Caltech | Center for Technology & Management Education

Post Graduate Program in Cloud Computing

Powerd by simplilearn

Caltech | Center for Technology & Management Education

PG CC - Microsoft Azure Architect Design: AZ:304

Powered by simplilearn

Design Security for Applications

Learning Objectives

By the end of this lesson, you will be able to:

- Justify the importance of Azure Key Vault and the benefits it offers
- Implement and configure managed identities
- Configure applications and integrate them into Azure AD

A Day in the Life of an Azure Architect

You are working as an Architect for an organization. The organization is looking to improve its security posture. As part of ensuring that all the resources are protected, you need to design a data protection solution that build an identity for usage with Azure applications, hosted services, and automated tools.

- The roles allocated to the solution should restrict this access, allowing you to control which resources may be accessed and to what degree.
- Also, they are looking for an Azure security solution for storing application secrets such as tokens, passwords, certificates, API keys, and other secrets such as certificates, keys, and secret management.

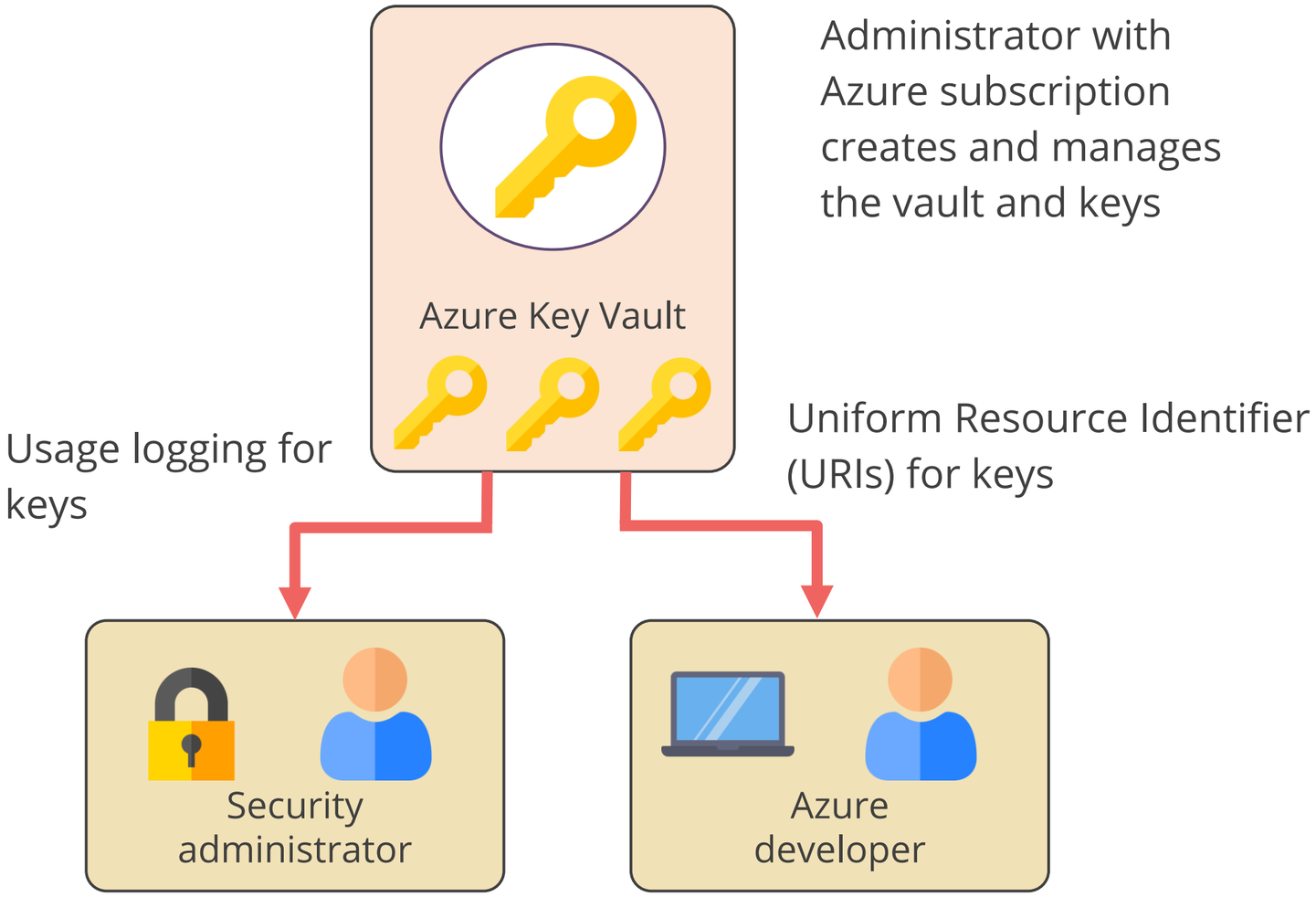
To achieve all of the above, along with some additional features, we would be learning a few concepts in this lesson that will help you find a solution for the above scenario.

Recommend a Solution That Includes Key Vault

Azure Key Vault

Azure Key Vault is a centralized cloud service for storing application secrets such as encryption keys, certificates, and server-side tokens.

