

Azure security fundamentals documentation

Learn how to secure your cloud solutions on Azure with our best practices and guidance.

About Azure security

OVERVIEW

[Introduction to Azure security](#)

[Security technical capabilities](#)

CONCEPT

[How Microsoft secures the Azure infrastructure](#)

Get started

OVERVIEW

[Shared responsibility in the cloud](#)

[Security services and technologies](#)

[Built-in security controls](#)

Mitigate threats

OVERVIEW

[Microsoft Defender for Cloud](#)

CONCEPT

[Management and monitoring](#)

[Threat protection](#)

[Recover from identity compromise](#)

Best practices for securing your cloud solutions

CONCEPT

[Network security](#)

[IaaS workloads](#)

[Identity management and access control](#)

[PaaS deployments](#)

[Data security and encryption](#)

[Operational security](#)

Protect your Azure resources

CONCEPT

[Encryption at rest](#)

[Data protection](#)

[Network security](#)

[Virtual machines security](#)

[Identity management security](#)

TRAINING

[Secure your cloud applications](#)

Build your security skills

TRAINING

[Implement network security](#)

[Manage identity and access](#)

[Implement resource management security](#)

Implement virtual machine host security

Introduction to Azure security

Security is critical in today's cloud environment. Cyber threats are constantly evolving, and protecting your data, applications, and infrastructure requires a comprehensive, multi-layered approach. We know that security is job one in the cloud and how important it is that you find accurate and timely information about Azure security.

This article provides a comprehensive look at the security available with Azure. For an end-to-end view of Azure security organized by protection, detection, and response capabilities, see [End-to-end security in Azure](#).

Azure's defense-in-depth security approach

Azure employs a defense-in-depth strategy, providing multiple layers of security protection across the entire stack—from physical datacenters to compute, storage, networking, applications, and identity. This multi-layered approach ensures that if one layer is compromised, additional layers continue to protect your resources.

Azure's infrastructure is meticulously crafted from the ground up, encompassing everything from physical facilities to applications, to securely host millions of customers simultaneously. This robust foundation empowers businesses to confidently meet their security requirements. For information on how Microsoft secures the Azure platform itself, see [Azure infrastructure security](#). For details on physical datacenter security, see [Azure physical security](#).

Azure is a public cloud service platform that supports a broad selection of operating systems, programming languages, frameworks, tools, databases, and devices. It can run Linux containers with Docker integration; build apps with JavaScript, Python, .NET, PHP, Java, and Node.js; build back-ends for iOS, Android, and Windows devices. Azure public cloud services support the same technologies millions of developers and IT professionals already rely on and trust.

Built-in platform security

Azure provides default security protections built into the platform that help protect your resources from the moment they're deployed. For comprehensive information on Azure's platform security capabilities, see [Azure platform security overview](#).

- **Network Protection:** Azure DDoS Protection automatically shields your resources from distributed denial-of-service attacks
- **Encryption by Default:** Data encryption at rest is enabled by default for Azure Storage, SQL Database, and many other services

- **Identity Security:** Microsoft Entra ID provides secure authentication and authorization for all Azure services
- **Threat Detection:** Built-in threat detection monitors for suspicious activities across your Azure resources
- **Compliance:** Azure maintains the largest compliance portfolio in the industry, helping you meet regulatory requirements

These foundational security controls work continuously in the background to protect your cloud infrastructure, with no additional configuration required for basic protection.

Shared responsibility in the cloud

While Azure provides robust platform security, security in the cloud is a shared responsibility between Microsoft and our customers. The division of responsibilities depends on your deployment model (IaaS, PaaS, or SaaS):

- **Microsoft's responsibility:** Azure secures the underlying infrastructure, including physical datacenters, hardware, network infrastructure, and the host operating system
- **Your responsibility:** You're responsible for securing your data, applications, identities, and access management

Every workload and application is different, with unique security requirements based on industry regulations, data sensitivity, and business needs. This is where Azure's advanced security services come into play. For more information about the shared responsibility model, see [Shared responsibility in the cloud](#).

ⓘ Note

The primary focus of this document is on customer-facing controls that you can use to customize and increase security for your applications and services.

Advanced security services for every workload

To meet your unique security requirements, Azure provides a comprehensive suite of advanced security services that you can configure and customize for your specific needs. These services are organized across six functional areas: Operations, Applications, Storage, Networking, Compute, and Identity. For a comprehensive catalog of security services and technologies, see [Azure security services and technologies](#).

In addition, Azure provides you with a wide array of configurable security options and the ability to control them so that you can customize security to meet the unique requirements of

your organization's deployments. This document helps you understand how Azure security capabilities can help you fulfill these requirements.

For a structured view of Azure security controls and baselines, see the [Microsoft cloud security benchmark](#), which provides comprehensive security guidance for Azure services. For information on Azure's technical security capabilities, see [Azure security technical capabilities](#).

Compute security

Securing your virtual machines and compute resources is fundamental to protecting your workloads in Azure. Azure provides multiple layers of compute security, from hardware-based protections to software-based threat detection. For detailed virtual machine security information, see [Azure Virtual Machines security overview](#).

Trusted launch

[Trusted launch](#) is the default for newly created Generation 2 Azure VMs and Virtual Machine Scale Sets. Trusted launch protects against advanced and persistent attack techniques including boot kits, rootkits, and kernel-level malware.

Trusted launch provides:

- **Secure Boot:** Protects against installation of malware-based rootkits and boot kits by ensuring only signed operating systems and drivers can boot
- **vTPM (virtual Trusted Platform Module):** A dedicated secure vault for keys and measurements that enables attestation and boot integrity verification
- **Boot Integrity Monitoring:** Uses attestation through Microsoft Defender for Cloud to verify boot chain integrity and alert on failures

Trusted launch can be enabled on existing VMs and Virtual Machine Scale Sets.

Azure confidential computing

[Azure confidential computing](#) provides the final, missing piece, of the data protection puzzle. It allows you to keep your data encrypted always. While at rest, when in motion through the network, and now, even while loaded in memory and in use. Additionally, by making [Remote Attestation](#) possible, it allows you to cryptographically verify that the VM you deploy booted securely and is configured correctly, before unlocking your data.

The spectrum of options ranges from enabling "lift and shift" scenarios of existing applications, to full control of security features. For Infrastructure as a Service (IaaS), you can use:

- **Confidential virtual machines powered by AMD SEV-SNP:** Hardware-based memory encryption with up to 256 GB encrypted memory
- **Confidential VMs with Intel TDX:** Intel Trust Domain Extensions providing enhanced performance and security
- **Confidential VMs with NVIDIA H100 GPUs:** GPU-accelerated confidential computing for AI/ML workloads
- **Confidential application enclaves with Intel SGX:** Application-level isolation for sensitive code and data

For Platform as a Service (PaaS), Azure offers multiple [container-based confidential computing options](#), including integrations with [Azure Kubernetes Service \(AKS\)](#).

Antimalware & Antivirus

With Azure IaaS, you can use antimalware software from security vendors such as Microsoft, Symantec, Trend Micro, McAfee, and Kaspersky to protect your virtual machines from malicious files, adware, and other threats. [Microsoft Antimalware](#) for Azure Virtual Machines is a protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems. Microsoft Antimalware can also be deployed using Microsoft Defender for Cloud.

ⓘ Note

For modern protection, consider [Microsoft Defender for Servers](#) which provides advanced threat protection including endpoint detection and response (EDR) through integration with Microsoft Defender for Endpoint.

Hardware Security Module

Encryption and authentication don't improve security unless the keys themselves are protected. You can simplify the management and security of your critical secrets and keys by storing them in [Azure Key Vault](#). Key Vault provides the option to store your keys in hardware Security modules (HSMs) certified to [FIPS 140-3 Level 3](#) standards. Your SQL Server encryption keys for backup or [transparent data encryption](#) can all be stored in Key Vault with any keys or secrets from your applications. Permissions and access to these protected items are managed through [Microsoft Entra ID](#).

For comprehensive information on key management options including Azure Key Vault, Managed HSM, and Payment HSM, see [Key management in Azure](#).

Virtual machine backup

[Azure Backup](#) is a solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications that can lead to security issues. With Azure Backup, your virtual machines running Windows and Linux are protected.

Azure Site Recovery

An important part of your organization's [business continuity/disaster recovery \(BCDR\)](#) strategy is figuring out how to keep corporate workloads and apps up and running when planned and unplanned outages occur. [Azure Site Recovery](#) helps orchestrate replication, failover, and recovery of workloads and apps so that they're available from a secondary location if your primary location goes down.

SQL VM TDE

Transparent data encryption (TDE) and column level encryption (CLE) are SQL server encryption features. This form of encryption requires customers to manage and store the cryptographic keys you use for encryption.

The Azure Key Vault (AKV) service is designed to improve the security and management of these keys in a secure and highly available location. The SQL Server Connector enables SQL Server to use these keys from Azure Key Vault.

If you're running SQL Server with on-premises machines, there are steps you can follow to access Azure Key Vault from your on-premises SQL Server instance. But for SQL Server in Azure VMs, you can save time by using the Azure Key Vault Integration feature. With a few Azure PowerShell cmdlets to enable this feature, you can automate the configuration necessary for a SQL VM to access your key vault.

For a comprehensive list of database security best practices, see [Azure database security checklist](#).

VM Disk Encryption

Important

Azure Disk Encryption for Virtual Machines and Virtual Machine Scale Sets will be retired on **September 15, 2028**. New customers should use [encryption at host](#) for all new VMs. Existing customers should plan to migrate current ADE-enabled VMs to encryption at host

before the retirement date to avoid service disruption -- see [Migrate from Azure Disk Encryption to encryption at host](#).

For modern virtual machine encryption, Azure offers:

- **Encryption at host:** Provides end-to-end encryption for VM data, including temp disks and OS/data disk caches
- **Confidential disk encryption:** Available with confidential VMs for hardware-based encryption
- **Server-side encryption with customer-managed keys:** Manage your own encryption keys through Azure Key Vault

For more information, see [Overview of managed disk encryption options](#).

Virtual networking

Virtual machines need network connectivity. To support that requirement, Azure requires virtual machines to be connected to an Azure Virtual Network. An Azure Virtual Network is a logical construct built on top of the physical Azure network fabric. Each logical [Azure Virtual Network](#) is isolated from all other Azure Virtual Networks. This isolation helps ensure that network traffic in your deployments isn't accessible to other Microsoft Azure customers.

Patch Updates

Patch Updates provide the basis for finding and fixing potential problems and simplify the software update management process, both by reducing the number of software updates you must deploy in your enterprise and by increasing your ability to monitor compliance.

Security policy management and reporting

[Defender for Cloud](#) helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated Security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Application security

Application security focuses on protecting your applications from threats throughout their lifecycle—from development to deployment and runtime. Azure provides comprehensive tools for secure development, testing, and protection of applications. For secure application

development guidance, see [Develop secure applications on Azure](#). For PaaS-specific security best practices, see [Securing PaaS deployments](#). For IaaS deployment security, see [Security best practices for IaaS workloads in Azure](#).

Penetration Testing

We don't perform [penetration testing](#) of your application for you, but we do understand that you want and need to perform testing on your own applications. Notification of Microsoft of pen testing activities is no longer required customers must still comply with the [Microsoft Cloud Penetration Testing Rules of Engagement](#) [↗](#).

Web Application firewall

The Web Application Firewall (WAF) in [Azure Application Gateway](#) provides protection for web applications against common web-based attacks such as SQL injection, cross-site scripting, and session hijacking. It is preconfigured to defend against the top 10 vulnerabilities identified by the [Open Web Application Security Project \(OWASP\)](#) [↗](#).

Authentication and authorization in Azure App Service

[App Service Authentication / Authorization](#) is a feature that provides a way for your application to sign in users so that you don't have to change code on the app backend. It provides an easy way to protect your application and work with per-user data.

Layered Security Architecture

Since [App Service Environments](#) provide an isolated runtime environment deployed into an [Azure Virtual Network](#), developers can create a layered security architecture providing differing levels of network access for each application tier. It's common to hide API back-ends from general Internet access, and only permit APIs to be called by upstream web apps. [Network Security groups \(NSGs\)](#) can be used on Azure Virtual Network subnets containing App Service Environments to restrict public access to API applications.

[App Service web apps](#) offer robust diagnostic capabilities for capturing logs from both the web server and the web application. These diagnostics are categorized into web server diagnostics and application diagnostics. Web server diagnostics include significant advancements for diagnosing and troubleshooting sites and applications.

The first new feature is real-time state information about application pools, worker processes, sites, application domains, and running requests. The second new advantages are the detailed trace events that track a request throughout the complete request-and-response process.

To enable the collection of these trace events, IIS 7 can be configured to automatically capture comprehensive trace logs in XML format for specific requests. The collection can be based on elapsed time or error response codes.

Storage security

Storage security is essential for protecting your data at rest and in transit. Azure provides multiple layers of encryption, access controls, and monitoring capabilities to ensure your data remains secure. For detailed information on data encryption, see [Azure encryption overview](#). For key management options, see [Key management in Azure](#). For data encryption best practices, see [Azure data security and encryption best practices](#).

Azure role-based access control (Azure RBAC)

You can secure your storage account with [Azure role-based access control \(Azure RBAC\)](#). Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce Security policies for data access. These access rights are granted by assigning the appropriate Azure role to groups and applications at a certain scope. You can use [Azure built-in roles](#), such as Storage Account Contributor, to assign privileges to users. Access to the storage keys for a storage account using the [Azure Resource Manager](#) model can be controlled through Azure RBAC.

Shared Access Signature

A [shared access signature \(SAS\)](#) provides delegated access to resources in your storage account. The SAS means that you can grant a client limited permissions to objects in your storage account for a specified period and with a specified set of permissions. You can grant these limited permissions without having to share your account access keys.

Encryption in Transit

Encryption in transit is a mechanism of protecting data when it's transmitted across networks. With Azure Storage, you can secure data using:

- [Transport-level encryption](#), such as HTTPS when you transfer data into or out of Azure Storage.
- [Wire encryption](#), such as [SMB 3.0 encryption](#) for [Azure File shares](#).
- Client-side encryption, to encrypt the data before it's transferred into storage and to decrypt the data after it's transferred out of storage.

Encryption at rest

For many organizations, data encryption at rest is a mandatory step towards data privacy, compliance, and data sovereignty. There are three Azure storage security features that provide encryption of data that is at rest:

- [Storage Service Encryption](#) allows you to request that the storage service automatically encrypt data when writing it to Azure Storage.
- [Client-side Encryption](#) also provides the feature of encryption at rest.
- [Azure Disk Encryption for Linux VMs](#) and [Azure Disk Encryption for Windows VMs](#) allows you to encrypt the OS disks and data disks used by an IaaS virtual machine.

Storage Analytics

[Azure Storage Analytics](#) performs logging and provides metrics data for a storage account. You can use this data to trace requests, analyze usage trends, and diagnose issues with your storage account. Storage Analytics logs detailed information about successful and failed requests to a storage service. This information can be used to monitor individual requests and to diagnose issues with a storage service. Requests are logged on a best-effort basis. The following types of authenticated requests are logged:

- Successful requests.
- Failed requests, including timeout, throttling, network, authorization, and other errors.
- Requests using a Shared Access Signature (SAS), including failed and successful requests.
- Requests to analytics data.

Enabling Browser-Based Clients Using CORS

[Cross-Origin Resource Sharing \(CORS\)](#) is a mechanism that allows domains to give each other permission for accessing each other's resources. The User Agent sends extra headers to ensure that the JavaScript code loaded from a certain domain is allowed to access resources located at another domain. The latter domain then replies with extra headers allowing or denying the original domain access to its resources.

Azure storage services now support CORS so that once you set the CORS rules for the service, a properly authenticated request made against the service from a different domain is evaluated to determine whether it's allowed according to the rules you have specified.

Network security

Network security controls how traffic flows to and from your Azure resources. Azure provides a comprehensive set of network security services, from basic firewalling to advanced threat protection and global load balancing. For comprehensive network security information, see [Azure network security overview](#). For network security best practices, see [Azure network security best practices](#).

Network Layer Controls

Network access control is the act of limiting connectivity to and from specific devices or subnets and represents the core of network security. The goal of network access control is to make sure that your virtual machines and services are accessible to only users and devices to which you want them accessible.

Network Security Groups

A [Network Security Group \(NSG\)](#) is a basic stateful packet filtering firewall and it enables you to control access based on a five-tuple. NSGs don't provide application layer inspection or authenticated access controls. They can be used to control traffic moving between subnets within an Azure Virtual Network and traffic between an Azure Virtual Network and the Internet.

Azure Firewall

[Azure Firewall](#) is a cloud-native and intelligent network firewall security service that provides threat protection for your cloud workloads running in Azure. It's a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability. It provides both east-west and north-south traffic inspection.

Azure Firewall is offered in three SKUs: Basic, Standard, and Premium:

- [Azure Firewall Basic](#) - Designed for small and medium-sized businesses, offering essential protection at an affordable price point
- [Azure Firewall Standard](#) - Provides L3-L7 filtering, threat intelligence feeds from Microsoft Cyber Security, and can scale to 30 Gbps
- [Azure Firewall Premium](#) - Advanced threat protection for highly sensitive and regulated environments with:
 - **TLS Inspection:** Decrypts outbound traffic, processes it for threats, then re-encrypts before sending to destination
 - **IDPS (Intrusion Detection and Prevention System):** Signature-based IDPS with over 67,000 signatures in more than 50 categories, updated with 20-40+ new rules daily
 - **URL Filtering:** Extends FQDN filtering to consider the entire URL path

- **Advanced Web Categories:** Enhanced categorization based on complete URLs for both HTTP and HTTPS traffic
- **Enhanced Performance:** Scales up to 100 Gbps with 10 Gbps fat flow support
- **PCI DSS Compliance:** Meets Payment Card Industry Data Security Standard requirements

Azure Firewall Premium is essential for protecting against ransomware, as it can detect and block Command and Control (C&C) connectivity used by ransomware to fetch encryption keys. Learn more about [ransomware protection with Azure Firewall](#).

Azure DDoS Protection

[Azure DDoS Protection](#), combined with application design best practices, offers enhanced features to defend against DDoS attacks. It is automatically tuned to protect your specific Azure resources in a virtual network. Enabling protection is simple on any new or existing virtual network and requires no changes to your applications or resources.

Azure DDoS Protection offers two tiers: DDoS Network Protection and DDoS IP Protection.

- **DDoS Network Protection** - Provides enhanced features to defend against Distributed Denial of Service (DDoS) attacks. It operates at network layers 3 and 4 and includes advanced features such as DDoS rapid response support, cost protection, and discounts on Web Application Firewall (WAF).
- **DDoS IP Protection** - Follows a pay-per-protected IP model. It includes the same core engineering features as DDoS Network Protection but does not offer the additional services like DDoS rapid response support, cost protection, and WAF discounts.

Route Control and Forced Tunneling

The ability to control routing behavior on your Azure Virtual Networks is a critical network security and access control capability. For example, if you want to make sure that all traffic to and from your Azure Virtual Network goes through that virtual security appliance, you need to be able to control and customize routing behavior. You can do this by configuring User-Defined Routes in Azure.

[User-Defined Routes](#) allow you to customize inbound and outbound paths for traffic moving into and out of individual virtual machines or subnets to ensure the most secure route possible. [Forced tunneling](#) is a mechanism you can use to ensure that your services aren't allowed to initiate a connection to devices on the Internet.

This is different from being able to accept incoming connections and then responding to them. Front-end web servers need to respond to requests from Internet hosts, and so Internet-

sourced traffic is allowed inbound to these web servers and the web servers can respond.

Forced tunneling is commonly used to force outbound traffic to the Internet to go through on-premises security proxies and firewalls.

Virtual Network Security Appliances

While Network Security Groups, User-Defined Routes, and forced tunneling provide you with a level of security at the network and transport layers of the [OSI model](#), there might be times when you want to enable security at higher levels of the stack. You can access these enhanced network security features by using an Azure partner network security appliance solution. You can find the most current Azure partner network security solutions by visiting the [Azure Marketplace](#) and searching for **security** and **network security**.

Azure Virtual Network

An Azure virtual network (VNet) is a representation of your own network in the cloud. It's a logical isolation of the Azure network fabric dedicated to your subscription. You can fully control the IP address blocks, DNS settings, security policies, and route tables within this network. You can segment your VNet into subnets and place Azure IaaS virtual machines (VMs) on Azure Virtual Networks.

Additionally, you can connect the virtual network to your on-premises network using one of the [connectivity options](#) available in Azure. In essence, you can expand your network to Azure, with complete control on IP address blocks with the benefit of enterprise scale Azure provides.

Azure networking supports various secure remote access scenarios. Some of these include:

- [Connect individual workstations to an Azure Virtual Network](#)
- [Connect on-premises network to an Azure Virtual Network with a VPN](#)
- [Connect on-premises network to an Azure Virtual Network with a dedicated WAN link](#)
- [Connect Azure Virtual Networks to each other](#)

Azure Virtual Network Manager

[Azure Virtual Network Manager](#) provides a centralized solution for managing and securing your virtual networks at scale. It uses [security admin rules](#) to centrally define and enforce security policies across your entire organization. Security admin rules take precedence over network security group (NSGs) rules and are applied on the virtual network. This allows organizations to enforce core policies with security admin rules, while still enabling

downstream teams to tailor NSGs according to their specific needs at the subnet and NIC levels.

Depending on the needs of your organization you can use **Allow**, **Deny**, or **Always Allow** rule actions to enforce security policies:

 [Expand table](#)

Rule Action	Description
Allow	Allows the specified traffic by default. Downstream NSGs still receive this traffic and might deny it.
Always Allow	Always allow the specified traffic, regardless of other rules with lower priority or NSGs. This can be used to ensure that monitoring agent, domain controller, or management traffic isn't blocked.
Deny	Block the specified traffic. Downstream NSGs won't evaluate this traffic after being denied by a security admin rule, ensuring your high-risk ports for existing and new virtual networks are protected by default.

In Azure Virtual Network Manager, [network groups](#) allow you to group virtual networks together for centralized management and enforcement of security policies. Network groups are a logical grouping of virtual networks based on your needs from a topology and security perspective. You can manually update the virtual network membership of your network groups or you can [define conditional statements with Azure Policy](#) to dynamically update network groups to automatically update your network group membership.

Azure Private Link

[Azure Private Link](#)  enables you to access Azure PaaS Services (for example, Azure Storage and SQL Database) and Azure hosted customer-owned/partner services privately in your virtual network over a [private endpoint](#). Setup and consumption using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services. Traffic from your virtual network to the Azure service always remains on the Microsoft Azure backbone network.

[Private Endpoints](#) allow you to secure your critical Azure service resources to only your virtual networks. Azure Private Endpoint uses a private IP address from your VNet to connect you privately and securely to a service powered by Azure Private Link, effectively bringing the service into your VNet. Exposing your virtual network to the public internet is no longer necessary to consume services on Azure.

You can also create your own private link service in your virtual network. [Azure Private Link service](#) is the reference to your own service that is powered by Azure Private Link. Your service

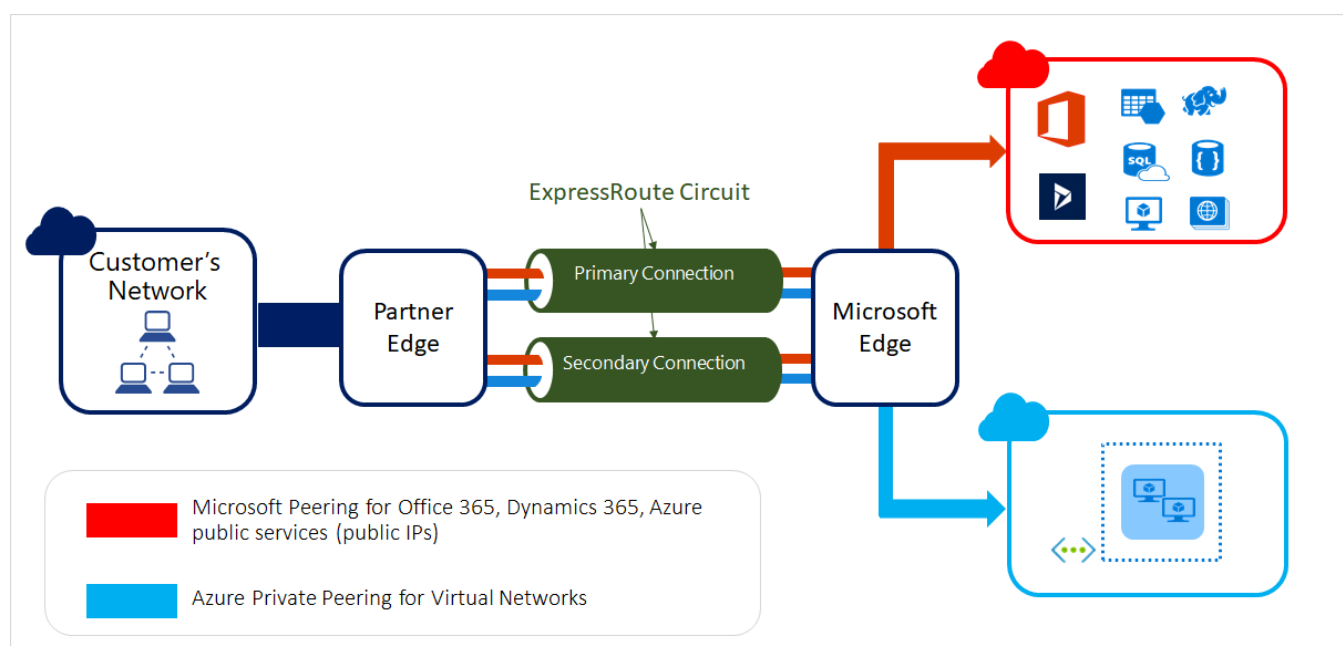
that is running behind Azure Standard Load Balancer can be enabled for Private Link access so that consumers to your service can access it privately from their own virtual networks. Your customers can create a private endpoint inside their virtual network and map it to this service. Exposing your service to the public internet is no longer necessary to render services on Azure.

VPN Gateway

To send network traffic between your Azure Virtual Network and your on-premises site, you must create a VPN gateway for your Azure Virtual Network. A [VPN gateway](#) is a type of virtual network gateway that sends encrypted traffic across a public connection. You can also use VPN gateways to send traffic between Azure Virtual Networks over the Azure network fabric.

Express Route

Microsoft Azure [ExpressRoute](#) is a dedicated WAN link that lets you extend your on-premises networks into the Microsoft cloud over a dedicated private connection facilitated by a connectivity provider.

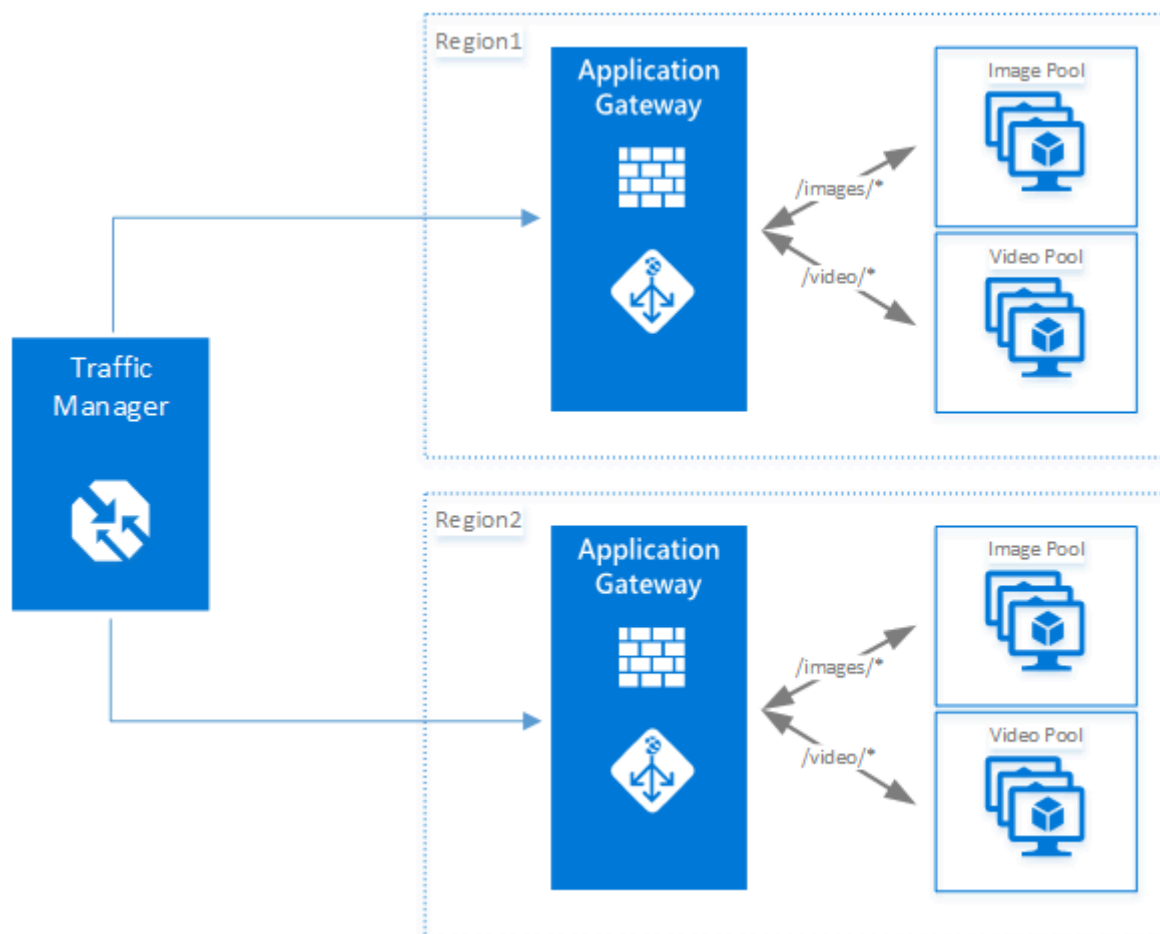


With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure and Microsoft 365. Connectivity can be from an any-to-any (IP VPN) network, a point-to-point Ethernet network, or a virtual cross-connection through a connectivity provider at a colocation facility.

ExpressRoute connections don't go over the public Internet and thus can be considered more secure than VPN-based solutions. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.

Application Gateway

Microsoft [Azure Application Gateway](#) provides an [Application Delivery Controller \(ADC\)](#) [↗](#) as a service, offering various layer 7 load balancing capabilities for your application.



It allows you to optimize web farm productivity by offloading CPU intensive TLS termination to the Application Gateway (also known as **TLS offload** or **TLS bridging**). It also provides other Layer 7 routing capabilities including round-robin distribution of incoming traffic, cookie-based session affinity, URL path-based routing, and the ability to host multiple websites behind a single Application Gateway. Azure Application Gateway is a layer-7 load balancer.

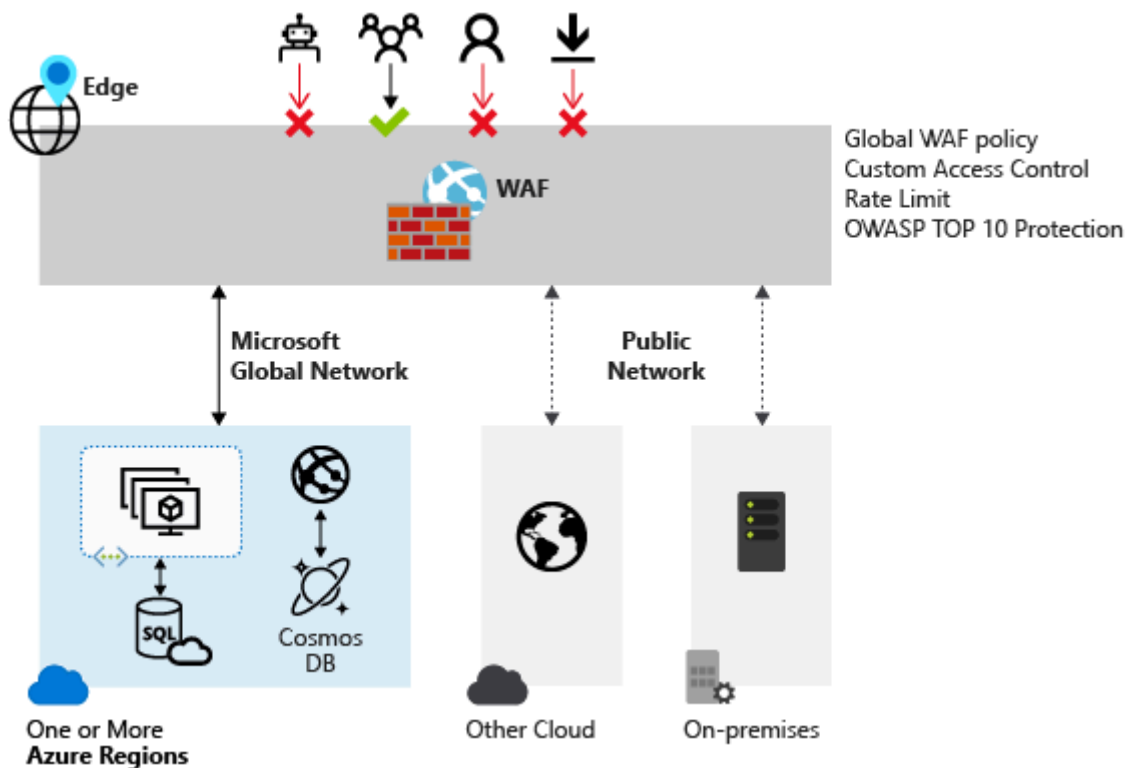
It provides failover, performance-routing HTTP requests between different servers, whether they are on the cloud or on-premises.

Application provides many Application Delivery Controller (ADC) features including HTTP load balancing, cookie-based session affinity, [TLS offload](#), custom health probes, support for multi-site, and many others.

Web Application Firewall

Web Application Firewall is a feature of [Azure Application Gateway](#) that provides protection to web applications that use application gateway for standard Application Delivery Control (ADC)

functions. Web application firewall does this by protecting them against most of the OWASP top 10 common web vulnerabilities.



- SQL injection protection
- Protection against common web attacks such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion
- Protection against HTTP protocol violations
- Protection against HTTP protocol anomalies, such as missing host, user-agent, and accept headers
- Prevention against bots, crawlers, and scanners
- Detection of common application misconfigurations (e.g., Apache, IIS)

A centralized web application firewall (WAF) simplifies security management and enhances protection against web attacks. It provides better assurance against intrusion threats and can respond faster to security threats by patching known vulnerabilities centrally, rather than securing each individual web application. Existing application gateways can be easily upgraded to include a web application firewall.

