

Describe Azure identity, access, and security

1. Introduction

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/1-introduction>

Introduction

Completed

In this module, you'll be introduced to the Azure identity, access, and security services and tools. You'll learn about directory services in Azure, authentication methods, and access control. You'll also cover Zero Trust, defense in depth, and how they keep your cloud safer. Finally, you'll review encryption concepts, key management with Azure Key Vault, and Microsoft Defender for Cloud.

Learning objectives

After completing this module, you'll be able to:

- Describe directory services in Azure, including Microsoft Entra ID and Microsoft Entra Domain Services.
- Describe authentication methods in Azure, including single sign-on (SSO), multifactor authentication (MFA), and passwordless.
- Describe external identities and guest access in Azure.
- Describe Microsoft Entra Conditional Access.
- Describe Azure Role Based Access Control (RBAC).
- Describe the concept of Zero Trust.
- Describe the purpose of the defense in depth model.
- Describe encryption concepts and key management options in Azure.
- Describe the purpose of Microsoft Defender for Cloud.

2. Describe Azure directory services

Describe Azure directory services

Completed

Microsoft Entra ID is Microsoft's cloud-based identity and access management service. It lets you sign in and access both Microsoft cloud applications and cloud applications that you develop.

If you've worked with on-premises Active Directory, Microsoft Entra ID will feel familiar. The key difference is that you control the identity accounts while Microsoft ensures the service is available globally.

Connecting the two unlocks extra protection. On its own, on-premises Active Directory doesn't monitor sign-in behavior. Once connected to Microsoft Entra ID, Microsoft can detect suspicious sign-in attempts at no extra cost — for example, sign-ins from unexpected locations or unknown devices.

Who uses Microsoft Entra ID?

Microsoft Entra ID is for:

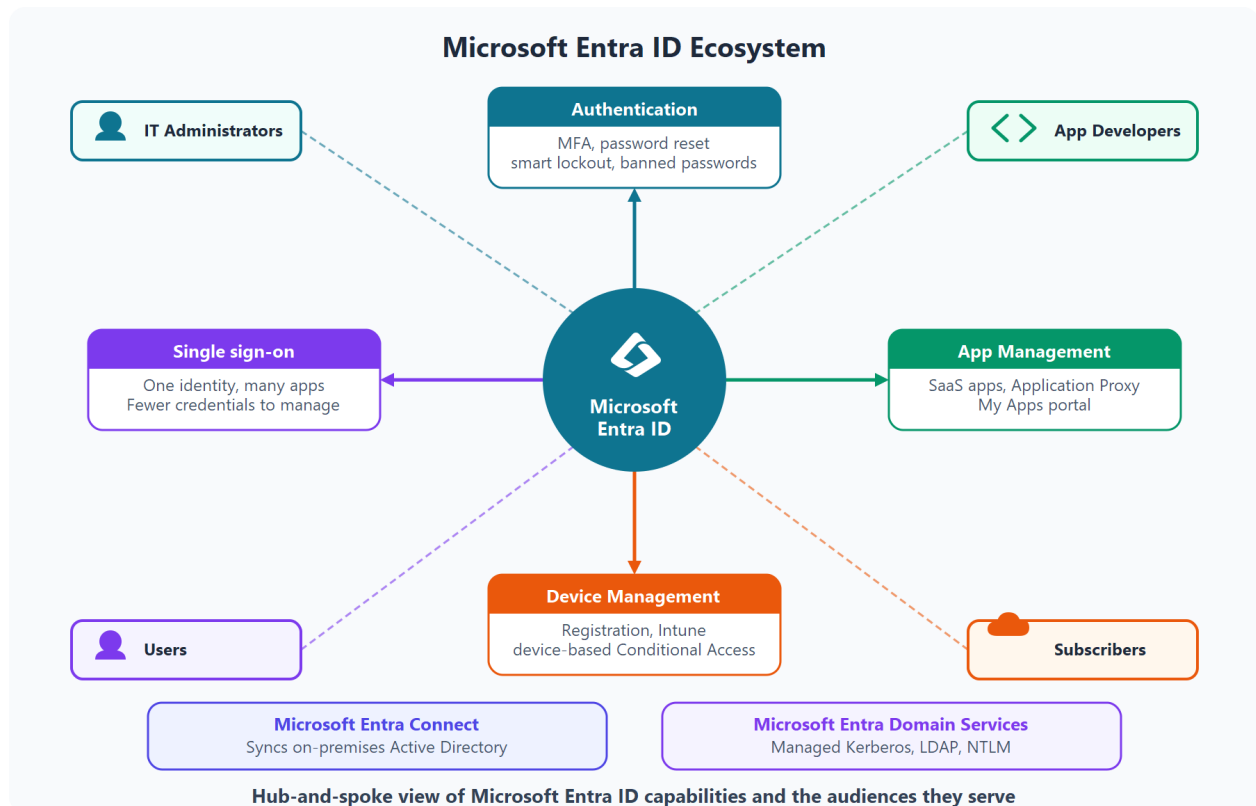
- **IT administrators.** Administrators can use Microsoft Entra ID to control access to applications and resources based on workload and security requirements.
- **App developers.** Developers can use Microsoft Entra ID to provide a standards-based approach for adding functionality to applications that they build, such as adding SSO functionality to an app or enabling an app to work with a user's existing credentials.
- **Users.** Users can manage their identities and take maintenance actions like self-service password reset.
- **Online service subscribers.** Microsoft 365, Microsoft Office 365, Azure, and Microsoft Dynamics CRM Online subscribers are already using Microsoft Entra ID to authenticate into their account.

What does Microsoft Entra ID do?

Microsoft Entra ID provides services such as:

- **Authentication** — Verifies identity before granting access. Includes self-service password reset, multifactor authentication, banned password lists, and smart logout.
- **Single sign-on (SSO)** — Lets one identity access multiple applications. SSO benefits and behavior are covered in the authentication methods unit.
- **Application management** — Manages cloud and on-premises apps through features like Application Proxy, SaaS app integration, and the My Apps portal.

- **Device management** — Supports device registration and management through tools like Microsoft Intune. Enables device-based Conditional Access policies that restrict access to known devices.



Can I connect my on-premises AD with Microsoft Entra ID?

Without a connection, an on-premises Active Directory deployment and a cloud Microsoft Entra ID deployment require you to maintain two separate identity sets. Microsoft Entra Connect bridges that gap.

Microsoft Entra Connect synchronizes user identities between on-premises Active Directory and Microsoft Entra ID. Because changes flow between both systems, users get a consistent experience — including SSO, multifactor authentication, and self-service password reset — whether they're accessing on-premises or cloud resources.

What is Microsoft Entra Domain Services?

Microsoft Entra Domain Services provides managed domain services — domain join, group policy, LDAP, and Kerberos/NTLM authentication — without requiring you to deploy or maintain domain controllers in the cloud.

This is especially useful for legacy applications that can't use modern authentication. You can lift and shift those applications from on-premises into a managed domain without managing an AD DS environment in the cloud.

Because Microsoft Entra Domain Services integrate with your existing Microsoft Entra tenant, users can sign in to the managed domain with their existing credentials. Existing groups and user accounts also carry over, providing a smoother migration path.

