - Microsoft Azure Architect Design

Jeff Wagner
Partner Technology Strategist



Design for Identity and Security (20-25%)

Agenda

1

Agenda

2

Speaker
Introduction

3

Feedback
Loop

4

Objective
Review

5

Open Mic

Series Agenda

1	Determine Workload Requirements (10-15%)
2	Design for Identity and Security (20-25%)
3	Design a Data Platform Solution (15-20%)
4	Design a Business Continuity Strategy (15-20%)
5	Design for Deployment, Migration, and Integration (10-15%)
6	Design an Infrastructure Strategy (15-20%)

<https://aka.ms/azurecsg>

Series Agenda

1	Determine Workload Requirements (10-15%)
2	Design for Identity and Security (20-25%)
3	Design a Data Platform Solution (15-20%)
4	Design a Business Continuity Strategy (15-20%)
5	Design for Deployment, Migration, and Integration (10-15%)
6	Design an Infrastructure Strategy (15-20%)

<https://aka.ms/azurecsg>

Speaker Introduction - Jeff Wagner

- Partner Technology Strategist based in Atlanta
- 21+ years with Microsoft, more in the industry
- Constant learner - *Ancora Imparo*
- Working on the same certifications that you are



Feedback Loop

Objectives

Design Identity Management

May include but not limited to: Choose an identity management approach; design an identity delegation strategy, identity repository (including directory, application, systems, etc.); design self-service identity management and user and persona provisioning; define personas and roles; recommend appropriate access control strategy (e.g., attribute-based, discretionary access, history-based, identity-based, mandatory, organization-based, role-based, rule-based, responsibility-based)

Design Authentication

May include but not limited to: Choose an authentication approach; design a single-sign on approach; logon, multi-factor, network access, and remote authentication

Design Authorization

May include but not limited to: Choose an authorization approach; define access permissions and privileges; design secure delegated access (e.g., OAuth, OpenID, etc.); recommend when and how to use API Keys.

Objectives (cont.)

Design for Risk Prevention for Identity

May include but not limited to: Design a risk assessment strategy (e.g., access reviews, RBAC policies, physical access); evaluate agreements involving services or products from vendors and contractors; update solution design to address and mitigate changes to existing security policies, standards, guidelines and procedures

Design a Monitoring Strategy for Identity and Security

May include but not limited to: Design for alert notifications; design an alert and metrics strategy; recommend authentication monitors

Design Identity Management



Platform Security

Azure offers a “shared responsibility” security model:

Microsoft Azure is built with end-to-end security in mind. Microsoft delivers a secure foundation and tooling to control your environment.

Customers own responsibility of their subscription governance, data, identities, and protection. The level of responsibility varies between IaaS and PaaS-based solutions.

Azure is architected for multi-tenancy, with separation based on:

The Azure Fabric Controller (FC) functioning as the kernel of the Azure platform

The host OS running a hardened installation of Windows Server

The hypervisor implemented as the Hyper-V server role

The guest VM OS, which can be Windows Server or an approved Linux distribution

Storage isolation: hypervisor maps storage blocks to separate storage accounts.

Network isolation: hypervisor switch blocks inter-tenant communication

Securing the Azure Platform

Azure Key Vault is a managed FIPS 140-2 Level 2 HSM-based service:

Safeguards cryptographic keys, secrets, and certificates

Facilitates RBAC and policy-based delegated administration and usage, including:

Creating and importing keys and secrets.

Revoking and deleting keys and secrets.

Authorizing users or applications to access the key vault.

Configuring key usage (e.g. for signing or encryption).

Monitoring key usage.

Azure Active Directory

Microsoft's cloud-based directory and identity management service:
Combines directory services, identity governance, and application access management
Facilitates a wide range of identity and access management features, including:

Single sign-on for access to SaaS applications

Multi-factor authentication

Security monitoring

Device registration

Reach auditing

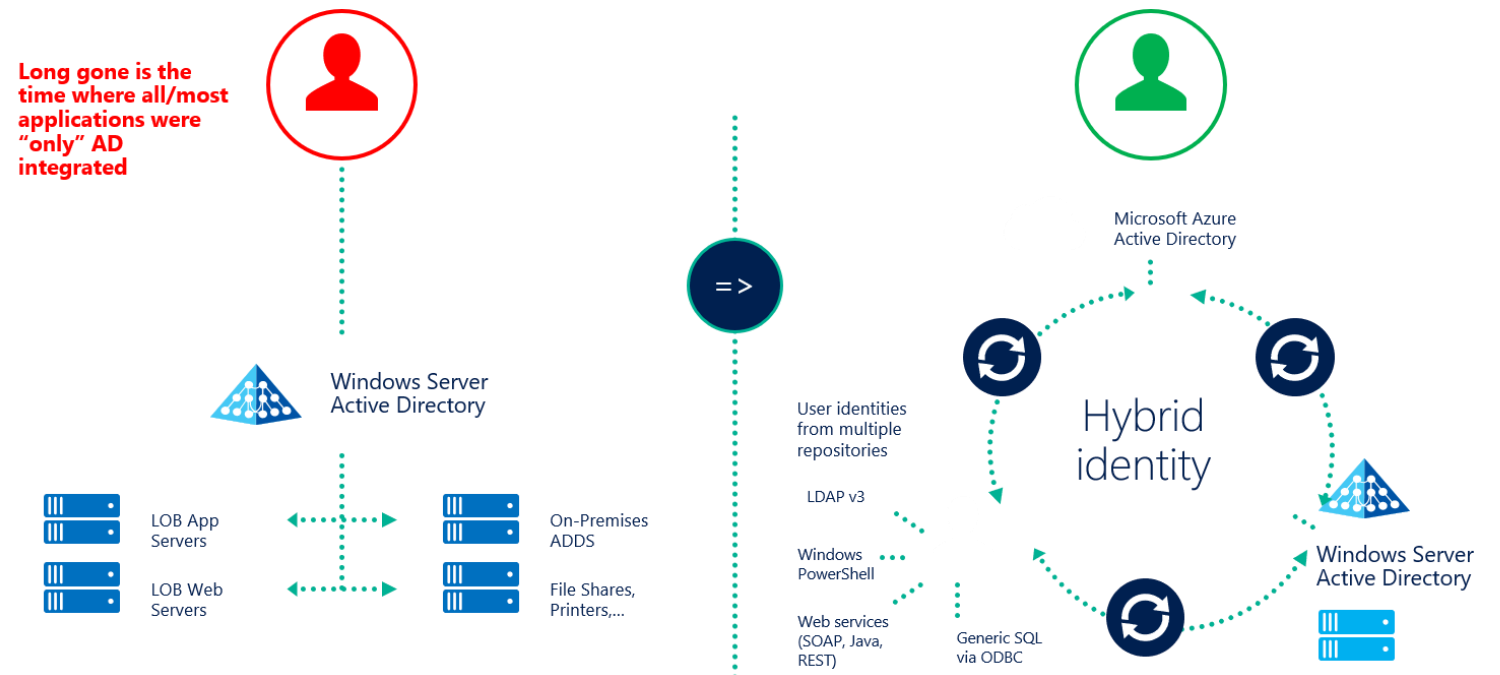
Alerting

Integrates with Active Directory:

SSO with Pass-through Authentication

SSO with federation (AD FS)

Password Hash Sync



Azure AD Authentication Strategies

Majority of hybrid scenarios integrate AD with Azure AD:

Integration relies on sync between AD and Azure AD by using Azure AD Connect

Azure AD Connect is a Microsoft-developed synchronization engine:

Supports a wide range of synchronization topologies, including multiple AD forests

Can be installed on any Windows Server with direct connectivity to an AD domain controller

Uses SQL Server Express included in the installation binaries or a separate SQL Server instance

Supports two-way sync, allowing changing and resetting AD user passwords by using Azure AD password management features

Azure AD B2B & B2C

Azure AD B2B:

Allows granting partner organizations access to local resources and applications

Allows adding partner organization users to local Azure AD groups

Supports partner organization user credentials

Is managed by the host organization:

Partner lifecycle

Security policy and compliance

Branding

Azure AD B2C:

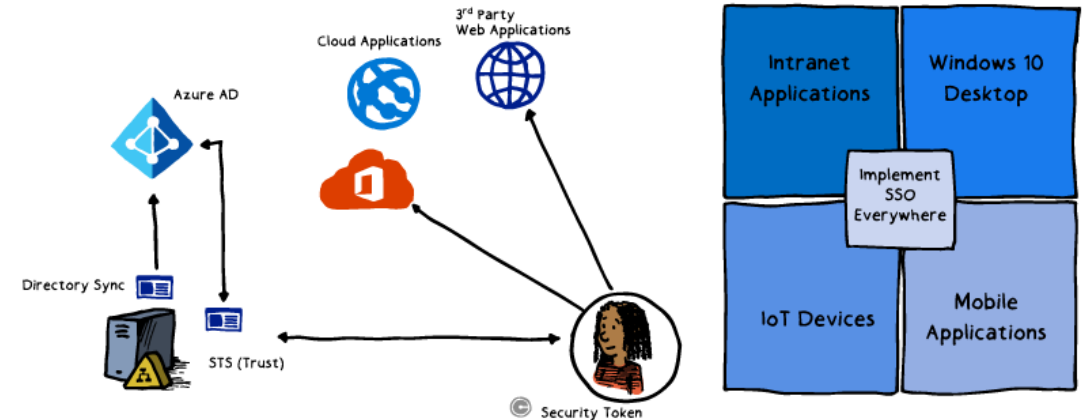
Allows granting individual customers access to local resources and applications

Supports social identities via federation but allows creating new Azure AD accounts

Requires a separate Azure AD tenant (separately from the host organization's tenant)

Provides SSO to customer's applications

Is managed by the host organization with app specific customization



Azure AD Identity Protection

Azure AD Identity Protection:

Protects all identities regardless of their privilege level.