Azure Front Door

[Azure Front Door](#) is a global, scalable entry point that uses Microsoft's global edge network to create fast, secure, and widely scalable web applications. Front Door provides:

- **Global load balancing:** Distribute traffic across multiple backends in different regions
- **Integrated Web Application Firewall:** Protect against common web vulnerabilities and attacks
- **DDoS protection:** Built-in protection against distributed denial-of-service attacks
- **SSL/TLS offload:** Centralized certificate management and traffic encryption
- **URL-based routing:** Route traffic to different backends based on URL patterns

Front Door combines content delivery, application acceleration, and security into a single service.

Traffic Manager

[Microsoft Azure Traffic Manager](#) allows you to control the distribution of user traffic for service endpoints in different datacenters. Service endpoints supported by Traffic Manager include Azure VMs, Web Apps, and Cloud services. You can also use Traffic Manager with external, non-Azure endpoints.

Traffic Manager uses the Domain Name System (DNS) to direct client requests to the most appropriate endpoint based on a [traffic-routing method](#) and the health of the endpoints. Traffic Manager provides a range of traffic-routing methods to suit different application needs, endpoint health [monitoring](#), and automatic failover. Traffic Manager is resilient to failure, including the failure of an entire Azure region.

Azure Load Balancer

[Azure Load Balancer](#) delivers high availability and network performance to your applications. It's a Layer 4 (TCP, UDP) load balancer that distributes incoming traffic among healthy instances of services defined in a load-balanced set. Azure Load Balancer can be configured to:

- Load balance incoming Internet traffic to virtual machines. This configuration is known as [public load balancing](#).
- Load balance traffic between virtual machines in a virtual network, between virtual machines in cloud services, or between on-premises computers and virtual machines in a cross-premises virtual network. This configuration is known as [internal load balancing](#).
- Forward external traffic to a specific virtual machine

Internal DNS

You can manage the list of DNS servers used in a VNet in the Management Portal, or in the network configuration file. Customer can add up to 12 DNS servers for each VNet. When

specifying DNS servers, it's important to verify that you list customer's DNS servers in the correct order for customer's environment. DNS server lists don't work round-robin. They're used in the order that they're specified. If the first DNS server on the list is able to be reached, the client uses that DNS server regardless of whether the DNS server is functioning properly or not. To change the DNS server order for customer's virtual network, remove the DNS servers from the list and add them back in the order that customer wants. DNS supports the availability aspect of the "CIA" security triad.

Azure DNS

The Domain Name System, or DNS, is responsible for translating (or resolving) a website or service name to its IP address. [Azure DNS](#) is a hosting service for DNS domains, providing name resolution using Microsoft Azure infrastructure. By hosting your domains in Azure, you can manage your DNS records using the same credentials, APIs, tools, and billing as your other Azure services. DNS supports the availability aspect of the "CIA" security triad.

Azure Monitor logs NSGs

You can enable the following diagnostic log categories for NSGs:

- Event: Contains entries for which NSG rules are applied to VMs and instance roles based on MAC address. The status for these rules is collected every 60 seconds.
- Rules counter: Contains entries for how many times each NSG rule is applied to deny or allow traffic.

Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) continuously analyzes the security state of your Azure resources for network security best practices. When Defender for Cloud identifies potential security vulnerabilities, it creates [recommendations](#) that guide you through the process of configuring the needed controls to harden and protect your resources.

Advanced Container Networking Services (ACNS)

[Advanced Container Networking Services \(ACNS\)](#) is a comprehensive suite designed to elevate the operational efficiency of your Azure Kubernetes Service (AKS) clusters. It provides advanced security and observability features, addressing the complexities of managing microservices infrastructure at scale.

These features are divided into two main pillars:

- **Security:** For clusters using Azure CNI Powered by Cilium, network policies include fully qualified domain name (FQDN) filtering for solving the complexities of maintaining configuration.
- **Observability:** This feature of the Advanced Container Networking Services suite brings the power of Hubble's control plane to both Cilium and non-Cilium Linux data planes, providing enhanced visibility into networking and performance.

Security operations and management

Managing and monitoring the security of your Azure environment is essential for maintaining a strong security posture. Azure provides comprehensive tools for security operations, threat detection, and incident response. For detailed coverage of security management and monitoring, see [Azure security management and monitoring overview](#). For operational security best practices, see [Azure operational security best practices](#). For comprehensive operational security overview, see [Azure operational security overview](#).

Microsoft Sentinel

[Microsoft Sentinel](#) is a scalable, cloud-native security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution. Microsoft Sentinel delivers intelligent security analytics and threat intelligence across the enterprise, providing a single solution for attack detection, threat visibility, proactive hunting, and threat response.

Microsoft Sentinel is now available in the Microsoft Defender portal for all customers, offering a unified security operations experience that streamlines workflows and enhances visibility. The integration with Security Copilot enables analysts to interact with Microsoft Sentinel data using natural language, generate hunting queries, and automate investigations for faster threat response.

Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. Microsoft Defender for Cloud provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Microsoft Defender for Cloud offers comprehensive protection with workload-specific plans including:

- **Defender for Servers** - Advanced threat protection for Windows and Linux servers
- **Defender for Containers** - Security for containerized applications and Kubernetes
- **Defender for Storage** - Threat detection with malware scanning and sensitive data discovery
- **Defender for Databases** - Protection for Azure SQL, Azure Database for MySQL, and PostgreSQL
- **Defender for AI Services** - Runtime protection for Azure AI services against jailbreak attempts, data exposure, and suspicious access patterns
- **Defender CSPM** - Cloud Security Posture Management with attack path analysis, security governance, and AI security posture management

In addition, Defender for Cloud helps with security operations by providing you with a single dashboard that surfaces alerts and recommendations that can be acted upon immediately. Security Copilot integration provides AI-generated summaries, remediation scripts, and delegation capabilities to accelerate risk remediation.

For comprehensive threat detection capabilities across Azure, see [Azure threat protection](#).

Azure Resource Manager

[Azure Resource Manager](#) enables you to work with the resources in your solution as a group. You can deploy, update, or delete all the resources for your solution in a single, coordinated operation. You use an [Azure Resource Manager template](#) for deployment and that template can work for different environments such as testing, staging, and production. Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.

Azure Resource Manager template-based deployments help improve the security of solutions deployed in Azure because standard security control settings and can be integrated into standardized template-based deployments. Templates reduce the risk of security configuration errors that might take place during manual deployments.

Application Insights

[Application Insights](#) is a flexible Application Performance Management (APM) service designed for web developers. It enables you to monitor your live web applications and automatically detect performance issues. With powerful analytics tools, you can diagnose problems and gain insights into user interactions with your apps. Application Insights monitors your application continuously, from development through testing and into production.

Application Insights generates insightful charts and tables that reveal peak user activity times, app responsiveness, and the performance of any external services it relies on.

If there are crashes, failures, or performance issues, you can search through the data in detail to diagnose the cause. And the service sends you emails if there are any changes in the availability and performance of your app. Application Insight thus becomes a valuable security tool because it helps with the availability in the confidentiality, integrity, and availability security triad.

Azure Monitor

[Azure Monitor](#) offers visualization, query, routing, alerting, auto scale, and automation on data both from the Azure subscription ([Activity Log](#)) and each individual Azure resource ([Resource Logs](#)). You can use Azure Monitor to alert you on security-related events that are generated in Azure logs.

Azure Monitor logs

[Azure Monitor logs](#) provides an IT management solution for both on-premises and third-party cloud-based infrastructure (such as Amazon Web Services) in addition to Azure resources. Data from Azure Monitor can be routed directly to Azure Monitor logs so you can see metrics and logs for your entire environment in one place.

Azure Monitor logs can be a useful tool in forensic and other security analysis, as the tool enables you to quickly search through large amounts of security-related entries with a flexible query approach. In addition, on-premises [firewall and proxy logs can be exported into Azure and made available for analysis using Azure Monitor logs](#).

Azure Advisor

[Azure Advisor](#) is a personalized cloud consultant that helps you to optimize your Azure deployments. It analyzes your resource configuration and usage data. It then recommends solutions to help improve the [performance](#), [security](#), and [reliability](#) of your resources while looking for opportunities to [reduce your overall Azure spend](#). Azure Advisor provides security recommendations, which can significantly improve your overall security posture for solutions you deploy in Azure. These recommendations are drawn from security analysis performed by [Microsoft Defender for Cloud](#).

Identity and access management

Identity is the primary security perimeter in cloud computing. Protecting identities and controlling access to resources is fundamental to securing your Azure environment. Microsoft Entra ID provides comprehensive identity and access management capabilities. For detailed

information, see [Azure identity management overview](#). For identity management best practices, see [Azure identity management and access control security best practices](#). For guidance on securing identity infrastructure, see [Five steps to securing your identity infrastructure](#).

Microsoft Entra ID

[Microsoft Entra ID](#) is Microsoft's cloud-based identity and access management service. It provides:

- **Single Sign-On (SSO)**: Enable users to access multiple applications with one set of credentials
- **Multi-Factor Authentication (MFA)**: Require multiple forms of verification to sign in
- **Conditional Access**: Control access to resources based on user, device, location, and risk
- **Identity Protection**: Detect and respond to identity-based risks
- **Privileged Identity Management (PIM)**: Provide just-in-time privileged access to Azure resources
- **Identity Governance**: Manage identity lifecycle and access rights

Role-Based Access Control (RBAC)

Azure role-based access control (RBAC) helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to. RBAC provides fine-grained access management for Azure resources, enabling you to grant users only the rights they need to perform their jobs.

Microsoft Entra Privileged Identity Management

[Microsoft Entra Privileged Identity Management \(PIM\)](#) enables you to manage, control, and monitor access to important resources in your organization. PIM provides time-based and approval-based role activation to mitigate the risks of excessive, unnecessary, or misused access permissions.

Managed Identities for Azure Resources

[Managed identities](#) for Azure resources provides Azure services with an automatically managed identity in Microsoft Entra ID. You can use this identity to authenticate to any service that supports Microsoft Entra authentication, without having credentials in your code.

Patch Updates provide the basis for finding and fixing potential problems and simplify the software update management process, both by reducing the number of software updates you must deploy in your enterprise and by increasing your ability to monitor compliance.

Security policy management and reporting

[Defender for Cloud](#) helps you prevent, detect, and respond to threats, and provides you increased visibility into, and control over, the security of your Azure resources. It provides integrated Security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Secure Identity

Microsoft uses multiple security practices and technologies across its products and services to manage identity and access.

- [Multifactor authentication](#)  requires users to use multiple methods for access, on-premises and in the cloud. It provides strong authentication with a range of easy verification options, while accommodating users with a simple sign-in process.
- [Microsoft Authenticator](#)  provides a user-friendly multifactor authentication experience that works with both Microsoft Entra ID and Microsoft accounts, and includes support for wearables and fingerprint-based approvals.
- [Password policy enforcement](#) increases the security of traditional passwords by imposing length and complexity requirements, forced periodic rotation, and account lockout after failed authentication attempts.
- [Token-based authentication](#) enables authentication via Microsoft Entra ID.
- [Azure role-based access control \(Azure RBAC\)](#) enables you to grant access based on the user's assigned role, making it easy to give users only the amount of access they need to perform their job duties. You can customize Azure RBAC per your organization's business model and risk tolerance.
- [Integrated identity management \(hybrid identity\)](#) enables you to maintain control of users' access across internal datacenters and cloud platforms, creating a single user identity for authentication and authorization to all resources.

Secure Apps and data

[Microsoft Entra ID](#) , a comprehensive identity, and access management cloud solution, helps secure access to data in applications on site and in the cloud, and simplifies the management of users and groups. It combines core directory services, advanced identity governance, security, and application access management, and makes it easy for developers to build policy-

based identity management into their apps. To enhance your Microsoft Entra ID, you can add paid capabilities using the Microsoft Entra Basic, Premium P1, and Premium P2 editions.

[Expand table](#)

Free / Common Features	Basic Features	Premium P1 Features	Premium P2 Features	Microsoft Entra join – Windows 10 only related features
Directory Objects, User/Group Management (add/update/delete)/ User-based provisioning, Device registration, single sign-on (SSO), Self-Service Password Change for cloud users, Connect (Sync engine that extends on-premises directories to Microsoft Entra ID), Security / Usage Reports	Group-based access management / provisioning, Self-Service Password Reset for cloud users, Company Branding (sign in Pages/Access Panel customization), Application Proxy, SLA 99.9%	Self-Service Group and app Management/Self-Service application additions/Dynamic Groups, Self-Service Password Reset/Change/Unlock with on-premises write-back, multifactor authentication (Cloud and On-premises (MFA Server)), MIM CAL + MIM Server, Cloud App Discovery, Connect Health, Automatic password rollover for group accounts	Identity Protection, Privileged Identity Management	Join a device to Microsoft Entra ID, Desktop SSO, Microsoft Passport for Microsoft Entra ID, Administrator BitLocker recovery, MDM autoenrollment, Self-Service BitLocker recovery, extra local administrators to Windows 10 devices via Microsoft Entra join

- [Cloud App Discovery](#) is a premium feature of Microsoft Entra ID that enables you to identify cloud applications that are used by the employees in your organization.
- [Microsoft Entra ID Protection](#) is a security service that uses Microsoft Entra anomaly detection capabilities to provide a consolidated view into risk detections and potential vulnerabilities that could affect your organization's identities.
- [Microsoft Entra Domain Services](#) [↗](#) enables you to join Azure VMs to a domain without the need to deploy domain controllers. Users sign in to these VMs by using their corporate Active Directory credentials, and can seamlessly access resources.
- [Microsoft Entra B2C](#) [↗](#) is a highly available, global identity management service for consumer-facing apps that can scale to hundreds of millions of identities and integrate across mobile and web platforms. Your customers can sign in to all your apps through customizable experiences that use existing social media accounts, or you can create new standalone credentials.

- [Microsoft Entra B2B Collaboration](#) is a secure partner integration solution that supports your cross-company relationships by enabling partners to access your corporate applications and data selectively by using their self-managed identities.
- [Microsoft Entra joined](#) enables you to extend cloud capabilities to Windows 10 devices for centralized management. It makes it possible for users to connect to the corporate or organizational cloud through Microsoft Entra ID and simplifies access to apps and resources.
- [Microsoft Entra application proxy](#) provides SSO and secure remote access for web applications hosted on-premises.

Next Steps

- Understand your [shared responsibility in the cloud](#).
- Learn how [Microsoft Defender for Cloud](#) can help you prevent, detect, and respond to threats with increased visibility and control over the security of your Azure resources.
- Explore [Azure security best practices and patterns](#) for additional security recommendations.
- Review the [Microsoft cloud security benchmark](#) for comprehensive security guidance.
- See [End-to-end security in Azure](#) for a protection, detection, and response view of Azure security architecture.

Last updated on 11/07/2025

End-to-end security in Azure

Azure provides comprehensive security capabilities across all layers of your cloud deployments. Microsoft Azure delivers confidentiality, integrity, and availability of customer data while enabling transparent accountability. This article introduces Azure's security architecture organized by protection, detection, and response capabilities.

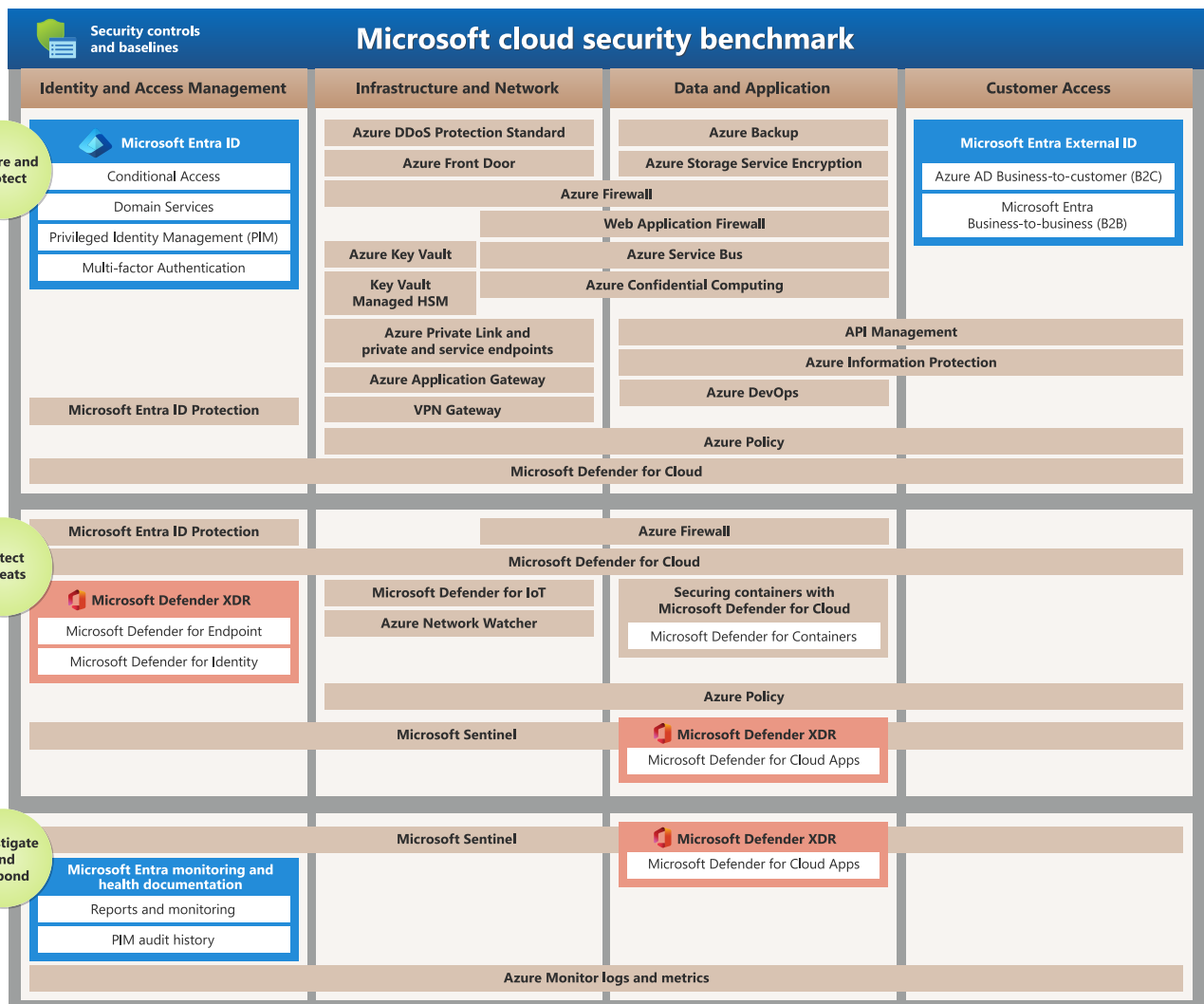
For a comprehensive introduction to Azure security capabilities organized by functional area, see [Introduction to Azure security](#). For detailed implementation guidance and best practices, refer to the domain-specific security overview articles linked throughout this document.

Microsoft security architecture

Azure security services are organized into three foundational categories:

- **Secure and protect:** Implement defense-in-depth strategies across identity, infrastructure, networks, and data
- **Detect threats:** Identify suspicious activities and potential security incidents
- **Investigate and respond:** Analyze security events and take corrective actions

The following diagram illustrates how Azure security services align with these categories and the resources they protect:



Security controls and baselines

The [Microsoft cloud security benchmark](#) provides comprehensive security guidance for Azure services:

- **Security controls:** High-level recommendations applicable across your Azure tenant and services
- **Service baselines:** Implementation of controls for individual Azure services with specific configuration recommendations

Use these controls and baselines to:

- Establish security standards for cloud deployments
- Assess compliance at scale using [Microsoft Defender for Cloud regulatory compliance dashboard](#)
- Map to industry frameworks including CIS, NIST, and PCI-DSS
- Implement secure configurations with [Azure Policy](#)

For governance and compliance capabilities, see [Azure security management and monitoring overview](#).

Secure and protect

Azure provides layered security controls across identity, infrastructure, networks, and data. For detailed implementation guidance, refer to the domain-specific overview articles.

Threat protection

[Microsoft Defender for Cloud](#) provides unified security management with continuous assessment and advanced threat protection. For comprehensive coverage, see [Azure threat protection](#).

Identity and access

- [Microsoft Entra ID](#) - Cloud identity and access management
- [Privileged Identity Management](#) - Just-in-time privileged access
- [Microsoft Entra ID Protection](#) - Automated risk detection and remediation

For details, see [Azure identity management security overview](#).

Network security

- [Azure Firewall](#) - Cloud-native network firewall with IDPS
- [Azure DDoS Protection](#) - Always-on DDoS mitigation
- [Azure VPN Gateway](#) - Encrypted cross-premises connectivity
- [Azure Front Door](#) - Global load balancer with integrated WAF
- [Azure Private Link](#) - Private connectivity to Azure services

For details, see [Azure network security overview](#).

Data protection

- [Azure Key Vault](#) - Secure key and secret storage with FIPS 140-2 Level 1 (Standard tier) and FIPS 140-3 Level 3 (Premium tier) validated HSMs
- [Key Vault Managed HSM](#) - Single-tenant FIPS 140-3 Level 3 HSM
- [Azure Storage Service Encryption](#) - Automatic encryption at rest
- [Azure Backup](#) - Independent and isolated backups
- [Azure confidential computing](#) - Hardware-based data protection in use with AMD SEV-SNP, Intel TDX, and NVIDIA H100 GPU support

For details, see [Azure encryption overview](#) and [Key management in Azure](#).

Governance

- [Azure Policy](#) - Enforce standards and assess compliance

For details, see [Azure security management and monitoring overview](#).

Detect threats

Azure threat detection services identify suspicious activities and security incidents across your environment.

- [Microsoft Defender for Cloud](#) - Advanced threat protection with workload-specific plans
- [Microsoft Sentinel](#) - Cloud-native SIEM and SOAR solution
- [Microsoft Defender XDR](#) - Unified endpoint, identity, email, and application protection
- [Azure Network Watcher](#) - Network monitoring and diagnostics
- [Microsoft Defender for Cloud Apps](#) - Cloud access security broker (CASB)

For comprehensive threat detection capabilities, see [Azure threat protection](#).

Investigate and respond

Azure provides tools to analyze security events and respond to incidents.

- [Microsoft Sentinel](#) - Threat hunting with search and query tools
- [Azure Monitor](#) - Comprehensive telemetry collection and analysis with Log Analytics workspaces
- [What is Microsoft Entra monitoring and health?](#) - Activity logs and audit history
- [Microsoft Defender for Cloud Apps](#) - Cloud environment investigation tools

For monitoring and operational guidance, see [Azure security management and monitoring overview](#).

Next steps

- Review [Introduction to Azure security](#) for a comprehensive overview organized by functional area
- Review [Azure security services and technologies](#) for a comprehensive list of security capabilities
- Understand [shared responsibility in the cloud](#)

- Explore [Azure security best practices and patterns](#)
 - Learn about [Microsoft cloud security benchmark](#)
-

Last updated on 11/07/2025

Shared responsibility in the cloud

As you consider and evaluate public cloud services, it's critical to understand the shared responsibility model and which security tasks the cloud provider handles and which tasks you handle. The workload responsibilities vary depending on whether the workload is hosted on Software as a Service (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS), or in an on-premises datacenter:

- **IaaS** (Infrastructure as a Service): You manage virtual machines, operating systems, and applications. Examples include Azure Virtual Machines, Azure Disk Storage, and virtual networks.
- **PaaS** (Platform as a Service): You deploy applications without managing VMs or operating systems. Examples include Azure App Service, Azure Functions, Azure SQL Database, and Azure Storage.
- **SaaS** (Software as a Service): You use ready-made applications. Examples include Microsoft 365, Dynamics 365, and other cloud applications.

Many Azure solutions use a combination of service models. For more detailed guidance on choosing compute services, see [Choose an Azure compute service](#).

Division of responsibility

In an on-premises datacenter, you own the whole stack. As you move to the cloud, some responsibilities transfer to Microsoft. The following diagram illustrates the areas of responsibility between you and Microsoft, according to the type of deployment of your stack.

Responsibility		SaaS	PaaS	IaaS	On-prem
Responsibility always retained by the customer	Information and data	Customer	Customer	Customer	Customer
	Devices (Mobile and PCs)	Customer	Customer	Customer	Customer
	Accounts and identities	Customer	Customer	Customer	Customer
Responsibility varies by type	Identity and directory infrastructure	Shared	Shared	Customer	Customer
	Applications	Customer	Shared	Customer	Customer
	Network controls	Customer	Shared	Customer	Customer
	Operating system	Customer	Customer	Customer	Customer
Responsibility transfers to cloud provider	Physical hosts	Microsoft	Microsoft	Microsoft	Customer
	Physical network	Microsoft	Microsoft	Microsoft	Customer
	Physical datacenter	Microsoft	Microsoft	Microsoft	Customer

 Microsoft  Customer  Shared

For all cloud deployment types, you own your data and identities. You're responsible for protecting the security of your data and identities, on-premises resources, and the cloud components you control. Cloud components you control vary by service type.

Responsibility matrix

The following table details the division of responsibility between you and Microsoft for each area of your stack:

 Expand table

Responsibility area	On-premises	IaaS	PaaS	SaaS
Customer data	Customer	Customer	Customer	Customer
Configurations and settings	Customer	Customer	Customer	Customer
Identities and users	Customer	Customer	Customer	Customer
Client devices	Customer	Customer	Customer	Shared
Applications	Customer	Customer	Shared	Shared
Network controls	Customer	Customer	Shared	Microsoft
Operating system	Customer	Customer	Microsoft	Microsoft
Physical hosts	Customer	Microsoft	Microsoft	Microsoft
Physical network	Customer	Microsoft	Microsoft	Microsoft
Physical datacenter	Customer	Microsoft	Microsoft	Microsoft

Responsibilities you always retain

Regardless of the type of deployment, you always retain the following responsibilities:

- **Data** - You're responsible for your data, including data classification, data protection, encryption decisions, and compliance with data governance requirements.
- **Endpoints** - You're responsible for protecting client devices and endpoints that access your cloud services, including mobile devices, laptops, and desktops.
- **Accounts** - You're responsible for managing user accounts, including creating, managing, and removing user access.
- **Access management** - You're responsible for implementing and managing access controls, including role-based access control (RBAC), multifactor authentication, and conditional access policies.

Shared responsibilities explained

Some responsibilities are shared between you and Microsoft, with the division varying by service model:

- **Applications** - In IaaS, you're fully responsible for deployed applications. In PaaS and SaaS, Microsoft manages parts of the application stack, but you're responsible for application configuration, code security, and access controls.
- **Network controls** - In IaaS, you configure all network security including firewalls and network segmentation. In PaaS, Microsoft provides baseline network security, but you configure application-level network controls. In SaaS, Microsoft manages network security.
- **Client devices** - In SaaS scenarios, Microsoft may provide some device management capabilities, but you're responsible for endpoint protection and compliance.

Microsoft responsibilities

Microsoft is responsible for the underlying cloud infrastructure, which includes:

- **Physical security** - Securing datacenters, including facilities, physical access controls, and environmental controls.
- **Physical network** - Managing network infrastructure, including routers, switches, and cables within datacenters.
- **Physical hosts** - Managing and maintaining the physical servers that host cloud services.
- **Hypervisor** - Managing the virtualization layer that enables virtual machines in IaaS and PaaS.
- **Platform services** - In PaaS and SaaS, Microsoft manages operating systems, runtime environments, and middleware.

AI Shared Responsibility

When using AI services, the shared responsibility model introduces unique considerations beyond traditional IaaS, PaaS, and SaaS. Microsoft is responsible for securing the AI infrastructure, model hosting, and platform-level safeguards. Customers, however, remain accountable for how AI is applied within their environment—this includes protecting sensitive data, managing prompt security, mitigating prompt injection risks, and ensuring compliance with organizational and regulatory requirements.

Because responsibilities differ significantly for AI workloads, you should review the [AI Shared Responsibility Model](#) for detailed guidance on roles, best practices, and risk management.

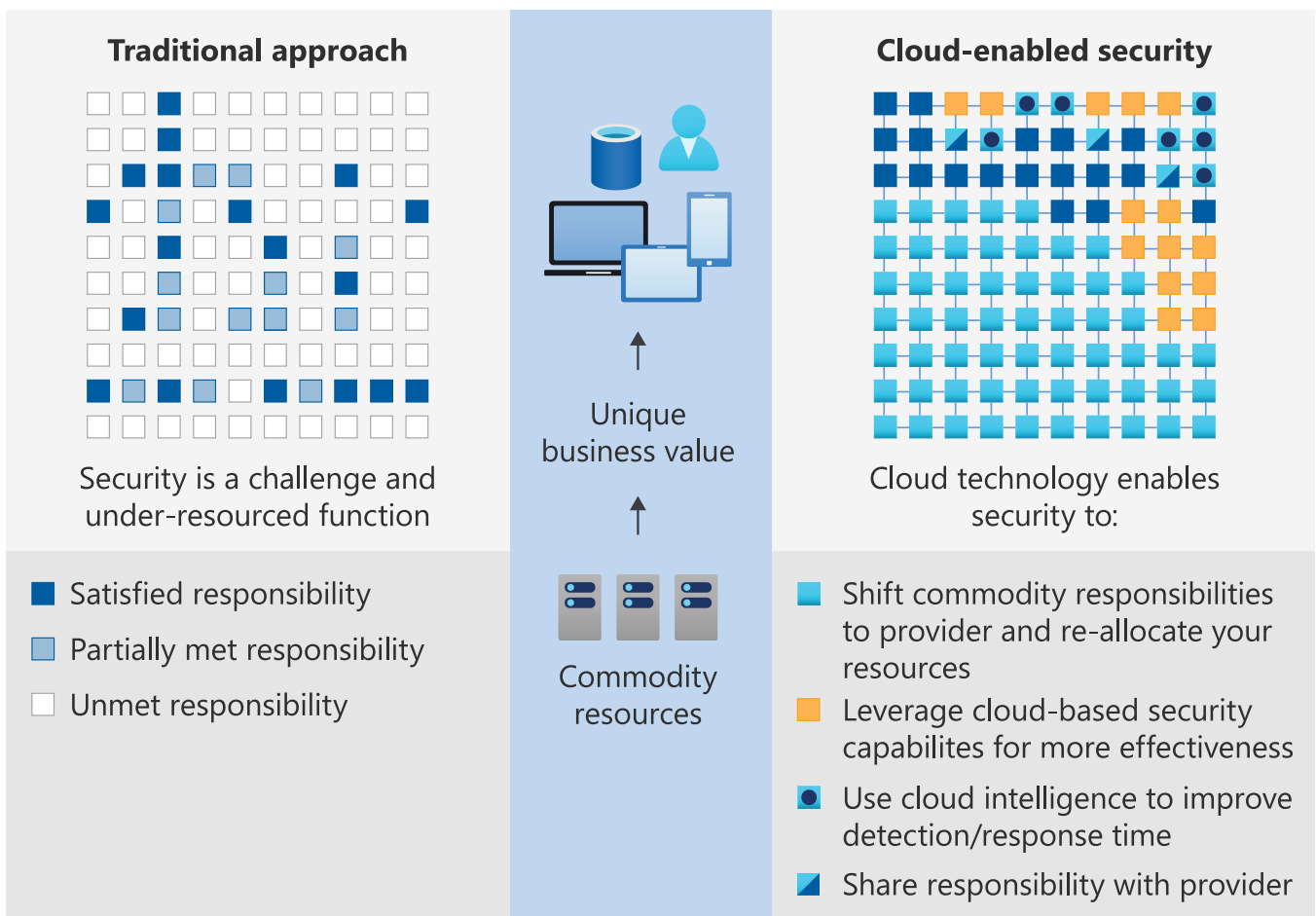
Cloud security advantages

The cloud offers significant advantages for solving long standing information security challenges. In an on-premises environment, organizations likely have unmet responsibilities and limited resources available to invest in security, which creates an environment where attackers are able to exploit vulnerabilities at all layers.

Common examples of unmet responsibilities in traditional on-premises environments include:

- **Delayed patching** - Security updates aren't applied promptly due to limited IT staff or concerns about system downtime, leaving known vulnerabilities exposed.
- **Inadequate physical security** - Server rooms may lack proper access controls, environmental monitoring, or surveillance due to budget constraints.
- **Incomplete network monitoring** - Organizations may not have tools or expertise to detect intrusions, monitor traffic anomalies, or respond to threats in real time.
- **Outdated hardware** - Aging infrastructure may no longer receive security updates from vendors, creating permanent security gaps.
- **Insufficient backup and disaster recovery** - Backups may be infrequent, untested, or stored on-site, leaving data vulnerable to ransomware or physical disasters.

The following diagram shows a traditional approach where many security responsibilities are unmet due to limited resources. In the cloud-enabled approach, you're able to shift day to day security responsibilities to your cloud provider and reallocate your resources.



In the cloud-enabled approach, you're also able to apply cloud-based security capabilities for more effectiveness and use cloud intelligence to improve your threat detection and response time. By shifting responsibilities to the cloud provider, organizations can get more security coverage, which enables them to reallocate security resources and budget to other business priorities.

Next step

Learn more about shared responsibility and strategies to improve your security posture in the Well-Architected Framework's [overview of the security pillar](#).