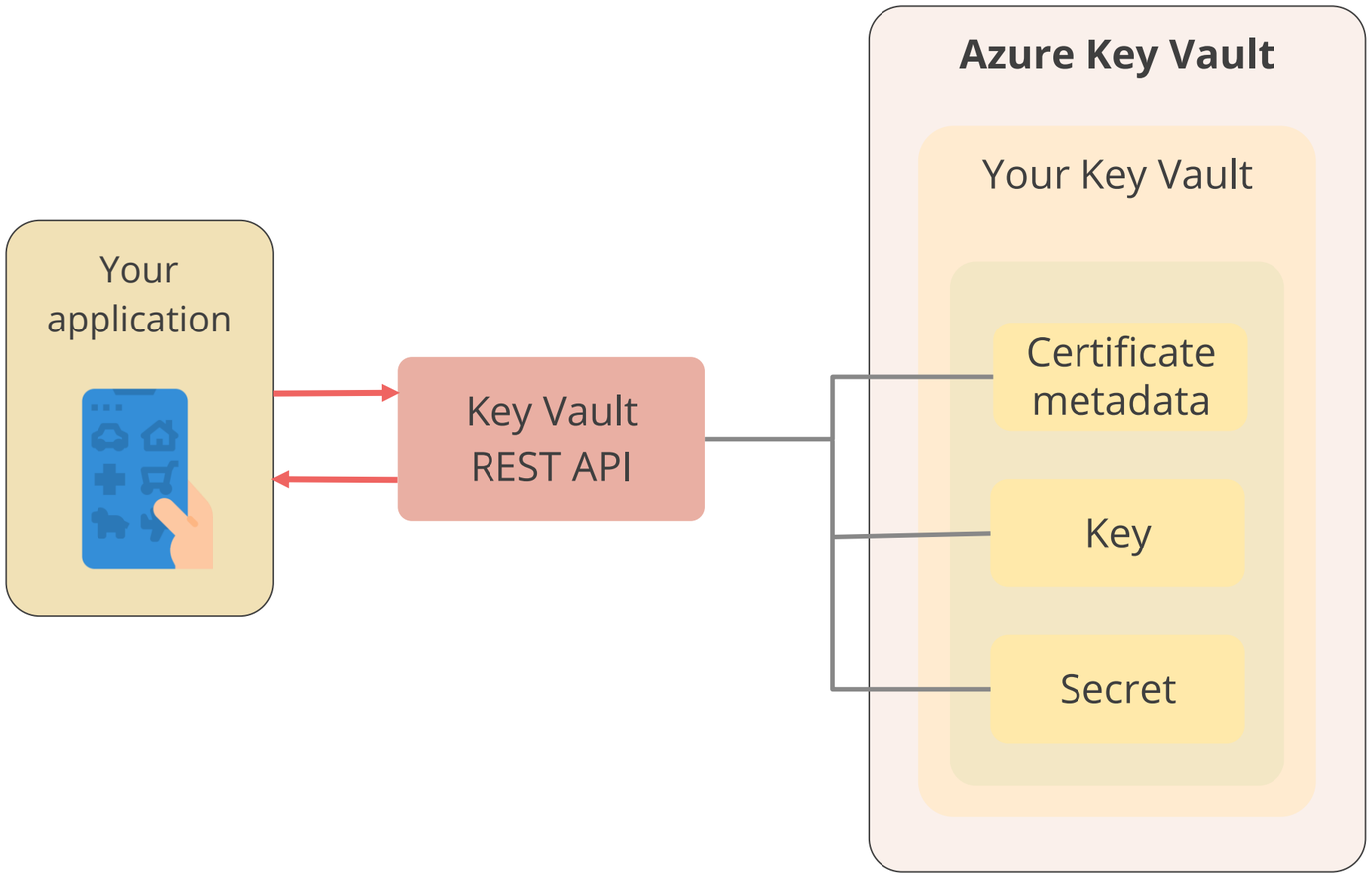
Azure Key Vault



Azure Key Vault offerings:

- Secrets management
- Key management
- Certificate management
- Secrets storage

Azure Key Vault Benefits

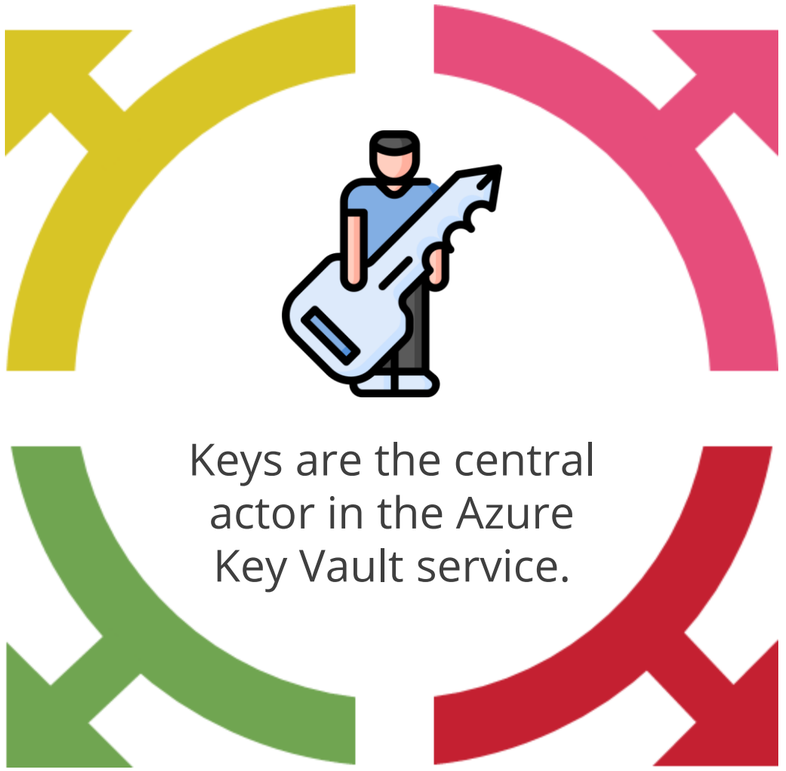


- Centralize application secrets
- Securely store secrets and keys
- Monitor access and use
- Simplified administration of application secrets
- Integrate with other Azure services

Keys

Keys can be single instanced (only one key exists)

Keys are cryptographic assets that serve a specific purpose



Keys are the central actor in the Azure Key Vault service.

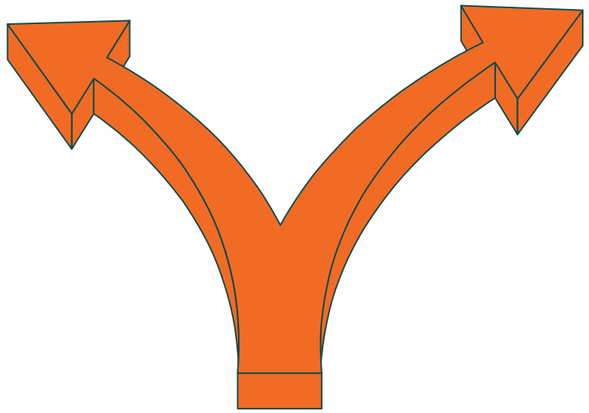
Keys can be versioned (mix of primary and secondary keys)

Keys are never directly accessible to applications

Keys

Hardware Protected Keys

Supports using HSMs that provide a hardened, tamper-resistant environment for cryptographic processing and key generation