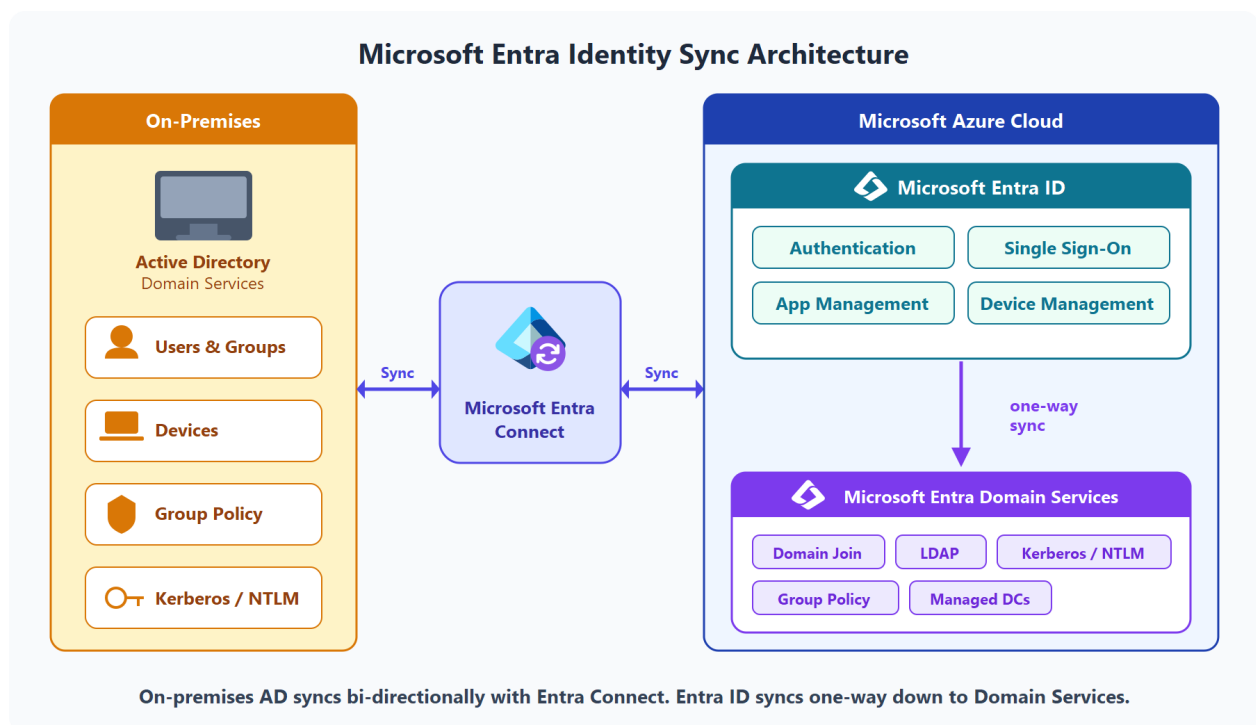
How does Microsoft Entra Domain Services work?

When you create a Microsoft Entra Domain Services managed domain, you define a unique namespace. This namespace is the domain name. Two Windows Server domain controllers are then deployed into your selected Azure region. This deployment of DCs is known as a replica set.

You don't need to manage, configure, or update these DCs. The Azure platform handles the DCs as part of the managed domain, including backups and encryption at rest using Azure Disk Encryption.

Is information synchronized?

A managed domain is configured to perform a one-way synchronization from Microsoft Entra ID to Microsoft Entra Domain Services. You can create resources directly in the managed domain, but they aren't synchronized back to Microsoft Entra ID. In a hybrid environment with an on-premises AD DS environment, Microsoft Entra Connect synchronizes identity information with Microsoft Entra ID, which is then synchronized to the managed domain.



Applications, services, and VMs in Azure that connect to the managed domain can then use common Microsoft Entra Domain Services features such as domain join, group policy, LDAP, and Kerberos/NTLM authentication.

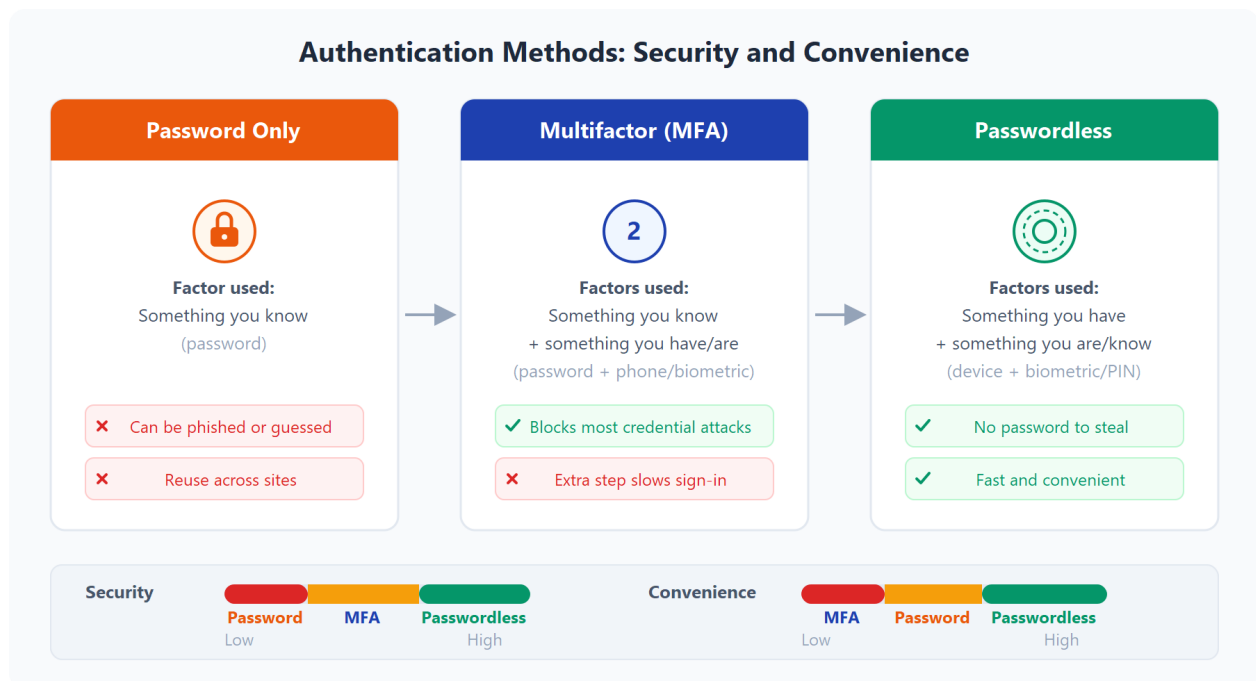
3. Describe Azure authentication methods

Describe Azure authentication methods

Completed

Authentication establishes the identity of a person, service, or device by requiring credentials. In Azure, common methods include passwords, single sign-on (SSO), multifactor authentication (MFA), and passwordless sign-in. Modern approaches are designed to improve both security and user convenience.

The following diagram shows the security level compared to the convenience. Notice Passwordless authentication is high security and high convenience while passwords on their own are low security but high convenience.



What's single sign-on?

Single sign-on (SSO) lets a user sign in once and access multiple trusted applications. SSO reduces password sprawl, which lowers the chance of credential-related incidents and decreases account lockout and reset overhead.

From an operations perspective, SSO also simplifies lifecycle management. Access is tied to one identity, making it easier to update or remove access when roles change.

Important

Single sign-on is only as secure as the initial authenticator because the subsequent connections are all based on the security of the initial authenticator.

What's multifactor authentication?

Multifactor authentication (MFA) prompts a user for an extra factor during sign-in, so a compromised password alone isn't enough for access. These factors fall into three categories:

- **Something the user knows** — a password or challenge question.
- **Something the user has** — a code sent to a mobile phone.
- **Something the user is** — a biometric signal such as a fingerprint or face scan.

With MFA enabled, an attacker who obtains a password still needs the second factor — such as a phone prompt or biometric signal — to complete sign-in.

What's Microsoft Entra multifactor authentication?

Microsoft Entra multifactor authentication is a Microsoft service that provides multifactor authentication capabilities. Microsoft Entra multifactor authentication enables users to choose an additional form of authentication during sign-in, such as a phone call or mobile app notification.