Last updated on 01/09/2026

Artificial intelligence (AI) shared responsibility model

Article • 09/29/2024

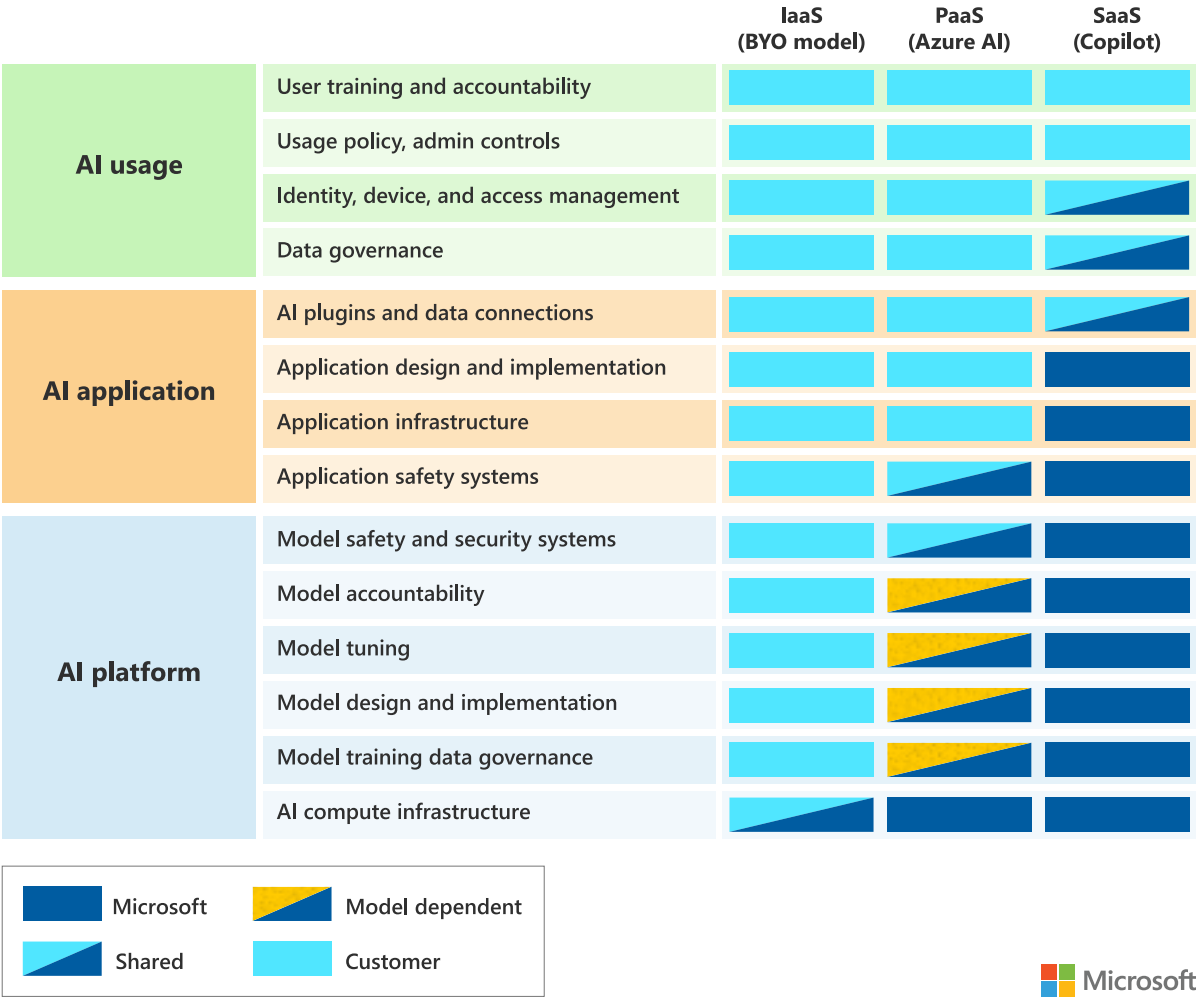
As you consider and evaluate AI enabled integration, it's critical to understand the shared responsibility model and which tasks the AI platform or application provider handle and which tasks you handle. The workload responsibilities vary depending on whether the AI integration is based on Software as a Service (SaaS), Platform as a Service (PaaS), or Infrastructure as a Service (IaaS).

Division of responsibility

As with cloud services, you have options when implementing AI capabilities for your organization. Depending on which option you choose, you take responsibility for different parts of the necessary operations and policies needed to use AI safely.

The following diagram illustrates the areas of responsibility between you and Microsoft according to the type of deployment.

AI shared responsibility model



AI layer overview

An AI enabled application consists of three layers of functionality that group together tasks, which you or an AI provider perform. The security responsibilities generally reside with whoever performs the tasks, but an AI provider might choose to expose security or other controls as a configuration option to you as appropriate. These three layers include:

AI platform

The AI platform layer provides the AI capabilities to the applications. At the platform layer, there's a need to build and safeguard the infrastructure that runs the AI model, training data, and specific configurations that change the behavior of the model, such as weights and biases. This layer provides access to functionality via APIs, which pass text known as a *Metaprompt* to the AI model for processing, then return the generated outcome, known as a *Prompt-Response*.

AI platform security considerations - To protect the AI platform from malicious inputs, a safety system must be built to filter out the potentially harmful instructions sent to the AI model (inputs). As AI models are generative, there's also a potential that some harmful content might be generated and returned to the user (outputs). Any safety system must first protect against potentially harmful inputs and outputs of many classifications including hate, jailbreaks, and others. These classifications will likely evolve over time based on model knowledge, locale, and industry.

Microsoft has built-in safety systems for both PaaS and SaaS offerings:

- PaaS - [Azure OpenAI Service](#)
- SaaS - [Microsoft Security Copilot](#) 

AI application

The AI application accesses the AI capabilities and provides the service or interface that the user consumes. The components in this layer can vary from relatively simple to highly complex, depending on the application. The simplest standalone AI applications act as an interface to a set of APIs taking a text-based user-prompt and passing that data to the model for a response. More complex AI applications include the ability to ground the user-prompt with extra context, including a persistence layer, semantic index, or via plugins to allow access to more data sources. Advanced AI applications might also interface with existing applications and systems. Existing applications and systems might work across text, audio, and images to generate various types of content.

AI application security considerations - An application safety system must be built to protect the AI application from malicious activities. The safety system provides deep inspection of the content being used in the Metaprompt sent to the AI model. The safety system also inspects the interactions with any plugins, data connectors, and other AI applications (known as AI Orchestration). One way you can incorporate this in your own IaaS/PaaS based AI application is to use the [Azure AI Content Safety](#)  service. Other capabilities are available depending on your needs.

AI usage

The AI usage layer describes how the AI capabilities are ultimately used and consumed. Generative AI offers a new type of user/computer interface that is fundamentally different from other computer interfaces, such as API, command-prompt, and graphical user interfaces (GUIs). The generative AI interface is both interactive and dynamic, allowing the computer capabilities to adjust to the user and their intent. The generative AI interface contrasts with previous interfaces that primarily force users to learn the

system design and functionality and adjust to it. This interactivity allows user input, instead of application designers, to have a high level of influence of the output of the system, making safety guardrails critical to protecting people, data, and business assets.

AI usage security considerations - Protecting AI usage is similar to any computer system as it relies on security assurances for identity and access controls, device protections and monitoring, data protection and governance, administrative controls, and other controls.

More emphasis is required on user behavior and accountability because of the increased influence users have on the output of the systems. It's critical to update acceptable use policies and educate users on the difference of standard IT applications to AI enabled applications. These should include AI specific considerations related to security, privacy, and ethics. Additionally, users should be educated on AI based attacks that can be used to trick them with convincing fake text, voices, videos, and more.

AI specific attack types are defined in:

- [Microsoft Security Response Center's \(MSRC\) vulnerability severity classification for AI systems](#) 
- [MITRE Adversarial Threat Landscape for Artificial-Intelligence Systems \(ATLAS\)](#) 
- [OWASP top 10 for Large Language Model \(LLM\) applications](#) 
- [OWASP Machine Learning \(ML\) security top 10](#) 
- [NIST AI risk management framework](#) 

Security lifecycle

As with security for other types of capability, it's critical to plan for a complete approach. A complete approach includes people, process, and technology across the full security lifecycle: identify, protect, detect, respond, recover, and govern. Any gap or weakness in this lifecycle could have you:

- Fail to secure important assets
- Experience easily preventable attacks
- Unable to handle attacks
- Unable to rapidly restore business critical services
- Apply controls inconsistently

To learn more about the unique nature of AI threat testing, read how [Microsoft AI Red Team is building the future of safer AI](#) .

Configure before customize

Microsoft recommends organizations start with SaaS based approaches like the Copilot model for their initial adoption of AI and for all subsequent AI workloads. This minimizes the level of responsibility and expertise your organization has to provide to design, operate, and secure these highly complex capabilities.

If the current "off the shelf" capabilities don't meet the specific needs for a workload, you can adopt a PaaS model by using AI services, such as [Azure OpenAI Service](#), to meet those specific requirements.

Custom model building should only be adopted by organizations with deep expertise in data science and the security, privacy, and ethical considerations of AI.

To help bring AI to the world, Microsoft is developing Copilot solutions for each of the main productivity solutions: from Bing and Windows, to GitHub and Office 365. Microsoft is developing full stack solutions for all types of productivity scenarios. These are offered as SaaS solutions. Built into the user interface of the product, they're tuned to assist the user with specific tasks to increase productivity.

Microsoft ensures that every Copilot solution is engineered following our strong principles for [AI governance](#).

Next steps

Learn more about Microsoft's product development requirements for responsible AI in the [Microsoft Responsible AI Standard](#).

Learn about [shared responsibilities for cloud computing](#).

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) | [Get help at Microsoft Q&A](#)

Zero Trust security in Azure

Zero Trust is a security strategy that assumes breach and verifies each request as though it originated from an uncontrolled network. This article explains how to apply Zero Trust principles to Microsoft Azure infrastructure and services.

For comprehensive information about Zero Trust as a security model and its application across Microsoft products, see [What is Zero Trust?](#).

Zero Trust principles for Azure

Today, organizations 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.

The Zero Trust security model is based on three guiding principles:

- **Verify explicitly** - Always authenticate and authorize based on all available data points, including user identity, location, device health, and service or workload.
- **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** - Minimize blast radius and segment access. Verify end-to-end encryption and use analytics to get visibility, drive threat detection, and improve defenses.

Applying principles to Azure workloads

When implementing Zero Trust in Azure, these principles translate into specific architectural patterns:

Verify explicitly means every access request to Azure resources must be authenticated and authorized using Microsoft Entra ID, with Conditional Access policies evaluating risk based on multiple signals including user, device, location, and workload context.

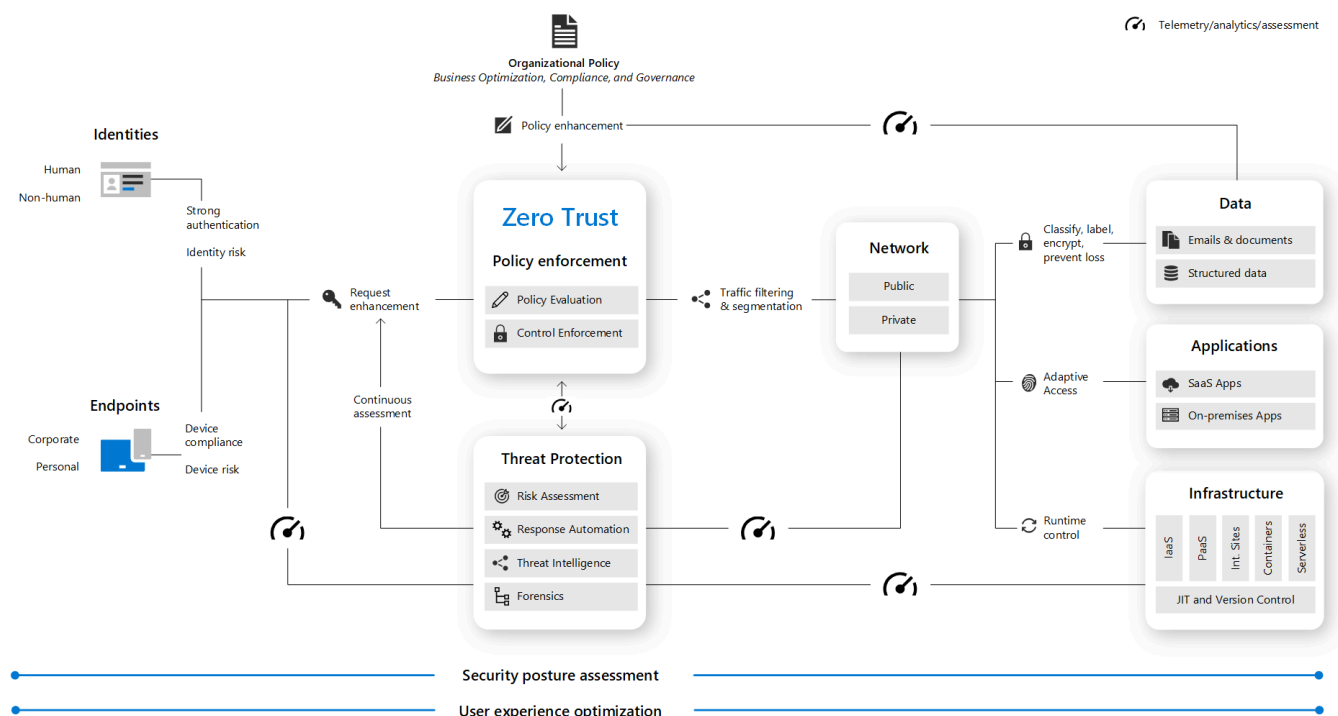
Use least privilege access requires using role-based access control (RBAC) with minimal permissions, Just-In-Time (JIT) access for administrative operations, and managed identities instead of storing credentials.

Assume breach drives network segmentation to limit lateral movement, encryption for data at rest and in transit, continuous monitoring and threat detection, and immutable backups to protect against destructive attacks.

Zero Trust architecture in Azure

A Zero Trust approach extends throughout the entire digital estate and serves as an integrated security philosophy and end-to-end strategy. When applied to Azure, it requires a multi-disciplinary approach that addresses infrastructure, networking, identity, and data protection systematically.

This illustration provides a representation of the primary elements that contribute to Zero Trust.



In the illustration:

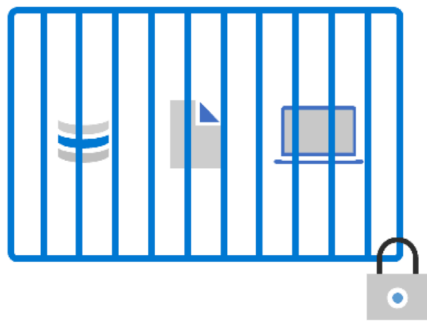
- Security policy enforcement is at the center of a Zero Trust architecture. This includes multifactor authentication with Conditional Access that takes into account user account risk, device status, and other criteria and policies that you set.
- **Identities**, **devices** (also called endpoints), **data**, **applications**, **network**, and other **infrastructure** components are all configured with appropriate security. Policies that are configured for each of these components are coordinated with your overall Zero Trust strategy.
- Threat protection and intelligence monitors the environment, surfaces current risks, and takes automated action to remediate attacks.

From perimeter-based to Zero Trust

The traditional approach of access control for IT has been based on restricting access to a corporate network perimeter. This model restricts all resources to a corporate-owned network connection and has become too restrictive to meet the needs of a dynamic enterprise.

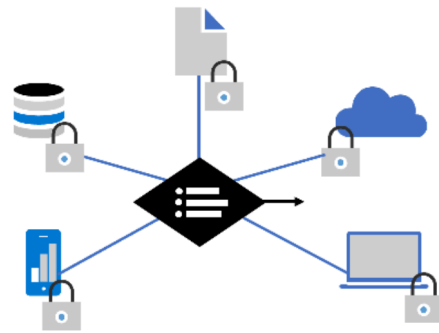
Secure assets where they are with Zero Trust

Simplify security and make it more effective



Classic Approach

Restrict everything to a 'secure' network



Zero Trust

Protect assets anywhere with central policy

In Azure environments, the shift to Zero Trust is particularly important because cloud resources exist outside traditional network perimeters. Organizations must embrace a Zero Trust approach to access control as they embrace remote work and use cloud technology to transform their business model.

Zero Trust principles help establish and continuously improve security assurances while maintaining the flexibility needed in modern cloud environments. Most Zero Trust journeys start with access control and focus on identity as the preferred and primary control. Network security technology remains a key element, but it's not the dominant approach in a complete access control strategy.

For more information on the Zero Trust transformation of access control in Azure, see the Cloud Adoption Framework's [access control](#).

Implementing Zero Trust for Azure infrastructure

Applying Zero Trust to Azure requires a methodical approach that addresses different layers of your infrastructure, from foundational elements up to complete workloads.

Azure IaaS and infrastructure components

Zero Trust for Azure IaaS addresses the complete infrastructure stack: storage services with encryption and access controls, virtual machines with trusted launch and disk encryption, spoke networks with microsegmentation, hub networks with centralized security services, and PaaS integration through private endpoints. For detailed guidance, see [Apply Zero Trust principles to Azure IaaS overview](#).

Azure networking

Network security focuses on four key areas: encryption of all network traffic, segmentation using network security groups and Azure Firewall, visibility through traffic monitoring, and discontinuing legacy VPN-based controls in favor of identity-centric approaches. For detailed guidance, see [Apply Zero Trust principles to Azure networking](#).

Identity as the control plane

Identity is the primary control plane for Zero Trust in Azure. Conditional Access serves as the main policy engine, evaluating access requests based on multiple signals to grant, limit, or block access. For more information, see [Conditional Access for Zero Trust](#) and [Azure identity management security overview](#).

Protecting data and ensuring availability

Data protection in Azure requires multiple layers: encryption at rest and in transit, identity-based access controls using managed identities and RBAC, and for highly sensitive workloads, confidential computing to protect data during processing. Resilience against destructive attacks requires resource locks, immutable backups, geo-replication, and protecting the recovery infrastructure itself. For detailed guidance, see [Protect your Azure resources from destructive cyberattacks](#).

Threat detection and response

Zero Trust requires continuous monitoring with the assumption that threats may already be present. Microsoft Defender for Cloud provides unified security management and threat protection for Azure resources, while integration with Microsoft Defender XDR enables correlated detection across your entire environment. For detailed information, see [Azure threat detection overview](#) and [Microsoft Sentinel and Microsoft Defender XDR](#).

Shared responsibility and Azure security

Security in Azure is a shared responsibility between Microsoft and customers. Microsoft secures the physical infrastructure and Azure platform, while customers are responsible for identity, data, and application security, with the division varying by service model (IaaS, PaaS, SaaS). Implementing Zero Trust requires coordinating platform-level controls with customer configuration choices. For more information, see [Shared responsibility in the cloud](#).

Azure security capabilities

While this article focuses on the conceptual application of Zero Trust to Azure, it's important to understand the breadth of security capabilities available. Azure provides comprehensive security services across all layers of your infrastructure.

For an overview of Azure's security capabilities organized by functional area, see [Introduction to Azure security](#). For a view of Azure security organized by protection, detection, and response capabilities, see [End-to-end security in Azure](#).

Additional detailed guidance is available for specific domains:

- **Identity and access** - [Azure identity management security overview](#)
- **Network security** - [Azure network security overview](#)
- **Data protection** - [Azure encryption overview](#) and [Azure Key Vault security](#)
- **Compute security** - [Azure Virtual Machines security overview](#)
- **Platform security** - [Azure platform security overview](#)
- **Threat detection** - [Azure threat detection overview](#)
- **Management and monitoring** - [Azure security management and monitoring overview](#)

Application development and Zero Trust

Applications deployed on Azure must authenticate and authorize every request rather than relying on implicit trust from network location. Key principles include using Microsoft Entra ID for identity verification, requesting minimum permissions, protecting sensitive data, and using managed identities instead of stored credentials. For comprehensive guidance, see [Develop using Zero Trust principles](#) and [Build Zero Trust-ready apps using Microsoft identity platform](#).

Next steps

To implement Zero Trust in your Azure environment, begin with these resources:

- [Apply Zero Trust principles to Azure services overview](#) - Start here for a comprehensive view of how to apply Zero Trust across different Azure service types
- [Apply Zero Trust principles to Azure IaaS overview](#) - Detailed guidance for infrastructure workloads
- [Apply Zero Trust principles to Azure networking](#) - Network security implementation guidance
- [Protect your Azure resources from destructive cyberattacks](#) - Resilience and recovery planning
- [What is Zero Trust?](#) - Comprehensive Zero Trust guidance across Microsoft products

For broader Microsoft Zero Trust resources:

- [Zero Trust deployment guidance](#) - Technology-area specific deployment objectives
 - [Zero Trust adoption framework](#) - Business-outcome focused implementation guidance
 - [Zero Trust for Microsoft 365](#) - SaaS and productivity workload guidance
-

Last updated on 11/07/2025

Ransomware protection in Azure

Ransomware and extortion are a high profit, low-cost business, which has a debilitating impact on targeted organizations, national/regional security, economic security, and public health and safety. What started as simple, single-PC ransomware grew to include various extortion techniques directed at all types of corporate networks and cloud platforms.

To ensure customers running on Azure are protected against ransomware attacks, Microsoft invests heavily in the security of our cloud platforms, and provides security controls you need to protect your Azure cloud workloads.

By using Azure native ransomware protections and implementing the best practices recommended in this article, you're taking measures that position your organization to prevent, protect, and detect potential ransomware attacks on your Azure assets.

This article lays out key Azure native capabilities and defenses for ransomware attacks and guidance on how to proactively use these to protect your assets on Azure cloud.

Tip

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#). For information about what ransomware is and how it works, see [What is ransomware?](#)

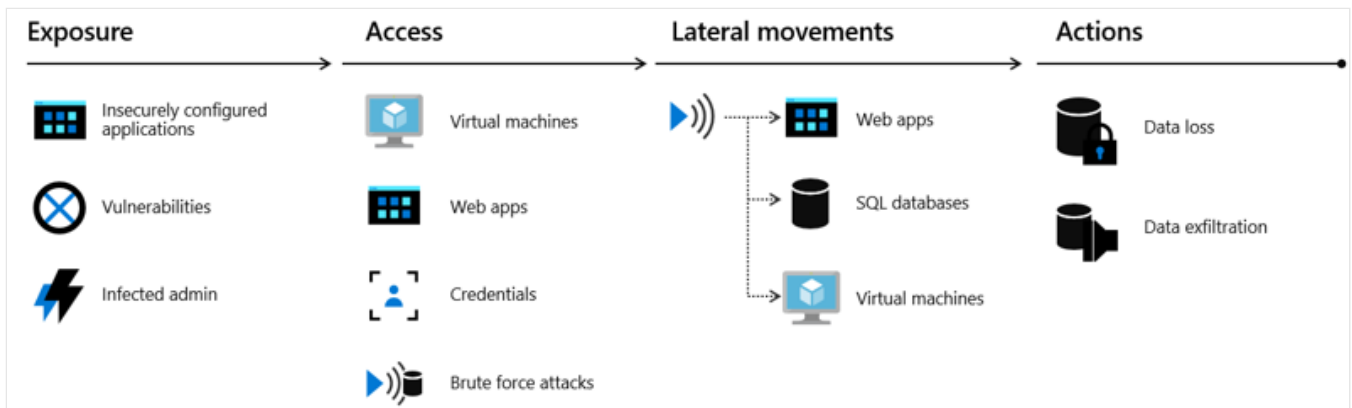
How Azure cloud resources are targeted

When attacking cloud infrastructure, adversaries often attack multiple resources to try to obtain access to customer data or company secrets. The cloud "kill chain" model explains how attackers attempt to gain access to any of your resources running in the public cloud through a four-step process: exposure, access, lateral movement, and actions.

1. Exposure is where attackers look for opportunities to gain access to your infrastructure. For example, attackers know customer-facing applications must be open for legitimate users to access them. Those applications are exposed to the Internet and therefore susceptible to attacks.
2. Attackers try to exploit an exposure to gain access to your public cloud infrastructure. This can be done through compromised user credentials, compromised instances, or misconfigured resources.
3. During the lateral movement stage, attackers discover what resources they have access to and what the scope of that access is. Successful attacks on instances give attackers access to databases and other sensitive information. The attacker then searches for other

credentials. Our Microsoft Defender for Cloud data shows that without a security tool to quickly notify you of the attack, it takes organizations on average 101 days to discover a breach. Meanwhile, in just 24-48 hours after a breach, the attacker usually have complete control of the network.

4. The actions an attacker takes after lateral movement are largely dependent on the resources they were able to gain access to during the lateral movement phase. Attackers can take actions that cause data exfiltration, data loss or launch other attacks. For enterprises, the average financial impact of data loss is now reaching \$1.23 million.



Azure-specific attack vectors

When targeting Azure environments, ransomware attackers often exploit:

- **Misconfigured Azure resources:** Publicly exposed storage accounts, databases, or virtual machines with weak access controls
- **Compromised Azure credentials:** Stolen Azure AD accounts, service principals, or managed identities that provide access to Azure resources
- **Vulnerable Azure VMs:** Unpatched virtual machines accessible via Remote Desktop Protocol (RDP) or SSH
- **Weak network security:** Improperly configured Network Security Groups (NSGs) or Azure Firewall rules
- **Inadequate backup protection:** Azure Backup configurations that lack immutability or MFA protection
- **Poor identity security:** Azure AD accounts without multifactor authentication or Conditional Access policies

Microsoft Defender for Cloud continuously monitors for these vulnerabilities in your Azure environment. For comprehensive guidance on attack techniques and defense strategies, see [What is ransomware?](#).

Azure native protections against ransomware

Azure provides built-in capabilities to defend against ransomware attacks at every stage of the attack lifecycle. The best defense against paying ransom is implementing preventive measures using Azure's robust security tools and ensuring the ability to recover impacted assets to restore business operations quickly.

Key Azure-native protection capabilities include:

- **Microsoft Defender for Cloud** - Provides threat detection and response (XDR) for Azure workloads, with ransomware-specific detection capabilities
- **Azure Backup** - Offers immutable backups with soft delete and MFA protection to ensure recovery options
- **Azure Firewall Premium** - Includes IDPS to detect and block ransomware Command & Control (C&C) communications
- **Microsoft Entra ID Protection** - Detects credential theft and suspicious authentication patterns targeting Azure resources
- **Azure Policy** - Enforces security configurations and compliance across Azure resources
- **Microsoft Sentinel** - Provides SIEM/SOAR capabilities with ransomware-specific detection analytics

For detailed information about Azure features that help protect, detect, and respond to ransomware, see [Azure features and resources that help you protect, detect, and respond](#).

Next steps

For comprehensive ransomware protection guidance across all Microsoft platforms, see [Protect your organization against ransomware and extortion](#).

Azure-specific ransomware protection articles:

- [Prepare for a ransomware attack](#)
- [Detect and respond to ransomware attack](#)
- [Azure features and resources that help you protect, detect, and respond](#)
- [Improve your security defenses for ransomware attacks with Azure Firewall Premium](#)

Additional resources:

- [What is ransomware?](#)

Prepare for a ransomware attack

This article provides Azure-specific guidance for preparing your organization to defend against and recover from ransomware attacks.

 Tip

This article focuses specifically on Azure capabilities and best practices. For comprehensive ransomware preparation guidance across all Microsoft platforms and services, see [Prepare your ransomware recovery plan](#).

Adopt a Cybersecurity framework

A good place to start is to adopt the [Microsoft cloud security benchmark \(MCSB\)](#) to secure the Azure environment. The Microsoft cloud security benchmark is the Azure security control framework, based on industry-based security control frameworks such as NIST SP800-53, CIS Controls v7.1.

NS-1: Establish network segmentation boundaries

CIS Controls v8 ID(s)	NIST SP 800-53 r4 ID(s)	PCI-DSS ID(s) v3.2.1
3.12, 13.4, 4.4	AC-4, SC-2, SC-7	1.1, 1.2, 1.3

The Microsoft cloud security benchmark provides organizations guidance on how to configure Azure and Azure Services and implement the security controls. Organizations can use [Microsoft Defender for Cloud](#) to monitor their live Azure environment status with all the MCSB controls.

Ultimately, the Framework is aimed at reducing and better managing cybersecurity risks.

 Expand table

Microsoft cloud security benchmark stack
Network security (NS)
Identity Management (IM)
Privileged Access (PA)
Data Protection (DP)

Microsoft cloud security benchmark stack
Asset Management (AM)
Logging and Threat Detection (LT)
Incident Response (IR)
Posture and Vulnerability Management (PV)
Endpoint Security (ES)
Backup and Recovery (BR)
DevOps Security (DS)
Governance and Strategy (GS)

Prioritize mitigation

Based on our experience with ransomware attacks on Azure environments, we find that prioritization should focus on:

1. Prepare - Have backups and recovery plans for your Azure resources
2. Limit - Protect privileged access to Azure resources
3. Prevent - Harden Azure security controls

This may seem counterintuitive, since most people want to prevent an attack and move on. Unfortunately, we must assume breach (a key Zero Trust principle) and focus on reliably mitigating the most damage first. This prioritization is critical because of the high likelihood of a worst-case scenario with ransomware. While it's not a pleasant truth to accept, we're facing creative and motivated human attackers who are adept at finding a way to control the complex real-world environments in which we operate. Against that reality, it's important to prepare for the worst and establish frameworks to contain and prevent attackers' ability to get what they're after.

While these priorities should govern what to do first, we encourage organizations to run steps in parallel where possible, including pulling quick wins forward from step 1 when you can.

For comprehensive guidance on the three-phase approach to ransomware protection, see [Protect your organization against ransomware and extortion](#).

Make it harder to get in

Prevent a ransomware attacker from entering your Azure environment and rapidly respond to incidents to remove attacker access before they can steal and encrypt data. This causes attackers to fail earlier and more often, undermining the profit of their attacks. While prevention is the preferred outcome, it's a continuous journey and may not be possible to achieve 100% prevention and rapid response across a real-world organization's complex multi-platform and multicloud estate with distributed IT responsibilities.

To achieve this, organizations should identify and execute quick wins to strengthen security controls for their Azure resources to prevent entry, and rapidly detect/evict attackers while implementing a sustained program that helps them stay secure. Microsoft recommends organizations follow the principles outlined in the Zero Trust strategy. Specifically, for Azure resources, organizations should prioritize:

- Improving security hygiene by focusing efforts on attack surface reduction and threat and vulnerability management for Azure resources.
- Implementing Protection, Detection and Response controls for Azure workloads that can protect against commodity and advanced threats, provide visibility, and alerting on attacker activity and respond to active threats.

For comprehensive guidance on making it harder for attackers to access your environment, see [Defend against ransomware attacks](#).

Limit scope of damage

Ensure you have strong controls (prevent, detect, respond) for privileged accounts with access to your Azure resources like IT Admins and other roles with control of business-critical systems. This slows and/or blocks attackers from gaining complete access to your Azure resources to steal and encrypt them. Taking away the attackers' ability to use IT Admin accounts as a shortcut to resources drastically lowers the chances they're successful at attacking you and demanding payment / profiting.

Organizations should have elevated security for privileged accounts with Azure access (tightly protect, closely monitor, and rapidly respond to incidents related to these roles). See Microsoft's Security rapid modernization plan, which covers:

- End to End Session Security (including multifactor authentication (MFA) for admins)
- Protect and Monitor Identity Systems
- Mitigate Lateral Traversal
- Rapid Threat Response

For comprehensive guidance on limiting the scope of damage, see [Limit the impact of ransomware attacks](#).

Prepare for the worst

Plan for the worst-case scenario and expect that it happens (at all levels of the organization). This helps your organization and others in the world you depend on:

- Limits damage for the worst-case scenario – While restoring all systems from backups is highly disruptive to business, this is more effective and efficient than trying to recovery using (low quality) attacker-provided decryption tools after paying to get the key. Note: Paying is an uncertain path – You have no formal or legal guarantee that the key works on all files, the tools work effectively, or that the attacker (who may be an amateur affiliate using a professional's toolkit) will act in good faith.
- Limit the financial return for attackers – If an organization can restore business operations without paying the attackers, the attack fails and results in zero return on investment (ROI) for the attackers. This makes it less likely that they'll target the organization in the future (and deprives them of more funding to attack others).

The attackers may still attempt to extort the organization through data disclosure or abusing/selling the stolen data, but this gives them less leverage than if they have the only access path to your data and systems.

To realize this, organizations should ensure they:

- Register Risk - Add ransomware to risk register as high likelihood and high impact scenario. Track mitigation status via Enterprise Risk Management (ERM) assessment cycle.
- Define and Backup Critical Business Assets – Define systems required for critical business operations and automatically back them up on a regular schedule (including correct backup of critical dependencies like Active Directory) Protect backups against deliberate erasure and encryption with offline storage, immutable storage, and/or out of band steps (MFA or PIN) before modifying/erasing online backups.
- Test 'Recover from Zero' Scenario – test to ensure your business continuity / disaster recovery (BC/DR) can rapidly bring critical business operations online from zero functionality (all systems down). Conduct practice exercises to validate cross-team processes and technical procedures, including out-of-band employee and customer communications (assume all email/chat/etc. is down).
It's critical to protect (or print) supporting documents and systems required for recovery including restoration procedure documents, CMDBs, network diagrams, SolarWinds instances, etc. Attackers destroy these regularly.
- Reduce on-premises exposure – by moving data to Azure cloud services with automatic backup & self-service rollback.

For comprehensive guidance on preparing for the worst case scenario, including awareness training and SOC readiness, see [Prepare your ransomware recovery plan](#).

Azure-specific technical controls for ransomware protection

Azure provides a wide variety of native technical controls to protect, detect, and respond to ransomware incidents with emphasis on prevention. Organizations running workloads in Azure should leverage these Azure-native capabilities:

Detection and prevention tools for Azure

- [Microsoft Defender for Cloud](#) - Unified security management providing threat protection for Azure workloads including VMs, containers, databases, and storage
- [Azure Firewall Premium](#) - Next-generation firewall with IDPS capabilities to detect and block ransomware C&C communications
- [Microsoft Sentinel](#) - Cloud-native SIEM/SOAR platform with built-in ransomware detection analytics and automated response
- [Azure Network Watcher](#) - Network monitoring and diagnostics to detect anomalous traffic patterns
- [Microsoft Defender for Endpoint](#) - For Azure VMs running Windows or Linux

Data protection for Azure resources

- [Azure Backup](#) with immutability and soft delete for Azure VMs, SQL databases, and file shares
- [Azure Storage immutable blobs](#) - WORM (Write Once, Read Many) storage that cannot be modified or deleted
- [Azure role-based access control \(RBAC\)](#) - Principle of least privilege for Azure resource access
- [Azure Policy](#) - Enforce backup policies and security configurations across Azure subscriptions
- **Regular backup verification** using Azure Site Recovery for disaster recovery testing

For comprehensive incident handling guidance, see [Prepare your ransomware recovery plan](#).

Azure backup and recovery capabilities

Ensure that you have appropriate processes and procedures in place for Azure workloads. Almost all ransomware incidents result in the need to restore compromised systems. Appropriate and tested backup and restore processes should be in place for Azure resources, along with suitable containment strategies to stop ransomware from spreading.