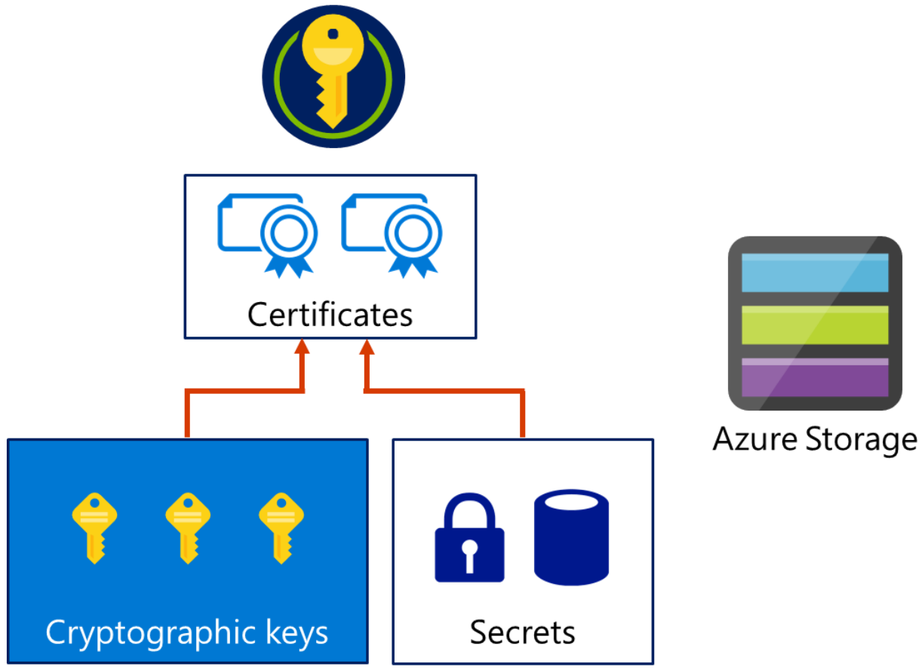
Software Protected Keys

Supports using software-based Rivest–Shamir–Adleman (RSA) and Elliptic curve cryptography (ECC) algorithms

Two types of keys in Key Vault

Secrets

Secrets are small (less than 10K) data blobs protected by HSM-generated key created with the Key Vault.



Types of Keys

Cryptographic keys

Key Vault supports multiple key types and algorithms and using hardware security modules (HSMs) for high-value keys.

Certificates

Key Vault supports certificates, which are built on top of keys and secrets and add an automated renewal feature.

Secrets

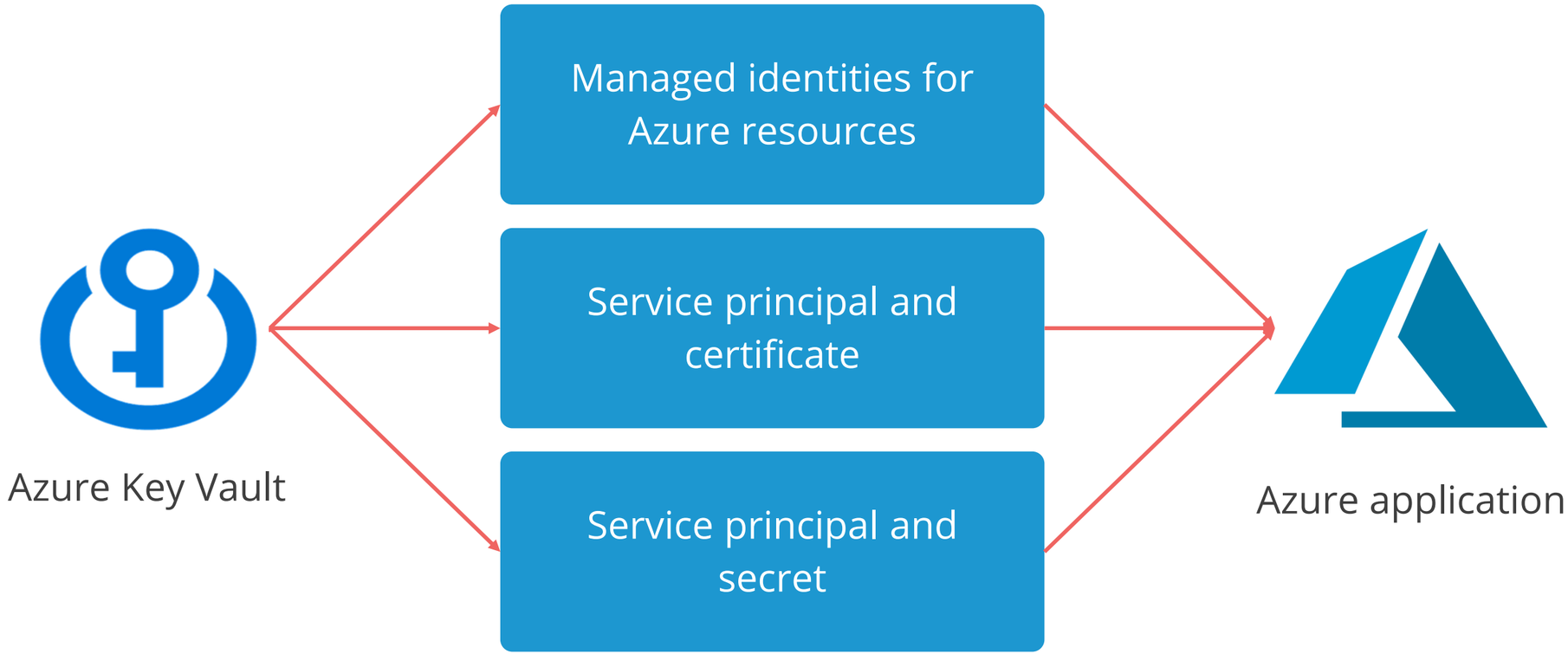
Key Vault provides secure storage of secrets, such as passwords and database connection strings.

Azure storage

Key Vault can manage the keys of an Azure Storage account. Internally, Key Vault can list (sync) keys with an Azure Storage Account and regenerate (rotate) the keys periodically.