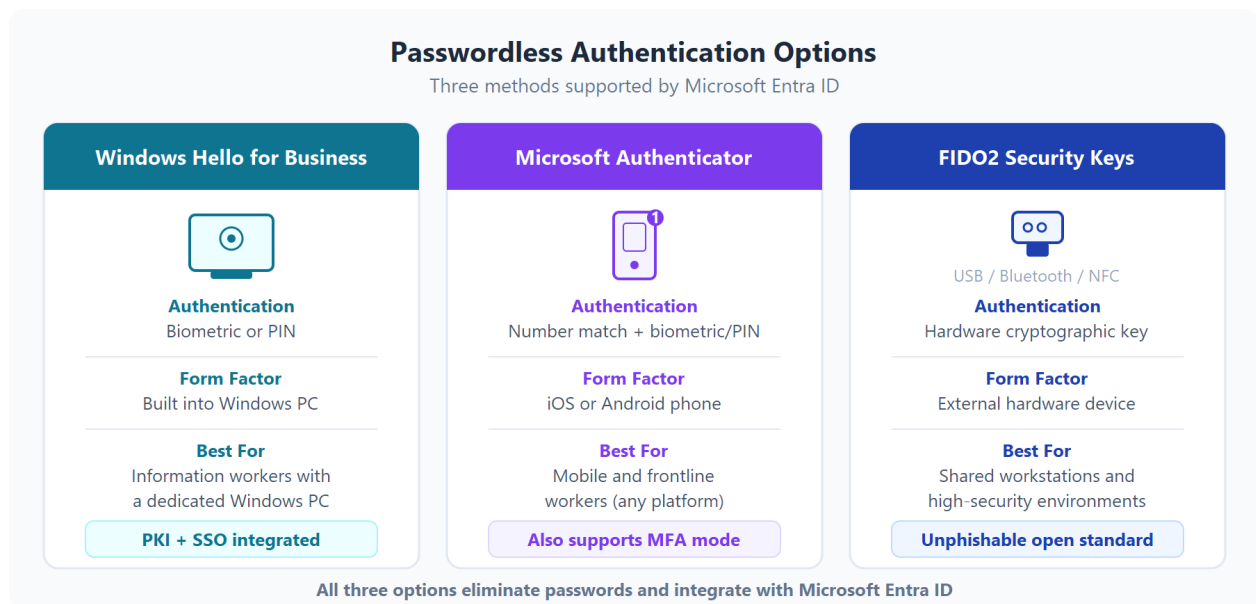
What's passwordless authentication?

While MFA adds security, passwords themselves remain a usability and risk challenge. Passwordless methods eliminate the password entirely, replacing it with a trusted device plus a biometric signal or PIN.

After initial registration, the user signs in with a factor they know or are — such as a PIN or fingerprint — instead of typing a password.

Microsoft Entra ID supports three passwordless options:

- Windows Hello for Business
- Microsoft Authenticator app
- FIDO2 security keys



Windows Hello for Business

Windows Hello for Business is ideal for information workers that have their own designated Windows PC. The biometric and PIN credentials are directly tied to the user's PC, which prevents access from anyone other than the owner. With public key infrastructure (PKI) integration and built-in support for single sign-on (SSO), Windows Hello for Business provides a convenient method for seamlessly accessing work resources on-premises and in the cloud.

Microsoft Authenticator app

The Microsoft Authenticator app can also serve as a passwordless credential, turning any iOS or Android phone into a strong sign-in factor.

To sign in, the user receives a notification on their phone, matches a number displayed on screen, and confirms with a biometric signal (touch or face) or PIN. No password is needed.

FIDO2 security keys

FIDO2 is an open standard for passwordless authentication built on the web authentication (WebAuthn) specification. FIDO2 security keys are unphishable hardware devices — typically USB, but also available with Bluetooth or NFC — that handle authentication without a username or password.

Users register a FIDO2 key and then select it at the sign-in screen as their primary authentication method. Because the hardware device handles authentication, there's no password that could be exposed or guessed.

4. Describe Azure external identities

Describe Azure external identities

Completed

An external identity is a person, device, or service that exists outside your tenant. Microsoft Entra External ID includes the capabilities used to securely interact with users beyond your tenant boundary.

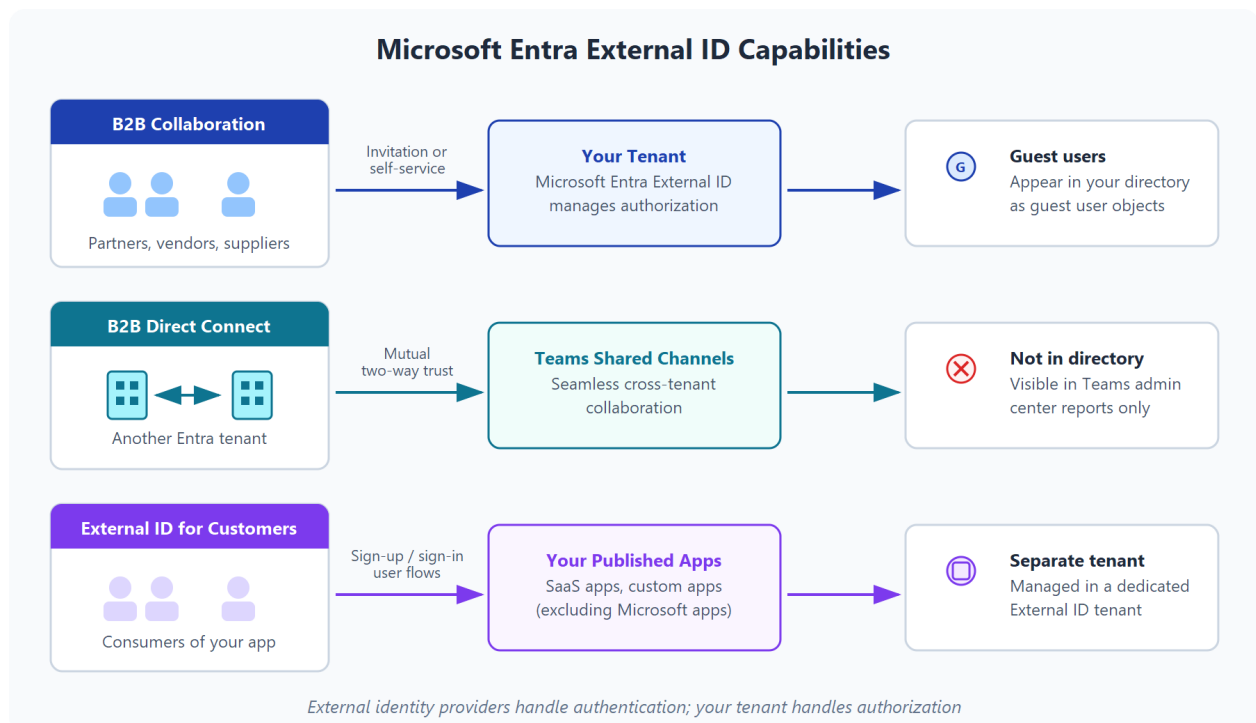
Why external identities matter

Organizations often need to collaborate with partners, suppliers, vendors, and contractors. External identities let those users access approved resources by using their existing credentials, while your team still enforces access policies.

For example, a project team can invite a supplier's users into specific collaboration spaces without creating and managing separate local credentials for each guest.

External identities can sound similar to single sign-on, but they're used for cross-tenant and consumer access scenarios. External users can bring their own identities, whether those are work accounts, government-issued digital identities, or social identities such as Google or Facebook.

The external identity provider manages authentication, and you manage authorization to your applications with Microsoft Entra ID or Azure AD B2C.



External ID capabilities

The following capabilities make up External Identities:

- **B2B collaboration** - Collaborate with external users by letting them use their preferred identity to sign in to your Microsoft applications or other internal applications (SaaS apps, custom-developed apps, etc.). B2B collaboration users are represented in your directory, typically as guest users.
- **B2B direct connect** - Establish a mutual, two-way trust with another Microsoft Entra tenant for seamless collaboration. B2B direct connect currently supports Teams shared channels, enabling external users to access your resources from within their home instances of Teams. B2B direct connect users aren't represented in your directory, but they're visible from within the Teams shared channel and can be monitored in Teams admin center reports.
- **Microsoft Entra External ID for customers (formerly Azure AD B2C)** - Publish modern SaaS apps or custom-developed apps (excluding Microsoft apps) to consumers and customers, while using Entra External ID for identity and access management.

External ID Capabilities at a Glance		
B2B Collaboration	B2B Direct Connect	External ID for Customers
WHO Partners, vendors, suppliers using work or social accounts	WHO Users from another Microsoft Entra tenant	WHO Consumers of your published SaaS or custom app
HOW Invitation or self-service sign-up into your tenant	HOW Mutual two-way trust between Entra tenants	HOW Sign-up and sign-in user flows with custom policies
DIRECTORY PRESENCE ✓ Represented as guest user objects	DIRECTORY PRESENCE ✗ Not in your directory; visible in Teams admin	DIRECTORY PRESENCE ✓ Managed in a separate External ID tenant
BEST FOR Sharing internal apps, SaaS apps, or custom apps	BEST FOR Teams shared channels across organizations	BEST FOR Consumer-facing apps (excluding Microsoft apps)

External providers authenticate; Microsoft Entra ID authorizes access to your resources

Depending on your collaboration goals and the resources being shared, you can combine these capabilities.