The Azure platform provides multiple backup and recovery options through Azure Backup and built-in capabilities within various Azure data services and workloads:

Isolated backups with Azure Backup

[Azure Backup](#) provides immutable, isolated backups with soft delete and MFA protection for:

- Azure Virtual Machines
- Databases in Azure VMs: SQL, SAP HANA
- Azure Database for PostgreSQL
- On-premises Windows Servers (back up to cloud using MARS agent)

Operational backups

- **Azure Files** - Share snapshots with point-in-time restore
- **Azure Blobs** - Soft delete, versioning, and immutable storage
- **Azure Disks** - Incremental snapshots

Built-in backups from Azure data services

Data services like Azure SQL Database, Azure Database for MySQL/MariaDB/PostgreSQL, Azure Cosmos DB, and Azure NetApp Files offer built-in backup capabilities with automated schedules.

For detailed guidance, see [Backup and restore plan to protect against ransomware](#).

What's Next

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#).

Other Azure ransomware articles:

- [Ransomware protection in Azure](#)
- [Detect and respond to ransomware attack](#)
- [Azure features and resources that help you protect, detect, and respond](#)
- [Improve your security defenses for ransomware attacks with Azure Firewall Premium](#)

Detect and respond to ransomware attacks

Ransomware incidents typically present with distinct warning signs that security teams can identify. Unlike other malware types, ransomware usually produces highly evident indicators that require minimal investigation before declaring an incident. These high-confidence triggers stand in contrast to more subtle threats that would demand extensive analysis before escalation. When ransomware strikes, the evidence is often unmistakable.

In general, such infections are obvious from basic system behavior, the absence of key system or user files, and the demand for ransom. In such cases, the analyst should consider whether to immediately declare and escalate the incident, including taking any automated actions to mitigate the attack.

Tip

For comprehensive ransomware detection and response guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#). This article focuses specifically on Azure-based detection and response capabilities.

Detecting ransomware attacks

Microsoft Defender for Cloud provides high-quality threat detection and response capabilities for Azure resources, also called Extended Detection and Response (XDR).

Ensure rapid detection and remediation of common attacks on Azure VMs, SQL Servers, Web applications, and identity in your Azure environment.

- **Prioritize Common Entry Points** – Ransomware (and other) operators favor Endpoint/Email/Identity + Remote Desktop Protocol (RDP)
 - **Integrated XDR** - Use integrated Extended Detection and Response (XDR) tools like Microsoft [Defender for Cloud](#)  to provide high quality alerts for Azure resources and minimize friction and manual steps during response
 - **Brute Force** - Monitor for brute-force attempts like [password spray](#) against Azure resources
- **Monitor for Adversary Disabling Security** – Often part of Human-Operated Ransomware (HumOR) attack chain
- **Event Logs Clearing** – especially the Security Event log and PowerShell Operational logs on Azure VMs
 - **Disabling of security tools/controls** (associated with some groups)

- **Don't Ignore Commodity Malware** - Ransomware attackers regularly purchase access to target organizations from dark markets
- **Integrate outside experts** – into processes to supplement expertise, such as the [Microsoft Incident Response team](#) .
- **Rapidly isolate** compromised Azure VMs using [Defender for Endpoint](#).

Responding to ransomware attacks

Incident declaration

Once a successful ransomware infection is confirmed, the analyst should verify whether it represents a new incident or if it might be related to an existing incident. Look for currently open tickets that indicate similar incidents. If so, update the current incident ticket with new information in the ticketing system. If it is a new incident, an incident should be declared in the relevant ticketing system and escalated to the appropriate teams or providers to contain and mitigate the incident. Be mindful that managing ransomware incidents might require actions taken by multiple IT and security teams. Where possible, ensure that the ticket is clearly identified as a ransomware incident to guide workflow.

Containment/Mitigation

In general, various server/endpoint anti-malware, email anti-malware, and network protection solutions should be configured to automatically contain and mitigate known ransomware. There might be cases, however, where the specific ransomware variant is able to bypass such protections and successfully infect target systems.

Microsoft provides extensive resources to help update your incident response processes on the [Top Azure Security Best Practices](#).

The following are recommended actions to contain or mitigate a declared incident involving ransomware where automated actions taken by anti-malware systems are unsuccessful:

1. Engage anti-malware vendors through standard support processes
2. Manually add hashes and other information associated with malware to anti-malware systems
3. Apply anti-malware vendor updates
4. Contain affected systems until they can be remediated
5. Disable compromised accounts
6. Perform root cause analysis
7. Apply relevant patches and configuration changes on affected systems

8. Block ransomware communications using internal and external controls
9. Purge cached content

Road to recovery

The Microsoft Incident Response team helps protect you from attacks.

Understanding and fixing the fundamental security issues that led to the compromise in the first place should be a priority for ransomware targets.

Integrate outside experts into processes to supplement expertise, such as [Microsoft Incident Response](#). Microsoft Incident Response engages with customers around the world, helping to protect and harden Azure environments against attacks before they occur, as well as investigating and remediating when an attack has occurred.

Customers can engage our security experts directly from within the Microsoft Defender Portal for timely and accurate response. Experts provide insights needed to better understand the complex threats affecting your organization, from alert inquiries, potentially compromised devices, root cause of a suspicious network connection, to additional threat intelligence regarding ongoing advanced persistent threat campaigns.

Microsoft is ready to assist your company in returning to safe operations.

Microsoft performs hundreds of compromise recoveries and has a tried-and-true methodology. Not only will it get you to a more secure position, it affords you the opportunity to consider your long-term strategy rather than reacting to the situation.

Microsoft provides Rapid Ransomware Recovery services. Under this, assistance is provided in all areas such as restoration of identity services, remediation and hardening, and monitoring deployment to help targets of ransomware attacks return to normal business in the shortest possible timeframe.

Our Rapid Ransomware Recovery services are treated as "Confidential" for the duration of the engagement. Rapid Ransomware Recovery engagements are exclusively delivered by the Microsoft Incident Response team. For more information, you can contact Microsoft Incident Response at [Request contact about Azure security](#).

What's next

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#).

Other Azure ransomware articles:

- [Ransomware protection in Azure](#)
 - [Prepare for a ransomware attack](#)
 - [Azure features and resources that help you protect, detect, and respond](#)
 - [Improve your security defenses for ransomware attacks with Azure Firewall Premium](#)
-

Last updated on 01/06/2026

Azure features & resources that help you protect, detect, and respond to ransomware attacks

Microsoft has invested in Azure native security capabilities that organizations can leverage to defeat ransomware attack techniques found in both high-volume, everyday attacks, and sophisticated targeted attacks.



Tip

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#). This article focuses specifically on Azure native features and resources.

Key capabilities include:

- **Native Threat Detection:** Microsoft Defender for Cloud provides high-quality threat detection and response capabilities, also called Extended Detection and Response (XDR). This helps you:
 - Avoid wasting time and talent of scarce security resources to build custom alerts using raw activity logs.
 - Ensure effective security monitoring, which often enables security teams to rapidly approve use of Azure services.
- **Passwordless and multifactor authentication:** Microsoft Entra multifactor authentication, Microsoft Entra Authenticator App, and Windows Hello provide these capabilities. This helps protect accounts against commonly seen password attacks (which account for 99.9% of the volume of identity attacks we see in Microsoft Entra ID). While no security is perfect, eliminating password-only attack vectors dramatically lowers the ransomware attack risk to Azure resources.
- **Native Firewall and Network Security:** Microsoft built native DDoS attack mitigations, Firewall, Web Application Firewall, and many other controls into Azure. These security 'as a service' help simplify the configuration and implementation of security controls. These give organizations the choice of using native services or virtual appliances versions of familiar vendor capabilities to simplify their Azure security.

Microsoft Defender for Cloud

Microsoft Defender for Cloud is a built-in tool that provides threat protection for workloads running in Azure, on-premises, and in other clouds. It protects your hybrid data, cloud native

services, and servers from ransomware and other threats; and integrates with your existing security workflows like your SIEM solution and Microsoft's vast threat intelligence to streamline threat mitigation.

Microsoft Defender for Cloud delivers protection for all resources from directly within the Azure experience and extends protection to on-premises and multi-cloud virtual machines and SQL databases using Azure Arc:

- Protects Azure services
- Protects hybrid workloads
- Streamline security with AI and automation
- Detects and blocks advanced malware and threats for Linux and Windows servers on any cloud
- Protects cloud-native services from threats
- Protects data services against ransomware attacks
- Protects your managed and unmanaged IoT and OT devices, with continuous asset discovery, vulnerability management, and threat monitoring

Microsoft Defender for Cloud provides you the tools to detect and block ransomware, advanced malware and threats for your resources

Keeping your resources safe is a joint effort between your cloud provider, Azure, and you, the customer. You have to make sure your workloads are secure as you move to the cloud, and at the same time, when you move to IaaS (infrastructure as a service) there is more customer responsibility than there was in PaaS (platform as a service), and SaaS (software as a service). Microsoft Defender for Cloud provides you the tools needed to harden your network, secure your services and make sure you're on top of your security posture.

Microsoft Defender for Cloud is a unified infrastructure security management system that strengthens the security posture of your data centers and provides advanced threat protection across your hybrid workloads in the cloud whether they're in Azure or not - as well as on premises.

Defender for Cloud's threat protection enables you to detect and prevent threats at the Infrastructure as a Service (IaaS) layer, non-Azure servers as well as for Platforms as a Service (PaaS) in Azure.

Defender for Cloud's threat protection includes fusion kill-chain analysis, which automatically correlates alerts in your environment based on cyber kill-chain analysis, to help you better understand the full story of an attack campaign, where it started and what kind of impact it had on your resources.

Key Features:

- Continuous security assessment: Identify Windows and Linux machines with missing security updates or insecure OS settings and vulnerable Azure configurations. Add optional watchlists or events you want to monitor.
- Actionable recommendations: Remediate security vulnerabilities quickly with prioritized, actionable security recommendations.
- Centralized policy management: Ensure compliance with company or regulatory security requirements by centrally managing security policies across all your hybrid cloud workloads.
- Industry's most extensive threat intelligence: Tap into the Microsoft Intelligent Security Graph, which uses trillions of signals from Microsoft services and systems around the globe to identify new and evolving threats.
- Advanced analytics and machine learning: Use built-in behavioral analytics and machine learning to identify known attack patterns and post-breach activity.
- Adaptive application control: Block malware and other unwanted applications by applying allowlist recommendations adapted to your specific workloads and powered by machine learning.
- Prioritized alerts and attack timelines: Focus on the most critical threats first with prioritized alerts and incidents that are mapped into a single attack campaign.
- Streamlined investigation: Quickly investigate the scope and impact of an attack with a visual, interactive experience. Use ad hoc queries for deeper exploration of security data.
- Automation and orchestration: Automate common security workflows to address threats quickly using built-in integration with Azure Logic Apps. Create security playbooks that can route alerts to existing ticketing system or trigger incident response actions.

Microsoft Sentinel

Microsoft Sentinel helps to create a complete view of a kill chain

With Sentinel, you can connect to any of your security sources using built-in connectors and industry standards and then take advantage of artificial intelligence to correlate multiple low fidelity signals spanning multiple sources to create a complete view of a ransomware kill chain and prioritized alerts so that defenders can accelerate their time to evict adversaries.

Microsoft Sentinel is your birds-eye view across the enterprise alleviating the stress of increasingly sophisticated attacks, increasing volumes of alerts, and long resolution time frames.

Collect data at cloud scale across all users, devices, applications, and infrastructure, both on-premises and in multiple clouds.

Detect previously undetected threats, and minimize [false positives](#) using Microsoft's analytics and unparalleled threat intelligence.

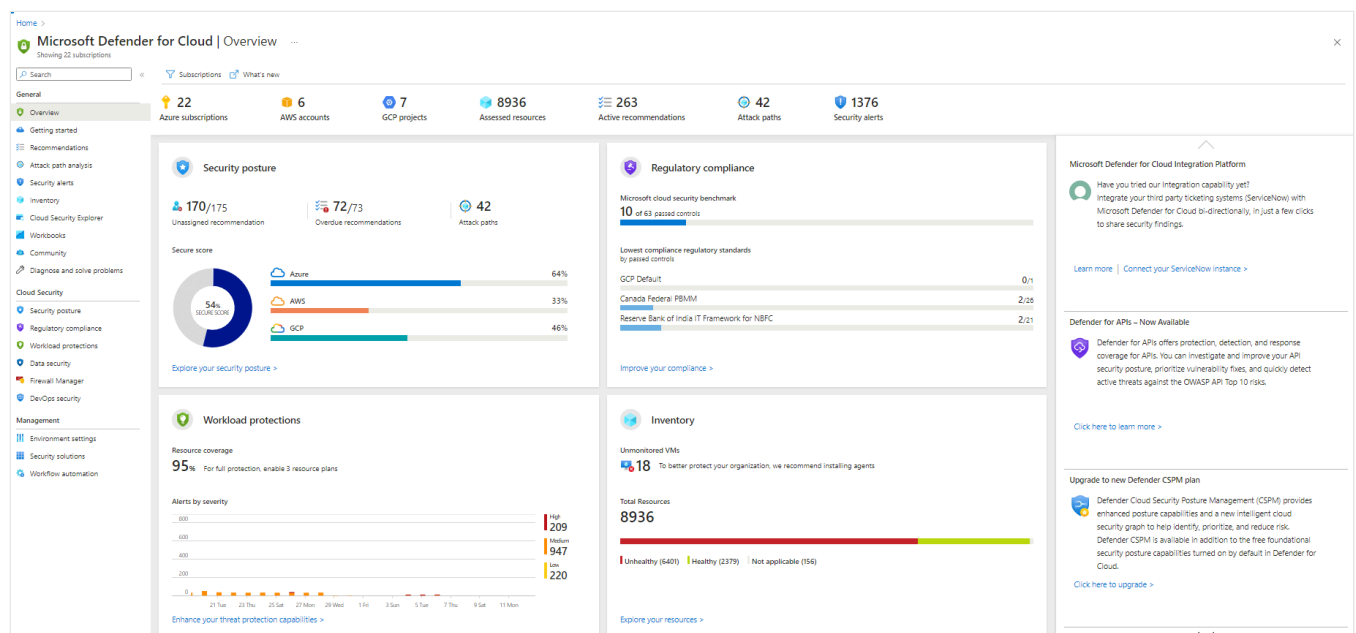
Investigate threats with artificial intelligence, and hunt for suspicious activities at scale, tapping into years of Cyber security work at Microsoft.

Respond to incidents rapidly with built-in orchestration and automation of common tasks.

Native threat prevention with Microsoft Defender for Cloud

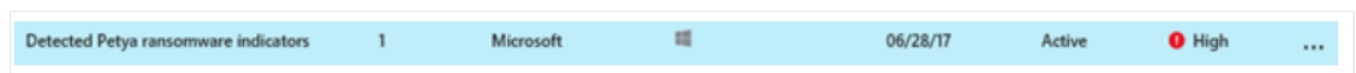
Microsoft Defender for Cloud scans virtual machines across an Azure subscription and makes a recommendation to deploy endpoint protection where an existing solution is not detected.

This recommendation can be accessed via the Recommendations section:



Microsoft Defender for Cloud provides security alerts and advanced threat protection for virtual machines, SQL databases, containers, web applications, your network, and more. When Microsoft Defender for Cloud detects a threat in any area of your environment, it generates a security alert. These alerts describe details of the affected resources, suggested remediation steps, and in some cases an option to trigger a logic app in response.

This alert is an example of a detected Petya ransomware alert:



Azure native backup solution protects Your data

One important way that organizations can help protect against losses in a ransomware attack is to have a backup of business-critical information in case other defenses fail. Since ransomware attackers have invested heavily into neutralizing backup applications and operating system

features like volume shadow copy, it is critical to have backups that are inaccessible to a malicious attacker. With a flexible business continuity and disaster recovery solution, industry-leading data protection and security tools, Azure cloud offers secure services to protect your data:

- **Azure Backup:** Azure Backup service provides simple, secure, and cost-effective solution to back up your Azure VM. Currently, Azure Backup supports backing up of all the disks (OS and Data disks) in a VM using backup solution for Azure virtual machine.
- **Azure Disaster Recovery:** With disaster recovery from on-prem to the cloud, or from one cloud to another, you can avoid downtime and keep your applications up and running.
- **Built-in Security and Management in Azure:** To be successful in the Cloud era, enterprises must have visibility/metrics and controls on every component to pinpoint issues efficiently, optimize and scale effectively, while having the assurance the security, compliance and policies are in place to ensure the velocity.

Guaranteed and Protected Access to Your Data

Azure has a lengthy period of experience managing global data centers, which are backed by Microsoft's \$15 billion-infrastructure investment that is under continuous evaluation and improvement – with ongoing investments and improvements, of course.

Key Features:

- Azure comes with Locally Redundant Storage (LRS), where data is stored locally, as well as Geo Redundant Storage (GRS) in a second region
- All data stored on Azure is protected by an advanced encryption process, and all Microsoft's data centers have two-tier authentication, proxy card access readers, biometric scanners
- Azure has more certifications than any other public cloud provider on the market, including ISO 27001, HIPAA, FedRAMP, SOC 1, SOC 2, and many international specifications

Additional resources

- [Microsoft Cloud Adoption Framework for Azure](#)
- [Build great solutions with the Microsoft Azure Well-Architected Framework](#)
- [Azure Top Security Best Practices](#)
- [Security Baselines](#) 
- [Microsoft Azure Resource Center](#) 
- [Azure Migration Guide](#)
- [Security Compliance Management](#)

- [Azure Security Control – Incident Response](#)
- [Zero Trust Guidance Center](#)
- [Azure Web Application Firewall](#)
- [Azure VPN gateway](#)
- [Microsoft Entra multifactor authentication \(MFA\)](#)
- [Microsoft Entra ID Protection](#)
- [Microsoft Entra Conditional Access](#)
- [Microsoft Defender for Cloud documentation](#)

Conclusion

Microsoft focuses heavily on both security of our cloud and providing you the security controls you need to protect your cloud workloads. As a leader in cybersecurity, we embrace our responsibility to make the world a safer place. This is reflected in our comprehensive approach to ransomware prevention and detection in our security framework, designs, products, legal efforts, industry partnerships, and services.

We look forward to partnering with you in addressing ransomware protection, detection, and prevention in a holistic manner.

Connect with us:

- AskAzureSecurity@microsoft.com
- www.microsoft.com/services [↗](#)

For detailed information on how Microsoft secures our cloud, visit the [service trust portal](#) [↗](#).

What's Next

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#).

Other Azure ransomware articles:

- [Ransomware protection in Azure](#)
- [Prepare for a ransomware attack](#)
- [Detect and respond to ransomware attack](#)
- [Improve your security defenses for ransomware attacks with Azure Firewall Premium](#)

Backup and restore plan to protect against ransomware

Ransomware attacks deliberately encrypt or erase data and systems to force your organization to pay money to attackers. These attacks target your data, your backups, and also key documentation required for you to recover without paying the attackers (as a means to increase the chances your organization will pay).

This article addresses what to do before an attack to protect your critical business systems and during an attack to ensure a rapid recovery of business operations.

What is ransomware?

Ransomware is a type of extortion attack that encrypts files and folders, preventing access to important data and systems. Attackers use ransomware to extort money from victims by demanding money, usually in the form of cryptocurrencies, in exchange for a decryption key or in exchange for not releasing sensitive data to the dark web or the public internet.

While early ransomware mostly used malware that spread with phishing or between devices, human-operated ransomware has emerged where a gang of active attackers, driven by human attack operators, target all systems in an organization (rather than a single device or set of devices). An attack can:

- Encrypt your data
- Exfiltrate your data
- Corrupt your backups

The ransomware leverages the attackers' knowledge of common system and security misconfigurations and vulnerabilities to infiltrate the organization, navigate the enterprise network, and adapt to the environment and its weaknesses as they go.

Ransomware can be staged to exfiltrate your data first, over several weeks or months, before the ransomware actually executes on a specific date.

Ransomware can also slowly encrypt your data while keeping your key on the system. With your key still available, your data is usable to you and the ransomware goes unnoticed. Your backups, though, are of the encrypted data. Once all of your data is encrypted and recent backups are also of encrypted data, your key is removed so you can no longer read your data.

The real damage is often done when the attack exfiltrates files while leaving backdoors in the network for future malicious activity—and these risks persist whether or not the ransom is paid. These attacks can be catastrophic to business operations and difficult to clean up, requiring

complete adversary eviction to protect against future attacks. Unlike early forms of ransomware that only required malware remediation, human-operated ransomware can continue to threaten your business operations after the initial encounter.

Impact of an attack

The impact of a ransomware attack on any organization is difficult to quantify accurately. Depending on the scope of the attack, the impact could include:

- Loss of data access
- Business operation disruption
- Financial loss
- Intellectual property theft
- Compromised customer trust or tarnished reputation
- Legal expenses

How can you protect yourself?

The best way to prevent falling victim to ransomware is to implement preventive measures and have tools that protect your organization from every step that attackers take to infiltrate your systems.

You can reduce your on-premises exposure by moving your organization to a cloud service. Microsoft has invested in native security capabilities that make Microsoft Azure resilient against ransomware attacks and helps organizations defeat ransomware attack techniques. For a comprehensive view of ransomware and extortion and how to protect your organization, use the information in the [Human-Operated Ransomware Mitigation Project Plan](#)  PowerPoint presentation.

You should assume that at some point in time you'll fall victim to a ransomware attack. One of the most important steps you can take to protect your data and avoid paying a ransom is to have a reliable backup and restore plan for your business-critical information. Since ransomware attackers have invested heavily into neutralizing backup applications and operating system features like volume shadow copy, it's critical to have backups that are inaccessible to a malicious attacker.

Azure Backup

[Azure Backup](#) provides security to your backup environment, both when your data is in transit and at rest. With Azure Backup, [you can back up](#):

- On-premises files, folders, and system state

- Entire Windows/Linux VMs
- Azure Managed Disks
- Azure file shares to a storage account
- SQL Server databases running on Azure VMs

The backup data is stored in Azure storage and the guest or attacker has no direct access to backup storage or its contents. With virtual machine backup, the backup snapshot creation and storage is done by Azure fabric where the guest or attacker has no involvement other than quiescing the workload for application consistent backups. With SQL and SAP HANA, the backup extension gets temporary access to write to specific blobs. In this way, even in a compromised environment, existing backups can't be tampered with or deleted by the attacker.

Azure Backup provides built-in monitoring and alerting capabilities to view and configure actions for events related to Azure Backup. Backup Reports serve as a one-stop destination for tracking usage, auditing of backups and restores, and identifying key trends at different levels of granularity. Using Azure Backup's monitoring and reporting tools can alert you to any unauthorized, suspicious, or malicious activity as soon as they occur.

Checks have been added to make sure only valid users can perform various operations. These include adding an extra layer of authentication. As part of adding an extra layer of authentication for critical operations, you're prompted to enter a security PIN before [modifying online backups](#).

Learn more about the [security features](#) built into Azure Backup.

Validate backups

Validate that your backup is good as your backup is created and before you restore. We recommend that you use a [Recovery Services vault](#), which is a storage entity in Azure that houses data. The data is typically copies of data, or configuration information for virtual machines (VMs), workloads, servers, or workstations. You can use Recovery Services vaults to hold backup data for various Azure services such as IaaS VMs (Linux or Windows) and Azure SQL databases as well as on-premises assets. Recovery Services vaults make it easy to organize your backup data and provide features such as:

- Enhanced capabilities to ensure you can secure your backups, and safely recover data, even if production and backup servers are compromised. [Learn more](#).
- Monitoring for your hybrid IT environment (Azure IaaS VMs and on-premises assets) from a central portal. [Learn more](#).
- Compatibility with Azure role-based access control (Azure RBAC), which restricts backup and restore access to a defined set of user roles. Azure RBAC provides various built-in

roles, and Azure Backup has three built-in roles to manage recovery points. [Learn more](#).

- Soft delete protection, even if a malicious actor deletes a backup (or backup data is accidentally deleted). Backup data is retained for 14 additional days, allowing the recovery of a backup item with no data loss. [Learn more](#).
- Cross Region Restore which allows you to restore Azure VMs in a secondary region, which is an Azure paired region. You can restore the replicated data in the secondary region any time. This enables you to restore the secondary region data for audit-compliance, and during outage scenarios, without waiting for Azure to declare a disaster (unlike the GRS settings of the vault). [Learn more](#).

ⓘ Note

There are two types of vaults in Azure Backup. In addition to the Recovery Services vaults, there are also [Backup vaults](#) that house data for newer workloads supported by Azure Backup.

What to do before an attack

As mentioned earlier, you should assume that at some point in time you'll fall victim to a ransomware attack. Identifying your business-critical systems and applying best practices before an attack will get you back up and running as quickly as possible.

Determine what is most important to you

Ransomware can attack while you're planning for an attack so your first priority should be to identify the business-critical systems that are most important to you and begin performing regular backups on those systems.

In our experience, the five most important applications to customers fall into the following categories in this priority order:

- Identity systems – required for users to access any systems (including all others described below) such as Active Directory, [Microsoft Entra Connect](#), AD domain controllers
- Human life – any system that supports human life or could put it at risk such as medical or life support systems, safety systems (ambulance, dispatch systems, traffic light control), large machinery, chemical/biological systems, production of food or personal products, and others
- Financial systems – systems that process monetary transactions and keep the business operating, such as payment systems and related databases, financial system for quarterly reporting

- Product or service enablement – any systems that are required to provide the business services or produce/deliver physical products that your customers pay you for, factory control systems, product delivery/dispatch systems, and similar
- Security (minimum) – You should also prioritize the security systems required to monitor for attacks and provide minimum security services. This should be focused on ensuring that the current attacks (or easy opportunistic ones) aren't immediately able to gain (or regain) access to your restored systems

Your prioritized back up list also becomes your prioritized restore list. Once you've identified your critical systems and are performing regular backups, then take steps to reduce your exposure level.

Steps to take before an attack

Apply these best practices before an attack.

 Expand table

Task	Detail
Identify the important systems that you need to bring back online first (using top five categories above) and immediately begin performing regular backups of those systems.	To get back up and running as quickly as possible after an attack, determine today what is most important to you.
Migrate your organization to the cloud. Consider purchasing a Microsoft Unified Support plan or working with a Microsoft partner to help support your move to the cloud.	Reduce your on-premises exposure by moving data to cloud services with automatic backup and self-service rollback. Microsoft Azure has a robust set of tools to help you back up your business-critical systems and restore your backups faster. Microsoft Unified Support  is a cloud services support model that is there to help you whenever you need it. Unified Support: Provides a designated team that is available 24x7 with as-needed problem resolution and critical incident escalation Helps you monitor the health of your IT environment and works proactively to make sure problems are prevented before they happen
Move user data to cloud solutions like OneDrive and SharePoint to take advantage of versioning and recycle bin capabilities .	User data in the Microsoft cloud can be protected by built-in security and data management features. It's good to teach users how to restore their own files but you need to be careful that your users don't restore the malware

Task	Detail
Educate users on how to recover their files by themselves to reduce delays and cost of recovery. For example, if a user's OneDrive files were infected by malware, they can restore their entire OneDrive to a previous time. Consider a defense strategy, such as Microsoft Defender XDR, before allowing users to restore their own files.	used to carry out the attack. You need to: Ensure your users don't restore their files until you're confident that the attacker has been evicted Have a mitigation in place in case a user does restore some of the malware Microsoft Defender XDR uses AI-powered automatic actions and playbooks to remediate impacted assets back to a secure state. Microsoft Defender XDR leverages automatic remediation capabilities of the suite products to ensure all impacted assets related to an incident are automatically remediated where possible.
Implement the Microsoft cloud security benchmark.	The Microsoft cloud security benchmark is our security control framework based on industry-based security control frameworks such as NIST SP800-53, CIS Controls v7.1. It provides organizations guidance on how to configure Azure and Azure services and implement the security controls. See Backup and Recovery.
Regularly exercise your business continuity/disaster recovery (BC/DR) plan. Simulate incident response scenarios. Exercises you perform in preparing for an attack should be planned and conducted around your prioritized backup and restore lists. Regularly test 'Recover from Zero' scenario to ensure your BC/DR can rapidly bring critical business operations online from zero functionality (all systems down).	Ensures rapid recovery of business operations by treating a ransomware or extortion attack with the same importance as a natural disaster. Conduct practice exercise(s) to validate cross-team processes and technical procedures, including out of band employee and customer communications (assume all email and chat is down).
Consider creating a risk register to identify potential risks and address how you'll mediate through preventative controls and actions. Add ransomware to risk register as high likelihood and high impact scenario.	A risk register can help you prioritize risks based on the likelihood of that risk occurring and the severity to your business should that risk occur. Track mitigation status via Enterprise Risk Management (ERM) assessment cycle.
Back up all critical business systems automatically on a regular schedule (including backup of critical	Allows you to recover data up to the last backup.

Task	Detail
dependencies like Active Directory).	
Validate that your backup is good as your backup is created.	
Protect (or print) supporting documents and systems required for recovery such as restoration procedure documents, CMDB, network diagrams, and SolarWinds instances.	Attackers deliberately target these resources because it impacts your ability to recover.
Ensure you have well-documented procedures for engaging any third-party support, particularly support from threat intelligence providers, antimalware solution providers, and from the malware analysis provider. Protect (or print) these procedures.	Third-party contacts may be useful if the given ransomware variant has known weaknesses or decryption tools are available.
Ensure backup and recovery strategy includes: Ability to back up data to a specific point in time. Multiple copies of backups are stored in isolated, offline (air-gapped) locations. Recovery time objectives that establish how quickly backed up information can be retrieved and put into production environment. Rapid restore of back up to a production environment/sandbox.	Backups are essential for resilience after an organization has been breached. Apply the 3-2-1 rule for maximum protection and availability: 3 copies (original + 2 backups), 2 storage types, and 1 offsite or cold copy.
Protect backups against deliberate erasure and encryption:	Backups that are accessible by attackers can be rendered unusable for business recovery.
Store backups in offline or off-site storage and/or immutable storage.	Offline storage ensures robust transfer of backup data without using any network bandwidth. Azure Backup supports offline backup , which transfers initial backup data offline, without the use of network bandwidth. It provides a mechanism to copy backup data onto physical storage devices. The devices are then shipped to a nearby Azure datacenter and uploaded onto a Recovery Services vault .
Require out of band steps (such as MFA or a security PIN) before permitting an online backup to be modified or erased.	
Create private endpoints within your Azure Virtual Network to securely back	Online immutable storage (such as Azure Blob) enables you to

Task	Detail
up and restore data from your Recovery Services vault.	store business-critical data objects in a WORM (Write Once, Read Many) state. This state makes the data non-erasable and non-modifiable for a user-specified interval. Multifactor authentication (MFA) should be mandatory for all admin accounts and is strongly recommended for all users. The preferred method is to use an authenticator app rather than SMS or voice where possible. When you set up Azure Backup you can configure your recovery services to enable MFA using a security PIN generated in the Azure portal. This ensures that a security pin is generated to perform critical operations such as updating or removing a recovery point.
Designate protected folders .	Makes it more difficult for unauthorized applications to modify the data in these folders.
Review your permissions: Discover broad write/delete permissions on file shares, SharePoint, and other solutions. Broad is defined as many users having write/delete permissions for business-critical data. Reduce broad permissions while meeting business collaboration requirements. Audit and monitor to ensure broad permissions don't reappear.	Reduces risk from broad access-enabling ransomware activities.
Protect against a phishing attempt: Conduct security awareness training regularly to help users identify a phishing attempt and avoid clicking on something that can create an initial entry point for a compromise. Apply security filtering controls to email to detect and minimize the likelihood of a successful phishing attempt.	The most common method used by attackers to infiltrate an organization is phishing attempts via email. Exchange Online Protection (EOP) is the cloud-based filtering service that protects your organization against spam, malware, and other email threats. EOP is included in all Microsoft 365 organizations with Exchange Online mailboxes. An example of a security filtering control for email is Safe Links. Safe Links is a feature in Defender for Office 365 that provides scanning and rewriting of URLs and links in email messages during inbound mail flow, and time-of-click verification of URLs and links in email messages and other locations (Microsoft Teams and Office documents). Safe Links scanning occurs in addition to the regular anti-spam and anti-malware protection in inbound email messages in EOP. Safe Links scanning can help protect your organization from malicious links that are used in phishing and other attacks.

Task	Detail
Learn more about anti-phishing protection .	

What to do during an attack

If you're attacked, your prioritized back up list becomes your prioritized restore list. Before you restore, validate again that your backup is good. You may be able to look for malware inside the backup.

Steps to take during an attack

Apply these best practices during an attack.

 Expand table

Task	Detail
Early in the attack, engage third-party support, particularly support from threat intelligence providers, antimalware solution providers and from the malware analysis provider.	These contacts may be useful if the given ransomware variant has a known weakness or decryption tools are available. The Microsoft Incident Response team can help protect you from attacks. Microsoft Incident Response engages with customers around the world, helping to protect and harden against attacks before they occur, as well as investigating and remediating when an attack has occurred. Microsoft also provides Rapid Ransomware Recovery services. Services are exclusively delivered by the Microsoft Global Compromise Recovery Security Practice (CRSP). The focus of this team during a ransomware attack is to restore authentication service and limit the impact of ransomware. Microsoft Incident Response is part of Microsoft's Industry Solutions Delivery security service line.
Contact your local or federal law enforcement agencies.	If you're in the United States, contact the FBI to report a ransomware breach using the IC3 Complaint Referral Form .
Take steps to remove malware or ransomware payload from your environment and stop the spread.	You can use Windows Defender. An alternative that will also help you remove ransomware or malware is the Malicious Software Removal Tool (MSRT).
Run a full, current antivirus scan on all suspected computers and devices to detect and remove the payload that's	

Task	Detail
associated with the ransomware. Scan devices that are synchronizing data, or the targets of mapped network drives.	
Restore business-critical systems first. Remember to validate again that your backup is good before you restore.	At this point, you don't need to restore everything. Focus on the top five business-critical systems from your restore list.
If you have offline backups, you can probably restore the encrypted data after you've removed the ransomware payload (malware) from your environment.	To prevent future attacks, ensure ransomware or malware is not on your offline backup before restoring.
Identify a safe point-in-time backup image that is known not to be infected. If you use Recovery Services vault, carefully review the incident timeline to understand the right point-in-time to restore a backup.	To prevent future attacks, scan backup for ransomware or malware before restoring.
Use a safety scanner and other tools for full operating system restore as well as data restore scenarios.	Microsoft Safety Scanner is a scan tool designed to find and remove malware from Windows computers. Simply download it and run a scan to find malware and try to reverse changes made by identified threats.
Ensure that your antivirus or endpoint detection and response (EDR) solution is up to date. You also need to have up-to-date patches.	An EDR solution, such as Microsoft Defender for Endpoint, is preferred.
After business-critical systems are up and running, restore other systems. As systems get restored, start collecting telemetry data so you can make formative decisions about what you're restoring.	Telemetry data should help you identify if malware is still on your systems.