Key Vault Authentication and Authorization

Illustration of authentication and authorization between the Azure Key Vault and Azure application:



Key Vault Authentication and Authorization

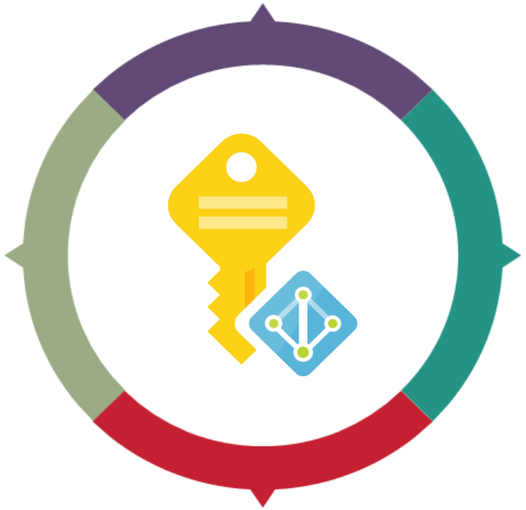
The procedure for accessing the Key Vault:

Authenticate

Azure Active Directory to authenticate users and applications

Read and write data in the Azure Key Vault

Uses a separate Key Vault access policy



Restrict network access

Services in the network can access the vault

Authorize

Role-based access control (RBAC)

Key Vault Availability and Redundancy

Availability and redundancy are distinct features in Azure Key Vault:

01 Key vault features multiple layers of redundancy

02 Contents of the Key Vault are replicated within the region and to a secondary region at least 150 miles away

03 If components within the Key Vault service fail, alternate components within the region step in

Key Vault Best Practices

The best practices of the Key Vault:

- Control what users can access
- Store certificates in the Key Vault

- Grant access to a specific scope
- Ensure that you can recover a deleted Key Vault or Key Vault objects

Assisted Practice

Azure Key Vault **Duration: 10 Min.**

Problem Statement:

As an Azure Architect, you've been asked to provide your organization with an Azure security solution for storing application secrets such as tokens, passwords, certificates, API keys, and other secrets such as certificates, keys, and secret management.

Assisted Practice: Guidelines

Steps to create an Azure key vault are:

1. Login to your Azure portal
2. Search for and select Key Vault
3. Select Create on key vault page
4. Provide the details and create an Azure key vault

Recommend a Solution that Includes Managed Identities

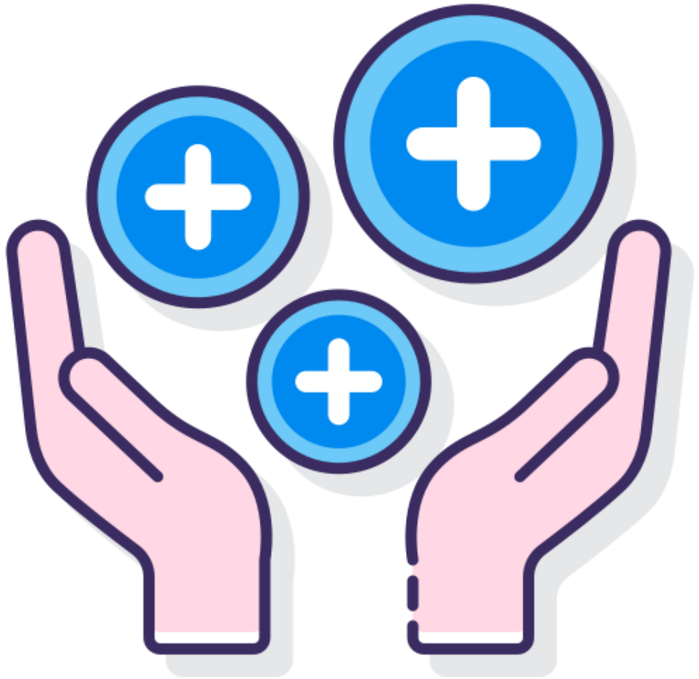
Azure Managed Identity

Azure managed identity provides identity for applications to connect with resources that support Azure AD authentication.

It combines Azure AD authentication and Azure RBAC.

Azure Managed Identity

The benefits of Azure managed identity are:



- No need to manage credentials
- Any resource can be authenticated
- No additional cost
- No need for rotating credentials or certificates

Azure Managed Identity Terminologies

Managed identities involve the use of these terms:

Client ID

A unique ID linked to the Azure AD application and service principal

Object ID

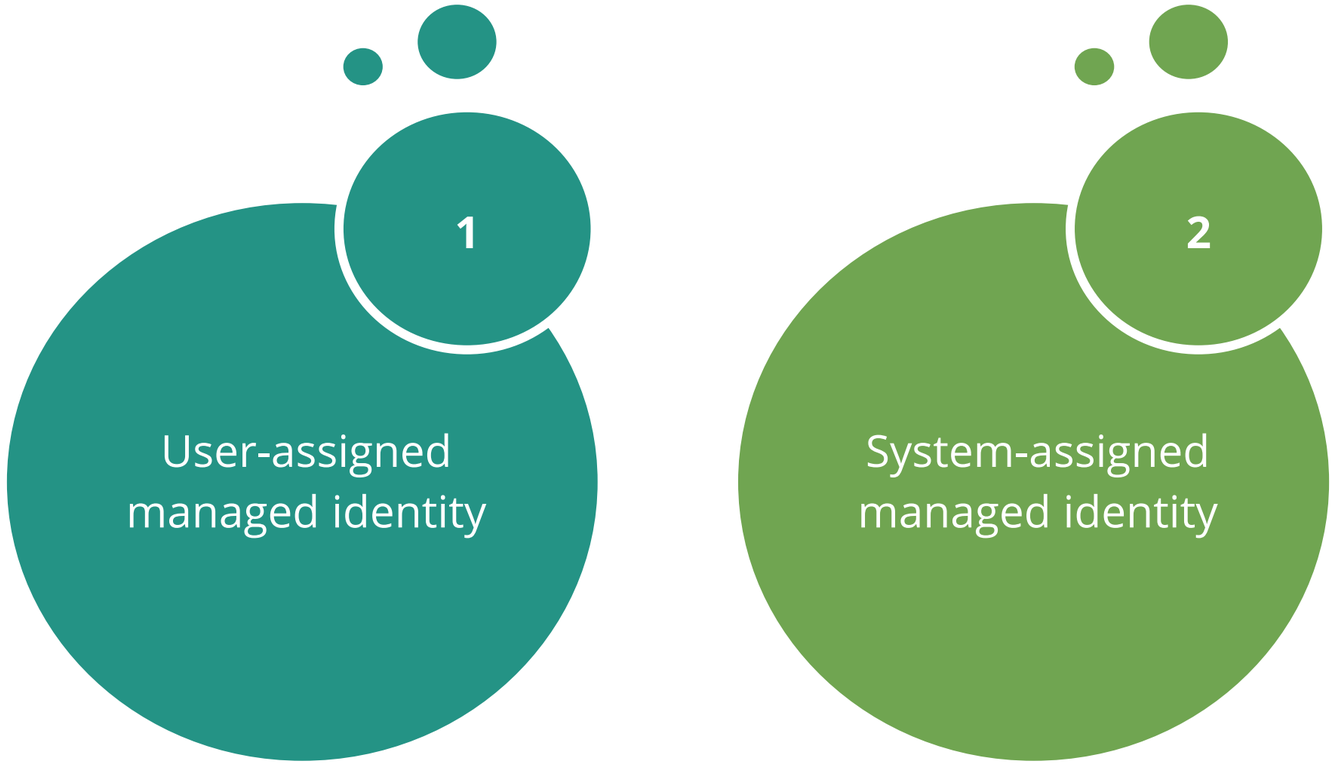
The service principal object of the managed identity

Azure Instance Metadata Service

A REST API that's enabled when Azure Resource Manager provisions a VM.