Detects threats based on adaptive machine learning and heuristics.

Generates reports and alerts to accelerate response and remediation.

Privileged Identity Management:

Identifies users with administrative roles in Azure AD.

Identifies users with privileged roles to manage Azure resources (Preview)

Enables on-demand, "just in time" administrative access to Microsoft Online Services like Office 365 and Intune, and to Azure resources (Preview).

Tracks history of privilege escalation, including changes to Azure resources (Preview).

Generates alerts in response to changes in administrator assignments.

Implements approval-based elevation to Azure AD privileged admin roles (Preview).

Generates reports identifying members of privileged roles.

Azure AD Domain Services

Provides managed AD Domain Services in Azure:

Supports domain join, group policies, LDAP, Kerberos/NTLM authentication

Implements two managed domain controllers in a VNET of an Azure subscription

Its identities are automatically replicated from an Azure AD tenant:

Uses Azure AD tenant associated with the Azure subscription hosting the VNET with two domain controllers.

If Azure AD is integrated with on-premises AD, users can sign in to Azure AD DS with their AD credentials.

In lift-and-shift scenarios, it eliminates the requirement for:

Provisioning hybrid connectivity between on-premises environments and Azure VNETs

Extending on-premises AD infrastructure to Azure by deploying replica domain controllers into Azure VNETs

Design Authentication



Implementing Authentication in Applications



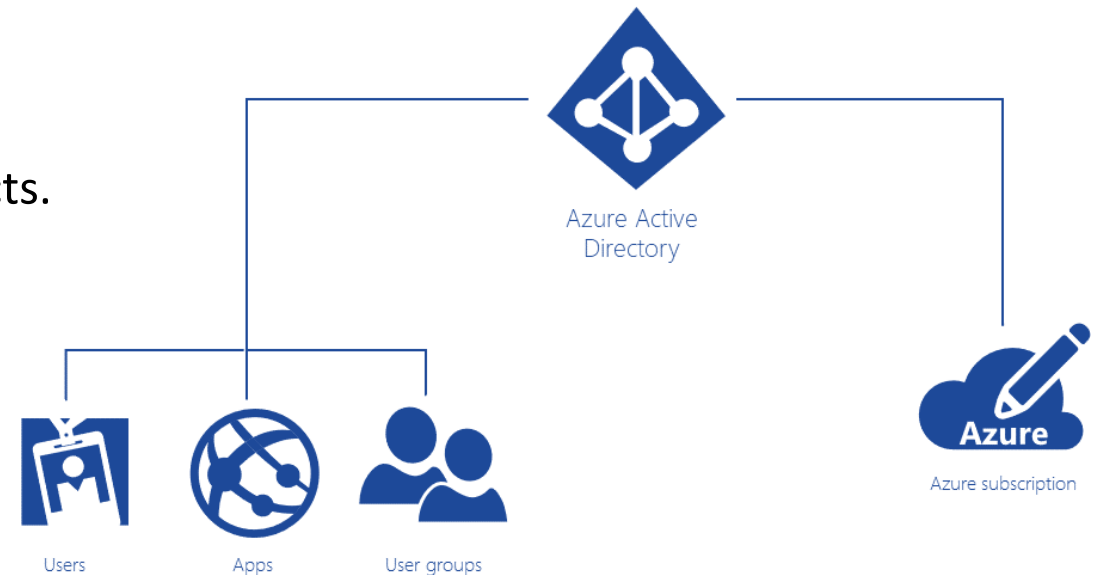
Certificate-based authentication

- Establishes identity by using a digital certificate:
 - Front-end applications interact with back-end services
 - Securing connections in hybrid scenarios over Transport Layer Security (TLS)
 - API Management protecting access to the back-end service



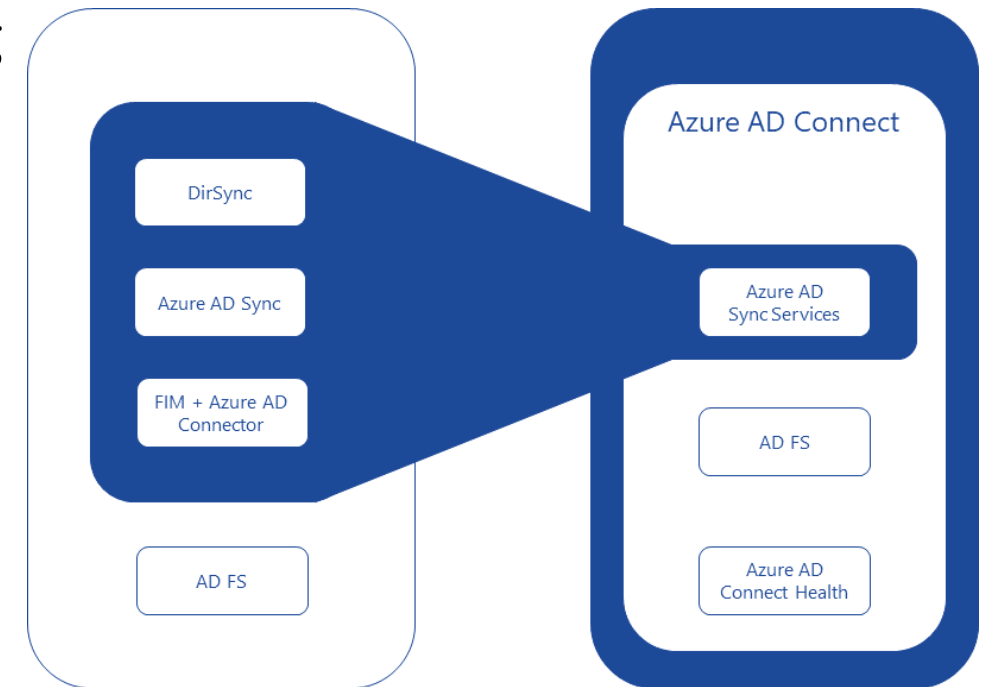
Azure Active Directory (Azure AD)

- Microsoft's cloud-based directory and identity management service:
 - Combines directory services, identity governance, and application access management
- AD DS vs. Azure AD:
 - Similarities:
 - Directory stores of user, group, and application objects.
 - Identity and authentication providers.
 - Differences
 - Single tenant vs. multi-tenant
 - A server role in Windows Server vs. cloud service
 - X.500-based hierarchical structure vs. flat structure
 - LDAP lookups vs. Graph API
 - Kerberos and NTLM vs. SAML, WS-Federation, and OpenID Connect/OAuth
 - Built-in GPO-based management capabilities vs. integration with management products such as Intune
 - Domains and forests trusts vs. federation



Azure AD Connect

- Integrates AD DS with Azure AD:
 - Implements a common identity for your users across Azure, Office 365, and SaaS apps
- Provides support for:
 - **Sync Services:** synchronize AD DS objects, such as users and groups.
 - **Health Monitoring:** offers centralized monitoring
 - **Federation:** simplifies configuration of AD FS



Legacy authentication methods

- **Forms-based authentication:**

- Requires use of a browser client
- Requires extra measures to prevent cross-site request forgery
- Sends user credentials in plaintext

- **Windows-based authentication:**

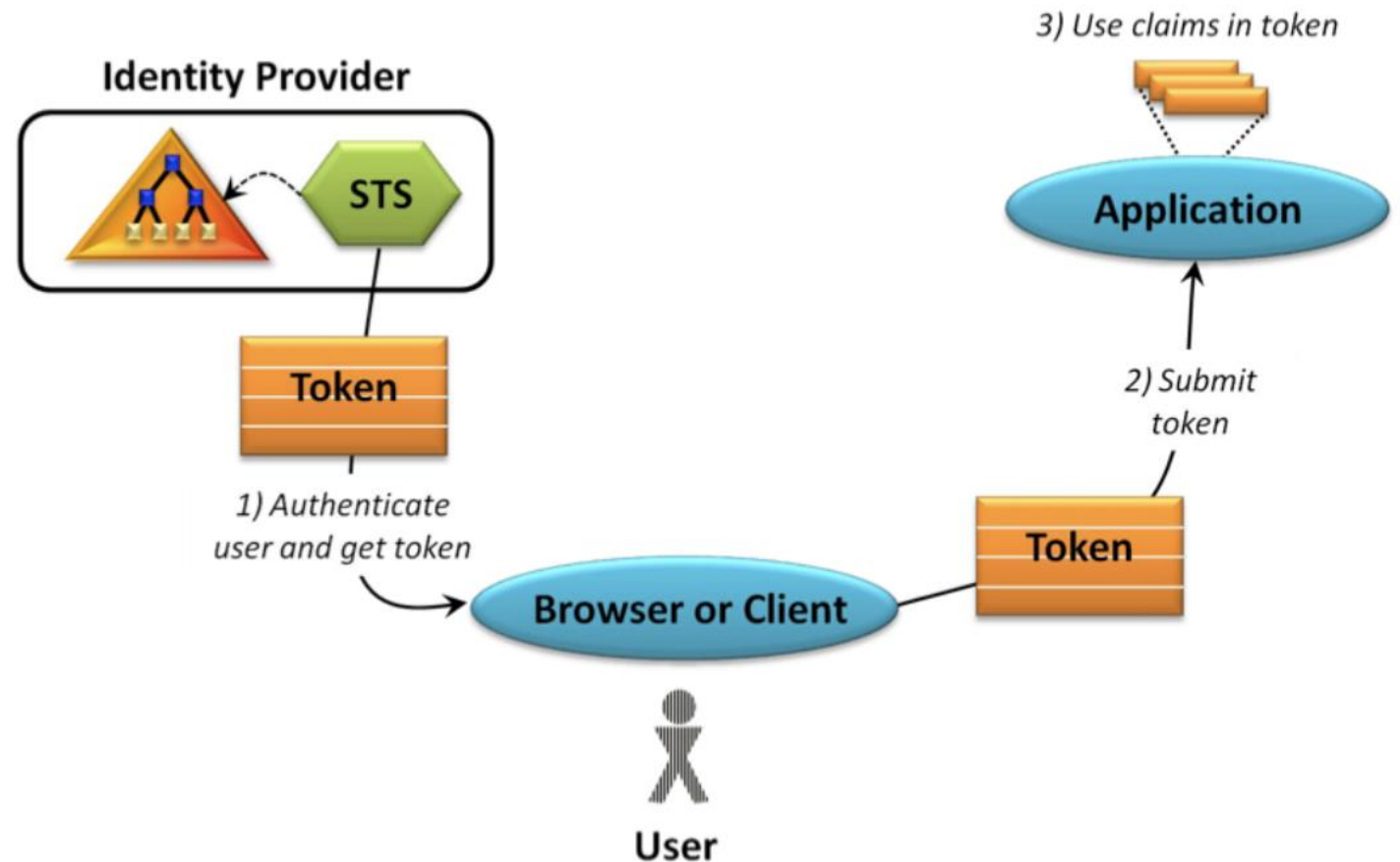
- Not suitable for internet applications
- Does not support BYOD scenarios
- Relies on Kerberos or NTLM
- Requires domain-joined computers