Post attack or simulation

After a ransomware attack or an incident response simulation, take the following steps to improve your backup and restore plans as well as your security posture:

1. Identify lessons learned where the process didn't work well (and opportunities to simplify, accelerate, or otherwise improve the process)
2. Perform root cause analysis on the biggest challenges (at enough detail to ensure solutions address the right problem — considering people, process, and technology)
3. Investigate and remediate the original breach (engage the [Microsoft Incident Response team \(formerly DART\)](#) [↗] to help)
4. Update your backup and restore strategy based on lessons learned and opportunities — prioritizing based on highest impact and quickest implementation steps first

Next steps

For best practices on deploying ransomware protection, see [Rapidly protect against ransomware and extortion](#).

Key industry information:

- [2023 Microsoft Digital Defense Report](#) [↗] (see pages 17-26)

Microsoft Azure:

- [Help protect from ransomware with Microsoft Azure Backup](#) [↗] (26-minute video)
- [Recovering from systemic identity compromise](#)
- [Advanced multistage attack detection in Microsoft Sentinel](#)

Microsoft 365:

- [Recover from a ransomware attack](#)
- [Malware and ransomware protection](#)
- [Protect your Windows 10 PC from ransomware](#) [↗]
- [Handling ransomware in SharePoint Online](#)

Microsoft Defender XDR:

- [Find ransomware with advanced hunting](#)

Improve your security defenses for ransomware attacks with Azure Firewall Premium

In this article, you learn how Azure Firewall Premium can help you protect against ransomware.

Tip

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#). This article focuses specifically on Azure Firewall Premium capabilities for ransomware protection.

What is ransomware?

Ransomware is a type of malicious software designed to block access to your computer system until a sum of money is paid. The attacker usually exploits an existing vulnerability in your system to penetrate your network and execute the malicious software on the target host.

Ransomware is often spread through phishing emails that contain malicious attachments or through drive-by downloading. Drive-by downloading occurs when a user unknowingly visits an infected website and then malware is downloaded and installed without the user's knowledge.

Protect from network malicious activity

A network intrusion detection and prevention system (IDPS) allows you to monitor your network for malicious activity, log information about this activity, report it, and optionally attempt to block it.

[Azure Firewall Premium](#) provides signature-based IDPS where every packet is inspected thoroughly, including all its headers and payload, to identify a malicious activity and to prevent it from penetrating into your network.

The IDPS signatures are applicable for both application and network level traffic (Layers 4-7), fully managed, and contain more than 65,000 signatures in over 50 different categories. To keep them (the IDPS signatures?) up to date with the dynamic ever-changing attack landscape:

- Azure Firewall has early access to vulnerability information from [Microsoft Active Protections Program \(MAPP\)](#) [↗] and [Microsoft Security Response Center \(MSRC\)](#) [↗].

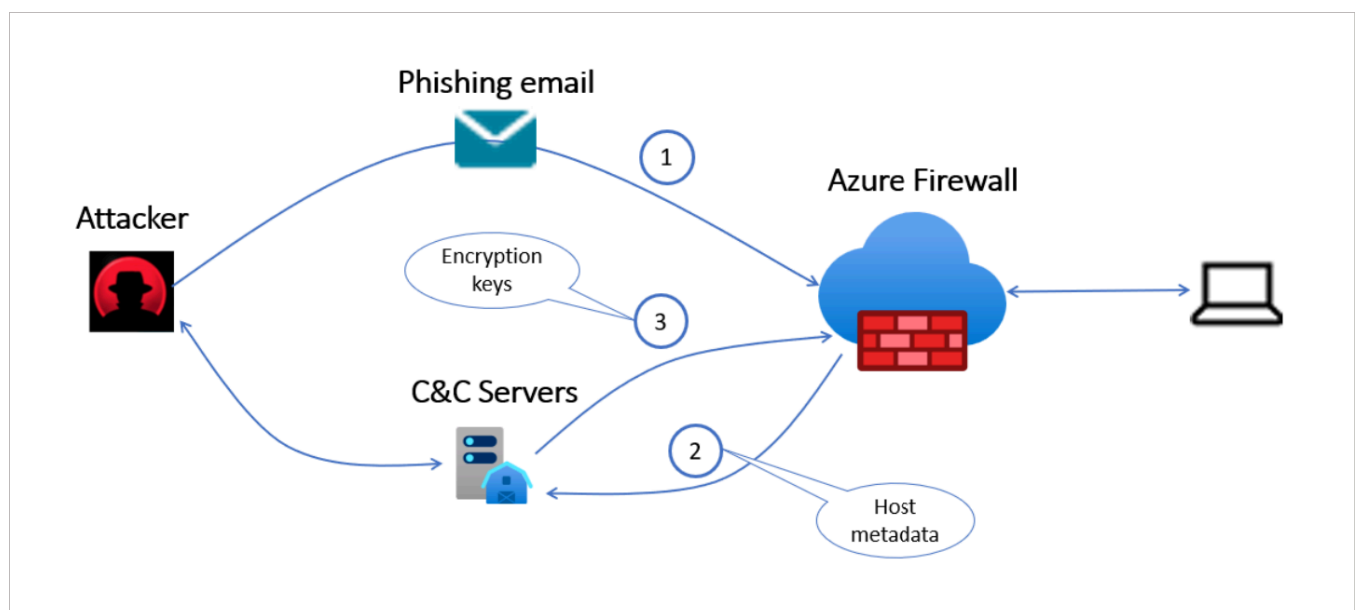
- Azure Firewall releases 30 to 50 new signatures each day.

Today, modern encryption (SSL/TLS) is used globally to secure Internet traffic. Attackers use encryption to carry their malicious software into the victim's network. Therefore, customers must inspect their encrypted traffic just like any other traffic.

Azure Firewall Premium IDPS allows you to detect attacks in all ports and protocols for non-encrypted traffic. However, when HTTPS traffic needs to be inspected, Azure Firewall can use its TLS inspection capability to decrypt the traffic and accurately detect malicious activities.

After the ransomware is installed on the target machine, it may try to encrypt the machine's data. The ransomware requires an encryption key and may use the Command and Control (C&C) to get the encryption key from the C&C server hosted by the attacker. CryptoLocker, WannaCry, TeslaCrypt, Cerber, and Locky are some of the ransomwares using C&C to fetch the required encryption keys.

Azure Firewall Premium has hundreds of signatures that are designed to detect C&C connectivity and block it to prevent the attacker from encrypting your data. The following diagram shows Azure Firewall protection against a ransomware attack using the C&C channel.



Fend off ransomware attacks

A holistic approach to fend off ransomware attacks is recommended. Azure Firewall operates in a default deny mode and blocks access unless explicitly allowed by the administrator. Enabling the Threat Intelligence (TI) feature in alert/deny mode blocks access to known malicious IPs and domains. Microsoft Threat Intel feed is updated continuously based on new and emerging threats.

Firewall Policy can be used for centralized configuration of firewalls. This helps with responding to threats rapidly. Customers can enable Threat Intel and IDPS across multiple firewalls with just a few clicks. Web categories lets administrators allow or deny user access to web categories such as gambling websites, social media websites, and others. URL filtering provides scoped access to external sites and can cut down risk even further. In other words, Azure Firewall has everything necessary for companies to defend comprehensively against malware and ransomware.

Detection is equally important as prevention. Azure Firewall solution for Azure Sentinel gets you both detection and prevention in the form of an easy-to-deploy solution. Combining prevention and detection allows you to ensure that you both prevent sophisticated threats when you can, while also maintaining an "assume breach mentality" to detect and quickly respond to cyberattacks.

Next steps

For comprehensive ransomware protection guidance across all Microsoft platforms and services, see [Protect your organization against ransomware and extortion](#).

See [Ransomware protection in Azure](#) to learn more about defenses for ransomware attacks in Azure and for guidance on how to proactively protect your assets.

Other Azure ransomware articles:

- [Prepare for a ransomware attack](#)
- [Detect and respond to ransomware attack](#)
- [Azure features and resources that help you protect, detect, and respond](#)

To learn more about Azure Firewall Premium, see:

- [Azure Firewall Premium features](#)
- [Optimize security with Azure Firewall solution for Azure Sentinel](#) [↗](#)

Azure threat protection

Azure provides comprehensive threat protection through services such as Microsoft Defender for Cloud, Microsoft Sentinel, Microsoft Entra ID Protection, and Microsoft Defender for Cloud Apps. This collection of security services and capabilities offers advanced detection, response, and threat intelligence to protect your Azure deployments.

Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) helps protect your hybrid cloud environment. By performing continuous security assessments of your connected resources, it provides detailed security recommendations and threat detection for discovered vulnerabilities.

Defender for Cloud's recommendations are based on the [Microsoft cloud security benchmark](#)—the Microsoft-authored, Azure-specific set of guidelines for security and compliance best practices based on common compliance frameworks.

Enabling Defender for Cloud's enhanced security features brings advanced, intelligent protection of your Azure, hybrid, and multicloud resources and workloads through specialized plans including:

- **Microsoft Defender for Servers** - Provides threat detection and advanced defenses for Windows and Linux machines
- **Microsoft Defender for Storage** - Detects unusual and potentially harmful attempts to access or exploit storage accounts
- **Microsoft Defender for SQL** - Protects databases against vulnerabilities, anomalous activities, and SQL injection threats
- **Microsoft Defender for Containers** - Secures containerized environments including Kubernetes clusters
- **Microsoft Defender for App Service** - Identifies attacks targeting applications running over App Service
- **Microsoft Defender for Key Vault** - Detects unusual and potentially harmful attempts to access key vault accounts
- **Microsoft Defender for Resource Manager** - Monitors resource management operations in your organization
- **Microsoft Defender for DNS** - Detects suspicious activities and anomalous DNS queries
- **Microsoft Defender for AI Services** - Provides runtime protection for Azure AI services against jailbreaks, data exposure, and suspicious access patterns

Security analytics and threat intelligence

Microsoft security researchers are constantly on the lookout for threats. They have access to an expansive set of telemetry gained from Microsoft's global presence in the cloud and on-premises. This wide-reaching and diverse collection of datasets enables Microsoft to discover new attack patterns and trends across its on-premises consumer and enterprise products, as well as its online services.

Defender for Cloud can rapidly update its detection algorithms as attackers release new and increasingly sophisticated exploits. This approach helps you keep pace with a fast-moving threat environment.

Defender for Cloud automatically collects security information from your resources, the network, and connected partner solutions. It analyzes this information, correlating data from multiple sources, to identify threats. Security alerts are prioritized in Defender for Cloud along with recommendations on how to remediate the threats.

Defender for Cloud employs advanced security analytics, which go far beyond signature-based approaches. Breakthroughs in big data and machine learning technologies are used to evaluate events across the entire cloud. Advanced analytics can detect threats that would be impossible to identify through manual approaches and predict the evolution of attacks.

For more information, see [Introduction to Microsoft Defender for Cloud](#).

Microsoft Sentinel

[Microsoft Sentinel](#) is a cloud-native security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution. Microsoft Sentinel provides intelligent security analytics and threat intelligence across the enterprise, offering a single solution for attack detection, threat visibility, proactive hunting, and threat response.

Microsoft Sentinel helps you:

- **Collect data at cloud scale** across all users, devices, applications, and infrastructure, both on-premises and in multiple clouds
- **Detect previously undetected threats** and minimize false positives using Microsoft's analytics and unparalleled threat intelligence
- **Investigate threats with artificial intelligence** and hunt for suspicious activities at scale, tapping into years of cybersecurity work at Microsoft
- **Respond to incidents rapidly** with built-in orchestration and automation of common tasks

Key capabilities include:

- **Advanced threat detection** using built-in machine learning, anomaly detection, and user and entity behavior analytics (UEBA)
- **Threat intelligence integration** from Microsoft and third-party sources to identify known threat actors and their techniques
- **Investigation and hunting** tools powered by AI to uncover hidden threats and pursue attackers across your environment
- **Automated response** through playbooks that can respond to threats in seconds
- **Microsoft Sentinel data lake** for scalable, cost-efficient long-term data retention and multi-modal analytics
- **Microsoft Sentinel graph** for unified graph analytics providing deeper context and threat reasoning

For more information, see [What is Microsoft Sentinel?](#).

Microsoft Entra ID Protection

[Microsoft Entra ID Protection](#) is a Microsoft Entra ID P2 feature that provides an overview of the risk detections and potential vulnerabilities that can affect your organization's identities. Identity Protection uses existing Microsoft Entra anomaly-detection capabilities and introduces new risk detection types that can detect real-time anomalies.

Identity Protection uses adaptive machine learning algorithms and heuristics to detect anomalies and risk detections that might indicate that an identity has been compromised. Using this data, Identity Protection generates reports and alerts so that you can investigate these risk detections and take appropriate remediation or mitigation action.

Identity Protection capabilities

Identity Protection helps you protect your organization's identities through:

Risk detection and assessment:

- Detect six risk detection types using machine learning and heuristic rules
- Calculate user risk levels
- Provide custom recommendations to improve overall security posture by highlighting vulnerabilities

Investigation capabilities:

- Send notifications for risk detections
- Investigate risk detections using relevant and contextual information
- Provide basic workflows to track investigations

- Provide easy access to remediation actions such as password reset

Risk-based conditional access policies:

- Mitigate risky sign-ins by blocking sign-ins or requiring multifactor authentication challenges
- Block or secure risky user accounts
- Require users to register for multifactor authentication

For more information, see [What is Microsoft Entra ID Protection?](#).

Microsoft Entra Privileged Identity Management

With [Microsoft Entra Privileged Identity Management \(PIM\)](#), you can manage, control, and monitor access within your organization. This feature includes access to resources in Microsoft Entra ID and other Microsoft online services, such as Microsoft 365 or Microsoft Intune.

PIM helps you:

- Get alerts and reports about Microsoft Entra administrators and just-in-time (JIT) administrative access to Microsoft online services
- Get reports about administrator access history and changes in administrator assignments
- Get alerts about access to a privileged role

For more information, see [What is Microsoft Entra Privileged Identity Management?](#).

Microsoft Defender for Cloud Apps

[Microsoft Defender for Cloud Apps](#) is a comprehensive solution that helps your organization take full advantage of the promise of cloud applications while maintaining control through improved visibility into activity and increased protection of critical data.

With tools that help uncover shadow IT, assess risk, enforce policies, investigate activities, and stop threats, your organization can more safely move to the cloud while maintaining control of critical data.

Defender for Cloud Apps integrates visibility with your cloud by:

- Using Cloud Discovery to map and identify your cloud environment and the cloud apps your organization is using
- Sanctioning and prohibiting apps in your cloud
- Using easy-to-deploy app connectors that take advantage of provider APIs for visibility and governance

- Providing continuous control by setting and continually fine-tuning policies
- Using behavioral analytics with dynamic threat detection powered by Microsoft Threat Intelligence

For more information, see [What is Microsoft Defender for Cloud Apps?](#).

Microsoft Defender for Storage

[Microsoft Defender for Storage](#) is an Azure-native layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit your storage accounts. It uses advanced threat detection capabilities and Microsoft Threat Intelligence data to provide contextual security alerts with steps to mitigate detected threats and prevent future attacks.

Key features include:

- **Malware scanning** - On-upload and on-demand malware scanning with automated remediation capabilities
- **Sensitive data threat detection** - Detects anomalous access to storage accounts containing sensitive data
- **Activity monitoring** - Provides aggregated storage activity logs for threat detection and investigation

For more information, see [Introduction to Microsoft Defender for Storage](#).

Microsoft Defender for SQL

Defender for SQL provides protection for databases against vulnerabilities, anomalous activities, and threats:

- **Vulnerability assessment** - Discovers, tracks, and helps remediate potential database vulnerabilities
- **Advanced threat protection** - Detects anomalous database activities indicating potential security threats such as SQL injection, brute-force attacks, and privilege abuse

For more information, see [Microsoft Defender for Azure SQL](#).

Microsoft Antimalware

[Microsoft Antimalware](#) for Azure is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. You can deploy protection based on the needs of your application workloads, with either basic secure-by-default or advanced custom configuration.

Microsoft Antimalware for Azure provides:

- **Real-time protection** - Monitors activity to detect and block malware execution
- **Scheduled scanning** - Performs targeted scanning to detect malware
- **Malware remediation** - Automatically acts on detected malware
- **Signature updates** - Automatically installs the latest protection signatures
- **Active protection** - Reports telemetry metadata about detected threats to Microsoft Azure

For more information, see [Microsoft Antimalware for Azure Cloud Services and Virtual Machines](#).

Azure Firewall

[Azure Firewall](#) is a cloud-native and intelligent network firewall security service that provides threat protection for your cloud workloads running in Azure. Azure Firewall inspects east-west and north-south traffic with built-in threat intelligence that can alert and deny traffic from/to known malicious IP addresses and domains.

Azure Firewall is available in three SKUs:

- **Azure Firewall Basic** - Simplified security for small to medium-sized businesses
- **Azure Firewall Standard** - Provides L3-L7 filtering and threat intelligence feeds from Microsoft Cyber Security
- **Azure Firewall Premium** - Advanced capabilities including signature-based IDPS, TLS inspection, and URL filtering

For more information, see [What is Azure Firewall?](#) and [Azure network security overview](#).

Web Application Firewall

[Web Application Firewall \(WAF\)](#) provides centralized protection of your web applications from common exploits and vulnerabilities. WAF is available through:

- **Azure Application Gateway** - Provides regional WAF protection
- **Azure Front Door** - Provides global WAF protection with protection against network-level DDoS attacks

WAF protects against common web vulnerabilities such as:

- SQL injection
- Cross-site scripting
- Other OWASP top 10 vulnerabilities

- Bot attacks
- HTTP protocol violations and anomalies

For more information, see [What is Azure Web Application Firewall?](#).

Next steps

- [Responding to today's threats](#) - Identify active threats and respond quickly
- [Azure security best practices and patterns](#) - Collection of security best practices
- [Microsoft Defender for Cloud documentation](#) - Comprehensive guide to Defender for Cloud
- [Microsoft Sentinel documentation](#) - Complete documentation for Microsoft Sentinel

Last updated on 11/07/2025

Azure security technical capabilities

05/23/2025

This article provides an introduction to security services in Azure that help you protect your data, resources, and applications in the cloud and meet the security needs of your business.

Azure platform

[Microsoft Azure](#)  is a cloud platform comprised of infrastructure and application services, with integrated data services and advanced analytics, and developer tools and services, hosted within Microsoft's public cloud data centers. Customers use Azure for many different capacities and scenarios, from basic compute, networking, and storage, to mobile and web app services, to full cloud scenarios like Internet of Things, and can be used with open-source technologies, and deployed as hybrid cloud or hosted within a customer's datacenter. Azure provides cloud technology as building blocks to help companies save costs, innovate quickly, and manage systems proactively. When you build on, or migrate IT assets to a cloud provider, you are relying on that organization's abilities to protect your applications and data with the services and the controls they provide to manage the security of your cloud-based assets.

Microsoft Azure is the only cloud computing provider that offers a secure, consistent application platform and infrastructure-as-a-service for teams to work within their different cloud skillsets and levels of project complexity, with integrated data services and analytics that uncover intelligence from data wherever it exists, across both Microsoft and non-Microsoft platforms, open frameworks and tools, providing choice for integrating cloud with on-premises as well deploying Azure cloud services within on-premises datacenters. As part of the Microsoft Trusted Cloud, customers rely on Azure for industry-leading security, reliability, compliance, privacy, and the vast network of people, partners, and processes to support organizations in the cloud.

With Microsoft Azure, you can:

- Accelerate innovation with the cloud
- Power business decisions & apps with insights
- Build freely and deploy anywhere
- Protect their business

Manage and control identity and user access

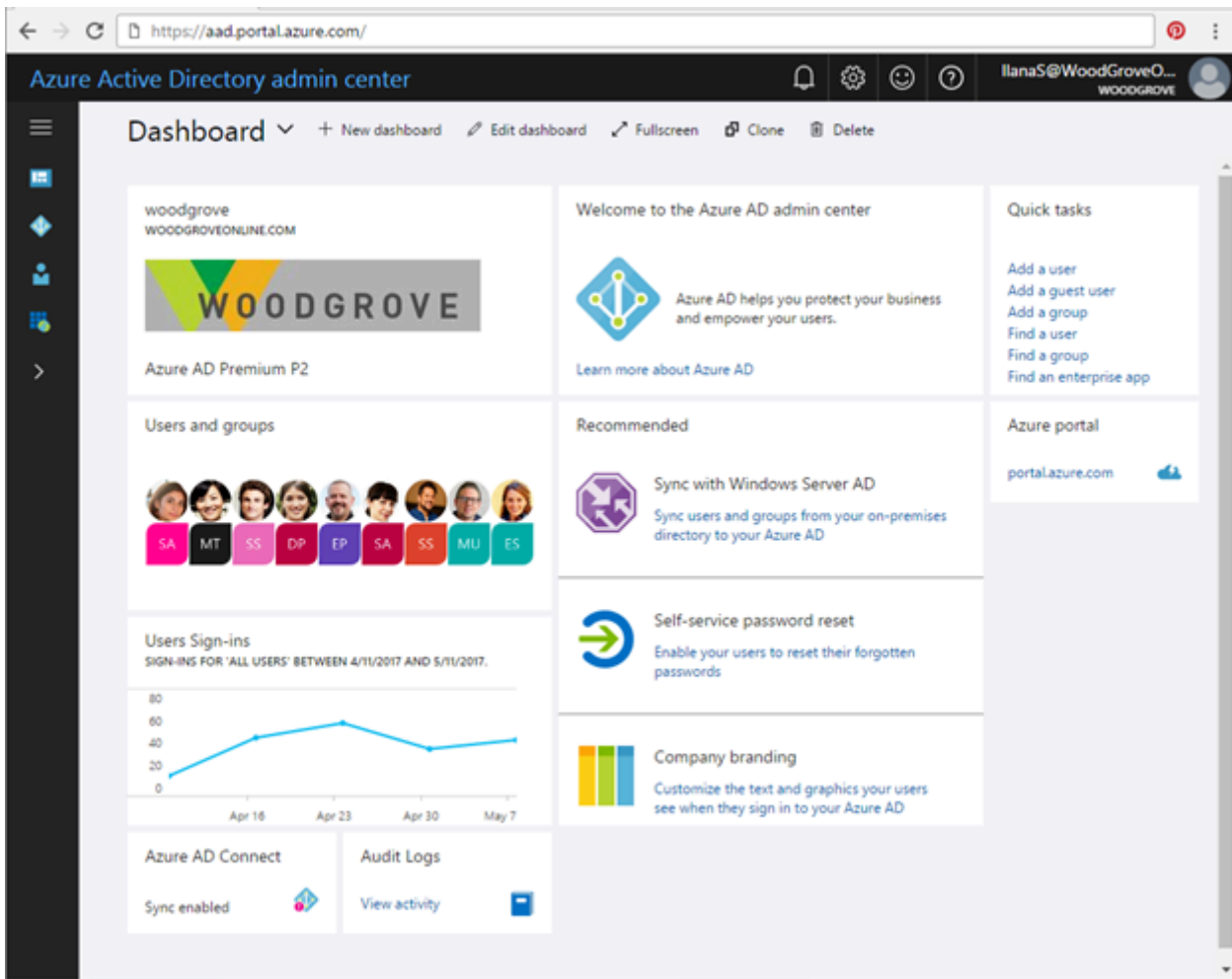
Azure helps you protect business and personal information by enabling you to manage user identities and credentials and control access.

Microsoft Entra ID

Microsoft identity and access management solutions help IT protect access to applications and resources across the corporate datacenter and into the cloud, enabling additional levels of validation such as multifactor authentication and Conditional Access policies. Monitoring suspicious activity through advanced security reporting, auditing and alerting helps mitigate potential security issues. [Microsoft Entra ID P1 or P2](#) provides single sign-on to thousands of cloud apps and access to web apps you run on-premises.

Security benefits of Microsoft Entra ID include the ability to:

- Create and manage a single identity for each user across your hybrid enterprise, keeping users, groups, and devices in sync.
- Provide single sign-on access to your applications including thousands of pre-integrated SaaS apps.
- Enable application access security by enforcing rules-based multifactor authentication for both on-premises and cloud applications.
- Provision secure remote access to on-premises web applications through Microsoft Entra application proxy.



The following are core Azure identity management capabilities:

- Single sign-on
- Multifactor authentication
- Security monitoring, alerts, and machine learning-based reports
- Consumer identity and access management
- Device registration
- Privileged identity management
- Identity protection

Single sign-on

Single sign-on (SSO) means being able to access all the applications and resources that you need to do business, by signing in only once using a single user account. Once signed in, you can access all the applications you need without being required to authenticate (for example, type a password) a second time.

Many organizations rely upon software as a service (SaaS) applications such as Microsoft 365, Box, and Salesforce for end-user productivity. Historically, IT staff needed to individually create and update user accounts in each SaaS application, and users had to remember a password for each SaaS application.

Microsoft Entra ID extends on-premises Active Directory into the cloud, enabling users to use their primary organizational account to not only sign in to their domain-joined devices and company resources, but also all the web and SaaS applications needed for their job.

Not only do users not have to manage multiple sets of usernames and passwords, application access can be automatically provisioned or de-provisioned based on organizational groups and their status as an employee. Microsoft Entra ID introduces security and access governance controls that enable you to centrally manage users' access across SaaS applications.

Multifactor authentication

[Microsoft Entra multifactor authentication \(MFA\)](#) is a method of authentication that requires the use of more than one verification method and adds a critical second layer of security to user sign-ins and transactions. [MFA helps safeguard](#) access to data and applications while meeting user demand for a simple sign-in process. It delivers strong authentication via a range of verification options—phone call, text message, or mobile app notification or verification code and third-party OAuth tokens.

Security monitoring, alerts, and machine learning-based reports

Security monitoring and alerts and machine learning-based reports that identify inconsistent access patterns can help you protect your business. You can use Microsoft Entra ID's access and usage reports to gain visibility into the integrity and security of your organization's directory. With this information, a directory admin can better determine where possible security risks may lie so that they can adequately plan to mitigate those risks.

In the [Azure portal](#) [↗](#), [reports](#) are categorized in the following ways:

- Anomaly reports – contain sign in events that we found to be anomalous. Our goal is to make you aware of such activity and enable you to be able to decide about whether an event is suspicious.
- Integrated application reports – provide insights into how cloud applications are being used in your organization. Microsoft Entra ID offers integration with thousands of cloud applications.
- Error reports – indicate errors that may occur when provisioning accounts to external applications.

- User-specific reports – display device and sign in activity data for a specific user.
- Activity logs – contain a record of all audited events within the last 24 hours, last 7 days, or last 30 days, and group activity changes, and password reset and registration activity.

Consumer identity and access management

[Microsoft Entra External ID](#) is a highly available, global, identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can log on to all your applications through customizable experiences by using their existing social accounts or by creating new credentials.

In the past, application developers who wanted to sign up and sign in consumers into their applications would have written their own code. And they would have used on-premises databases or systems to store usernames and passwords. Azure Microsoft Entra External ID offers your organization a better way to integrate consumer identity management into applications with the help of a secure, standards-based platform, and a large set of extensible policies.

When you use Microsoft Entra External ID, your consumers can sign up for your applications by using their existing social accounts (Facebook, Google, Amazon, LinkedIn) or by creating new credentials (email address and password, or username and password).

Device registration

[Microsoft Entra device registration](#) is the foundation for device-based [Conditional Access](#) scenarios. When a device is registered, Microsoft Entra device registration provides the device with an identity that is used to authenticate the device when the user signs in. The authenticated device, and the attributes of the device, can then be used to enforce Conditional Access policies for applications that are hosted in the cloud and on-premises.

When combined with a [mobile device management \(MDM\)](#) [↗](#) solution such as Intune, the device attributes in Microsoft Entra ID are updated with additional information about the device. This allows you to create Conditional Access rules that enforce access from devices to meet your standards for security and compliance.

Privileged identity management

[Microsoft Entra Privileged Identity Management](#) lets you manage, control, and monitor your privileged identities and access to resources in Microsoft Entra ID as well as other Microsoft online services like Microsoft 365 or Microsoft Intune.

Sometimes users need to carry out privileged operations in Azure or Microsoft 365 resources, or other SaaS apps. This often means organizations have to give them permanent privileged access in Microsoft Entra ID. This is a growing security risk for cloud-hosted resources because organizations can't sufficiently monitor what those users are doing with their admin privileges. Additionally, if a user account with privileged access is compromised, that one breach could impact their overall cloud security. Microsoft Entra Privileged Identity Management helps to resolve this risk.

Microsoft Entra Privileged Identity Management lets you:

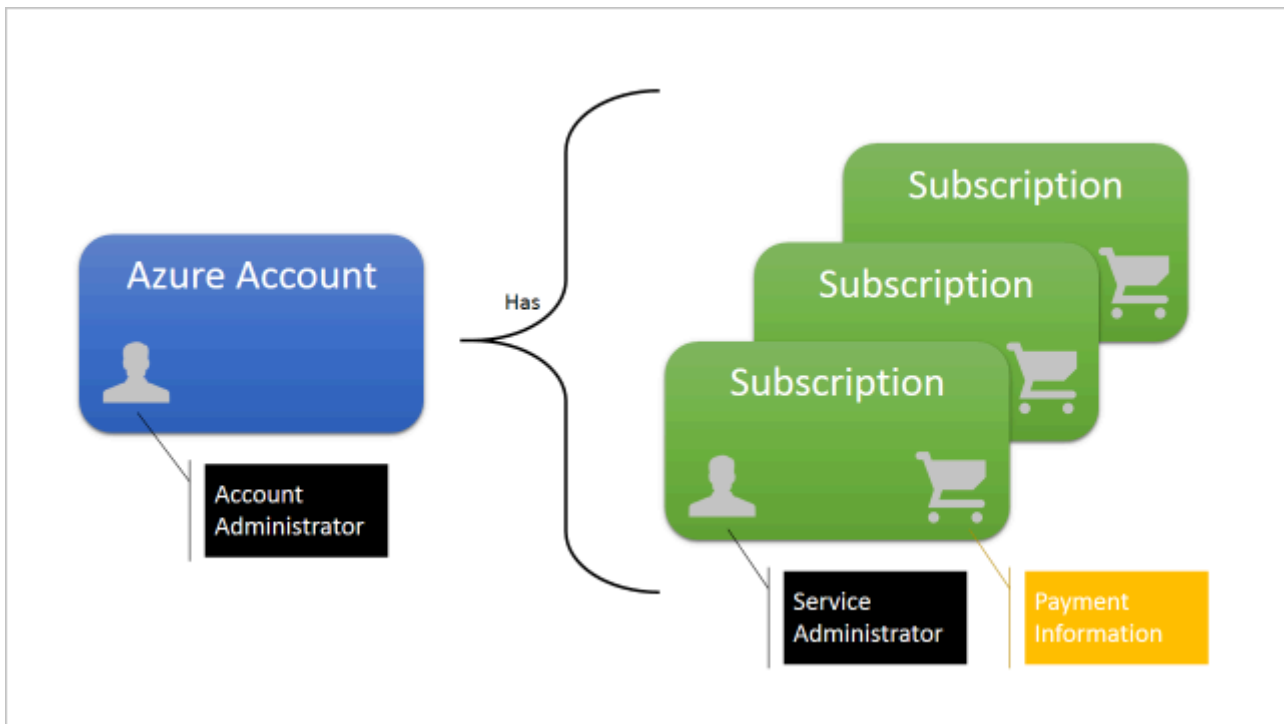
- See which users are Microsoft Entra admins
- Enable on-demand, "just in time" administrative access to Microsoft Online Services like Microsoft 365 and Intune
- Get reports about administrator access history and changes in administrator assignments
- Get alerts about access to a privileged role

Identity protection

[Microsoft Entra ID Protection](#) is a security service that provides a consolidated view into risk detections and potential vulnerabilities affecting your organization's identities. Identity Protection uses existing Microsoft Entra ID's anomaly detection capabilities (available through Microsoft Entra ID's Anomalous Activity Reports), and introduces new risk detection types that can detect anomalies in real time.

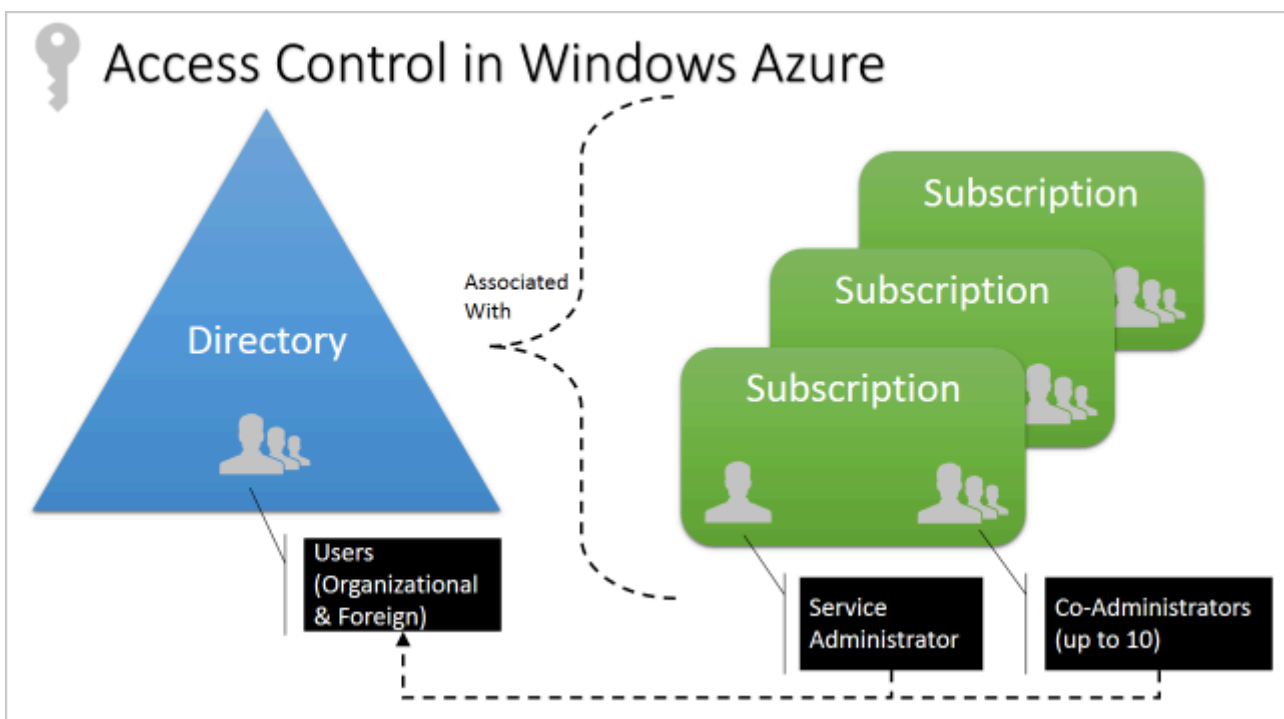
Secure resource access

Access control in Azure starts from a billing perspective. The owner of an Azure account, accessed by visiting the Azure portal, is the Account Administrator (AA). Subscriptions are a container for billing, but they also act as a security boundary: each subscription has a Service Administrator (SA) who can add, remove, and modify Azure resources in that subscription by using the Azure portal. The default SA of a new subscription is the AA, but the AA can change the SA in the Azure portal.