Types of Managed Identity

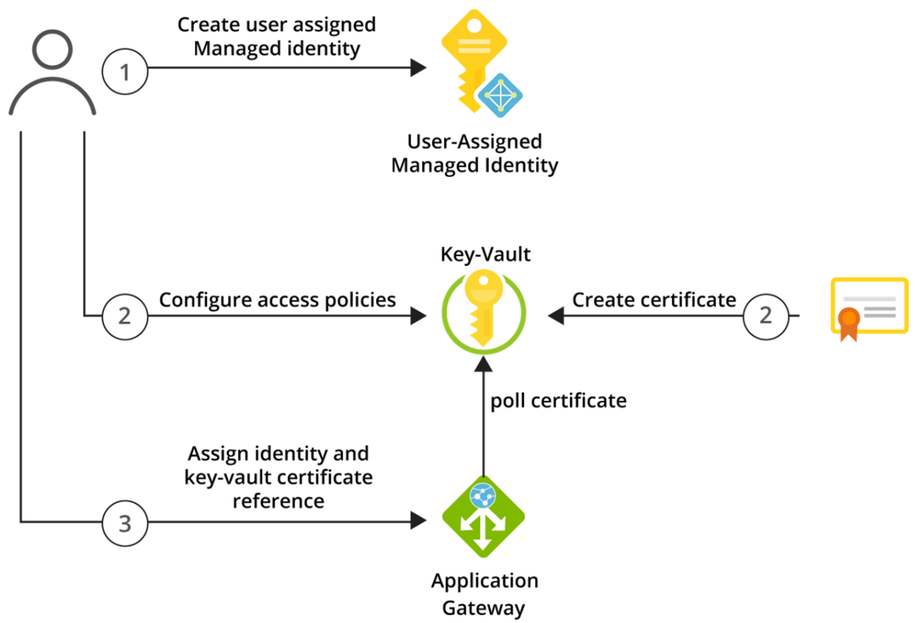
There are two types of managed identities:



User-Assigned Managed Identity

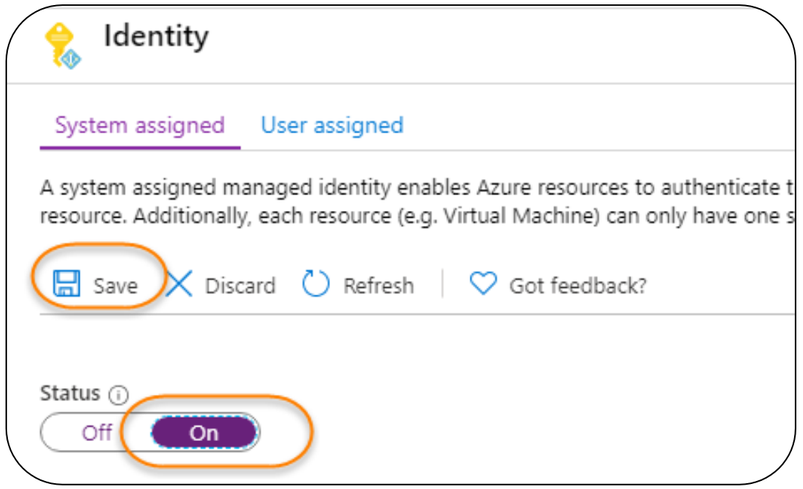
Azure creates a service principal just as it does for a system-assigned identity.

It is created as a standalone Azure resource.



System-Assigned Managed Identity

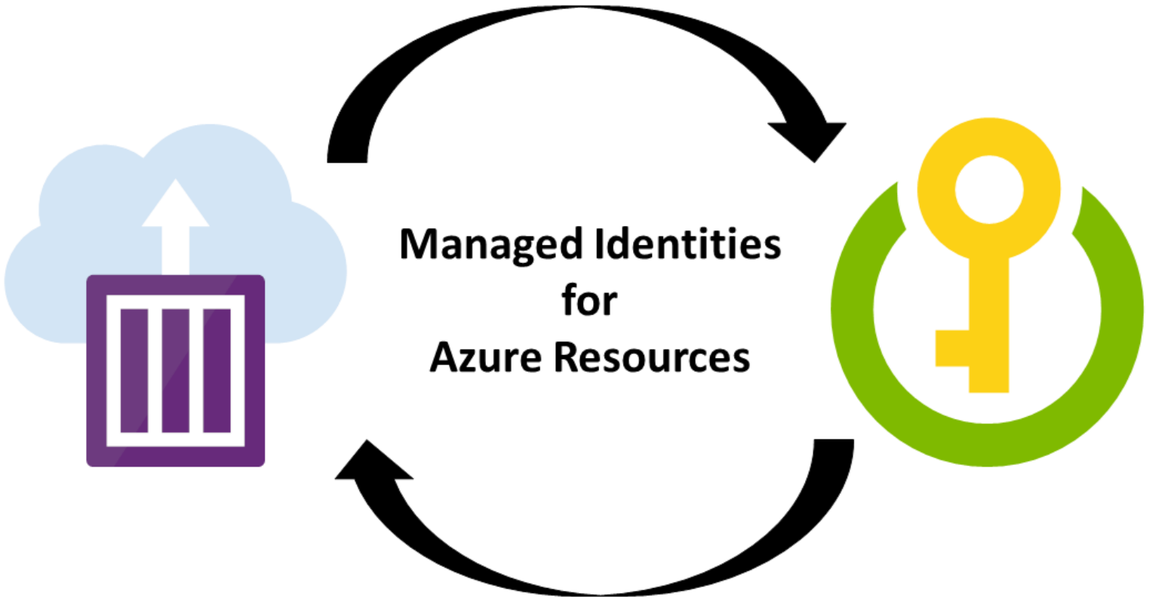
When a user enables the identity, Azure creates a service principal through Azure Resource Manager.