Example:

If you are moving an ASP.NET forms-based auth application to Azure, change connection string pointing to your on-premise SQL Server database used to store forms auth data to Azure SQL Server.

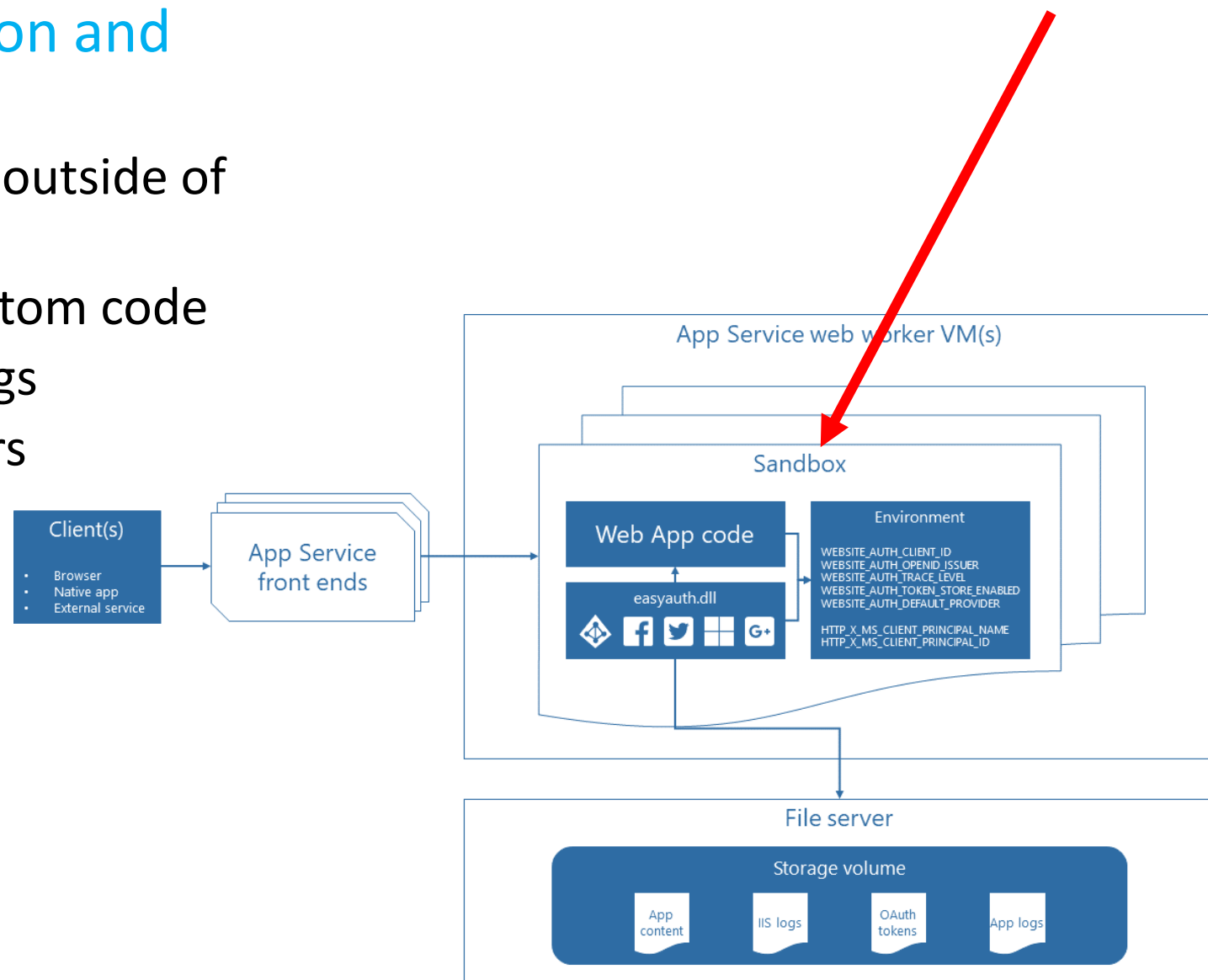
Token-based authentication

- Claims-based authentication in .NET:
 - **ASP.NET Identity** provides a unified identity platform for ASP.NET applications
 - Can be used with web, phone, store and hybrid applications
 - Ideal for token-based auth because:
 - Provider model for logins
 - Supports claims-based authentication



Token-based authentication

- Azure App Service authentication and authorization:
 - Runs in the worker sandbox and outside of the web app code
 - Available with minimal or no custom code
 - Configurable by using app settings
 - Injects claim into request headers
 - Provides built-in token store