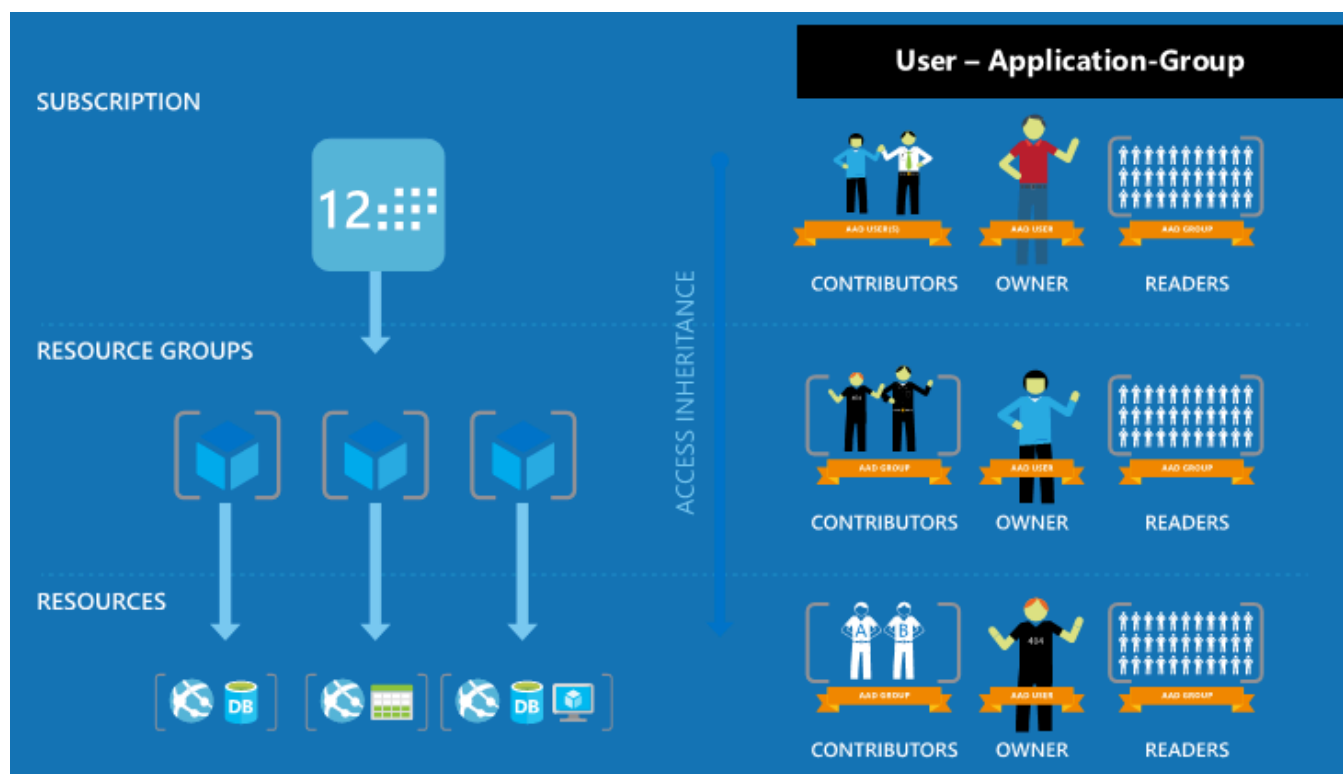


Subscriptions also have an association with a directory. The directory defines a set of users. These can be users from the work or school that created the directory, or they can be external users (that is, Microsoft Accounts). Subscriptions are accessible by a subset of those directory users who have been assigned as either Service Administrator (SA) or Co-Administrator (CA); the only exception is that, for legacy reasons, Microsoft Accounts (formerly Windows Live ID) can be assigned as SA or CA without being present in the directory.

Security-oriented companies should focus on giving employees the exact permissions they need. Too many permissions can expose an account to attackers. Too few permissions mean that employees can't get their work done efficiently. [Azure role-based access control \(Azure RBAC\)](#) helps address this problem by offering fine-grained access management for Azure.



Using Azure RBAC, you can segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, you can allow only certain actions. For example, use Azure RBAC to let one employee manage virtual machines in a subscription, while another can manage SQL databases within the same subscription.



Data security and encryption

One of the keys to data protection in the cloud is accounting for the possible states in which your data may occur, and what controls are available for that state. For Azure data security and encryption best practices the recommendations be around the following data's states.

- **At-rest:** This includes all information storage objects, containers, and types that exist statically on physical media, be it magnetic or optical disk.
- **In-transit:** When data is being transferred between components, locations or programs, such as over the network, across a service bus (from on-premises to cloud and vice-versa, including hybrid connections such as ExpressRoute), or during an input/output process, it is thought of as being in-motion.

Encryption at rest

Encryption at rest is discussed in detail in [Azure Data Encryption at Rest](#).

Encryption in-transit

Protecting data in transit should be an essential part of your data protection strategy. Since data is moving back and forth from many locations, the general recommendation is that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you may want to isolate the entire communication channel between your on-premises and cloud infrastructure by using a virtual private network (VPN).

For data moving between your on-premises infrastructure and Azure, you should consider appropriate safeguards such as HTTPS or VPN.

For organizations that need to secure access from multiple workstations located on-premises to Azure, use [Azure site-to-site VPN](#).

For organizations that need to secure access from one workstation located on-premises to Azure, use [Point-to-Site VPN](#).

Larger data sets can be moved over a dedicated high-speed WAN link such as [ExpressRoute](#). If you choose to use ExpressRoute, you can also encrypt the data at the application-level using SSL/TLS or other protocols for added protection.

If you are interacting with Azure Storage through the Azure portal, all transactions occur via HTTPS. [Storage REST API](#) over HTTPS can also be used to interact with [Azure Storage](#) and [Azure SQL Database](#).

You can learn more about Azure VPN option by reading the article [Planning and design for VPN Gateway](#).

Enforce file level data encryption

[Azure Rights Management](#) (Azure RMS) uses encryption, identity, and authorization policies to help secure your files and email. Azure RMS works across multiple devices—phones, tablets, and PCs by protecting both within your organization and outside your organization. This capability is possible because Azure RMS adds a level of protection that remains with the data, even when it leaves your organization's boundaries.

Secure your application

While Azure is responsible for securing the infrastructure and platform that your application runs on, it is your responsibility to secure your application itself. In other words, you need to develop, deploy, and manage your application code and content in a secure way. Without this, your application code or content can still be vulnerable to threats.

Web application firewall

[Web application firewall \(WAF\)](#) is a feature of [Application Gateway](#) and [Azure Front Door](#) that provides centralized protection of your web applications from common exploits and vulnerabilities.

Web application firewall is based on rules from the [OWASP core rule sets](#) [↗]. Web applications are increasingly targets of malicious attacks that exploit common known vulnerabilities. Common among these exploits are SQL injection attacks, cross site scripting attacks to name a few. Preventing such attacks in application code can be challenging and may require rigorous maintenance, patching and monitoring at multiple layers of the application topology. A centralized web application firewall helps make security management much simpler and gives better assurance to application administrators against threats or intrusions. A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each of individual web applications. Existing application gateways can be converted to a web application firewall enabled application gateway easily.

Some of the common web vulnerabilities which web application firewall protects against includes:

- SQL injection protection
- Cross site scripting protection
- Common Web Attacks Protection such as command injection, HTTP request smuggling, HTTP response splitting, and remote file inclusion attack
- Protection against HTTP protocol violations
- Protection against HTTP protocol anomalies such as missing host user-agent and accept headers
- Prevention against bots, crawlers, and scanners
- Detection of common application misconfigurations (that is, Apache, IIS, etc.)

 Note

For a more detailed list of rules and their protections see the following core rule sets for [Azure Application Gateway](#) and [Azure Front Door](#).

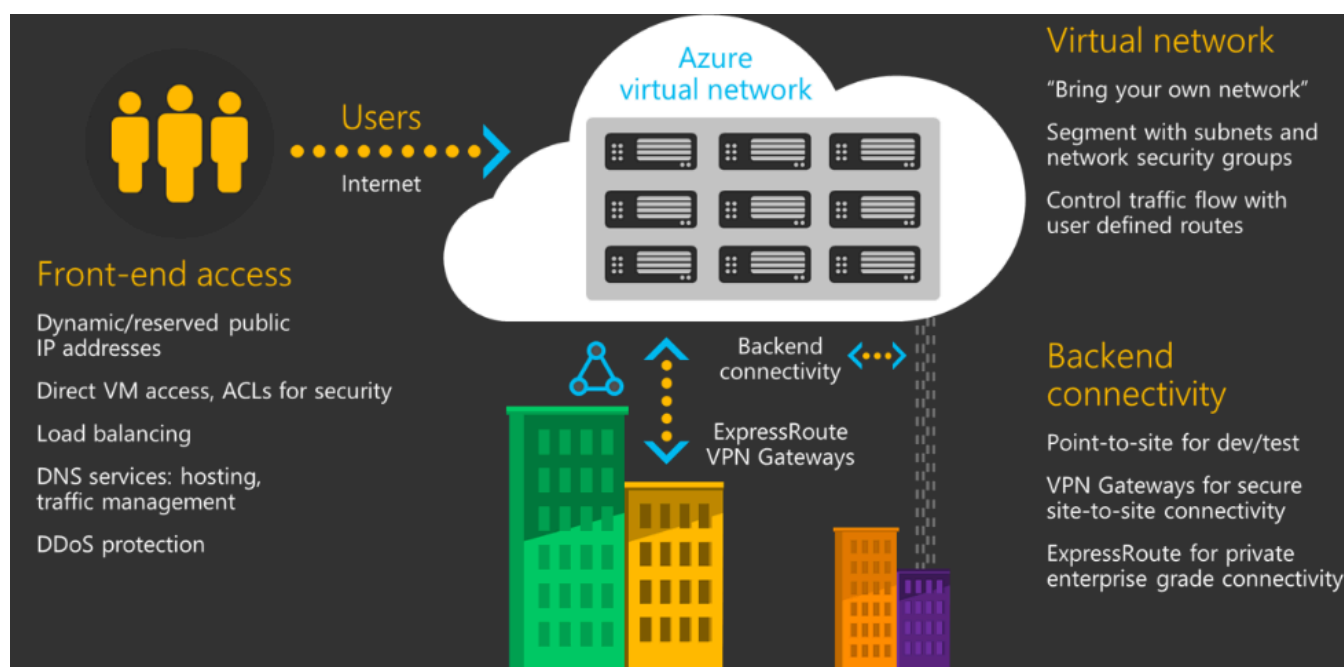
Azure provides several easy-to-use features to help secure both inbound and outbound traffic for your app. Azure helps customers secure their application code by providing externally provided functionality to scan your web application for vulnerabilities. See [Azure App Services](#) to learn more.

Azure App Service uses the same Antimalware solution used by Azure Cloud Services and Virtual Machines. To learn more about this refer to our [Antimalware documentation](#).

Secure your network

Microsoft Azure includes a robust networking infrastructure to support your application and service connectivity requirements. Network connectivity is possible between resources located in Azure, between on-premises and Azure hosted resources, and to and from the Internet and Azure.

The Azure network infrastructure enables you to securely connect Azure resources to each other with [virtual networks \(VNETs\)](#). A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud network dedicated to your subscription. You can connect VNets to your on-premises networks.



If you need basic network level access control (based on IP address and the TCP or UDP protocols), then you can use [Network Security Groups](#). A Network Security Group (NSG) is a basic stateful packet filtering firewall that enables you to control access.

[Azure Firewall](#) is a cloud-native and intelligent network firewall security service that provides threat protection for your cloud workloads running in Azure. It's a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability. It provides both east-west and north-south traffic inspection.

Azure Firewall is offered in three SKUs: Basic, Standard and Premium. [Azure Firewall Basic](#) offers simplified security similar to the Standard SKU but without advanced features. [Azure Firewall Standard](#) provides L3-L7 filtering and threat intelligence feeds directly from Microsoft Cyber

Security. [Azure Firewall Premium](#) provides advanced capabilities include signature-based IDPS to allow rapid detection of attacks by looking for specific patterns.

Azure networking supports the ability to customize the routing behavior for network traffic on your Azure Virtual Networks. You can do this by configuring [User-Defined Routes](#) in Azure.

[Forced tunneling](#)  is a mechanism you can use to ensure that your services are not allowed to initiate a connection to devices on the Internet.

Azure supports dedicated WAN link connectivity to your on-premises network and an Azure Virtual Network with [ExpressRoute](#). The link between Azure and your site uses a dedicated connection that does not go over the public Internet. If your Azure application is running in multiple datacenters, you can use [Azure Traffic Manager](#) to route requests from users intelligently across instances of the application. You can also route traffic to services not running in Azure if they are accessible from the Internet.

Azure also supports private and secure connectivity to your PaaS resources (for example, Azure Storage and SQL Database) from your Azure Virtual Network with [Azure Private Link](#). PaaS resource is mapped to a [private endpoint](#) in your virtual network. The link between private endpoint in your virtual network and your PaaS resource uses Microsoft backbone network and does not go over the public Internet. Exposing your service to the public internet is no longer necessary. You can also use Azure Private Link to access Azure hosted customer-owned and partner services in your virtual network. In addition, Azure Private Link enables you to create your own [private link service](#) in your virtual network and deliver it to your customers privately in their virtual networks. Setup and consumption using Azure Private Link is consistent across Azure PaaS, customer-owned, and shared partner services.

Virtual machine security

[Azure Virtual Machines](#) lets you deploy a wide range of computing solutions in an agile way. With support for Microsoft Windows, Linux, Microsoft SQL Server, Oracle, IBM, SAP, and Azure BizTalk Services, you can deploy any workload and any language on nearly any operating system.

With Azure, you can use [antimalware software](#) from security vendors such as Microsoft, Symantec, Trend Micro, and Kaspersky to protect your virtual machines from malicious files, adware, and other threats.

Microsoft Antimalware for Azure Cloud Services and Virtual Machines is a real-time protection capability that helps identify and remove viruses, spyware, and other malicious software. Microsoft Antimalware provides configurable alerts when known malicious or unwanted software attempts to install itself or run on your Azure systems.

[Azure Backup](#) is a scalable solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. With Azure Backup, your virtual machines running Windows and Linux are protected.

[Azure Site Recovery](#) helps orchestrate replication, failover, and recovery of workloads and apps so that they are available from a secondary location if your primary location goes down.

Ensure compliance: Cloud services due diligence checklist

Microsoft developed [the Cloud Services Due Diligence Checklist](#) [↗](#) to help organizations exercise due diligence as they consider a move to the cloud. It provides a structure for an organization of any size and type—private businesses and public-sector organizations, including government at all levels and nonprofits—to identify their own performance, service, data management, and governance objectives and requirements. This allows them to compare the offerings of different cloud service providers, ultimately forming the basis for a cloud service agreement.

The checklist provides a framework that aligns clause-by-clause with a new international standard for cloud service agreements, ISO/IEC 19086. This standard offers a unified set of considerations for organizations to help them make decisions about cloud adoption, and create a common ground for comparing cloud service offerings.

The checklist promotes a thoroughly vetted move to the cloud, providing structured guidance and a consistent, repeatable approach for choosing a cloud service provider.

Cloud adoption is no longer simply a technology decision. Because checklist requirements touch on every aspect of an organization, they serve to convene all key internal decision-makers—the CIO and CISO as well as legal, risk management, procurement, and compliance professionals. This increases the efficiency of the decision-making process and ground decisions in sound reasoning, thereby reducing the likelihood of unforeseen roadblocks to adoption.

In addition, the checklist:

- Exposes key discussion topics for decision-makers at the beginning of the cloud adoption process.
- Supports thorough business discussions about regulations and the organization's own objectives for privacy, personal information and data security.
- Helps organizations identify any potential issues that could affect a cloud project.

- Provides a consistent set of questions, with the same terms, definitions, metrics, and deliverables for each provider, to simplify the process of comparing offerings from different cloud service providers.

Azure infrastructure and application security validation

[Azure Operational Security](#) refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure.

Insight & Analytics

- Quickly diagnose root cause across the full stack of modern applications and underlying infrastructure
- Monitor and alert on key metrics and KPIs in real time to rapidly identify problems
- Collect, process and analyze petabytes of data
- Create and share data insights across your company in minutes
- Integrate with and extend the value of existing monitoring tools

Protection & Recovery

- Protection of Cloud Assets (DR/Backup for IAAS, Backup of SQL PaaS)
- Enhanced Capacity Planning and Monitoring with Log Analytics
- Enterprise coverage with Linux distros, SQL AG, Encryption at rest
- Faster, Cheaper, Compact Backup Storage (Xcool, De-dup, ReFS)
- Centralized hybrid backup monitoring and reporting in Azure
- Workload protection for public, hybrid, and private cloud
- Enterprise grade VMware VM Backup

Automation & Control

- Trigger immediate action in response to issues automatically or on-demand
- Maintain the state of IT resources and resolve configuration drifts
- Keep IT systems updated with minimal downtime
- Track and manage changes with ease

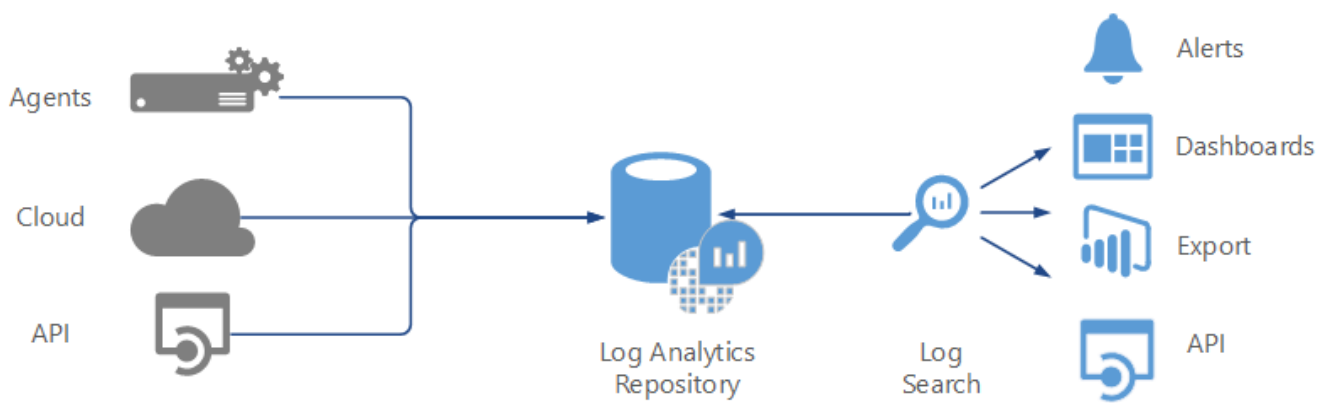
Security & Compliance

- Collection of security data from virtually any source
- Insight into security status (antimalware, system updates)
- Correlations to detect malicious activities and search for rapid investigation
- Integrates operational and security management
- Threat detection using advanced analytics

Azure Operational Security is built on a framework that incorporates the knowledge gained through a various capabilities that are unique to Microsoft, including the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Center program, and deep awareness of the cybersecurity threat landscape.

Microsoft Azure Monitor

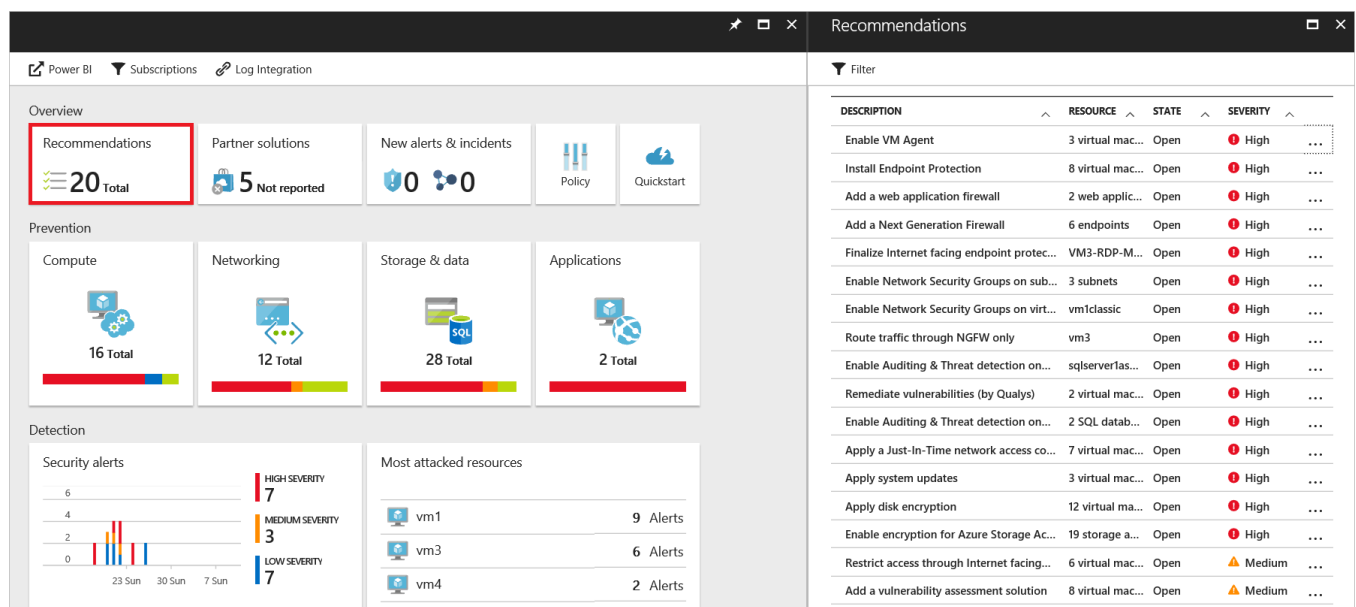
[Azure Monitor](#) is the IT management solution for the hybrid cloud. Used alone or to extend your existing System Center deployment, Azure Monitor logs gives you the maximum flexibility and control for cloud-based management of your infrastructure.



With Azure Monitor, you can manage any instance in any cloud, including on-premises, Azure, AWS, Windows Server, Linux, VMware, and OpenStack, at a lower cost than competitive solutions. Built for the cloud-first world, Azure Monitor offers a new approach to managing your enterprise that is the fastest, most cost-effective way to meet new business challenges and accommodate new workloads, applications and cloud environments.

Azure Monitor logs

[Azure Monitor logs](#) provides monitoring services by collecting data from managed resources into a central repository. This data could include events, performance data, or custom data provided through the API. Once collected, the data is available for alerting, analysis, and export.



This method allows you to consolidate data from a variety of sources, so you can combine data from your Azure services with your existing on-premises environment. It also clearly separates the collection of the data from the action taken on that data so that all actions are available to all kinds of data.

Microsoft Sentinel

[Microsoft Sentinel](#) is a scalable, cloud-native, security information and event management (SIEM) and security orchestration, automation, and response (SOAR) solution. Microsoft Sentinel delivers intelligent security analytics and threat intelligence across the enterprise, providing a single solution for attack detection, threat visibility, proactive hunting, and threat response.

Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats with increased visibility into and control over the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with a broad ecosystem of security solutions.

Defender for Cloud analyzes the security state of your Azure resources to identify potential security vulnerabilities. A list of recommendations guides you through the process of configuring needed controls.

Examples include:

- Provisioning antimalware to help identify and remove malicious software
- Configuring network security groups and rules to control traffic to VMs
- Provisioning of web application firewalls to help defend against attacks that target your web applications
- Deploying missing system updates
- Addressing OS configurations that do not match the recommended baselines

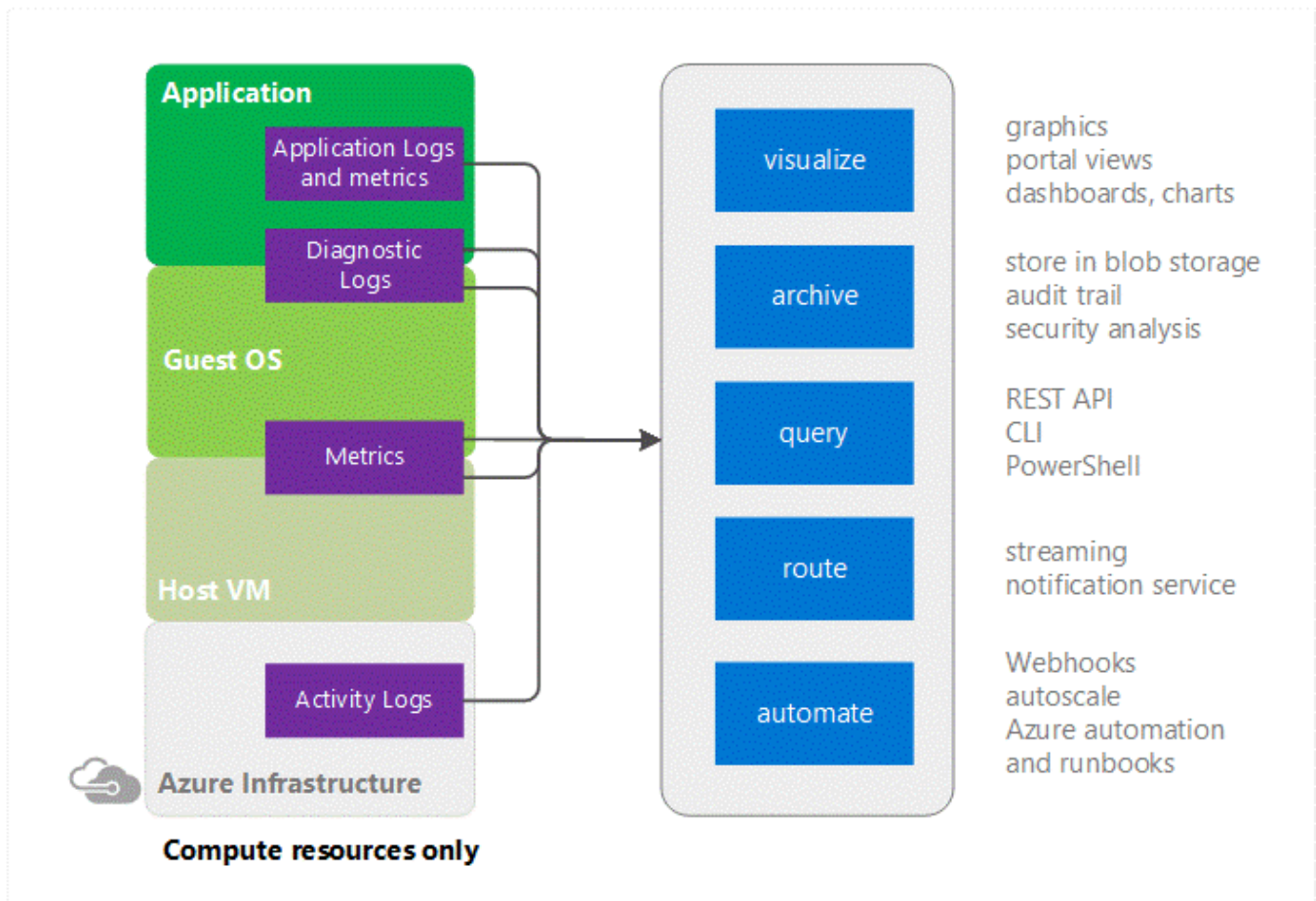
Defender for Cloud automatically collects, analyzes, and integrates log data from your Azure resources, the network, and partner solutions like antimalware programs and firewalls. When threats are detected, a security alert is created. Examples include detection of:

- Compromised VMs communicating with known malicious IP addresses
- Advanced malware detected by using Windows error reporting
- Brute force attacks against VMs
- Security alerts from integrated antimalware programs and firewalls

Azure monitor

[Azure Monitor](#) provides pointers to information on specific types of resources. It offers visualization, query, routing, alerting, auto scale, and automation on data both from the Azure infrastructure (Activity Log) and each individual Azure resource (Diagnostic Logs).

Cloud applications are complex with many moving parts. Monitoring provides data to ensure that your application stays up and running in a healthy state. It also helps you to stave off potential problems or troubleshoot past ones.



In addition, you can use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Auditing your network security is vital for detecting network vulnerabilities and ensuring compliance with your IT security and regulatory governance model. With Security Group view, you can retrieve the configured Network Security Group and security rules, as well as the effective security rules. With the list of rules applied, you can determine the ports that are open and ss network vulnerability.

Network watcher

[Network Watcher](#) is a regional service that enables you to monitor and diagnose conditions at a network level in, to, and from Azure. Network diagnostic and visualization tools available with Network Watcher help you understand, diagnose, and gain insights to your network in Azure.

This service includes packet capture, next hop, IP flow verify, security group view, NSG flow logs. Scenario level monitoring provides an end to end view of network resources in contrast to individual network resource monitoring.

Storage analytics

[Storage Analytics](#) can store metrics that include aggregated transaction statistics and capacity data about requests to a storage service. Transactions are reported at both the API operation level as well as at the storage service level, and capacity is reported at the storage service level. Metrics data can be used to analyze storage service usage, diagnose issues with requests made against the storage service, and to improve the performance of applications that use a service.

Application Insights

[Application Insights](#) is an extensible Application Performance Management (APM) service for web developers on multiple platforms. Use it to monitor your live web application. It will automatically detect performance anomalies. It includes powerful analytics tools to help you diagnose issues and to understand what users do with your app. It's designed to help you continuously improve performance and usability. It works for apps on a wide variety of platforms including .NET, Node.js and Java EE, hosted on-premises or in the cloud. It integrates with your DevOps process, and has connection points to a various development tools.

It monitors:

- **Request rates, response times, and failure rates** - Find out which pages are most popular, at what times of day, and where your users are. See which pages perform best. If your response times and failure rates go high when there are more requests, then perhaps you have a resourcing problem.
- **Dependency rates, response times, and failure rates** - Find out whether external services are slowing you down.
- **Exceptions** - Analyze the aggregated statistics, or pick specific instances and drill into the stack trace and related requests. Both server and browser exceptions are reported.
- **Page views and load performance** - reported by your users' browsers.
- **AJAX calls from web pages** - rates, response times, and failure rates.
- **User and session counts.**
- **Performance counters** from your Windows or Linux server machines, such as CPU, memory, and network usage.

- **Host diagnostics** from Docker or Azure.
- **Diagnostic trace logs** from your app - so that you can correlate trace events with requests.
- **Custom events and metrics** that you write yourself in the client or server code, to track business events such as items sold, or games won.

The infrastructure for your application is typically made up of many components – maybe a virtual machine, storage account, and virtual network, or a web app, database, database server, and 3rd party services. You do not see these components as separate entities, instead you see them as related and interdependent parts of a single entity. You want to deploy, manage, and monitor them as a group. [Azure Resource Manager](#) enables you to work with the resources in your solution as a group.

You can deploy, update, or delete all the resources for your solution in a single, coordinated operation. You use a template for deployment and that template can work for different environments such as testing, staging, and production. Resource Manager provides security, auditing, and tagging features to help you manage your resources after deployment.

The benefits of using Resource Manager

Resource Manager provides several benefits:

- You can deploy, manage, and monitor all the resources for your solution as a group, rather than handling these resources individually.
- You can repeatedly deploy your solution throughout the development lifecycle and have confidence your resources are deployed in a consistent state.
- You can manage your infrastructure through declarative templates rather than scripts.
- You can define the dependencies between resources, so they are deployed in the correct order.
- You can apply access control to all services in your resource group because Azure role-based access control (Azure RBAC) is natively integrated into the management platform.
- You can apply tags to resources to logically organize all the resources in your subscription.
- You can clarify your organization's billing by viewing costs for a group of resources sharing the same tag.

Resource Manager provides a new way to deploy and manage your solutions. If you used the earlier deployment model and want to learn about the changes, see [Understanding Resource Manager Deployment and classic deployment](#).

Next step

The [Microsoft cloud security benchmark](#) includes a collection of security recommendations you can use to help secure the services you use in Azure.

Azure infrastructure security

Article • 05/01/2025

Microsoft Azure runs in datacenters managed and operated by Microsoft. These geographically dispersed datacenters comply with key industry standards, such as ISO/IEC 27001:2013 and NIST SP 800-53, for security and reliability. The datacenters are managed, monitored, and administered by Microsoft operations staff. The operations staff has years of experience in delivering the world's largest online services with 24 x 7 continuity.

Securing the Azure infrastructure

This series of articles provides information about what Microsoft does to secure the Azure infrastructure. The articles address:

- [Physical security](#)
- [Availability](#)
- [Components and boundaries](#)
- [Network architecture](#)
- [Production network](#)
- [SQL Database](#)
- [Operations](#)
- [Monitoring](#)
- [Integrity](#)
- [Data protection](#)

Next steps

- Understand your [shared responsibility in the cloud](#).
- Learn how [Microsoft Defender for Cloud](#) can help you prevent, detect, and respond to threats with increased visibility and control over the security of your Azure resources.

Azure facilities, premises, and physical security

Article • 04/16/2025

This article describes what Microsoft does to secure the Azure infrastructure.

Datacenter infrastructure

Azure is composed of a [globally distributed datacenter infrastructure](#), supporting thousands of online services and spanning more than 100 highly secure facilities worldwide.

The infrastructure is designed to bring applications closer to users around the world, preserving data residency, and offering comprehensive compliance and resiliency options for customers. Azure has over 60 regions worldwide, and is available in 140 countries/regions.