System-assigned managed identity is enabled on an Azure service instance, such as a VM.

Managed Identity for Azure Resources

It manages the credentials to authenticate cloud services when building cloud applications.



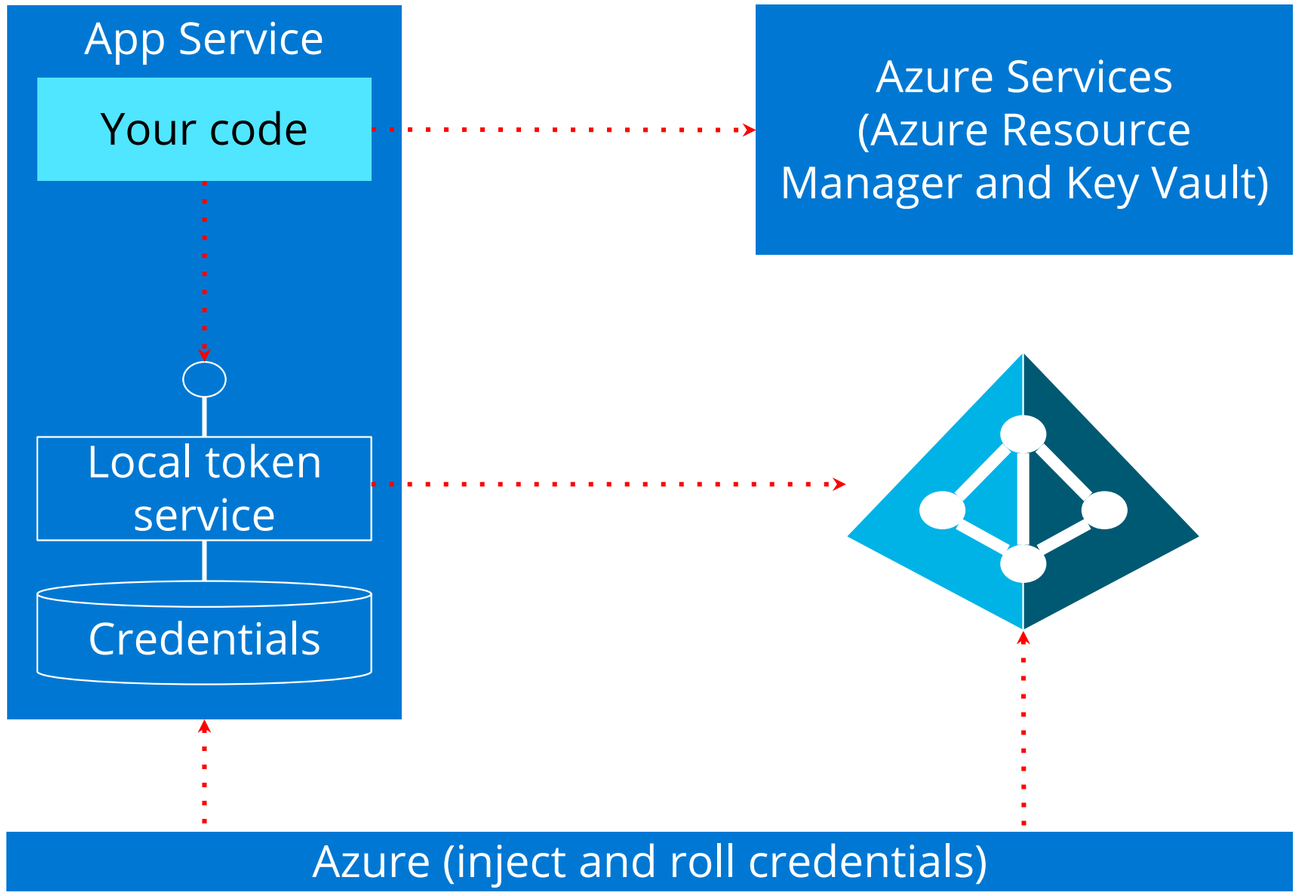
Managed Identity for Azure Resources

These are the advantages of managed identity when used for Azure resources:

- Keeps credentials out of code
- Uses a local MSI endpoint to get access tokens from Azure AD
- Manages the identity in Azure AD for Azure resources automatically
- Offers direct authentication with services or retrieval of credentials from the Azure Key Vault

Workflow of Managed Identity

The following diagram shows the internal workflow of Managed Identity:



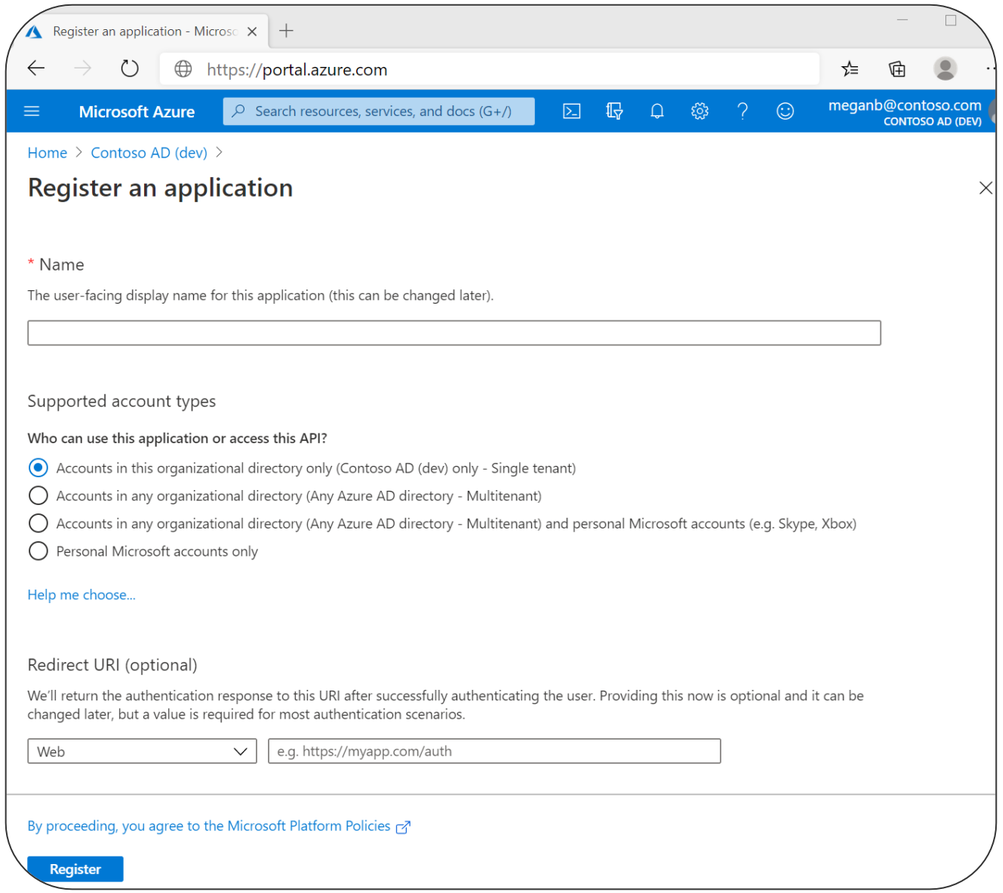
Recommend a Solution for Integrating Applications into Azure AD