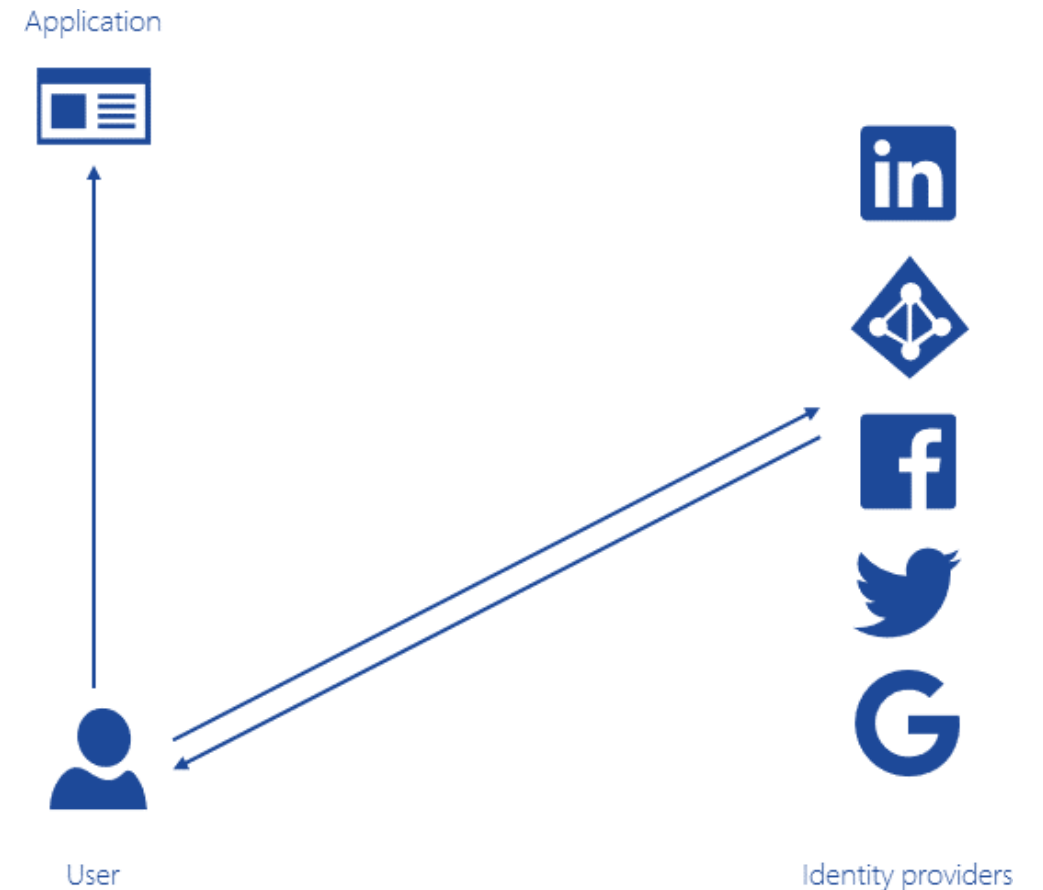


Claims-based Authorization



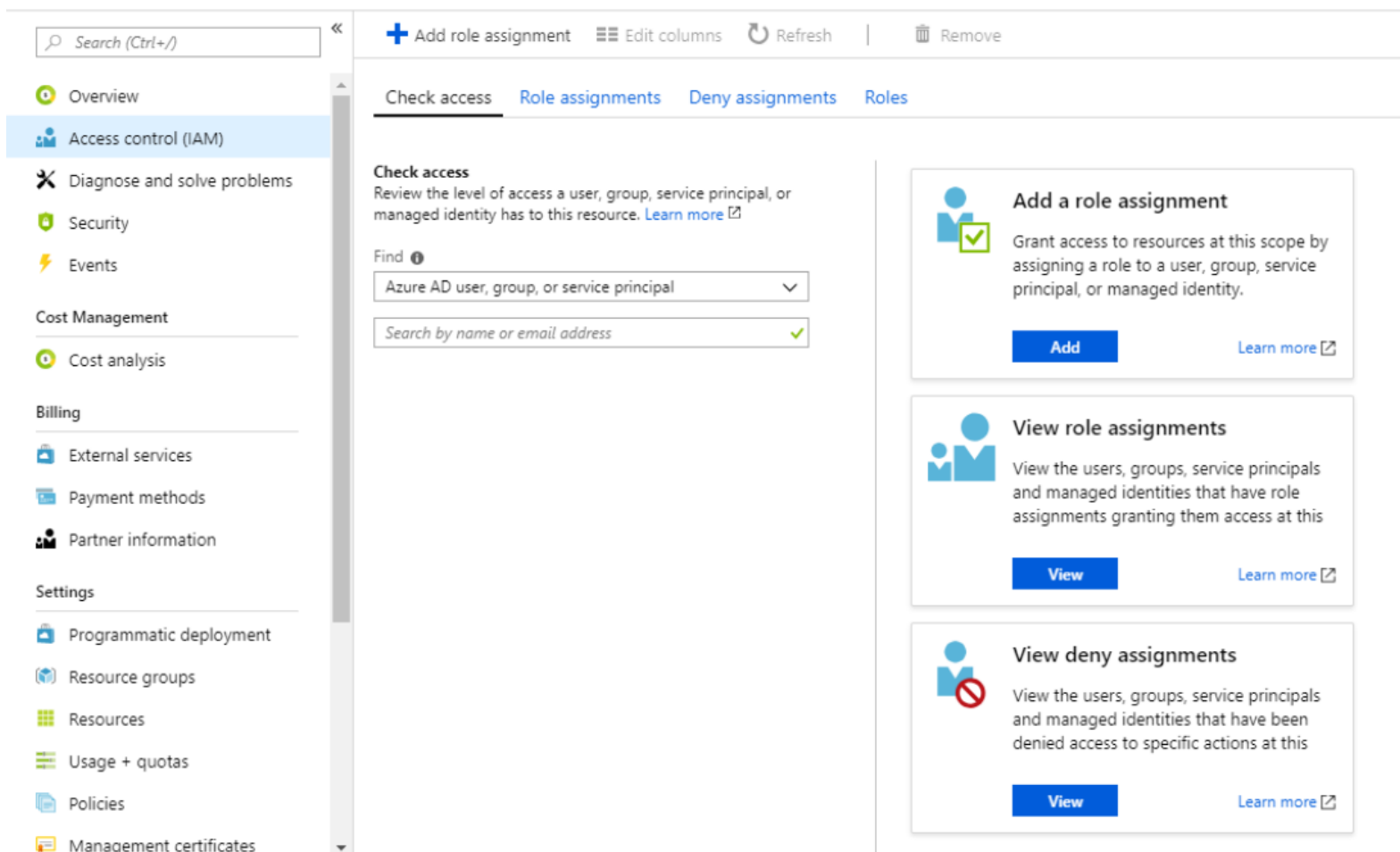
Claims

- A name/value pair representing an identity and its properties
 - Generated by an identity provider:
 - Azure AD, Facebook, Google, LinkedIn, Twitter, etc.
 - Serves as the basic for authorization:
 - Handled by a resource provider
 - Determines access to resources



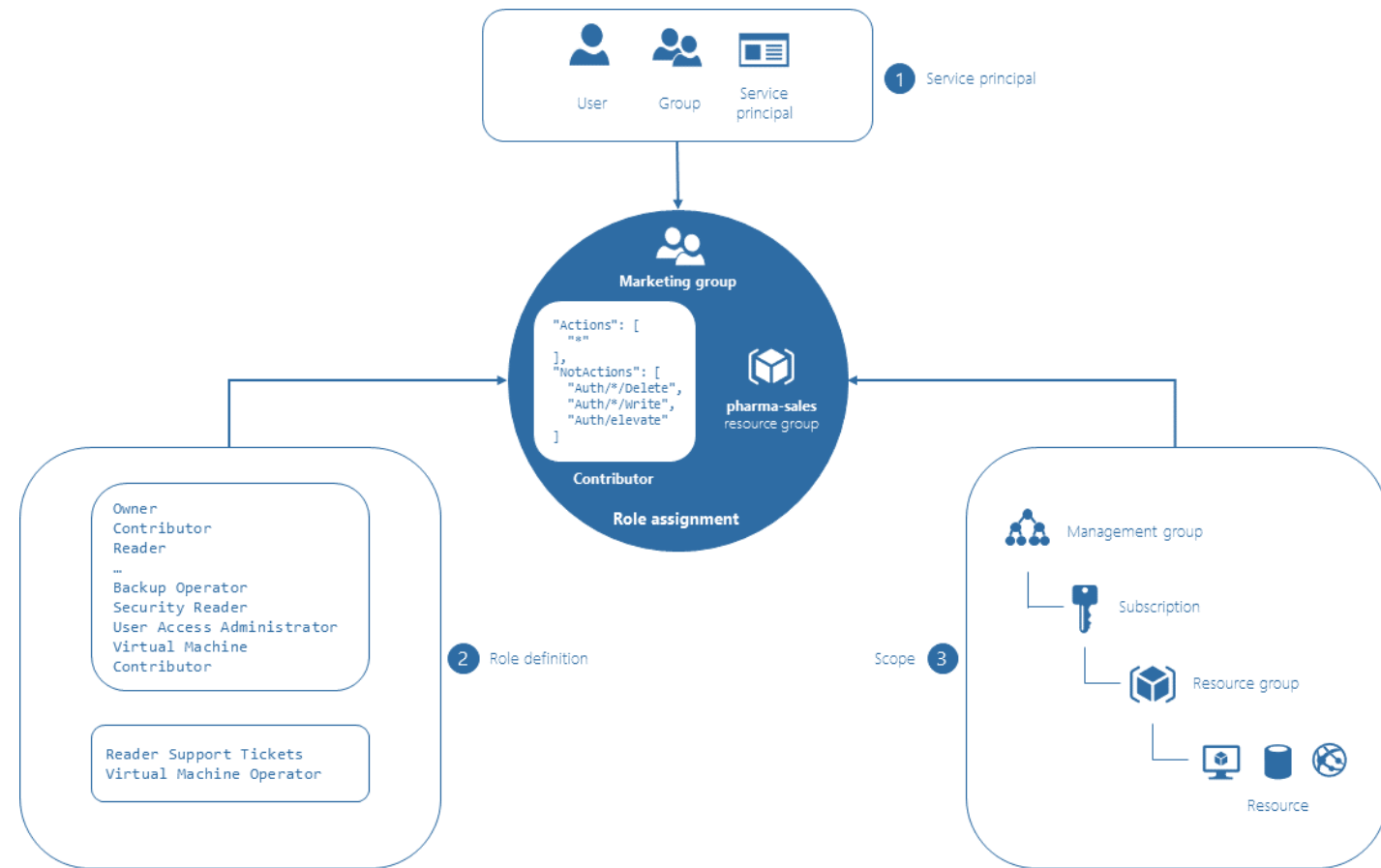
Role-based authorization

- Manages and enforces permissions based on user roles:
 - An identity may belong to one or more roles



Role-based access control (RBAC)

- Provides fine-grained access management of resources in Azure
- Facilitates segregation of duties
- Role assignments bind a role definition to a security principal, at a specific scope (or boundary) for the purpose of granting access.
- Supports four scope types:
 - A management group
 - A subscription
 - A resource group
 - A resource
- Includes built-in roles



Design Authorization



Choose an authorization approach

Authorization is the process of determining which entities have permission to change, view, or otherwise access a computer resource.

A claim is a name/value pair that represents what the subject is and not what the subject can do.

Claims-based authorization is an approach where the authorization decision to grant or deny access is based on arbitrary logic that uses data available in claims to make the decision.

Claims-based authorization, at its simplest, checks the value of a claim and allows access to a resource based on that value.

Choose an authorization approach

Claim-based authorization checks are declarative—the developer embeds them within their code, against a controller or an action within a controller, specifying claims that the current user must possess and optionally the value the claim must hold to access the requested resource.

Claims requirements are policy based; the developer must build and register a policy expressing the claims requirements.

*Let's say you want access to a bar. Here's an example:
The door security officer evaluates the value of your date of birth claim and whether they trust the issuer (the driving license authority) before granting you access.*

Choose an authorization approach

Role-based authorization is an authorization approach in which user permissions are managed and enforced by an application based on user roles.

If a user has a role that is required to perform an action, access is granted; otherwise, access is denied. When an identity is created, it may belong to one or more roles.

For example, Holly may belong to the Administrator and User roles, whereas Adam may belong only to the User role. How these roles are created and managed depends on the backing store of the authorization process.

Role-based access control (RBAC) is a system that provides fine-grained access management of resources in Azure. Using RBAC, you can segregate duties within your team and grant only the amount of access to users that they need to perform their jobs

Choose an authorization approach

Roles are exposed to the developer through the *IsInRole* method on the *ClaimsPrincipal* class.

Role-based authorization checks are declarative—the developer embeds them within their code, against a controller or an action within a controller, specifying roles that the current user must be a member of to access the requested resource.

You can mix and match both claims-based authorization and role-based authorization. Is it typical to see the role defined as a special claim.

Choose an authorization approach

A role assignment consists of three elements: a security principal, a role definition, and the scope.

A security principal is an object that represents a user, group, or service principal that is requesting access to Azure resources.