Manage guest access over time

With Microsoft Entra ID, you can enable collaboration across tenant boundaries by using B2B features. Guest users from other tenants can be invited by administrators or authorized users. This capability also applies to social identities such as Microsoft accounts.

You also can easily ensure that guest users have appropriate access. You can ask the guests themselves or a decision maker to participate in an access review and recertify (or attest) to the guests' access. The reviewers can give their input on each user's need for continued access, based on

suggestions from Microsoft Entra ID. When an access review is finished, you can then make changes and remove access for guests who no longer need it.

5. Describe Azure conditional access

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/5-conditional-access>

Describe Azure conditional access

Completed

Conditional Access is a tool that Microsoft Entra ID uses to allow (or deny) access to resources based on identity signals. These signals include who the user is, where the user is, and what device the user is requesting access from.

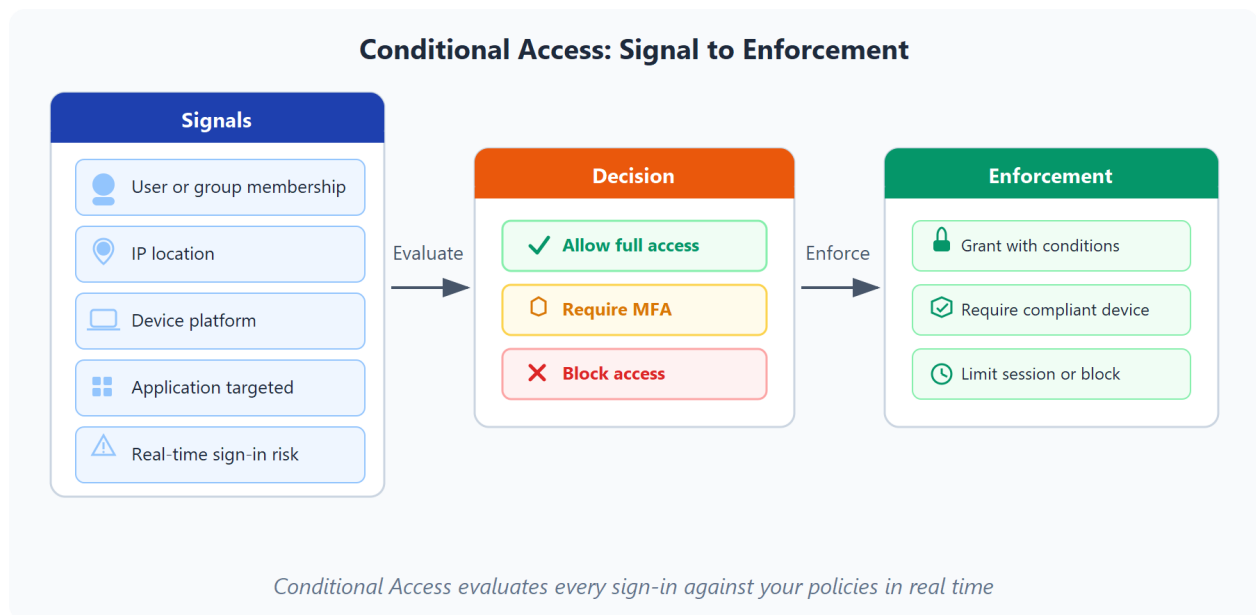
Conditional Access helps IT administrators:

- Empower users to be productive wherever and whenever.
- Protect critical assets.

Conditional Access also provides a more granular multifactor authentication experience for users. For example, a user might not be challenged for a second authentication factor if they're at a known location. However, they might be challenged for a second authentication factor if their sign-in signals are unusual or they're at an unexpected location.

During sign-in, Conditional Access collects signals from the user, makes decisions based on those signals, and then enforces that decision by allowing or denying the access request or challenging for a multifactor authentication response.

The following diagram illustrates this flow:



Here, the signal might be the user's location, the user's device, or the application that the user is trying to access.

Based on these signals, the decision might be to allow full access if the user is signing in from their usual location. If the user is signing in from an unusual location or a location that's marked as high risk, then access might be blocked entirely or possibly granted after the user provides a second form of authentication.

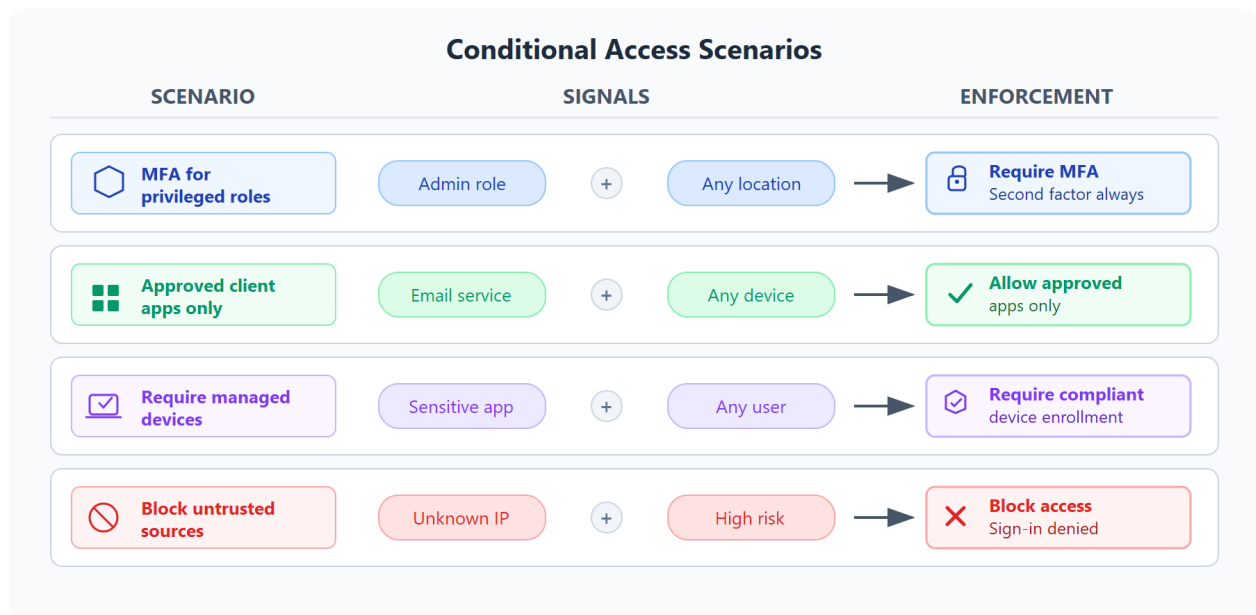
Enforcement is the action that carries out the decision. For example, the action is to allow access or require the user to provide a second form of authentication.

When can I use Conditional Access?

Conditional Access is useful when you need to:

- Require multifactor authentication (MFA) to access an application depending on the requester's role, location, or network. For example, you could require MFA for administrators, or for people connecting from outside trusted network locations.
- Require access to services only through approved client applications. For example, you could limit which email applications are able to connect to your email service.
- Require users to access your application only from managed devices. A managed device is a device that meets your standards for security and compliance.
- Block access from untrusted sources, such as access from unknown or unexpected locations.

The following diagram shows how these common scenarios map signals to enforcement actions:



Common IT tasks include requiring MFA for privileged roles, restricting access to sensitive apps from unmanaged devices, and blocking risky sign-ins from unexpected locations.

6. Describe Azure role-based access control

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/6-role-based-access-control>

Describe Azure role-based access control

Completed

When you have multiple IT and engineering teams, how can you control what access they have to the resources in your cloud environment? The principle of least privilege says you should only grant access up to the level needed to complete a task. If you only need read access to a storage blob, then you should only be granted read access to that storage blob — not write access, and not access to other blobs. It's a good security practice to follow.

However, managing that level of permissions for an entire team would become tedious. Instead of defining the detailed access requirements for each individual, and then updating access requirements when new resources are created or new people join the team, Azure enables you to control access through Azure role-based access control (Azure RBAC).