App Registration

When an application is registered, it provides authentication and authorization services for the application and its users.

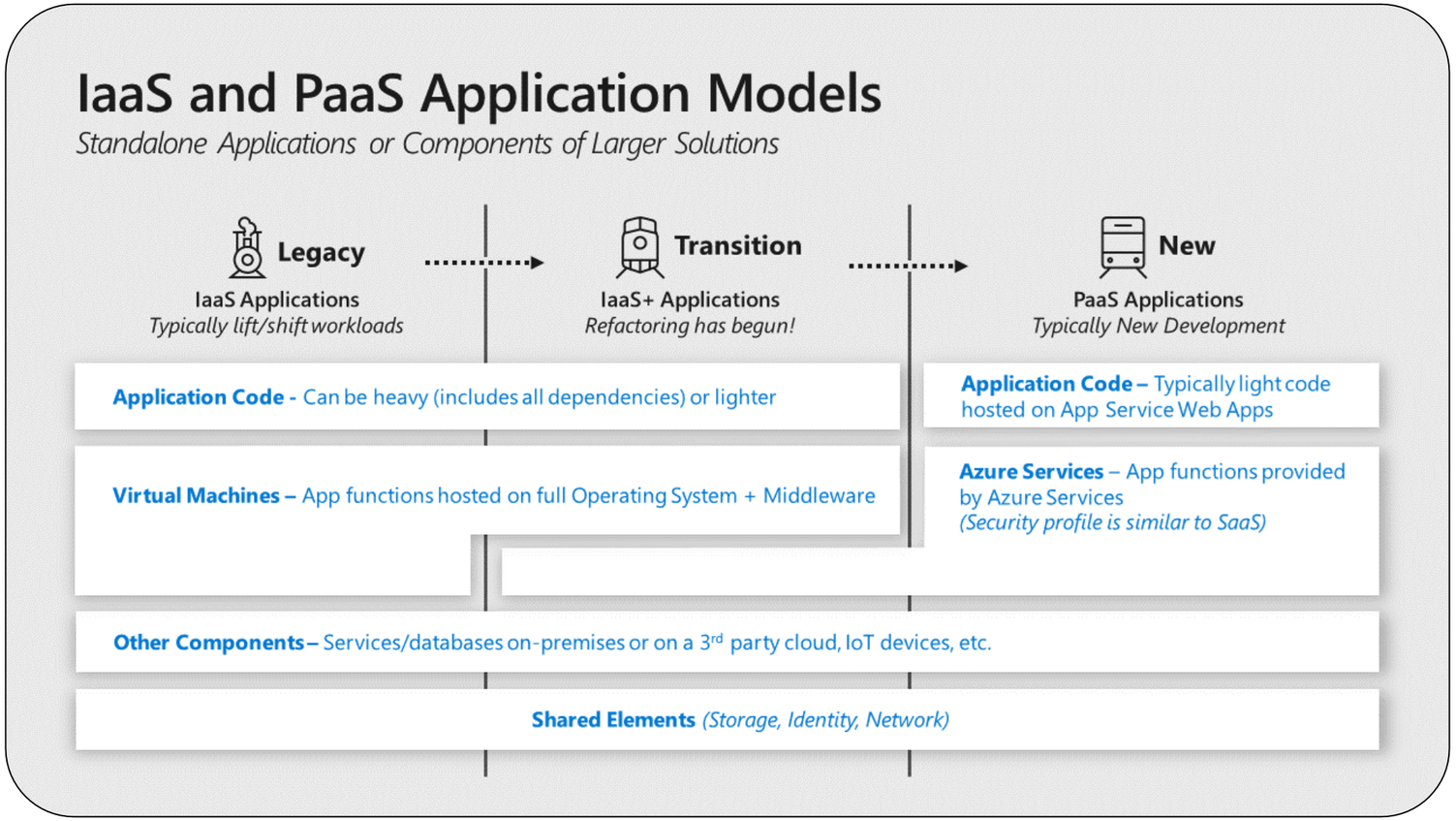
App Registration

A user can register an application under Azure Active Directory.



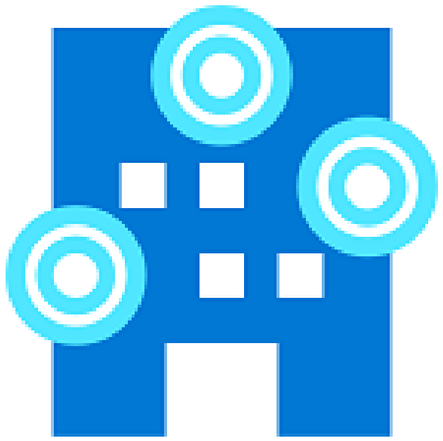
Security for Application and Services

The following figure illustrates the architecture of application models:



Identify and Classify Business Critical Applications

Identify and classify applications that have high impact or exposure to risk:



Business critical data

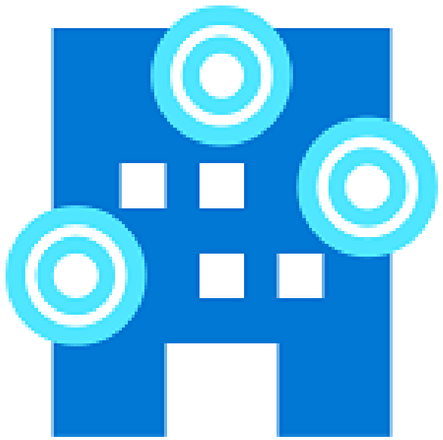
Applications that handle data would have a significant negative impact on the company if confidentiality, integrity, or availability were compromised.