A role definition is a collection of permissions.

The Scope is the boundary that the access applies to.

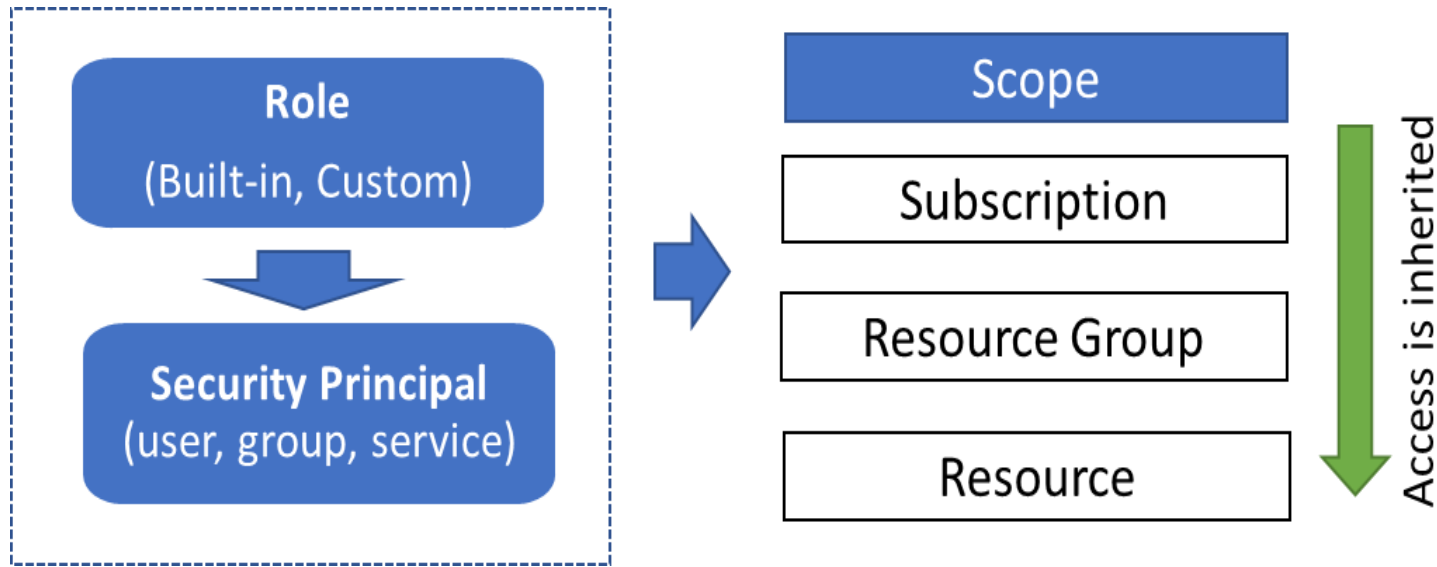
Design for Risk Prevention for Identity



Role-based Access Control

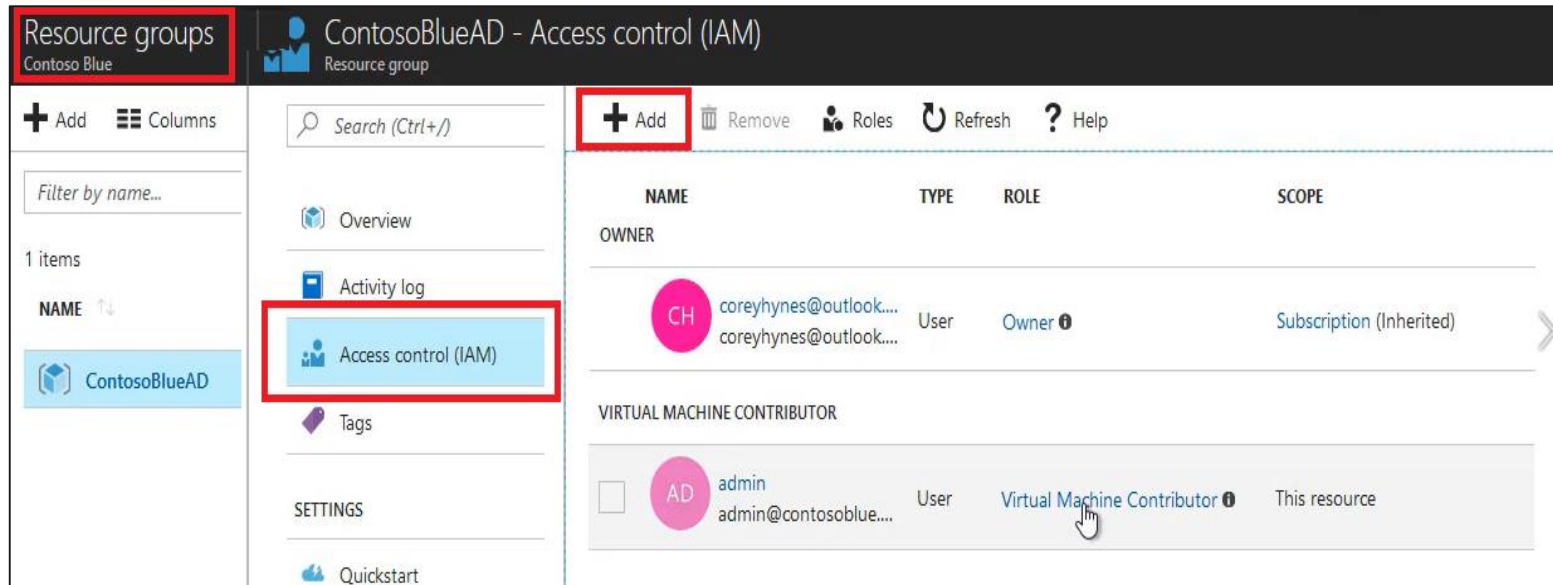


Role-Based Access Control



1. Select a role (the definition of what actions are allowed and/or denied)
2. Associating the role with a security principal (user, group, or service)
3. Scope to a subscription, a resource group, or a specific resource










RBAC in the Portal



- You can use the Azure Portal to make your role assignments
- You can add or remove roles as you need
- You can add synced users and groups to Azure roles, which enables organizations to centralize the granting of access

Built-in Roles

- Azure AD has many [built-in roles](#)
- Owner has full access to all resources including the right to delegate access to others.
- Contributor can create and manage all types of Azure resources but can't grant access to others
- Reader can view existing Azure resources

Roles		
ASH		
NAME	USERS	GROUPS
 Owner ⓘ	0	1
 Contributor ⓘ	4	0
 Reader ⓘ	1	0
 AcrImageSigner ⓘ	0	0
 AcrQuarantineReader ⓘ	0	0
 AcrQuarantineWriter ⓘ	0	0
 API Management Service Contributor ⓘ	0	0
 API Management Service Operator Role ⓘ	0	0
 API Management Service Reader Role ⓘ	0	0

Role Definitions

- Each role has a role definition defined in a JSON file
- The **Actions** and **NotActions** properties allow or deny actions
- The **AssignableScopes** property specifies the affected subscriptions, resource groups, or resources

Name: Owner

ID: 8e3af657-a8ff-443c-a75c-2fe8c4bcb65

IsCustom: False

Description: Manage everything, including access to resources

Actions: {*}

NotActions: {}

AssignableScopes: {/}

Role Assignments (PowerShell and CLI)

- For large numbers of role assignments, use PowerShell or the CLI

#Role assignment properties

\$roleName = "Contributor"

\$assigneeName = josh@microsoft.com

\$resourceGroupName = "contosoblue"

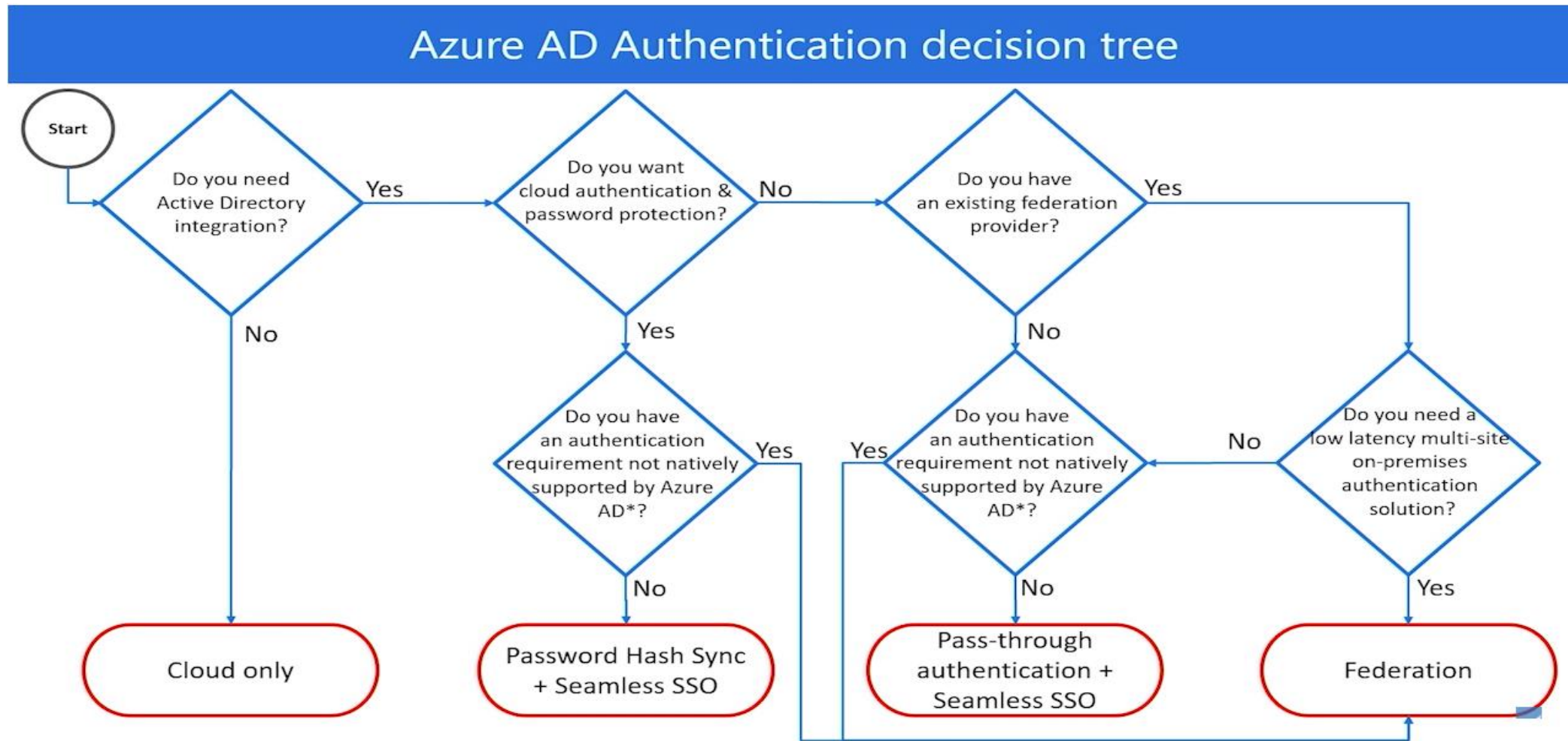
- Azure PowerShell

```
New-AzureRmRoleAssignment -RoleDefinitionName $roleName -SignInName  
$assigneeName -ResourceGroupName $resourceGroupName
```