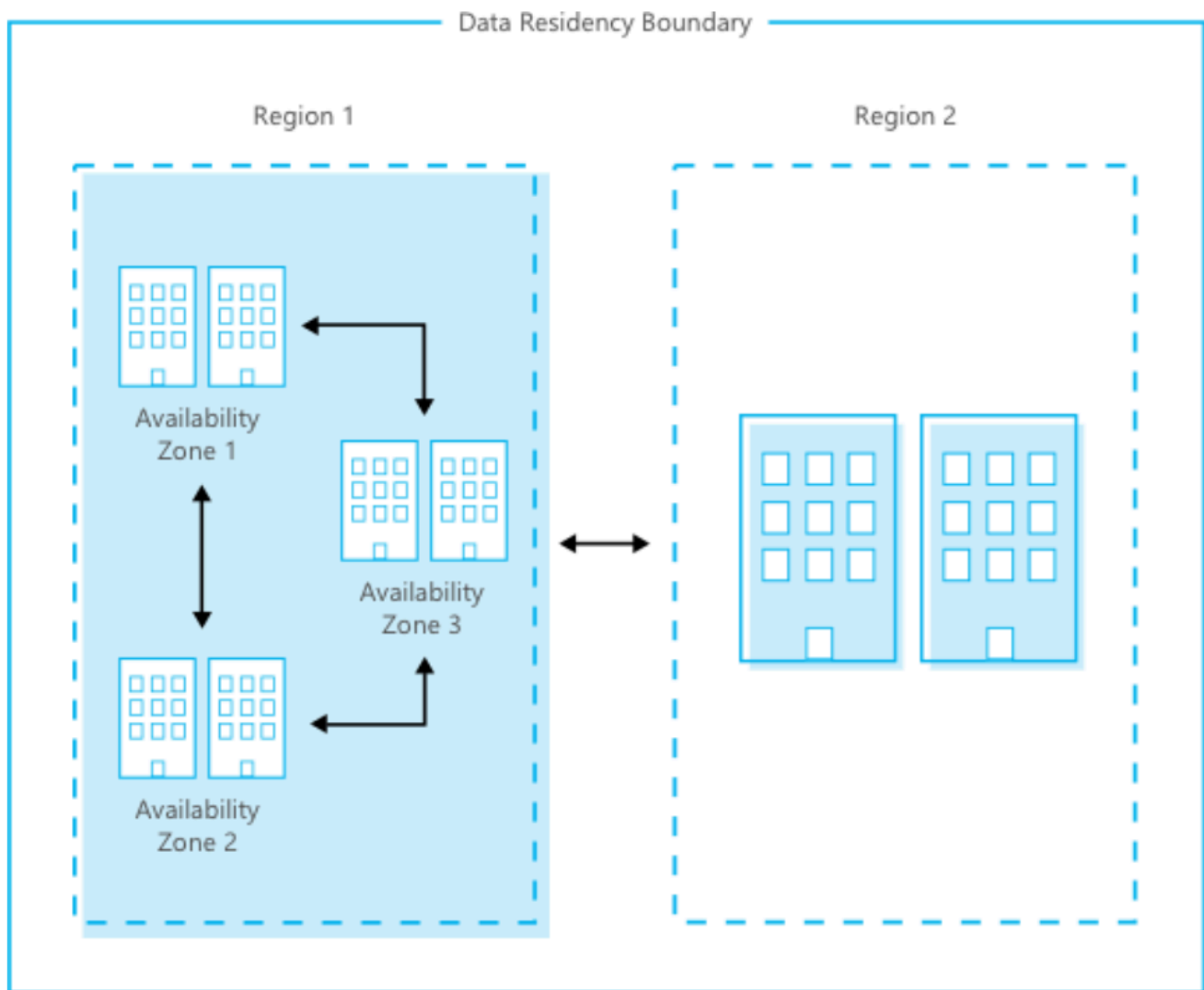
A region is a set of datacenters that is interconnected via a massive and resilient network. The network includes content distribution, load balancing, redundancy, and [data-link layer encryption by default](#) for all Azure traffic within a region or traveling between regions. With more global regions than any other cloud provider, Azure gives you the flexibility to deploy applications where you need them.

Azure regions are organized into geographies. An Azure geography ensures that data residency, sovereignty, compliance, and resiliency requirements are honored within geographical boundaries.

Geographies allow customers with specific data-residency and compliance needs to keep their data and applications close. Geographies are fault-tolerant to withstand complete region failure, through their connection to the dedicated, high-capacity networking infrastructure.

Availability zones are physically separate locations within an Azure region. Each availability zone is made up of one or more datacenters equipped with independent power, cooling, and networking. Availability zones allow you to run mission-critical applications with high availability and low-latency replication.

The following figure shows how the Azure global infrastructure pairs region and availability zones within the same data residency boundary for high availability, disaster recovery, and backup.



Geographically distributed datacenters enables Microsoft to be close to customers, to reduce network latency and allow for geo-redundant backup and failover.

Physical security

Microsoft designs, builds, and operates datacenters in a way that strictly controls physical access to the areas where your data is stored. Microsoft understands the importance of protecting your data, and is committed to helping secure the datacenters that contain your data. We have an entire division at Microsoft devoted to designing, building, and operating the physical facilities supporting Azure. This team is invested in maintaining state-of-the-art physical security.

Microsoft takes a layered approach to physical security, to reduce the risk of unauthorized users gaining physical access to data and the datacenter resources. Datacenters managed by Microsoft have extensive layers of protection: access approval at the facility's perimeter, at the building's perimeter, inside the building, and on the datacenter floor. Layers of physical security are:

- **Access request and approval.** You must request access prior to arriving at the datacenter. You're required to provide a valid business justification for your visit, such as compliance or auditing purposes. All requests are approved on a need-to-access basis by Microsoft employees. A need-to-access basis helps keep the number of individuals needed to complete a task in the datacenters to the bare minimum. After Microsoft grants permission, an individual only has access to the discrete area of the datacenter required, based on the approved business justification. Permissions are limited to a certain period of time, and then expire.
- **Visitor access.** Temporary access badges are stored within the access-controlled SOC and inventoried at the beginning and end of each shift. All visitors that have approved access to the datacenter are designated as *Escort Only* on their badges and are required to always remain with their escorts. Escorted visitors do not have any access levels granted to them and can only travel on the access of their escorts. The escort is responsible for reviewing the actions and access of their visitor during their visit to the datacenter. Microsoft requires visitors to surrender badges upon departure from any Microsoft facility. All visitor badges have their access levels removed before they are reused for future visits.
- **Facility's perimeter.** When you arrive at a datacenter, you're required to go through a well-defined access point. Typically, tall fences made of steel and concrete encompass every inch of the perimeter. There are cameras around the datacenters, with a security team monitoring their videos at all times. Security guard patrols ensure entry and exit are restricted to designated areas. Bollards and other measures protect the datacenter exterior from potential threats, including unauthorized access.
- **Building entrance.** The datacenter entrance is staffed with professional security officers who have undergone rigorous training and background checks. These security officers also routinely patrol the datacenter, and monitor the videos of cameras inside the datacenter at all times.
- **Inside the building.** After you enter the building, you must pass two-factor authentication with biometrics to continue moving through the datacenter. If your identity is validated, you can enter only the portion of the datacenter that you have approved access to. You can stay there only for the duration of the time approved.
- **Datacenter floor.** You are only allowed onto the floor that you're approved to enter. You are required to pass a full body metal detection screening. To reduce the risk of unauthorized data entering or leaving the datacenter without our knowledge, only approved devices can make their way into the datacenter floor. Additionally, video cameras monitor the front and back of every server rack. When you exit the datacenter

floor, you again must pass through full body metal detection screening. To leave the datacenter, you're required to pass through an additional security scan.

Physical security reviews

Periodically, we conduct physical security reviews of the facilities, to ensure the datacenters properly address Azure security requirements. The datacenter hosting provider personnel do not provide Azure service management. Personnel can't sign in to Azure systems and don't have physical access to the Azure collocation room and cages.

Data bearing devices

Microsoft uses best practice procedures and a wiping solution that is [NIST 800-88 compliant](#). For hard drives that can't be wiped, we use a destruction process that destroys it and renders the recovery of information impossible. This destruction process can be to disintegrate, shred, pulverize, or incinerate. We determine the means of disposal according to the asset type. We retain records of the destruction.

Equipment disposal

Upon a system's end-of-life, Microsoft operational personnel follow rigorous data handling and hardware disposal procedures to assure that hardware containing your data is not made available to untrusted parties. We use a secure erase approach for hard drives that support it. For hard drives that can't be wiped, we use a destruction process that destroys the drive and renders the recovery of information impossible. This destruction process can be to disintegrate, shred, pulverize, or incinerate. We determine the means of disposal according to the asset type. We retain records of the destruction. All Azure services use approved media storage and disposal management services.

Compliance

We design and manage the Azure infrastructure to meet a broad set of international and industry-specific compliance standards, such as ISO 27001, HIPAA, FedRAMP, SOC 1, and SOC 2. We also meet country-/region-specific standards, including Australia IRAP, UK G-Cloud, and Singapore MTCS. Rigorous third-party audits, such as those done by the British Standards Institute, verify adherence to the strict security controls these standards mandate.

For a full list of compliance standards that Azure adheres to, see the [Compliance offerings](#).

Next steps

To learn more about what Microsoft does to help secure the Azure infrastructure, see:

- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

Azure infrastructure availability

Article • 04/23/2025

This article provides information about what Microsoft does to secure the Azure infrastructure and provide maximum availability of customers' data. Azure provides robust availability, based on extensive redundancy achieved with virtualization technology.

Temporary outages and natural disaster

The Microsoft Cloud Infrastructure and Operations team designs, builds, operates, and improves the security of the cloud infrastructure. This team ensures that the Azure infrastructure is delivering high availability and reliability, high efficiency, and smart scalability. The team provides a more secure, private, and trusted cloud.

Uninterruptible power supplies and vast banks of batteries ensure that electricity remains continuous if a short-term power disruption occurs. Emergency generators provide backup power for extended outages and planned maintenance. If a natural disaster occurs, the datacenter can use onsite fuel reserves.

High-speed and robust fiber optic networks connect datacenters with other major hubs and internet users. Compute nodes host workloads closer to users to reduce latency, provide geo-redundancy, and increase overall service resiliency. A team of engineers works around the clock to ensure services are persistently available.

Microsoft ensures high availability through advanced monitoring and incident response, service support, and backup failover capability. Geographically distributed Microsoft operations centers operate 24/7/365. The Azure network is one of the largest in the world. The fiber optic and content distribution network connects datacenters and edge nodes to ensure high performance and reliability.

Disaster recovery

Azure keeps your data durable in two locations. You can choose the location of the backup site. In the primary location, Azure constantly maintains three healthy replicas of your data.

Database availability

Azure ensures that a database is internet accessible through an internet gateway with sustained database availability. Monitoring assesses the health and state of the active databases at five-minute time intervals.

Storage availability

Azure delivers storage through a highly scalable and durable storage service, which provides connectivity endpoints. This means that an application can access the storage service directly. The storage service processes incoming storage requests efficiently, with transactional integrity.

Next steps

To learn more about what Microsoft does to help secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

Azure information system components and boundaries

Article • 04/23/2025

This article provides a general description of the Azure architecture and management. The Azure system environment is made up of the following networks:

- Microsoft Azure production network (Azure network)
- Microsoft corporate network (corpnet)

Separate IT teams are responsible for operations and maintenance of these networks.

Azure architecture

Azure is a cloud computing platform and infrastructure for building, deploying, and managing applications and services through a network of datacenters. Microsoft manages these datacenters. Based on the number of resources you specify, Azure creates virtual machines (VMs) based on resource need. These VMs run on an Azure hypervisor, which is designed for use in the cloud and isn't accessible to the public.

On each Azure physical server node, there's a hypervisor that runs directly over the hardware. The hypervisor divides a node into a variable number of guest VMs. Each node also has one root VM, which runs the host operating system. Windows Firewall is enabled on each VM. You define which ports are addressable by configuring the service definition file. These ports are the only ones open and addressable, internally or externally. All traffic and access to the disk and network is mediated by the hypervisor and root operating system.

At the host layer, Azure VMs run a customized and hardened version of the latest Windows Server. Azure uses a version of Windows Server that includes only those components necessary to host VMs. This improves performance and reduces attack surface. Machine boundaries are enforced by the hypervisor, which doesn't depend on the operating system security.

Azure management by fabric controllers

In Azure, VMs running on physical servers (blades/nodes) are grouped into clusters of about 1000. The VMs are independently managed by a scaled-out and redundant platform software component called the fabric controller (FC).

Each FC manages the lifecycle of applications running in its cluster, and provisions and monitors the health of the hardware under its control. It runs autonomic operations, such as reincarnating VM instances on healthy servers when it determines that a server has failed. The

FC also performs application-management operations, such as deploying, updating, and scaling out applications.

The datacenter is divided into clusters. Clusters isolate faults at the FC level, and prevent certain classes of errors from affecting servers beyond the cluster in which they occur. FCs that serve a particular Azure cluster are grouped into an FC cluster.

Hardware inventory

The FC prepares an inventory of Azure hardware and network devices during the bootstrap configuration process. Any new hardware and network components entering the Azure production environment must follow the bootstrap configuration process. The FC is responsible for managing the entire inventory listed in the `datacenter.xml` configuration file.

FC-managed operating system images

The operating system team provides images, in the form of Virtual Hard Disks, deployed on all host and guest VMs in the Azure production environment. The team constructs these base images through an automated offline build process. The base image is a version of the operating system in which the kernel and other core components have been modified and optimized to support the Azure environment.

There are three types of fabric-managed operating system images:

- Host: A customized operating system that runs on host VMs.
- Native: A native operating system that runs on tenants (for example, Azure Storage). This operating system doesn't have any hypervisor.
- Guest: A guest operating system that runs on guest VMs.

The host and native FC-managed operating systems are designed for use in the cloud, and aren't publicly accessible.

Host and native operating systems

Host and native are hardened operating system images that host the fabric agents, and run on a compute node (runs as first VM on the node) and storage nodes. The benefit of using optimized base images of host and native is that it reduces the surface area exposed by APIs or unused components. These can present high security risks and increase the footprint of the operating system. Reduced-footprint operating systems only include the components necessary to Azure.

Guest operating system

Azure internal components running on guest operating system VMs have no opportunity to run Remote Desktop Protocol. Any changes to baseline configuration settings must go through the change and release management process.

Azure datacenters

The Microsoft Cloud Infrastructure and Operations (MCIO) team manages the physical infrastructure and datacenter facilities for all Microsoft online services. MCIO is primarily responsible for managing the physical and environmental controls within the datacenters, as well as managing and supporting outer perimeter network devices (such as edge routers and datacenter routers). MCIO is also responsible for setting up the bare minimum server hardware on racks in the datacenter. Customers have no direct interaction with Azure.

Service management and service teams

Various engineering groups, known as service teams, manage the support of the Azure service. Each service team is responsible for an area of support for Azure. Each service team must make an engineer available 24x7 to investigate and resolve failures in the service. Service teams don't, by default, have physical access to the hardware operating in Azure.

The service teams are:

- Application Platform
- Microsoft Entra ID
- Azure Compute
- Azure Net
- Cloud Engineering Services
- ISSD: Security
- Multifactor Authentication
- SQL Database
- Storage

Types of users

Employees (or contractors) of Microsoft are considered to be internal users. All other users are considered to be external users. All Azure internal users have their employee status categorized with a sensitivity level that defines their access to customer data (access or no access). User privileges to Azure (authorization permission after authentication takes place) are described in the following table:

Role	Internal or external	Sensitivity level	Authorized privileges and functions performed	Access type
Azure datacenter engineer	Internal	No access to customer data	Manage the physical security of the premises. Conduct patrols in and out of the datacenter, and monitor all entry points. Escort into and out of the datacenter certain non-cleared personnel who provide general services (such as dining or cleaning) or IT work within the datacenter. Conduct routine monitoring and maintenance of network hardware. Perform incident management and break-fix work by using various tools. Conduct routine monitoring and maintenance of the physical hardware in the datacenters. Access to environment on demand from property owners. Capable to perform forensic investigations, log incident reports, and require mandatory security training and policy requirements. Operational ownership and maintenance of critical security tools, such as scanners and log collection.	Persistent access to the environment.
Azure incident triage (rapid response engineers)	Internal	Access to customer data	Manage communications among MCIO, support, and engineering teams. Triage platform incidents, deployment issues, and service requests.	Just-in-time access to the environment, with limited persistent access to non-customer systems.
Azure deployment engineers	Internal	Access to customer data	Deploy and upgrade platform components, software, and scheduled configuration changes in support of Azure.	Just-in-time access to the environment, with limited persistent access to non-customer systems.
Azure customer outage support (tenant)	Internal	Access to customer data	Debug and diagnose platform outages and faults for individual compute tenants and Azure accounts. Analyze faults. Drive critical fixes to the platform or customer, and drive technical improvements across support.	Just-in-time access to the environment, with limited persistent

Role	Internal or external	Sensitivity level	Authorized privileges and functions performed	Access type
				access to non-customer systems.
Azure live site engineers (monitoring engineers) and incident	Internal	Access to customer data	Diagnose and mitigate platform health by using diagnostic tools. Drive fixes for volume drivers, repair items resulting from outages, and assist outage restoration actions.	Just-in-time access to the environment, with limited persistent access to non-customer systems.
Azure customers	External	N/A	N/A	N/A

Azure uses unique identifiers to authenticate organizational users and customers (or processes acting on behalf of organizational users). This applies to all assets and devices that are part of the Azure environment.

Azure internal authentication

Communications between Azure internal components are protected with TLS encryption. In most cases, the X.509 certificates are self-signed. Certificates with connections that can be accessed from outside the Azure network are an exception, as are certificates for the FCs. FCs have certificates issued by a Microsoft Certificate of Authority (CA) that is backed by a trusted root CA. This allows FC public keys to be rolled over easily. Additionally, Microsoft developer tools use FC public keys. When developers submit new application images, the images are encrypted with an FC public key in order to protect any embedded secrets.

Azure hardware device authentication

The FC maintains a set of credentials (keys and/or passwords) used to authenticate itself to various hardware devices under its control. Microsoft uses a system to prevent access to these credentials. Specifically, the transport, persistence, and use of these credentials is designed to prevent Azure developers, administrators, and backup services and personnel access to sensitive, confidential, or private information.

Microsoft uses encryption based on the FC's master identity public key. This occurs at FC setup and FC reconfiguration times, to transfer the credentials used to access networking hardware devices. When the FC needs the credentials, the FC retrieves and decrypts them.

Network devices

The Azure networking team configures network service accounts to enable an Azure client to authenticate to network devices (routers, switches, and load balancers).

Secure service administration

Azure operations personnel are required to use secure admin workstations (SAWs). Customers can implement similar controls by using privileged access workstations. With SAWs, administrative personnel use an individually assigned administrative account that is separate from the user's standard user account. The SAW builds on that account separation practice by providing a trustworthy workstation for those sensitive accounts.

Next steps

To learn more about what Microsoft does to help secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

Azure network architecture

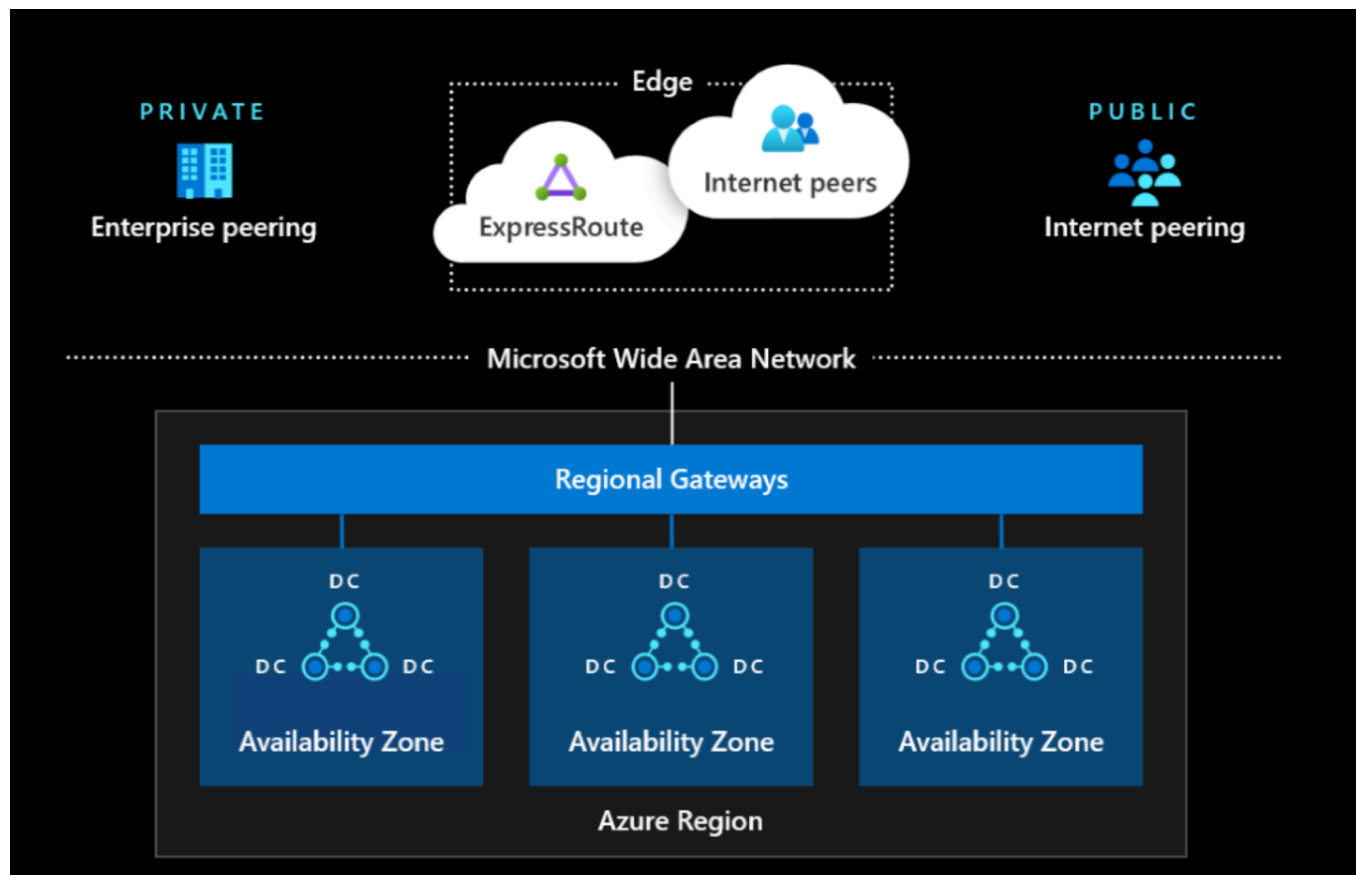
Article • 05/01/2025

The Azure network architecture provides connectivity from the Internet to the Azure datacenters. Any workload deployed (IaaS, PaaS, and SaaS) on Azure is leveraging the Azure datacenter network.

Network topology

The network architecture of an Azure datacenter consists of the following components:

- Edge network
- Wide area network
- Regional gateways network
- Datacenter network



Network components

A brief description of the network components.

- Edge network
 - Demarcation point between Microsoft networking and other networks (for example, Internet, Enterprise network)

- Provides Internet and [ExpressRoute](#) peering into Azure
- Wide area network
 - Microsoft intelligent backbone network covering the globe
 - Provides connectivity between [Azure regions](#) [↗](#)
- Regional gateway
 - Point of aggregation for all of the datacenters in an Azure region
 - Provides massive connectivity between datacenters within an Azure region (for example, multi hundred terabits per datacenter)
- Datacenter network
 - Provides connectivity between servers within the datacenter with low oversubscribed bandwidth

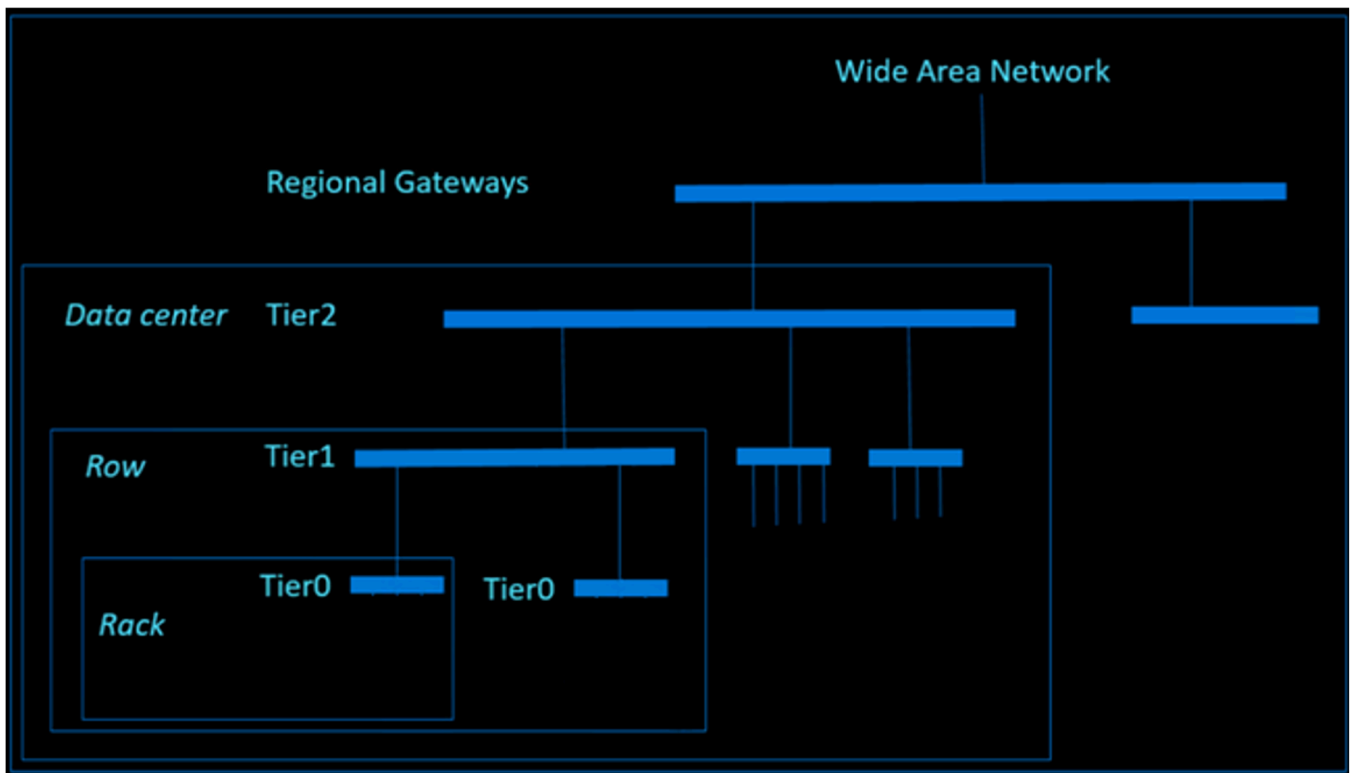
The above network components are designed to provide maximum availability to support always-on, always-available cloud business. The redundancy is designed and built into the network from the physical aspect all the way up to control protocol.

Datacenter network resiliency

Let's illustrate the resiliency design principle using datacenter network.

The datacenter network is a modified version of a [Clos network](#) [↗](#), providing high bi-sectional bandwidth for cloud scale traffic. The network is constructed using a large number of commodity devices to reduce the impact caused by individual hardware failure. These devices are strategically located in different physical locations with separate power and cooling domain to reduce impact of an environment event. On the control plane, all network devices are running as OSI model Layer 3 routing mode, which eliminates the historical issue of traffic loop. All paths between different tiers are active to provide high redundancy and bandwidth using Equal-Cost Multi-Path (ECMP) Routing.

The following diagram demonstrates that the datacenter network is constructed by different tiers of network devices. The bars in the diagram represent groups of network devices which provide redundancy and high bandwidth connectivity.



Next steps

To learn more about what Microsoft does to help secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

The Azure production network

The users of the Azure production network include both external customers who access their own Azure applications and internal Azure support personnel who manage the production network. This article discusses the security access methods and protection mechanisms for establishing connections to the Azure production network.

Internet routing and fault tolerance

A globally redundant internal and external Azure Domain Name Service (DNS) infrastructure, combined with multiple primary and secondary DNS server clusters, provides fault tolerance. At the same time, additional Azure network security controls, such as NetScaler, are used to prevent distributed denial of service (DDoS) attacks and protect the integrity of Azure DNS services.

The Azure DNS servers are located at multiple datacenter facilities. The Azure DNS implementation incorporates a hierarchy of secondary and primary DNS servers to publicly resolve Azure customer domain names. The domain names usually resolve to a CloudApp.net address, which wraps the virtual IP (VIP) address for the customer's service. Unique to Azure, the VIP that corresponds to internal dedicated IP (DIP) address of the tenant translation is done by the Microsoft load balancers responsible for that VIP.

Azure is hosted in geographically distributed Azure datacenters within the US, and it's built on state-of-the-art routing platforms that implement robust, scalable architectural standards.

Among the notable features are:

- Multiprotocol Label Switching (MPLS)-based traffic engineering, which provides efficient link utilization and graceful degradation of service if there is an outage.
- Networks are implemented with "need plus one" (N+1) redundancy architectures or better.
- Externally, datacenters are served by dedicated, high-bandwidth network circuits that redundantly connect properties with over 1,200 internet service providers globally at multiple peering points. This connection provides in excess of 2,000 gigabytes per second (GBps) of edge capacity.

Because Microsoft owns its own network circuits between datacenters, these attributes help the Azure offering achieve 99.9+ percent network availability without the need for traditional third-party internet service providers.

Connection to production network and associated firewalls

The Azure network internet traffic flow policy directs traffic to the Azure production network that's located in the nearest regional datacenter within the US. Because the Azure production datacenters maintain consistent network architecture and hardware, the traffic flow description that follows applies consistently to all datacenters.

After internet traffic for Azure is routed to the nearest datacenter, a connection is established to the access routers. These access routers serve to isolate traffic between Azure nodes and customer-instantiated VMs. Network infrastructure devices at the access and edge locations are the boundary points where ingress and egress filters are applied. These routers are configured through a tiered access-control list (ACL) to filter unwanted network traffic and apply traffic rate limits, if necessary. Traffic that is allowed by ACL is routed to the load balancers. Distribution routers are designed to allow only Microsoft-approved IP addresses, provide anti-spoofing, and establish TCP connections that use ACLs.

External load-balancing devices are located behind the access routers to perform network address translation (NAT) from internet-routable IPs to Azure internal IPs. The devices also route packets to valid production internal IPs and ports, and they act as a protection mechanism to limit exposing the internal production network address space.

By default, Microsoft enforces Hypertext Transfer Protocol Secure (HTTPS) for all traffic that's transmitted to customers' web browsers, including sign-in and all traffic thereafter. The use of TLS v1.2 enables a secure tunnel for traffic to flow through. ACLs on access and core routers ensure that the source of the traffic is consistent with what is expected.

An important distinction in this architecture, when it's compared to traditional security architecture, is that there are no dedicated hardware firewalls, specialized intrusion detection or prevention devices, or other security appliances that are normally expected before connections are made to the Azure production environment. Customers usually expect these hardware firewall devices in the Azure network; however, none are employed within Azure. Almost exclusively, those security features are built into the software that runs the Azure environment to provide robust, multi-layered security mechanisms, including firewall capabilities. Additionally, the scope of the boundary and associated sprawl of critical security devices is easier to manage and inventory, because it is managed by the software that's running Azure.

Core security and firewall features

Azure implements robust software security and firewall features at various levels to enforce security features that are usually expected in a traditional environment to protect the core Security Authorization boundary.

Azure security features

Azure implements host-based software firewalls inside the production network. Several core security and firewall features reside within the core Azure environment. These security features reflect a defense-in-depth strategy within the Azure environment. Customer data in Azure is protected by the following firewalls:

Hypervisor firewall (packet filter): This firewall is implemented in the hypervisor and configured by the fabric controller (FC) agent. This firewall protects the tenant that runs inside the VM from unauthorized access. By default, when a VM is created, all traffic is blocked and then the FC agent adds rules and exceptions in the filter to allow authorized traffic.

Two categories of rules are programmed here:

- **Machine config or infrastructure rules:** By default, all communication is blocked. Exceptions exist that allow a VM to send and receive Dynamic Host Configuration Protocol (DHCP) communications and DNS information, and send traffic to the "public" internet outbound to other VMs within the FC cluster and OS Activation server. Because the VMs' allowed list of outgoing destinations does not include Azure router subnets and other Microsoft properties, the rules act as a layer of defense for them.
- **Role configuration file rules:** Defines the inbound ACLs based on the tenants' service model. For example, if a tenant has a web front end on port 80 on a certain VM, port 80 is opened to all IP addresses. If the VM has a worker role running, the worker role is opened only to the VM within the same tenant.

Native host firewall: Azure Service Fabric and Azure Storage run on a native OS, which has no hypervisor and, therefore, Windows Firewall is configured with the preceding two sets of rules.

Host firewall: The host firewall protects the host partition, which runs the hypervisor. The rules are programmed to allow only the FC and jump boxes to talk to the host partition on a specific port. The other exceptions are to allow DHCP response and DNS replies. Azure uses a machine configuration file, which contains a template of firewall rules for the host partition. A host firewall exception also exists that allows VMs to communicate to host components, wire server, and metadata server, through specific protocol/ports.

Guest firewall: The Windows Firewall piece of the guest OS, which is configurable by customers on customer VMs and storage.

Additional security features that are built into the Azure capabilities include:

- Infrastructure components that are assigned IP addresses that are from DIPs. An attacker on the internet cannot address traffic to those addresses because it would not reach Microsoft. Internet gateway routers filter packets that are addressed solely to internal

addresses, so they would not enter the production network. The only components that accept traffic that's directed to VIPs are load balancers.

- Firewalls that are implemented on all internal nodes have three primary security architecture considerations for any given scenario:
 - Firewalls are placed behind the load balancer and accept packets from anywhere. These packets are intended to be externally exposed and would correspond to the open ports in a traditional perimeter firewall.
 - Firewalls accept packets only from a limited set of addresses. This consideration is part of the defensive in-depth strategy against DDoS attacks. Such connections are cryptographically authenticated.
 - Firewalls can be accessed only from select internal nodes. They accept packets only from an enumerated list of source IP addresses, all of which are DIPs within the Azure network. For example, an attack on the corporate network could direct requests to these addresses, but the attacks would be blocked unless the source address of the packet was one in the enumerated list within the Azure network.
 - The access router at the perimeter blocks outbound packets that are addressed to an address that's inside the Azure network because of its configured static routes.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

Azure SQL Database security features

Azure SQL Database provides a relational database service in Azure. To protect customer data and provide strong security features that customers expect from a relational database service, SQL Database has its own sets of security capabilities. These capabilities build upon the controls that are inherited from Azure.