Azure provides built-in roles that describe common access rules for cloud resources. You can also define your own roles. Each role has an associated set of access permissions that relate to that role.

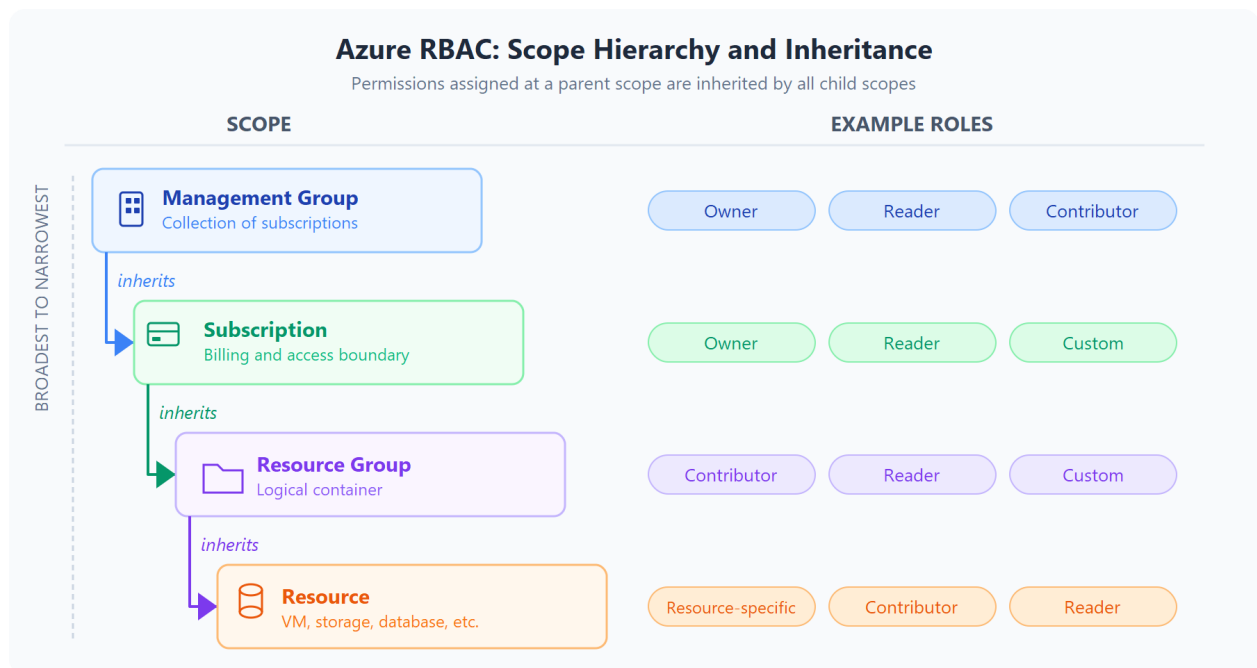
When you assign individuals or groups to one or more roles, they receive all the associated access permissions.

So, if you hire a new engineer and add them to the Azure RBAC group for engineers, they automatically get the same access as the other engineers in the same Azure RBAC group. Similarly, if you add additional resources and point Azure RBAC at them, everyone in that Azure RBAC group will now have those permissions on the new resources as well as the existing resources.

How is role-based access control applied to resources?

Role-based access control is applied to a scope, which is a resource or set of resources that this access applies to.

The following diagram shows the relationship between roles and scopes. A management group, subscription, or resource group might be given the role of owner, so they have increased control and authority. An observer, who isn't expected to make any updates, might be given a role of Reader for the same scope, enabling them to review or observe the management group, subscription, or resource group.



Scopes include:

- A management group (a collection of multiple subscriptions).
- A single subscription.
- A resource group.
- A single resource.

Observers, users managing resources, admins, and automated processes illustrate the kinds of users or accounts that would typically be assigned each of the various roles.

Azure RBAC is hierarchical, in that when you grant access at a parent scope, those permissions are inherited by all child scopes. For example:

- When you assign the Owner role to a user at the management group scope, that user can manage everything in all subscriptions within the management group.
- When you assign the Reader role to a group at the subscription scope, the members of that group can view every resource group and resource within the subscription.

How is Azure RBAC enforced?

Azure RBAC is enforced on any action that's initiated against an Azure resource that passes through Azure Resource Manager. Resource Manager is a management service that provides a way to organize and secure your cloud resources.

You typically access Resource Manager from the Azure portal, Azure Cloud Shell, Azure PowerShell, and the Azure CLI. Azure RBAC doesn't enforce access permissions at the application or data level. Application security must be handled by your application.

Azure RBAC uses an allow model. When you're assigned a role, Azure RBAC allows you to perform actions within the scope of that role. If one role assignment grants you read permissions to a resource group and a different role assignment grants you write permissions to the same resource group, you have both read and write permissions on that resource group.

7. Describe Zero Trust model

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/7-describe-zero-trust-model>

Describe Zero Trust model

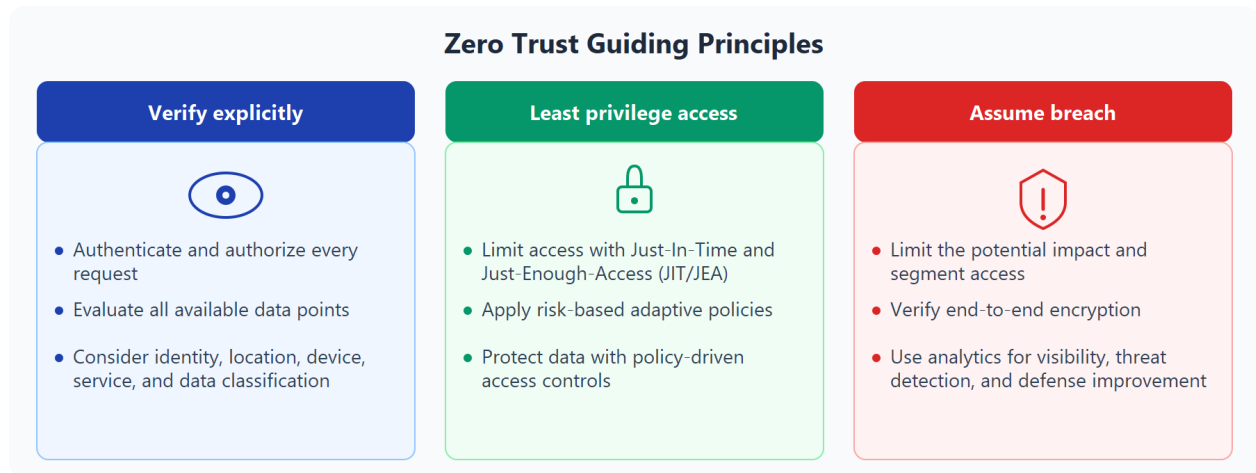
Completed

Zero Trust is a security model that assumes the worst case scenario and protects resources with that expectation. Zero Trust assumes breach at the outset, and then verifies each request as though it originated from an uncontrolled network.

Today, IT teams need a security model that effectively adapts to the complexity of the modern environment; embraces the mobile workforce; and protects people, devices, applications, and data wherever they're located.

To address this new world of computing, Microsoft highly recommends the Zero Trust security model, which is based on these guiding principles:

- **Verify explicitly** - Always authenticate and authorize based on all available data points.
- **Use least privilege access** - Limit user access with Just-In-Time and Just-Enough-Access (JIT/JEA), risk-based adaptive policies, and data protection.
- **Assume breach** - Limit the potential impact and segment access. Verify end-to-end encryption. Use analytics to get visibility, drive threat detection, and improve defenses.