- CLI

```
az role assignment create --role $roleName --assignee $assigneeName --  
resource-group $resourceGroupName
```

Video: Azure AD Authentication Options



Security Best Practices

- Centralize your identity management
- Enable Single Sign-On (SSO)
- Deploy password management
- Enforce MFA for users
- Use role-based access control (RBAC)
- Control Resource Manager resource locations
- Guide developers to leverage identity capabilities for SaaS apps
- Actively monitor for suspicious activities

Privileged Identity Management



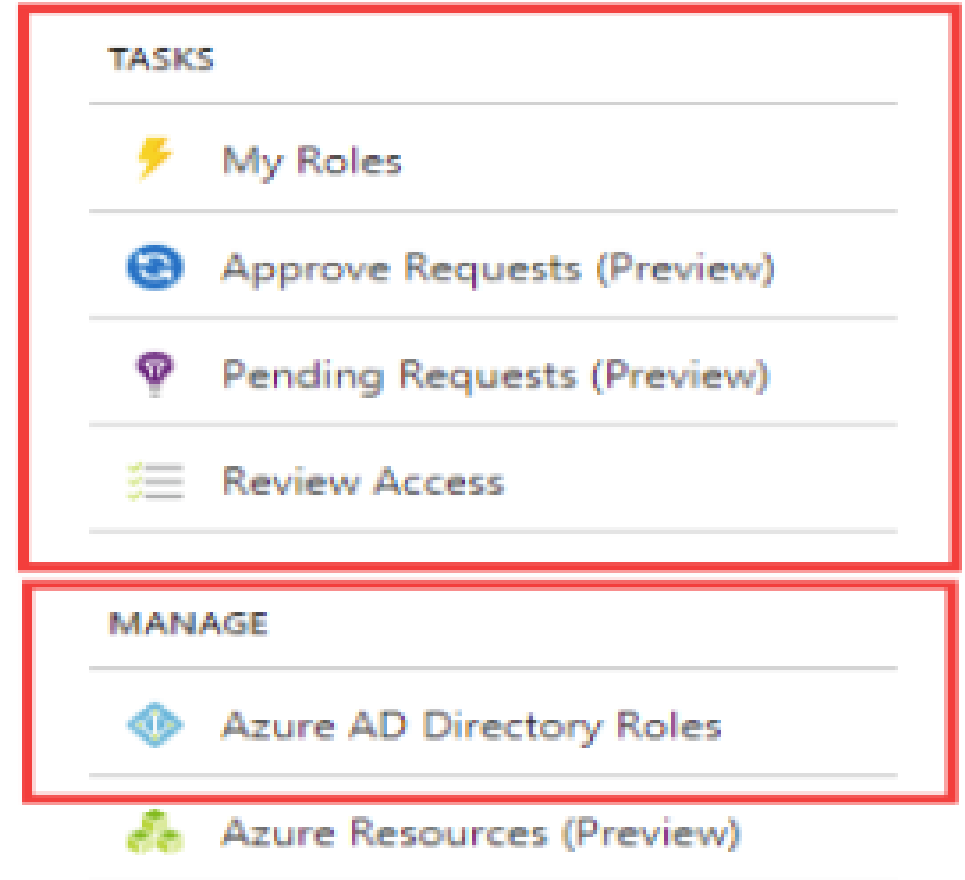
Azure AD PIM

- PIM, just-in-time administration, is a cloud-based service
- Minimize the number of users who can execute privileged operations
- Identify users are assigned privileged roles
- View administrator activation
- Require approval to activate
- Review membership of administrative roles



PIM Tasks

- Display a list of eligible and active roles assigned to you
- Display a list of requests to activate eligible Azure AD directory roles
- Display pending requests to activate eligible role assignments.
- List active access reviews
- Display the dashboard



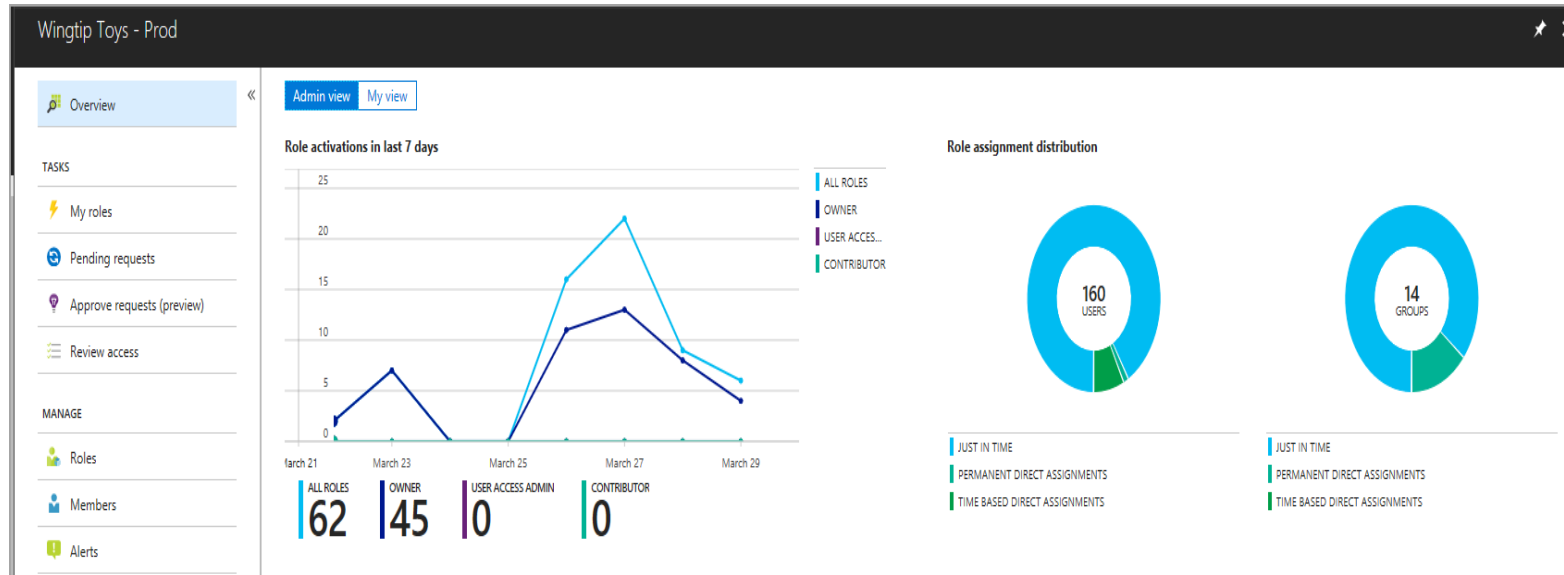
PIM Access

The screenshot displays the Azure AD Privileged Identity Management (PIM) console. The left sidebar shows the 'Manage privileged roles' option highlighted with a red box. The main area shows a summary of roles and a table of role details. The 'Privileged Role Administrator' role is highlighted in the table. The right pane shows the 'Privileged Role Administrator' role details, with the '+ Add' button highlighted.

