**Manage Azure identities and governance**

**Azure AD**

* Concept of **Tenants**
* Internet-based applications by using HTTP and HTTPS communications - Azure AD uses the REST API

it does not use Kerberos authentication. Instead, it uses HTTP and HTTPS protocols such as SAML, WS-Federation, and OpenID Connect for authentication (and OAuth for authorization)

- Flat structure

- Editions: Free, M O365, Premium1, Premium2

-**Azure AD join** enables single sign-on to devices, apps, and services from anywhere

- **Registering** a device to Azure AD enables you to manage a device’s identity

you can use the identity to enable or disable a device

Registration combined with a mobile device management (MDM) solution such as

Microsoft Intune provides additional device attributes in Azure AD

- **Joining** a device is an extension to registering a device.

Joining provides the benefits of registering and changes the local state of a device

Changing the local state enables your users to sign-in to a device using an organizational work or school account instead of a personal account

**self-service password reset**

Authentication methods: email notification, a text, or code sent to user’s mobile or office

phone, or a set of security questions

**Users and groups**

**3 types of AAD users**

**Cloud identities -** exist only in Azure AD

**Directory-synchronized identities** – on premise users, sync via **Azure AD connect** – Source: Windows Server AD

**Guest users** - Examples are accounts from other cloud providers and Microsoft accounts such as an Xbox LIVE account

To consider when **Managing users**

Must be **Global Administrator** or **User Administrator** to manage users.

User profile (picture, job, contact info) is optional.

**Deleted** users can be restored for 30 days.

Sign in and audit log information is available.

Ways to **Add Cloud identities**

**Azure Portal –** Create user or Invite User

Users can also be added to Azure AD through Microsoft 365 Admin Center, Microsoft Intune

admin console, and the CLI

**Create bulk user accounts –** fill out **CSV** template on Portal or **Powershell**

**Group Accounts**

**Security Groups -** used to manage member and computer access to shared resources

**Microsoft 365 Groups -** collaboration opportunities to sharepoint, files, shared mailbox, etc.

**Adding Members to a Group**

**Assigned –** add specific users

**Dynamic User –** dynamic membership rule

**Dynamic Device (security groups only) –** dynamic rules

**Administrative Unit –** administer only a certain group/unit

**Subscriptions**

logical unit of Azure services that is linked to an Azure account. Billing for Azure services is done on a

per-subscription basis

Azure **Cost Management** and **Billing** features - billing administrative tasks , costs, monitor and control spending

Cost analysis, Budgets, Recommendations, and Exporting cost management data

Apply **Resource Tagging**

Tags to resources to logically organize

Resource or resource group has max of 50 tags

RG tags are not inherited by resources

**Cost savings** – by **Reservations, Hybrid Benfits, Azure Credits, Azure Regions, Budgets, Pricing Calculator**

**Management Groups**

manage access, policies, and compliance for subscriptions included in the management group

subscriptions automatically inherit the conditions applied to the management group

**Create Management Groups by:** Portal, Powershell and Azure CLI

**Azure Policies -** enforce different rules over your resources

**Enforcement and compliance**. Turn on built-in policies or build custom ones for all resource types. Real-time policy evaluation and enforcement. Periodic and on-demand compliance evaluation.

**Apply policies at scale**. Apply policies to a Management Group with control across your entire organization. Apply multiple policies and aggregate policy states with policy initiative. Define an exclusion scope.

**Remediation**. Real-time remediation, and remediation on existing resources.

Steps to implement:

View **Policy Definitions** – what to evaluate and what actions to take

Create **Initiative Definitions** – Set of policy definitions for larger goal

**Scope Initiate Definitions –** limit the scope, to MG, Subs, RG

View **Policy Evaluation results –** evaluate state of compliance

**RBAC**

RBAC is an authorization system built on Azure Resource Manager

**Role Definition** - defined via JSON file,

Actions – Allowable permission

NotActions – denied permissions

Scope – Subs, RG, Resource

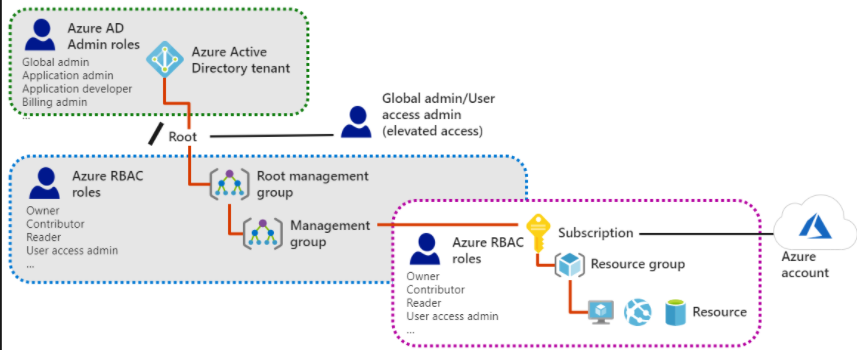
**Role Assignment -** scoping a role definition to a user, group, service principal, or managed identity

A resource inherits role assignments from its parent resource

Differences between **Azure Roles** and **Azure AD roles**

**Azure Roles Azure AD Roles**

Manage access to Azure resources Manage access to AAD resources

Scope = MG, Subs, RG, resource Scope = tenant level

**Owner**. Has full access to all resources including the right to delegate access to others. The **Service Administrator** and **Co-Administrators** are assigned the Owner role at the subscription scope.

**Contributor**. Can create and manage all types of Azure resources but can’t grant access to others.

**Reader**. Can view existing Azure resources.

**User Access Administrator**. Lets you manage user access to Azure resources, rather than to managing resources.