Adjusting to Zero Trust

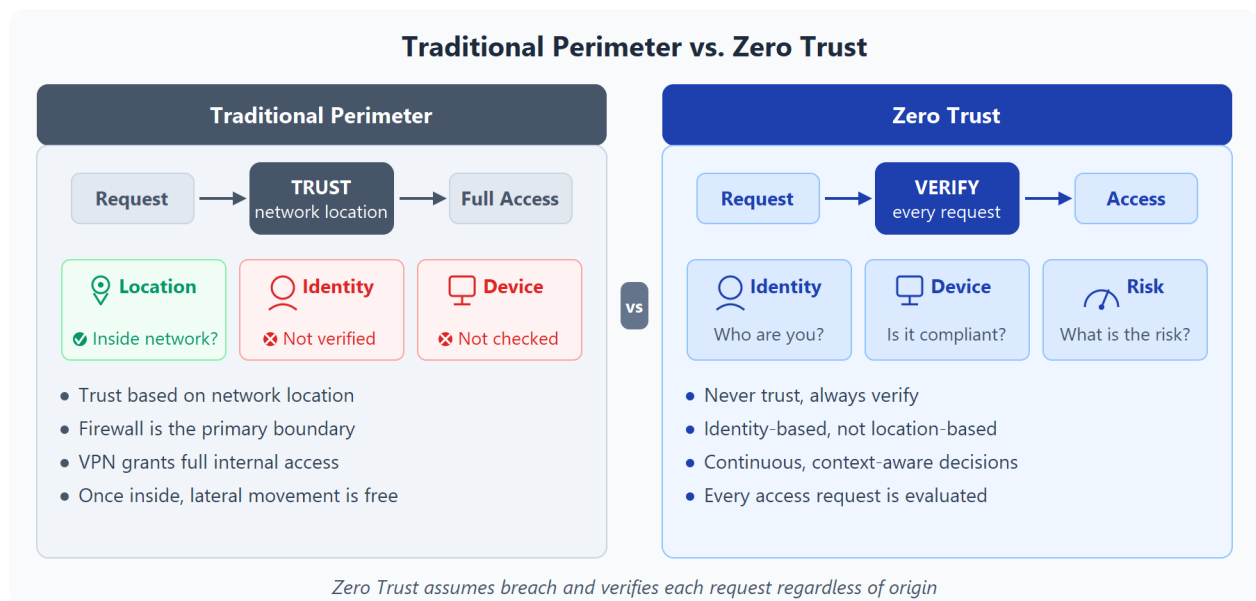
Traditionally, perimeter-based networks were restricted, protected, and generally assumed safe. Only managed computers could join the network, VPN access was tightly controlled, and personal devices were frequently restricted or blocked.

The Zero Trust model flips that scenario. Instead of assuming that a device is safe because it's within a trusted internal network, it requires everyone to authenticate. Then grants access based on authentication rather than location.

Example in practice

Suppose a user signs in from an unmanaged device on a public network. A Zero Trust approach still evaluates identity signals, device state, and risk conditions before granting access. You might allow access to low-risk apps, require MFA for sensitive systems, or block sign-in entirely when risk is too high.

At a fundamentals level, Zero Trust means access decisions are continuous and context-aware, not based only on where the request originates.



8. Describe defense-in-depth

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/8-describe-defense-depth>

Describe defense-in-depth

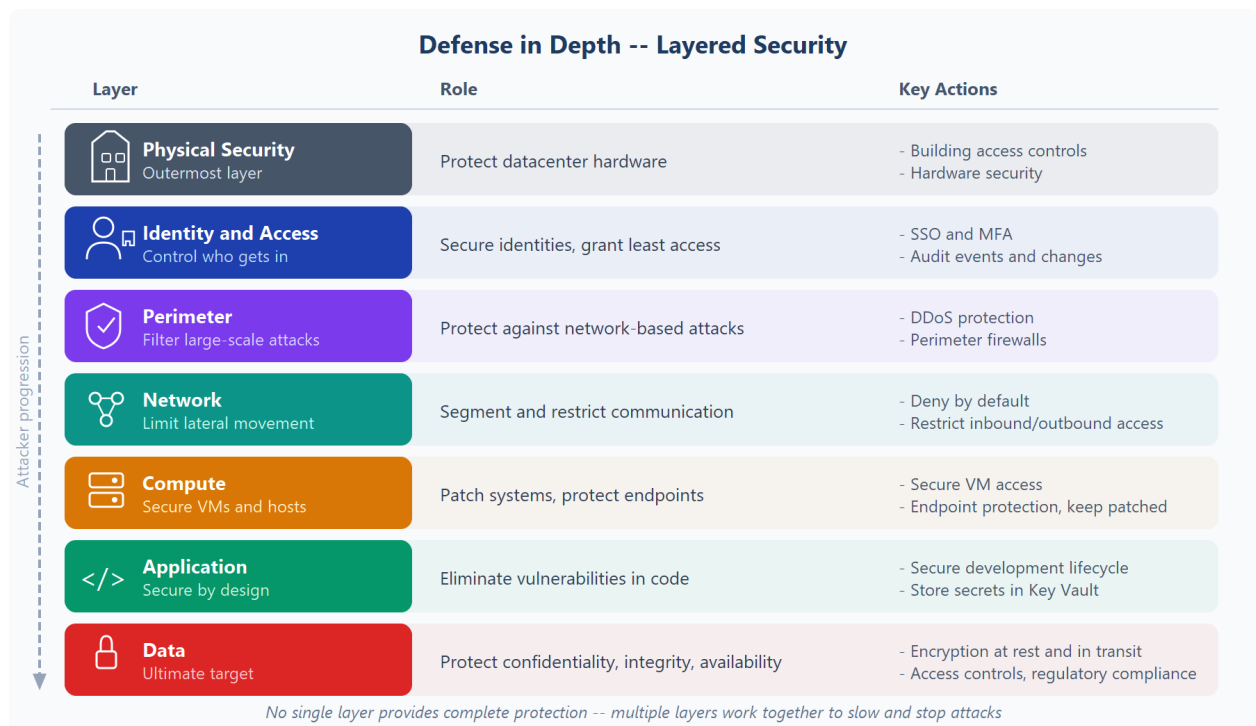
Completed

The objective of defense-in-depth is to protect information and prevent it from being stolen by those who aren't authorized to access it.

A defense-in-depth strategy uses a series of mechanisms to slow the advance of an attack that aims at acquiring unauthorized access to data.

Layers of defense-in-depth

You can visualize defense-in-depth as a set of layers, with the data to be secured at the center and all the other layers functioning to protect that central data layer.



Each layer provides protection so that if one layer is breached, a subsequent layer is already in place to prevent further exposure. This approach removes reliance on any single layer of protection. It slows down an attack and provides alert information that security teams can act upon, either automatically or manually.

Here's a brief overview of the role of each layer:

- The physical security layer is the first line of defense to protect computing hardware in the datacenter.
- The identity and access layer controls access to infrastructure and change control.
- The perimeter layer uses distributed denial of service (DDoS) protection to filter large-scale attacks before they can cause a denial of service for users.
- The network layer limits communication between resources through segmentation and access controls.
- The compute layer secures access to virtual machines.
- The application layer helps ensure that applications are secure and free of security vulnerabilities.
- The data layer controls access to operational and customer data that you need to protect.

These layers provide a guideline for you to help make security configuration decisions in all of the layers of your applications.

Azure provides security tools and features at every level of the defense-in-depth concept. Let's take a closer look at each layer:

Physical security

Physically securing access to buildings and controlling access to computing hardware within the datacenter are the first line of defense.

With physical security, the intent is to provide physical safeguards against access to assets. These safeguards ensure that other layers can't be bypassed, and loss or theft is handled appropriately. Microsoft uses various physical security mechanisms in its cloud datacenters.

Identity and access

The identity and access layer is all about ensuring that identities are secure, that access is granted only to what's needed, and that sign-in events and changes are logged.

At this layer, it's important to:

- Control access to infrastructure and change control.
- Use single sign-on (SSO) and multifactor authentication.
- Audit events and changes.

Perimeter

The network perimeter protects from network-based attacks against your resources. Identifying these attacks, eliminating their impact, and alerting you when they happen are important ways to keep your network secure.

At this layer, it's important to:

- Use DDoS protection to filter large-scale attacks before they can affect the availability of a system for users.
- Use perimeter firewalls to identify and alert on malicious attacks against your network.

Network

At this layer, the focus is on limiting the network connectivity across all your resources to allow only what's required. By limiting this communication, you reduce the risk of an attack spreading to other systems in your network.

At this layer, it's important to:

- Limit communication between resources.
- Deny by default.
- Restrict inbound internet access and limit outbound access where appropriate.
- Implement secure connectivity to on-premises networks.

Compute

Malware, unpatched systems, and improperly secured systems open your environment to attacks. The focus in this layer is on making sure that your compute resources are secure and that you have the proper controls in place to minimize security issues.