ROLE NAME	MFA ENABLED	USERS	ACTIVE	ELIGIBLE
Security Reader	Yes	1	1 (100%)	0 (0%)
Global Administrator	Yes	9	5 (56%)	4 (44%)
Privileged Role Administrator	Yes	2	2 (100%)	0 (0%)
Security Administrator	Yes	9	2 (22%)	7 (78%)
Password Administrator	Yes	2	0 (0%)	2 (100%)
User Administrator	Yes	2	0 (0%)	2 (100%)

- The global administrator who enables PIM automatically gets role assignments and access to PIM
- Other global administrators, security administrators, and security readers have read-only access to Azure AD PIM
- The first user can assign others to the Privileged role administrator

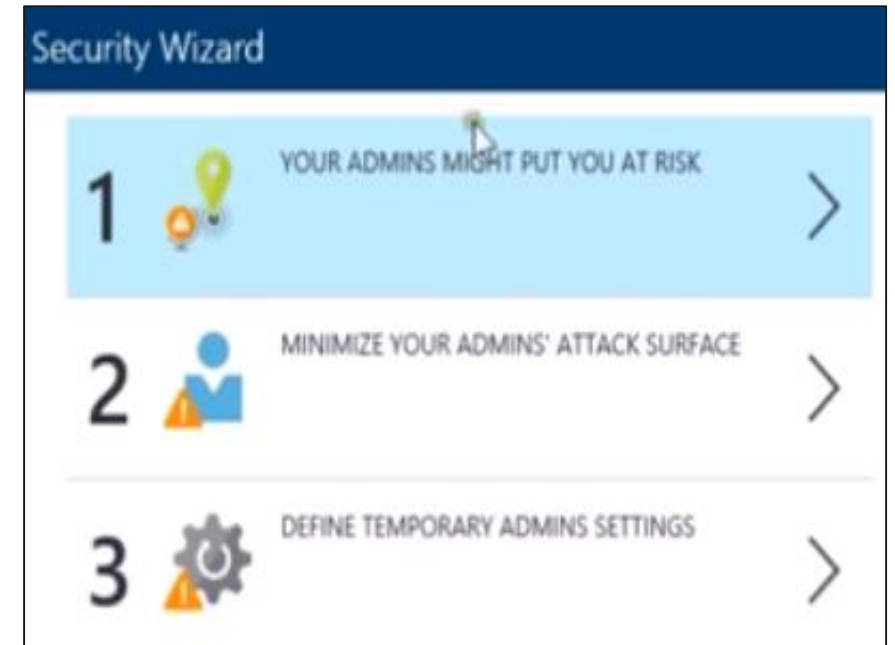
PIM Dashboard



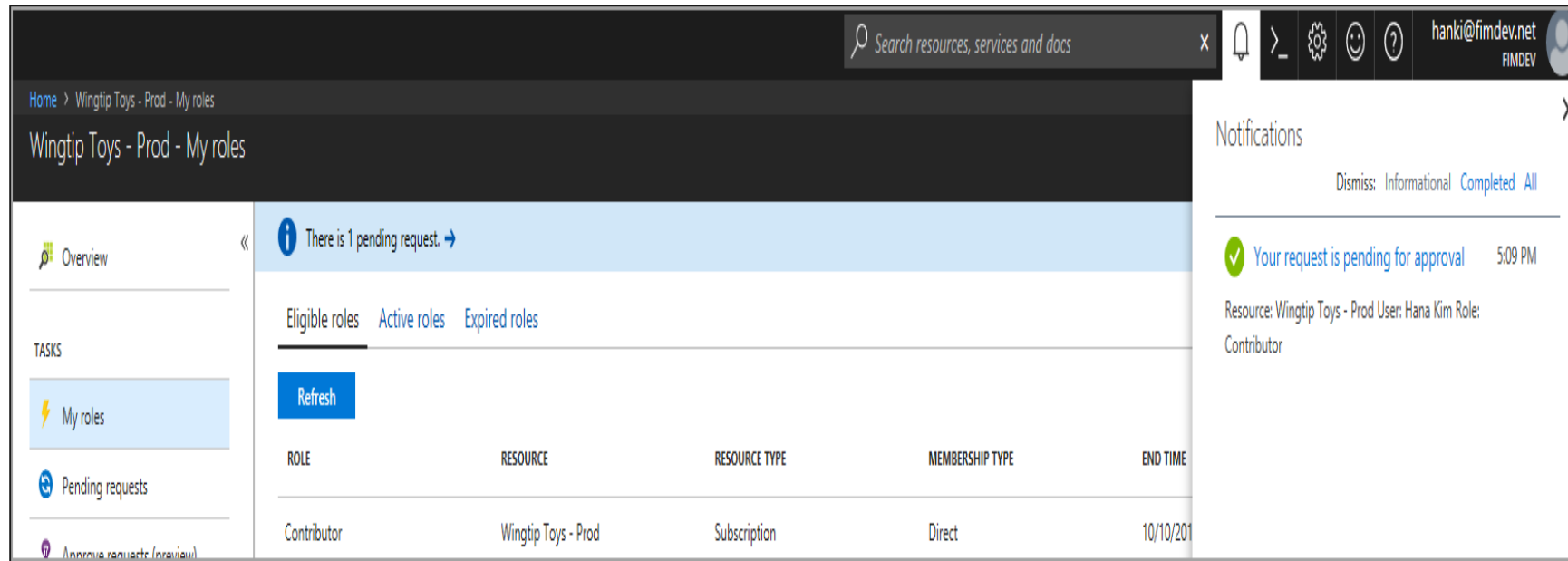
- A graphical representation of resource role activations.
- Two charts that display the distribution of role assignments by assignment type
- A data area pertaining to new role assignments.

PIM Security Wizard

- The first person to run PIM will see the wizard. After you have made changes, the wizard will no longer show up.
- Helps you understand the security risks of privileged identities and how to use PIM to reduce those risks
- Have at least one global administrator, and more than one privileged role administrator



Activate Roles



The screenshot displays the 'Wingtip Toys - Prod - My roles' page. At the top, there is a search bar and a user profile for 'hanki@fimdev.net'. A notification panel on the right indicates 'Your request is pending for approval' at 5:09 PM, with details: 'Resource: Wingtip Toys - Prod User: Hana Kim Role: Contributor'. The main content area shows a table of roles with columns: ROLE, RESOURCE, RESOURCE TYPE, MEMBERSHIP TYPE, and END TIME. A 'Refresh' button is located above the table.

ROLE	RESOURCE	RESOURCE TYPE	MEMBERSHIP TYPE	END TIME
Contributor	Wingtip Toys - Prod	Subscription	Direct	10/10/201

- PIM is expanding to non-administrator resource role assignments
- Use Just Enough Administration to assign users and groups to reduced subscription or resource scope
- Select a specific activation duration, including a future time

Assign Roles

- Just in time provides the user or group members with eligible but not persistent access to the role for a specified period or indefinitely (if configured in role settings)
- Direct does not require the user or group members to activate the role assignment (known as persistent access)

The screenshot displays two side-by-side windows from a role management application.

New assignment window:

- Header: New assignment
- Message: The Role assignment already exists.
- Step 1: Select a role. The selected role is Custom Role 3.
- Step 2: Select a user or group. The selected user is Albert Almora.
- Step 3: Set membership settings. The selected setting is Default setting is selected.

Membership settings window:

- Header: Membership settings
- Assignment type: A dropdown menu with 'Just in time' selected and 'Direct' as an option.
- Maximum allowed eligible duration is permanent.
- ☒ Permanently eligible
- * Eligible starts: 2018-03-29 3:03:15 PM
- * Eligible ends: 2018-06-27 3:03:15 PM

PIM Resource Alerts

Wingtip Toys - Prod - Alerts

Overview

TASKS

My roles

Pending requests

Approve requests (preview)

Review access

Settings

Scan

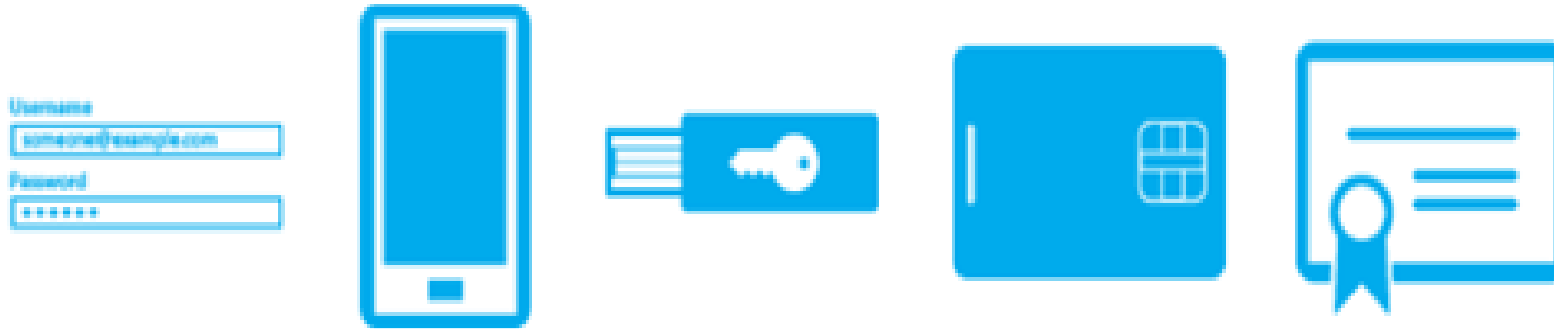
RISK LEVEL	COUNT	ALERT
Medium	46	Too many owners assigned to a resource
Medium	1	Too many permanent owners assigned to a resource

- Same severity levels: **High, Medium, and Low**
- "Too many owners assigned to a resource" alert
- "Too many permanent owners assigned to a resource" alert
- "Duplicate role created" alert

Multi-Factor Authentication for Secure Access



Azure MFA Concepts



- The security of MFA two-step verification lies in its layered approach
- Authentication methods include:
 - Something you know (typically a password)
 - Something you have (a trusted device that is not easily duplicated, like a phone)
 - Something you are (biometrics)

Azure MFA Features

- Get more security with less complexity
- Mitigate threats with real-time monitoring and alerts
- Deploy on-premises or on Azure
- Use with Office 365, Salesforce, and more
- Add protection for Azure administrator accounts