Regulated data

Applications that process monetary instruments and confidential information are regulated by standards.

Identify and Classify Business Critical Applications

Identify and classify applications that have high impact or exposure to risk:



Business critical availability

Applications whose functionality is vital to a firm's business objective, such as life-saving technologies or services, and other important functions.

Significant Access

Applications that gain access to systems with a high impact to risk via technological techniques.

Use Cloud Services Instead of Custom Applications

These are the capabilities that should be addressed first due to high-security impact:

Identity

Avoid using homegrown authentication solutions and favor mature capabilities

Data Protection

Use cloud providers native encryption in cloud services to encrypt and protect data

Use Cloud Services Instead of Custom Applications

These are the capabilities that should be addressed first due to high-security impact:

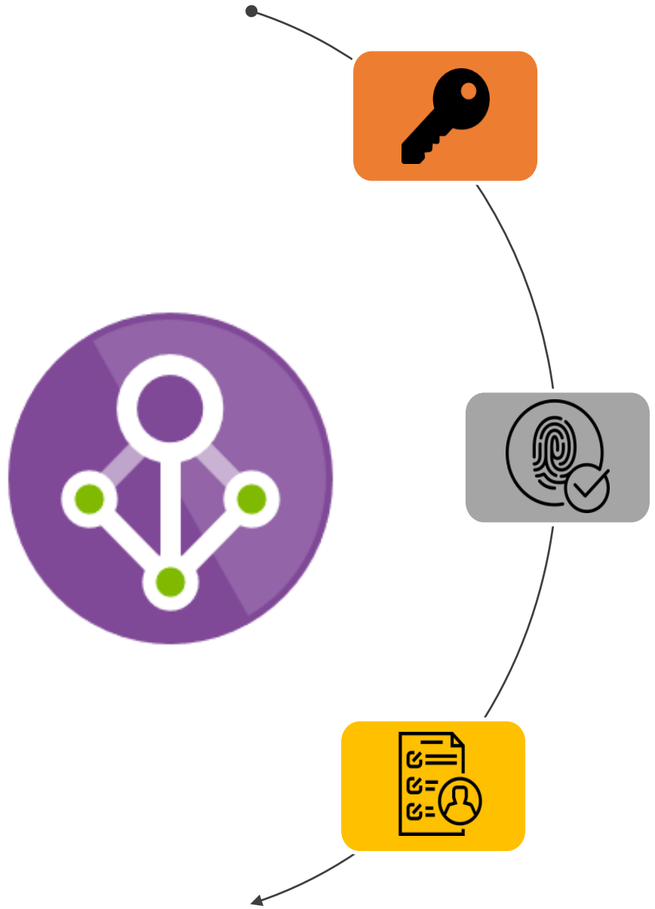
Key Management

Leverage key management services to manage and secure keys rather than attempting to handle keys in the application code

Configurations

Provides a service to centrally manage application settings and feature flags, which helps mitigate this risk

Preference for Identity Authentication Instead of Keys



Managing keys securely with application code is difficult and leads to mistakes like accidentally publishing sensitive access keys

Authenticate with identity services rather than cryptographic keys when available

For services that offer the Azure AD authentication like Azure Storage and Azure App Service use authentication and authorization

Unassisted Practice

Create a Service Principal

Duration: 10 Min.

Problem Statement:

As an Azure Architect, you've been asked to provide your company with an Azure security solution that can build an identity for usage with Azure applications, hosted services, and automated tools. The roles allocated to the solution should restrict this access, allowing you to control which resources may be accessed and to what degree.

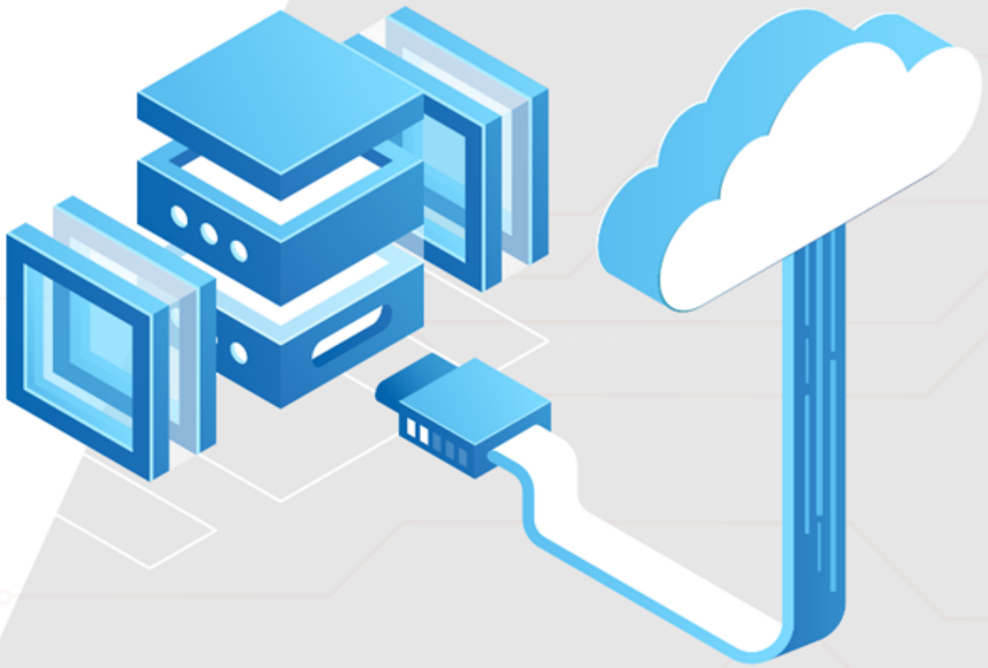
Unassisted Practice: Guidelines

Steps to create a service principal are:

1. Login to your Azure portal
2. Select Azure Active Directory
3. Select App registrations
4. Enter the required details
5. Azure AD application and service principal is created

Key Takeaways

- Azure Key Vault is a centralized cloud service for storing application secrets such as encryption keys, certificates, and server-side tokens.
- Secrets are small (less than 10K) data blobs protected by HSM-generated key created with the Key Vault.
- Azure managed identity provides identity for applications to connect with resources that support Azure AD authentication.
- Registering an application provides authentication and authorization services for the application and its users



Thank you