At this layer, it's important to:

- Secure access to virtual machines.
- Implement endpoint protection on devices and keep systems patched and current.

Application

Integrating security into the application development lifecycle helps reduce the number of vulnerabilities introduced in code. Every development team should ensure that its applications are secure by default.

At this layer, it's important to:

- Ensure that applications are secure and free of vulnerabilities.
- Store sensitive application secrets in a secure storage medium, such as Azure Key Vault.
- Make security a design requirement for all application development.

Data

Those who store and control access to data are responsible for ensuring that it's properly secured. Often, regulatory requirements dictate the controls and processes that must be in place to ensure the confidentiality, integrity, and availability of the data.

At this layer, it's also important to protect data with encryption at rest and encryption in transit. Azure services provide encryption capabilities, and Azure Key Vault can help secure and manage the keys and secrets used by your applications.

In almost all cases, attackers are after data:

- Stored in a database.
- Stored on disk inside virtual machines.
- Stored in software as a service (SaaS) applications, such as Office 365.
- Managed through cloud storage.

9. Describe encryption and key management in Azure

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/9a-describe-encryption-key-management>

Describe encryption and key management in Azure

Completed

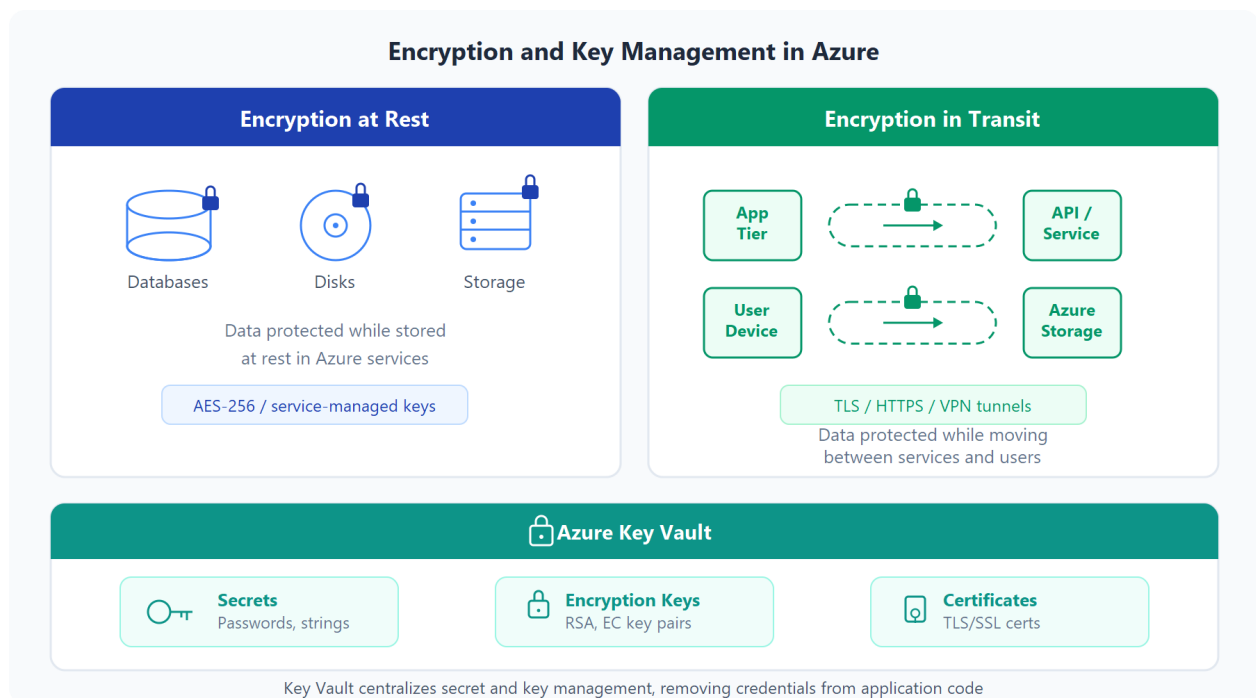
Encryption helps protect data confidentiality by making data unreadable to unauthorized users.

Encryption at rest and in transit

In Azure, encryption is commonly discussed in two forms:

- Encryption at rest protects data when it is stored, such as in databases, disks, and storage accounts.
- Encryption in transit protects data while it moves between services, applications, and users.

A strong security posture generally includes both.



Practical example

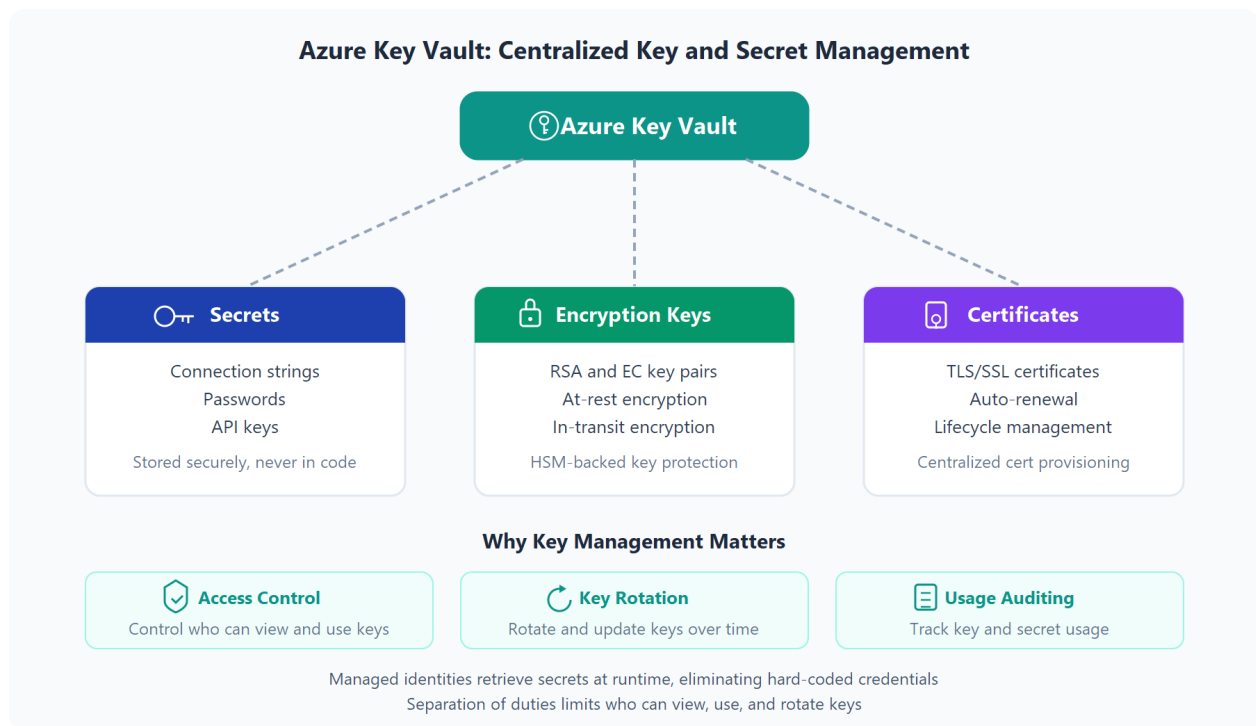
Consider a line-of-business application that stores customer records in Azure Storage and Azure SQL. Data should be encrypted while stored in those services and while moving between application tiers, APIs, and user devices.

Key management with Azure Key Vault

Azure Key Vault is a service for securely storing and controlling access to:

- Secrets (such as connection strings and passwords)
- Encryption keys
- Certificates

Using Key Vault helps centralize secret and key management instead of storing sensitive values directly in application code or configuration files.



Why key management matters

Key management supports security and compliance goals by helping you:

- Control who can access keys and secrets
- Rotate and update keys over time
- Audit key and secret usage

You can also separate duties by limiting who can view, use, and rotate keys. This approach helps reduce risk and supports compliance reporting.

Another common practice is to set key rotation policies and alerting so teams are notified before keys expire. This helps avoid service disruption while maintaining secure key hygiene.

When applications retrieve secrets from Key Vault at runtime by using managed identities, teams can reduce hard-coded credentials and improve overall secret governance.

At a fundamentals level, remember that Azure Key Vault is the primary Azure service for secure key and secret storage.