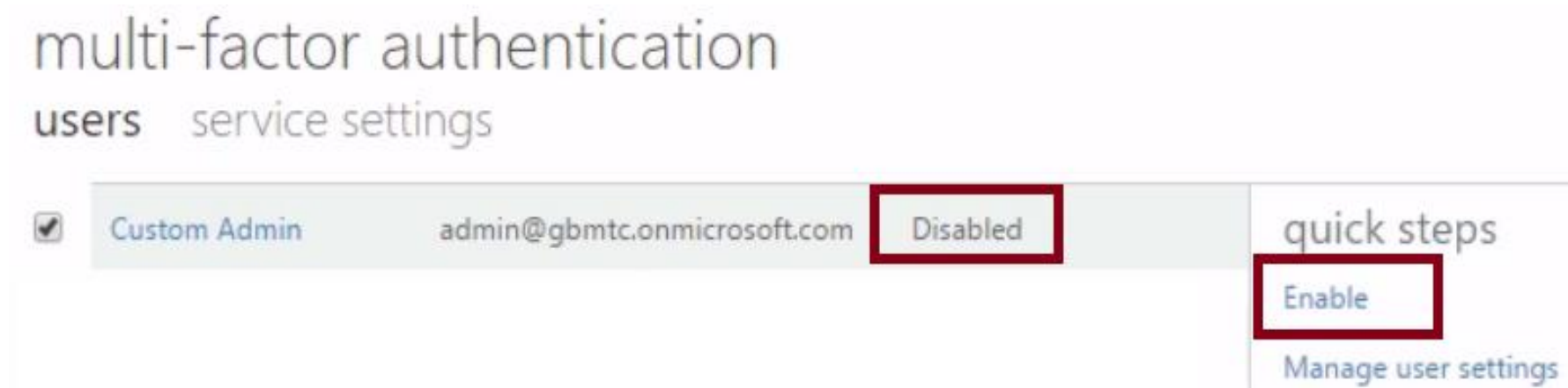
MFA Licensing and Pricing

- There are three pricing methods for Azure MFA.
- Consumption based billing
 - **Per user.** You can pay per user. Each user has unlimited authentications. Use this model if you know how many users you have and can accurately estimate your costs.
 - **Per authentication.** You can pay for a bundle (10) of authentications. Use this model when you are unsure how many users will participate in MFA authentication.
- MFA licenses included in other products
- Direct and Volume licensing

Microsoft Authenticator App

- Prevent unauthorized access to accounts
- Stop fraudulent transactions by giving you an additional level of security
- Use either as a second verification method or as a replacement for your password when using phone sign-in
- The app can work in one of two ways:
 - **Notification.** The app sends a notification to your device, then Verify or Deny
 - **Verification code.** Open the app and copy the verification code onto the sign-in screen

MFA for Global Admins



- Free of charge for global administrator security
- Added level of security when managing and creating Azure resources, like virtual machines
- Secondary authentication includes phone call, text message, and the authenticator app
- Use the portal to enable MFA for administrators

What are You Trying to Secure?

What are you trying to secure	Azure MFA	MFA Server
First-party Microsoft apps	•	•
SaaS apps in the app gallery	•	
Web applications published through Azure AD App Proxy	•	
IIS applications not published through Azure AD App Proxy		•
Remote access such as VPN, RDG	•	•

Where Are Your Users Located?

User Location	Azure MFA	MFA Server
Azure Active Directory	●	
Azure AD and on-premises AD using federation with AD FS	●	●
Azure AD and on-premises AD using Azure AD Connect - no password hash sync or pass-through authentication	●	●
Azure AD and on-premises AD using Azure AD Connect - with password hash sync or pass-through authentication	●	
On-premises Active Directory		●

What Features Do You Need?

Feature	Azure MFA	MFA Server
Mobile app notification and mobile app verification code as a second factor	●	●
Mobile app verification code as a second factor	●	●
Phone call or one-way SMS as second factor	●	●
Hardware Tokens as second factor		●
PIN mode		●
Fraud alert and MFA reports	●	●
Remember MFA for trusted devices	●	
Conditional access	●	●

Trusted IPs

- Allows federated users or IP address ranges to bypass two-step authentication
- For managed tenants, you can specify IP ranges that can skip MFA
- For federated tenants, you can specify IP ranges and you can also exempt AD FS claims users

multi-factor authentication

users service settings

trusted ips [\(learn more\)](#)

☐ Skip multi-factor authentication for requests from federated users on my intranet

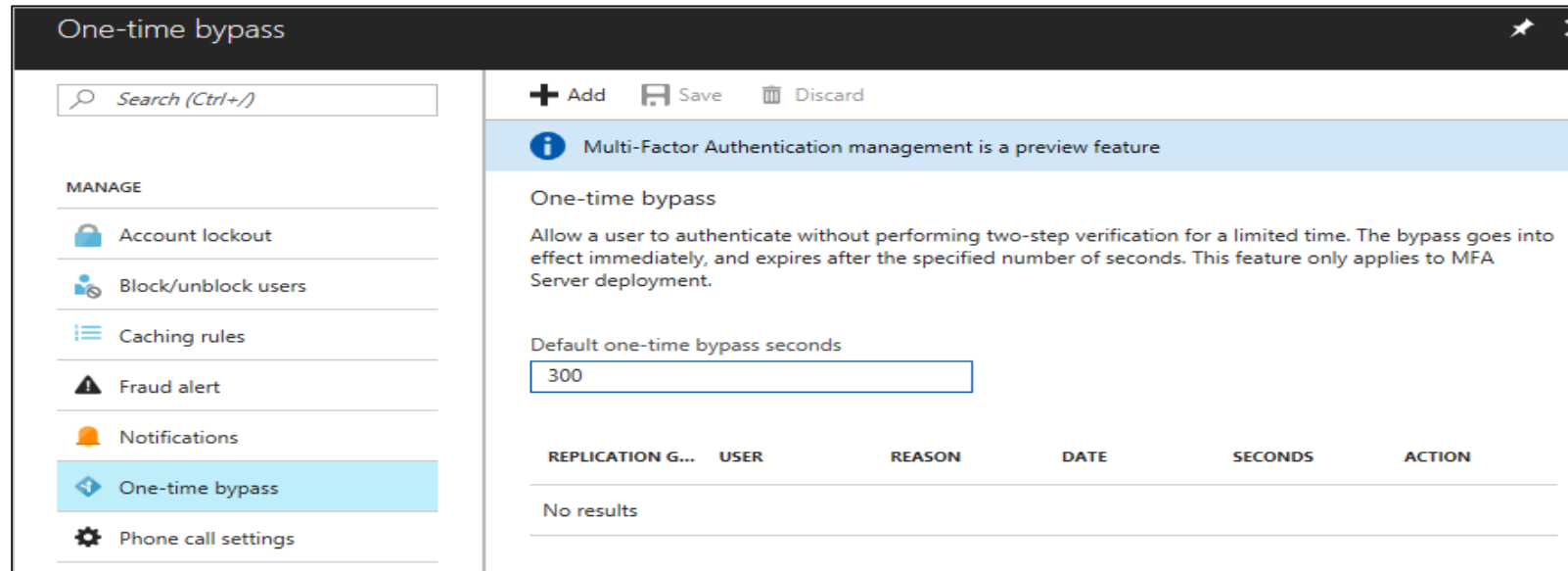
Skip multi-factor authentication for requests from following range of IP address subnets

192.168.1.0/27

192.168.1.0/27

192.168.1.0/27

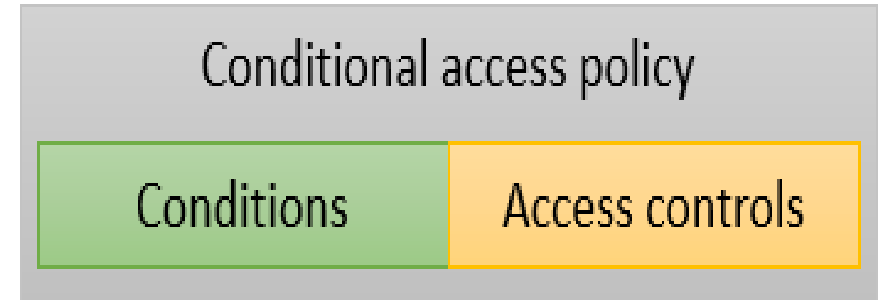
One-time Bypass



- Allows a user to authenticate a single time without performing two-step verification
- The bypass is temporary and expires after a specified number of seconds.

Conditional Access

- Enables you to enforce controls on access to apps based on specific conditions
- The combination of your conditions with your access controls represents a conditional access policy



Conditions – “When this happens”


Access controls – “Then do this”


Fraud Alert


- Users can report fraudulent attempts to access their resources
- Report fraud attempts by using the mobile app or through their phone
- Block user when fraud is reported


Fraud alert


MANAGE


 Account lockout

 Block/unblock users


 Caching rules

 **Fraud alert**

 Notifications

 One-time bypass

Save Discard

 Multi-Factor Authentication management is a preview feature

Fraud alert

Allow your users to report fraud if they receive a two-step verification request that they didn't initiate.

Allow users to submit fraud alerts

On

Off

Automatically block users who report fraud

On

Off

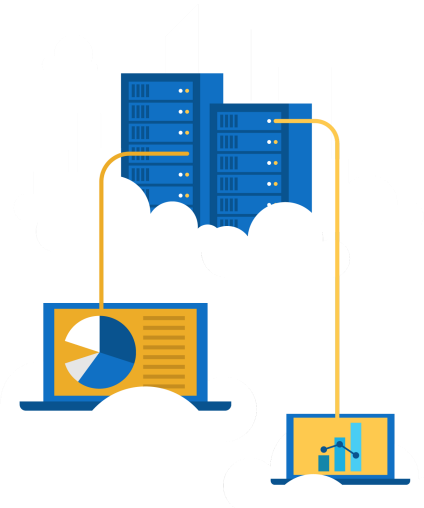
Code to report fraud during initial greeting



Questions?

Homework Assignment

<https://aka.ms/az301asg>



Open Mic