10. Describe Microsoft Defender for Cloud

Describe Microsoft Defender for Cloud

Completed

Defender for Cloud is a security posture management and threat protection service. It monitors cloud, on-premises, hybrid, and multicloud resources and provides recommendations and alerts to improve security posture.

Defender for Cloud helps you harden resources, track risk, and respond to threats. In Azure, it's natively integrated and can be enabled quickly.

Coverage across cloud and hybrid environments

Because Defender for Cloud is Azure-native, many Azure services can be monitored without extra deployment. For hybrid and multicloud estates, Defender for Cloud extends visibility so security teams can work from one control plane.

When needed, Defender for Cloud can deploy data collection components for security insights. Azure Arc can extend Microsoft Defender plans to non-Azure machines, and Cloud Security Posture Management (CSPM) capabilities can assess multicloud resources agentlessly.

Azure-native protections

Defender for Cloud helps you detect threats across:

- Azure PaaS services - Detects threats across services such as App Service, Azure SQL, and Azure Storage.
- Azure data services - Provides data security assessments and recommendations for services such as Azure SQL and Storage.
- Networks - Helps reduce brute-force exposure through controls such as just-in-time VM access and restrictive port policies.

Defend your hybrid resources

In addition to Azure coverage, Defender for Cloud can protect non-Azure servers in hybrid environments and prioritize alerts based on your environment.

To extend protection to on-premises machines, deploy Azure Arc and enable Defender for Cloud's enhanced security features.

Defend resources running on other clouds

Defender for Cloud can also protect resources in other clouds, including AWS and GCP.

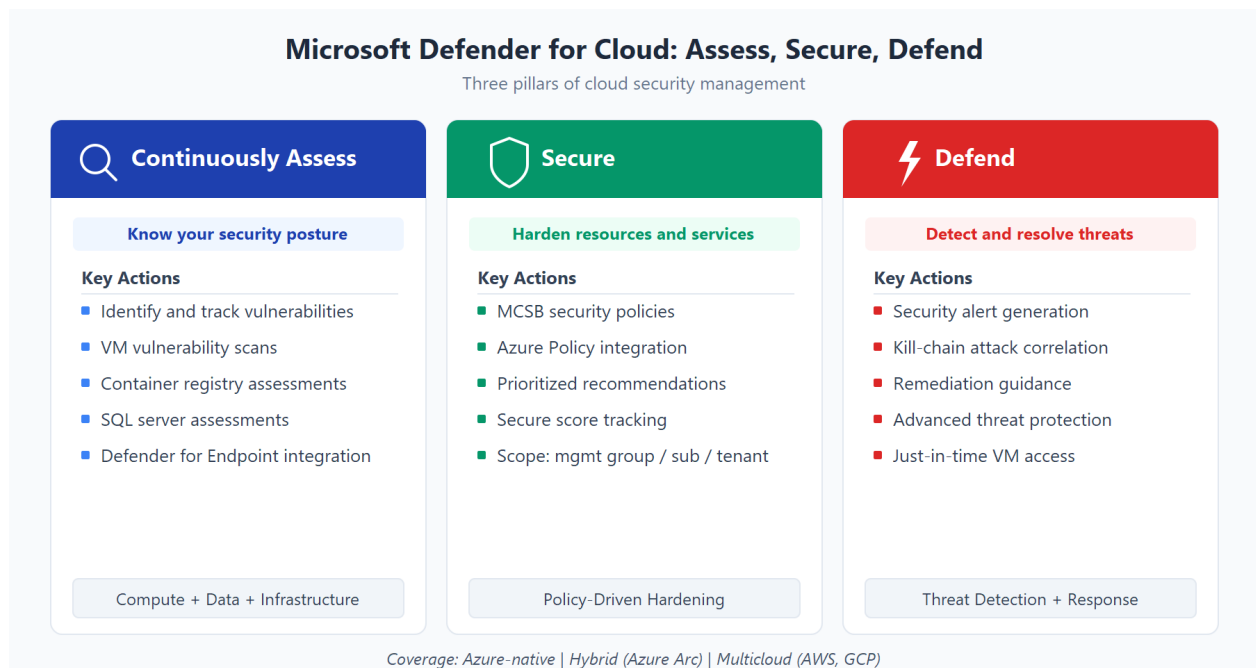
For connected AWS environments, Defender for Cloud can:

- Extend CSPM recommendations and compliance assessments to AWS resources.
- Extend Defender for Containers protections to Amazon EKS clusters.
- Extend Defender for Servers protections to EC2 instances.

Assess, Secure, and Defend

Defender for Cloud fills three vital needs as you manage the security of your resources and workloads in the cloud and on-premises:

- Continuously assess – Know your security posture. Identify and track vulnerabilities.
- Secure – Harden resources and services with Microsoft cloud security benchmark (MCSB).
- Defend – Detect and resolve threats to resources, workloads, and services.



Continuously assess

Defender for Cloud helps you continuously assess your environment. It includes vulnerability assessment for virtual machines, container registries, and SQL servers.

Microsoft Defender for servers includes automatic, native integration with Microsoft Defender for Endpoint. With this integration enabled, you'll have access to the vulnerability findings from Microsoft Defender Vulnerability Management.

Together, these tools provide regular vulnerability visibility across compute, data, and infrastructure from a unified experience.

Secure

To secure workloads, you need policies that align to your environment. Defender for Cloud builds on Azure Policy, so controls can be scoped at management group, subscription, or tenant levels.

As new resources are deployed, Defender for Cloud evaluates them against best practices and surfaces prioritized recommendations. These recommendations align to Microsoft cloud security benchmark (MCSB) guidance.

Defender for Cloud groups recommendations into security controls and calculates a secure score so teams can quickly understand posture and prioritize improvements.

Defend

The first two areas focused on assessing, monitoring, and maintaining your environment. Defender for Cloud also defends your environment by providing security alerts and advanced threat protection features.

Security alerts

When Defender for Cloud detects a threat in any area of your environment, it generates a security alert. Security alerts:

- Describe details of the affected resources
- Suggest remediation steps
- Provide, in some cases, an option to trigger a logic app in response

Whether an alert is generated by Defender for Cloud or received from an integrated security product, you can export it. Defender for Cloud also uses kill-chain analysis to automatically correlate related alerts, helping you see the full story of an attack — where it started, which resources were affected, and what impact it had.

Advanced threat protection

Defender for Cloud provides advanced threat protection for resources such as virtual machines, SQL databases, containers, web applications, and networks. Protections include just-in-time VM access and adaptive application controls.

11. Module assessment

Module assessment

Completed

12. Summary

<https://learn.microsoft.com/en-us/training/modules/describe-azure-identity-access-security/11-summary>

Summary

Completed

In this module, you learned about Azure identity, access, and security services and tools. You covered authentication methods, including which ones are more secure. You learned about restricting access based on a role to help create a more secure environment. You also reviewed encryption concepts and key management options in Azure. And, you learned about the Defense In Depth and Zero Trust models.

Learning objectives

You should now be able to:

- Describe directory services in Azure, including Microsoft Entra ID and Microsoft Entra Domain Services.
- Describe authentication methods in Azure, including single sign-on (SSO), multifactor authentication (MFA), and passwordless.
- Describe external identities and guest access in Azure.
- Describe Microsoft Entra Conditional Access.
- Describe Azure Role Based Access Control (RBAC).
- Describe the concept of Zero Trust.
- Describe the purpose of the defense in depth model.
- Describe encryption concepts and key management options in Azure.
- Describe the purpose of Microsoft Defender for Cloud.

Additional resources

The following resources provide more information on topics in this module or related to this module. [Microsoft Certified: Security, Compliance, and Identity Fundamentals](#) is an entire certification, with associated training, dedicated to helping you better understand and manage Security, Compliance, and identity.

Explore with Copilot

Tip

Try one of these prompts in Copilot Chat:

- "Use one end-to-end scenario to show how SSO, MFA, Conditional Access, and RBAC work together in a Zero Trust design."
- "Explain the difference between Microsoft Entra ID, Microsoft Entra Domain Services, and external identities with practical examples."
- "Simulate a security incident and show how encryption, key management, defense in depth, and Microsoft Defender for Cloud reduce risk."