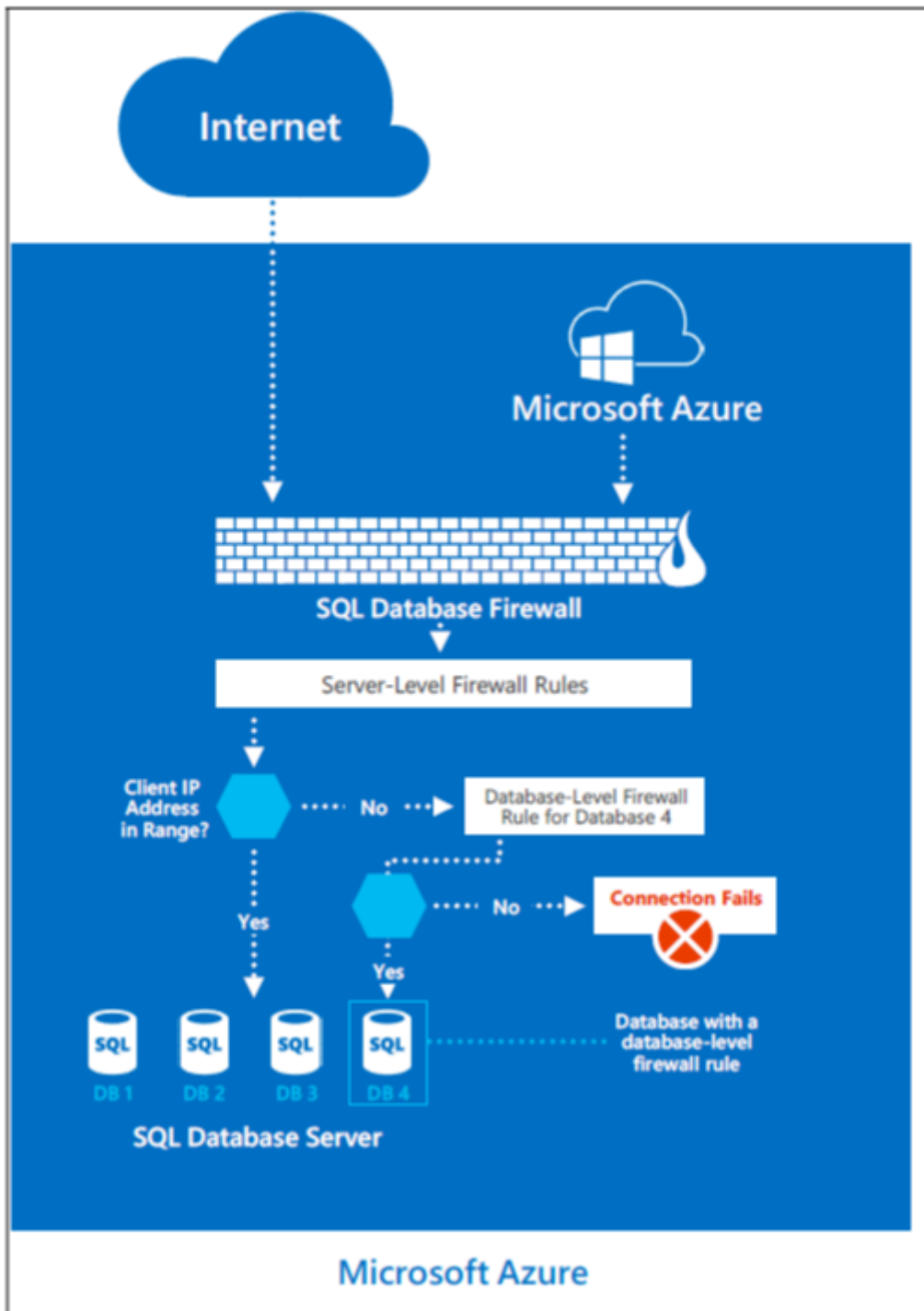
Security capabilities

Usage of the TDS protocol

Azure SQL Database supports only the tabular data stream (TDS) protocol, which requires the database to be accessible over only the default port of TCP/1433.

Azure SQL Database firewall

To help protect customer data, Azure SQL Database includes a firewall functionality, which by default prevents all access to SQL Database.



The gateway firewall can limit addresses, which allows customers granular control to specify ranges of acceptable IP addresses. The firewall grants access based on the originating IP address of each request.

Customers can achieve firewall configuration by using a management portal or programmatically using the Azure SQL Database Management REST API. The Azure SQL Database gateway firewall by default prevents all customer TDS access to Azure SQL Database. Customers must configure access by using access-control lists (ACLs) to permit Azure SQL Database connections by source and destination internet addresses, protocols, and port numbers.

DoSGuard

DosGuard, a SQL Database gateway service, reduces denial of service (DoS) attacks. DoSGuard actively tracks failed logins from IP addresses. If there are multiple failed logins from an IP address within a period of time, the IP address is blocked from accessing any resources in the service for a predefined time period.

In addition, the Azure SQL Database gateway performs:

- Secure channel capability negotiations to implement TDS FIPS 140-2 validated encrypted connections when it connects to the database servers.
- Stateful TDS packet inspection while it accepts connections from clients. The gateway validates the connection information. The gateway passes on the TDS packets to the appropriate physical server based on the database name that's specified in the connection string.

The overarching principle for network security of the Azure SQL Database offering is to allow only the connection and communication that is necessary to allow the service to operate. All other ports, protocols, and connections are blocked by default. Virtual local area networks (VLANs) and ACLs are used to restrict network communications by source and destination networks, protocols, and port numbers.

Mechanisms that are approved to implement network-based ACLs include ACLs on routers and load balancers. These mechanisms are managed by Azure networking, guest VM firewall, and Azure SQL Database gateway firewall rules configured by the customer.

Data segregation and customer isolation

The Azure production network is structured such that publicly accessible system components are segregated from internal resources. Physical and logical boundaries exist between web servers that provide access to the public-facing Azure portal and the underlying Azure virtual infrastructure, where customer application instances and customer data reside.

All publicly accessible information is managed within the Azure production network. The production network is:

- Subject to two-factor authentication and boundary protection mechanisms
- Uses the firewall and security feature set described in the previous section
- Uses data isolation functions noted in the next sections

Unauthorized systems and isolation of the FC

Because the fabric controller (FC) is the central orchestrator of the Azure fabric, significant controls are in place to mitigate threats to it, especially from potentially compromised FAs within customer applications. The FC doesn't recognize any hardware whose device information (for example, MAC address) isn't preloaded within the FC. The DHCP servers on the FC have configured lists of MAC addresses of the nodes they're willing to boot. Even if unauthorized systems are connected, they're not incorporated into fabric inventory, and therefore not connected or authorized to communicate with any system within the fabric inventory. This reduces the risk of unauthorized systems' communicating with the FC and gaining access to the VLAN and Azure.

VLAN isolation

The Azure production network is logically segregated into three primary VLANs:

- The main VLAN: Interconnects untrusted customer nodes.
- The FC VLAN: Contains trusted FCs and supporting systems.
- The device VLAN: Contains trusted network and other infrastructure devices.

Packet filtering

The IPFilter and the software firewalls that are implemented on the root OS and guest OS of the nodes enforce connectivity restrictions and prevent unauthorized traffic between VMs.

Hypervisor, root OS, and guest VMs

The hypervisor and the root OS manages the isolation of the root OS from the guest VMs and the guest VMs from one another.

Types of rules on firewalls

A rule is defined as:

{Src IP, Src Port, Destination IP, Destination Port, Destination Protocol, In/Out, Stateful/Stateless, Stateful Flow Timeout}.

Synchronous idle character (SYN) packets are allowed in or out only if any one of the rules permits it. For TCP, Azure uses stateless rules where the principle is that it allows only all non-SYN packets into or out of the VM. The security premise is that any host stack is resilient of ignoring a non-SYN if it hasn't seen a SYN packet previously. The TCP protocol itself is stateful, and in combination with the stateless SYN-based rule achieves an overall behavior of a stateful implementation.

For User Datagram Protocol (UDP), Azure uses a stateful rule. Every time a UDP packet matches a rule, a reverse flow is created in the other direction. This flow has a built-in timeout.

Customers are responsible for setting up their own firewalls on top of what Azure provides. Here, customers are able to define the rules for inbound and outbound traffic.

Production configuration management

Standard secure configurations are maintained by respective operations teams in Azure and Azure SQL Database. All configuration changes to production systems are documented and tracked through a central tracking system. Software and hardware changes are tracked through the central tracking system. Networking changes that relate to ACL are tracked using an ACL management service.

All configuration changes to Azure are developed and tested in the staging environment, and they're thereafter deployed in production environment. Software builds are reviewed as part of testing. Security and privacy checks are reviewed as part of entry checklist criteria. Changes are deployed on scheduled intervals by the respective deployment team. Releases are reviewed and signed off by the respective deployment team personnel before they're deployed into production.

Changes are monitored for success. On a failure scenario, the change is rolled back to its previous state or a hotfix is deployed to address the failure with approval of the designated personnel. Source Depot, Git, TFS, Master Data Services (MDS), runners, Azure security monitoring, the FC, and the WinFabric platform are used to centrally manage, apply, and verify the configuration settings in the Azure virtual environment.

Similarly, hardware and network changes have established validation steps to evaluate their adherence to the build requirements. The releases are reviewed and authorized through a coordinated change advisory board (CAB) of respective groups across the stack.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)

- [Azure infrastructure integrity](#)
 - [Azure customer data protection](#)
-

Last updated on 12/03/2025

Management and operation of the Azure production network

This article describes how Microsoft manages and operates the Azure production network to secure the Azure datacenters.

Monitor, log, and report

The management and operation of the Azure production network is a coordinated effort between the operations teams of Azure and Azure SQL Database. The teams use several system and application performance-monitoring tools in the environment. And they use appropriate tools to monitor network devices, servers, services, and application processes.

To ensure the secure execution of services running in the Azure environment, the operations teams implement multiple levels of monitoring, logging, and reporting, including the following actions:

- Primarily, the Microsoft Monitoring Agent (MMA) gathers monitoring and diagnostic log information from many places, including the fabric controller (FC) and the root operating system (OS), and writes it to log files. The agent eventually pushes a digested subset of the information into a pre-configured Azure storage account. In addition, the freestanding monitoring and diagnostic service reads various monitoring and diagnostic log data and summarizes the information. The monitoring and diagnostic service writes the information to an integrated log. Azure uses the custom-built Azure security monitoring, which is an extension to the Azure monitoring system. It has components that observe, analyze, and report on security-pertinent events from various points in the platform.
- The Azure SQL Database Windows Fabric platform provides management, deployment, development, and operational oversight services for Azure SQL Database. The platform offers distributed, multi-step deployment services, health monitoring, automatic repairs, and service version compliance. It provides the following services:
 - Service modeling capabilities with high-fidelity development environment (datacenter clusters are expensive and scarce).
 - One-click deployment and upgrade workflows for service bootstrap and maintenance.
 - Health reporting with automated repair workflows to enable self-healing.
 - Real time monitoring, alerting, and debugging facilities across the nodes of a distributed system.
 - Centralized collection of operational data and metrics for distributed root cause analysis and service insight.
 - Operational tooling for deployment, change management, and monitoring.

- The Azure SQL Database Windows Fabric platform and watchdog scripts run continuously and monitor in real time.

If any anomalies occur, the incident response process followed by the Azure incident triage team is activated. The appropriate Azure support personnel are notified to respond to the incident. Issue tracking and resolution are documented and managed in a centralized ticketing system. System uptime metrics are available under the non-disclosure agreement (NDA) and upon request.

Corporate network and multi-factor access to production

The corporate network user base includes Azure support personnel. The corporate network supports internal corporate functions and includes access to internal applications that are used for Azure customer support. The corporate network is both logically and physically separated from the Azure production network. Azure personnel access the corporate network by using Azure workstations and laptops. All users must have a Microsoft Entra account, including a username and password, to access corporate network resources. Corporate network access uses Microsoft Entra accounts, which are issued to all Microsoft personnel, contractors, and vendors and managed by Microsoft Information Technology. Unique user identifiers distinguish personnel based on their employment status at Microsoft.

Access to internal Azure applications is controlled through authentication with Active Directory Federation Services (AD FS). AD FS is a service hosted by Microsoft Information Technology that provides authentication of corporate network users through applying a secure token and user claims. AD FS enables internal Azure applications to authenticate users against the Microsoft corporate active directory domain. To access the production network from the corporate network environment, users must authenticate by using multi-factor authentication.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)

- [Azure customer data protection](#)
-

Last updated on 12/03/2025

Azure infrastructure monitoring

Configuration and change management

Azure reviews and updates configuration settings and baseline configurations of hardware, software, and network devices annually. Changes are developed, tested, and approved prior to entering the production environment from a development and/or test environment.

The baseline configurations that are required for Azure-based services are reviewed by the Azure security and compliance team and by service teams. A service team review is part of the testing that occurs before the deployment of their production service.

Vulnerability management

Security update management helps protect systems from known vulnerabilities. Azure uses integrated deployment systems to manage the distribution and installation of security updates for Microsoft software. Azure is also able to draw on the resources of the Microsoft Security Response Center (MSRC). The MSRC identifies, monitors, responds to, and resolves security incidents and cloud vulnerabilities around the clock, every day of the year.

Vulnerability scanning

Vulnerability scanning is performed on server operating systems, databases, and network devices. The vulnerability scans are performed on a quarterly basis at minimum. Azure contracts with independent assessors to perform penetration testing of the Azure boundary. Red-team exercises are also routinely performed and the results are used to make security improvements.

Protective monitoring

Azure security has defined requirements for active monitoring. Service teams configure active monitoring tools in accordance with these requirements. Active monitoring tools include the Microsoft Monitoring Agent (MMA) and System Center Operations Manager. These tools are configured to provide time alerts to Azure security personnel in situations that require immediate action.

Incident management

Microsoft implements a security incident management process to facilitate a coordinated response to incidents, should one occur.

If Microsoft becomes aware of unauthorized access to customer data that's stored on its equipment or in its facilities, or it becomes aware of unauthorized access to such equipment or facilities resulting in loss, disclosure, or alteration of customer data, Microsoft takes the following actions:

- Promptly notifies the customer of the security incident.
- Promptly investigates the security incident and provides customers detailed information about the security incident.
- Takes reasonable and prompt steps to mitigate the effects and minimize any damage resulting from the security incident.

An incident management framework has been established that defines roles and allocates responsibilities. The Azure security incident management team is responsible for managing security incidents, including escalation, and ensuring the involvement of specialist teams when necessary. Azure operations managers are responsible for overseeing the investigation and resolution of security and privacy incidents.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure integrity](#)
- [Azure customer data protection](#)

Azure infrastructure integrity

Software installation

All components in the software stack that are installed in the Azure environment are custom built following the Microsoft Security Development Lifecycle (SDL) process. All software components, including operating system (OS) images and SQL Database, are deployed as part of the change management and release management process. The OS that runs on all nodes is a customized version. The exact version is chosen by the fabric controller (FC) according to the role it intends for the OS to play. In addition, the host OS doesn't allow installation of any unauthorized software components.

Some Azure components are deployed as Azure customers on a guest VM running on a guest OS.

Virus scans on builds

Azure software component (including OS) builds have to undergo a virus scan that uses the Endpoint Protection anti-virus tool. Each virus scan creates a log within the associated build directory, detailing what was scanned and the results of the scan. The virus scan is part of the build source code for every component within Azure. Code isn't moved to production without having a clean and successful virus scan. If issues are noted, the build is frozen. The build goes to the security teams within Microsoft Security to identify where the "rogue" code entered the build.

Closed and locked environment

By default, Azure infrastructure nodes and guest VMs don't have user accounts created on them. In addition, default Windows administrator accounts are also disabled. Administrators from Azure live support can, with proper authentication, log in to these machines and administer the Azure production network for emergency repairs.

Azure SQL Database authentication

As with any implementation of SQL Server, user account management must be tightly controlled. Azure SQL Database supports only SQL Server authentication. To complement a customer's data security model, user accounts with strong passwords and configured with specific rights should be used as well.

ACLs and firewalls between the Microsoft corporate network and an Azure cluster

Access-control lists (ACLs) and firewalls between the service platform and the Microsoft corporate network protect SQL Database instances from unauthorized insider access. Further, only users from IP address ranges from the Microsoft corporate network can access the Windows Fabric platform-management endpoint.

ACLs and firewalls between nodes in a SQL Database cluster

As part of the defense-in depth-strategy, ACLs and a firewall have been implemented between nodes in a SQL Database cluster. All communication inside the Windows Fabric platform cluster and all running code is trusted.

Custom monitoring agents

SQL Database employs custom monitoring agents (MAs), also called watchdogs, to monitor the health of the SQL Database cluster.

Web protocols

Role instance monitoring and restart

Azure ensures that all deployed, running roles (internet-facing web, or back-end processing worker roles) are subject to sustained health monitoring. Health monitoring ensures that they effectively and efficiently deliver the services for which they've been provisioned. If a role becomes unhealthy, by either a critical fault in the application that's being hosted or an underlying configuration problem within the role instance itself, the FC detects the problem within the role instance and initiates a corrective state.

Compute connectivity

Azure ensures that the deployed application or service is reachable via standard web-based protocols. Virtual instances of internet-facing web roles have external internet connectivity and are reachable directly by web users. To protect the sensitivity and integrity of the operations that worker roles perform on behalf of the publicly accessible web role virtual instances, virtual

instances of back-end processing worker roles have external internet connectivity but can't be accessed directly by external web users.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure customer data protection](#)

Last updated on 12/03/2025

Azure customer data protection

Access to customer data by Microsoft operations and support personnel is denied by default. When access to data related to a support case is granted, it's only granted using a just-in-time (JIT) model using policies that are audited and vetted against our compliance and privacy policies. The access-control requirements are established by the following Azure Security Policy:

- No access to customer data, by default.
- No user or administrator accounts on customer virtual machines (VMs).
- Grant the least privilege that's required to complete task; audit and log access requests.

Azure support personnel are assigned unique corporate Active Directory accounts by Microsoft. Azure relies on Microsoft corporate Active Directory, managed by Microsoft Information Technology (MSIT), to control access to key information systems. Multifactor authentication is required, and access is granted only from secure consoles.

Data protection

Azure provides customers with strong data security, both by default and as customer options.

Data segregation: Azure is a multitenant service, which means that multiple customer deployments and VMs are stored on the same physical hardware. Azure uses logical isolation to segregate each customer's data from the data of others. Segregation provides the scale and economic benefits of multitenant services while rigorously preventing customers from accessing one another's data.

At-rest data protection: Customers are responsible for ensuring that data stored in Azure is encrypted in accordance with their standards. Azure offers a wide range of encryption capabilities, giving customers the flexibility to choose the solution that best meets their needs. Azure Key Vault helps customers easily maintain control of keys that are used by cloud applications and services to encrypt data. Azure Disk Encryption enables customers to encrypt VMs. Azure Storage Service Encryption makes it possible to encrypt all data placed into a customer's storage account.

In-transit data protection: Microsoft provides many options that can be utilized by customers for securing data in transit internally within the Azure network and externally across the Internet to the end user. These include communication through Virtual Private Networks (utilizing IPsec/IKE encryption), Transport Layer Security (TLS) 1.2 or later (via Azure components such as Application Gateway or Azure Front Door), protocols directly on the Azure virtual machines (such as Windows IPsec or SMB), and more.

Additionally, "encryption by default" using MACsec (an IEEE standard at the data-link layer) is enabled for all Azure traffic traveling between Azure datacenters to ensure confidentiality and integrity of customer data.

Data redundancy: Microsoft helps ensure that data is protected if there's a cyberattack or physical damage to a datacenter. Customers may opt for:

- In-country/region storage for compliance or latency considerations.
- Out-of-country/region storage for security or disaster recovery purposes.

Data can be replicated within a selected geographic area for redundancy but can't be transmitted outside it. Customers have multiple options for replicating data, including the number of copies and the number and location of replication datacenters.

When you create your storage account, select one of the following replication options:

- **Locally redundant storage (LRS):** Locally redundant storage maintains three copies of your data. LRS is replicated three times within a single facility in a single region. LRS protects your data from normal hardware failures, but not from a failure of a single facility.
- **Zone-redundant storage (ZRS):** Zone-redundant storage maintains three copies of your data. ZRS is replicated three times across two to three facilities to provide higher durability than LRS. Replication occurs within a single region or across two regions. ZRS helps ensure that your data is durable within a single region.
- **Geo-redundant storage (GRS):** Geo-redundant storage is enabled for your storage account by default when you create it. GRS maintains six copies of your data. With GRS, your data is replicated three times within the primary region. Your data is also replicated three times in a secondary region hundreds of miles away from the primary region, providing the highest level of durability. If a failure at the primary region, Azure Storage fails over to the secondary region. GRS helps ensure that your data is durable in two separate regions.

Data destruction: When customers delete data or leave Azure, Microsoft follows strict standards for deleting data and the physical destruction of decommissioned hardware. Microsoft executes a complete deletion of data on customer request and on contract termination. For more information, see [Data management at Microsoft](#).

Customer data ownership

Microsoft doesn't inspect, approve, or monitor applications that customers deploy to Azure. Moreover, Microsoft doesn't know what kind of data customers choose to store in Azure. Microsoft doesn't claim data ownership over the customer information entered into Azure.

Records management

Azure established internal records-retention requirements for back-end data. Customers are responsible for identifying their own record retention requirements. For records that are stored in Azure, customers are responsible for extracting their data and retaining their content outside of Azure for a customer-specified retention period.

Azure allows customers to export data and audit reports from the product. The exports are saved locally to retain the information for a customer-defined retention time period.

Electronic discovery (e-discovery)

Azure customers are responsible for complying with e-discovery requirements in their use of Azure services. If Azure customers must preserve their customer data, they may export and save the data locally. Additionally, customers can request exports of their data from the Azure Customer Support department. In addition to allowing customers to export their data, Azure conducts extensive logging and monitoring internally.

Next steps

To learn more about what Microsoft does to secure the Azure infrastructure, see:

- [Azure facilities, premises, and physical security](#)
- [Azure infrastructure availability](#)
- [Azure information system components and boundaries](#)
- [Azure network architecture](#)
- [Azure production network](#)
- [Azure SQL Database security features](#)
- [Azure production operations and management](#)
- [Azure infrastructure monitoring](#)
- [Azure infrastructure integrity](#)

Platform integrity and security overview

The Azure fleet is composed of millions of servers (hosts) with thousands more added on a daily basis. Thousands of hosts also undergo maintenance on a daily basis through reboots, operating system refreshes, or repairs. Before a host can join the fleet and begin accepting customer workloads, Microsoft verifies that the host is in a secure and trustworthy state. This verification ensures that malicious or inadvertent changes have not occurred on boot sequence components during the supply chain or maintenance workflows.

Securing Azure hardware and firmware

This series of articles describe how Microsoft ensures integrity and security of hosts through various stages in their lifecycle, from manufacturing to sunset. The articles address:

- [Firmware security](#)
- [Platform code integrity](#)
- [UEFI Secure Boot](#)
- [Measured boot and host attestation](#)
- [Project Cerberus](#)
- [Encryption at rest](#)
- [Hypervisor security](#)

Next steps

- Learn how Microsoft actively partners within the cloud hardware ecosystem to drive continuous [firmware security improvements](#).
- Understand your [shared responsibility in the cloud](#).

Last updated on 11/04/2025

Firmware security

Article • 11/11/2022

This article describes how Microsoft secures the cloud hardware ecosystem and supply chains.

Securing the cloud hardware ecosystem

Microsoft actively partners within the cloud hardware ecosystem to drive continuous security improvements by:

- Collaborating with Azure hardware and firmware partners (such as component manufacturers and system integrators) to meet Azure hardware and firmware security requirements.
- Enabling partners to perform continuous assessment and improvement of their products' security posture using Microsoft-defined requirements in areas such as:
 - Firmware secure boot
 - Firmware secure recovery
 - Firmware secure update
 - Firmware cryptography
 - Locked down hardware
 - Granular debug telemetry
 - System support for TPM 2.0 hardware to enable measured boot
- Engaging in and contributing to the [Open Compute Project \(OCP\)](#) security project through the development of specifications. Specifications promote consistency and clarity for secure design and architecture in the ecosystem.

ⓘ Note

An example of our contribution to the OCP Security Project is the [Hardware Secure Boot](#) specification.

Securing hardware and firmware supply chains

Cloud hardware suppliers and vendors for Azure are also required to adhere to supply chain security processes and requirements developed by Microsoft. Hardware and

firmware development and deployment processes are required to follow the Microsoft [Security Development Lifecycle](#) (SDL) processes such as:

- Threat modeling
- Secure design reviews
- Firmware reviews and penetration testing
- Secure build and test environments
- Security vulnerability management and incident response

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Platform code integrity](#)
- [Secure boot](#)
- [Measured boot and host attestation](#)
- [Project Cerberus](#)
- [Encryption at rest](#)
- [Hypervisor security](#)

Platform code integrity

A significant challenge in operating a complex system like Microsoft Azure is ensuring that only authorized software is running in the system. Unauthorized software presents several risks to any business:

- Security risks such as dedicated attack tools, custom malware, and third-party software with known vulnerabilities
- Compliance risks when the approved change management process isn't used to bring in new software
- Quality risk from externally developed software, which may not meet the operational requirements of the business

In Azure, we face the same challenge and at significant complexity. We have thousands of servers running software developed and maintained by thousands of engineers. This presents a large attack surface that cannot be managed through business processes alone.

Adding an authorization gate

Azure uses a rich engineering process that implements gates on the security, compliance, and quality of the software we deploy. This process includes access control to source code, conducting peer code reviews, doing static analysis for security vulnerabilities, following Microsoft's [Security Development Lifecycle](#) (SDL), and conducting functional and quality testing. We need to guarantee that the software we deploy has flowed through this process. Code integrity helps us achieve that guarantee.

Code integrity as an authorization gate

Code integrity is a kernel level service that became available starting in Windows Server 2016. Code integrity can apply a strict execution control policy whenever a driver or a dynamically linked library (DLL) is loaded, an executable binary is executed, or a script is run. Similar systems, such as [DM-Verity](#), exist for Linux. A code integrity policy consists of a set of authorization indicators, either code signing certificates or [SHA256](#) file hashes, which the kernel matches before loading or executing a binary or script.

Code Integrity allows a system administrator to define a policy that authorizes only binaries and scripts that have been signed by particular certificates or match specified SHA256 hashes. The kernel enforces this policy by blocking execution of everything that doesn't meet the set policy.

A concern with a code integrity policy is that unless the policy is perfectly correct, it can block critical software in production and cause an outage. Given this concern, one may ask why it isn't sufficient to use security monitoring to detect when unauthorized software has executed. Code integrity has an audit mode that, instead of preventing execution, can alert when unauthorized software is run. Alerting certainly can add much value in addressing compliance risks, but for security risks such as ransomware or custom malware, delaying the response by even a few seconds can be the difference between protection and an adversary gaining a persistent foothold in your fleet. In Azure, we've invested significantly to manage any risk of code integrity contributing to a customer impacting outage.

Build process

As discussed above, the Azure build system has a rich set of tests to ensure software changes are secure and compliant. Once a build has progressed through validation, the build system signs it using an Azure build certificate. The certificate indicates the build has passed through the entire change management process. The final test that the build goes through is called Code Signature Validation (CSV). CSV confirms the newly built binaries meet the code integrity policy before we deploy to production. This gives us high confidence that we won't cause a customer impacting outage because of incorrectly signed binaries. If CSV finds a problem, the build breaks and the relevant engineers are paged to investigate and fix the issue.

Safety during deployment

Even though we perform CSV for every build, there's still a chance that some change or inconsistency in production may cause a code integrity related outage. For example, a machine may be running an old version of the code integrity policy or it may be in an unhealthy state that produces false positives in code integrity. (At Azure scale, we've seen it all.) As such, we need to continue to protect against the risk of an outage during deployment.

All changes in Azure are required to deploy through a series of stages. The first of these are internal Azure testing instances. The next stage is used only to serve other Microsoft product teams. The final stage serves third-party customers. When a change is deployed, it moves to each of these stages in turn, and pauses to measure the health of the stage. If the change is found to have no negative impact, then it moves to the next stage. If we make a bad change to a code integrity policy, the change is detected during this staged deployment and rolled back.

Incident response

Even with this layered protection, it's still possible that some server in the fleet may block properly authorized software and cause a customer facing issue, one of our worst-case

scenarios. Our final layer of defense is human investigation. Each time code integrity blocks a file, it raises an alert for the on-call engineers to investigate. The alert allows us to start security investigations and intervene, whether the issue is an indicator of a real attack, a false positive, or other customer-impacting situation. This minimizes the time it takes to mitigate any code integrity related issues.

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Firmware security](#)
- [Secure boot](#)
- [Measured boot and host attestation](#)
- [Project Cerberus](#)
- [Encryption at rest](#)
- [Hypervisor security](#)

Last updated on 12/03/2025

Secure Boot

Article • 09/29/2024

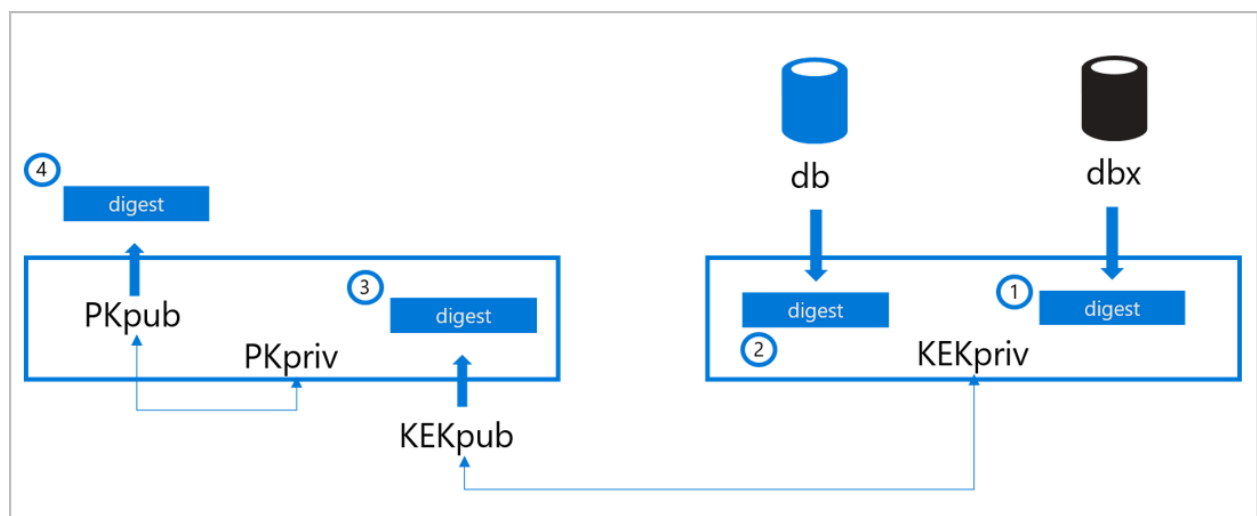
Secure Boot is a feature of the [Unified Extensible Firmware Interface \(UEFI\)](#) that requires all low-level firmware and software components to be verified before loading. During boot, UEFI Secure Boot checks the signature of each piece of boot software, including UEFI firmware drivers (also known as option ROMs), Extensible Firmware Interface (EFI) applications, and the operating system drivers and binaries. If the signatures are valid or trusted by the Original Equipment Manufacturer (OEM), the machine boots and the firmware gives control to the operating system.

Components and process

Secure Boot relies on these critical components:

- Platform key (PK) - Establishes trust between the platform owner (Microsoft) and the firmware. The public half is PKpub and the private half is PKpriv.
- Key enrollment key database (KEK) - Establishes trust between the OS and the platform firmware. The public half is KEKpub and the private half is KEKpriv.
- Signature database (db) - Holds the digests for trusted signers (public keys and certificates) of the firmware and software code modules authorized to interact with platform firmware.
- Revoked signatures database (dbx) – Holds revoked digests of code modules that are identified to be malicious, vulnerable, compromised, or untrusted. If a hash is in the signature db and the revoked signatures db, the revoked signatures database takes precedent.

The following figure and process explains how these components are updated:



The OEM stores the Secure Boot digests on the machine's nonvolatile RAM (NV-RAM) at the time of manufacturing.

1. The signature db is populated with the signers or image hashes of UEFI applications, operating system loaders (such as the Microsoft Operating System Loader or Boot Manager), and UEFI drivers that are trusted.
2. The revoked signatures dbx is populated with digests of modules that are no longer trusted.
3. The key enrollment key (KEK) database is populated with signing keys that can be used to update the signature database and revoked signatures database. The databases can be edited via updates that are signed with the correct key or via updates by a physically present authorized user using firmware menus.
4. After the db, dbx, and KEK databases are added and final firmware validation and testing is complete, the OEM locks the firmware from editing and generates a platform key (PK). The PK can be used to sign updates to the KEK or to turn off Secure Boot.

During each stage in the boot process, the digests of the firmware, bootloader, operating system, kernel drivers, and other boot chain artifacts are calculated and compared to acceptable values. Firmware and software that are discovered to be untrusted aren't allowed to load. Thus, low-level malware injection or preboot malware attacks can be blocked.

Secure Boot on the Azure fleet

Today, every machine that is onboarded and deployed to the Azure compute fleet to host customer workloads comes from factory floors with Secure Boot enabled. Targeted tooling and processes are in place at every stage in the hardware buildout and integration pipeline to ensure that Secure Boot enablement isn't reverted by accident or by malicious intent.

Validating that the db and dbx digests are correct ensures:

- Bootloader is present in one of the db entries
- Bootloader's signature is valid
- Host boots with trusted software

By validating the signatures of KEKpub and PKpub, we can confirm that only trusted parties have permission to modify the definitions of what software is considered trusted. Lastly, by ensuring that secure boot is active, we can validate that these definitions are being enforced.

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Firmware security](#)
 - [Platform code integrity](#)
 - [Measured boot and host attestation](#)
 - [Project Cerberus](#)
 - [Encryption at rest](#)
 - [Hypervisor security](#)
-

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

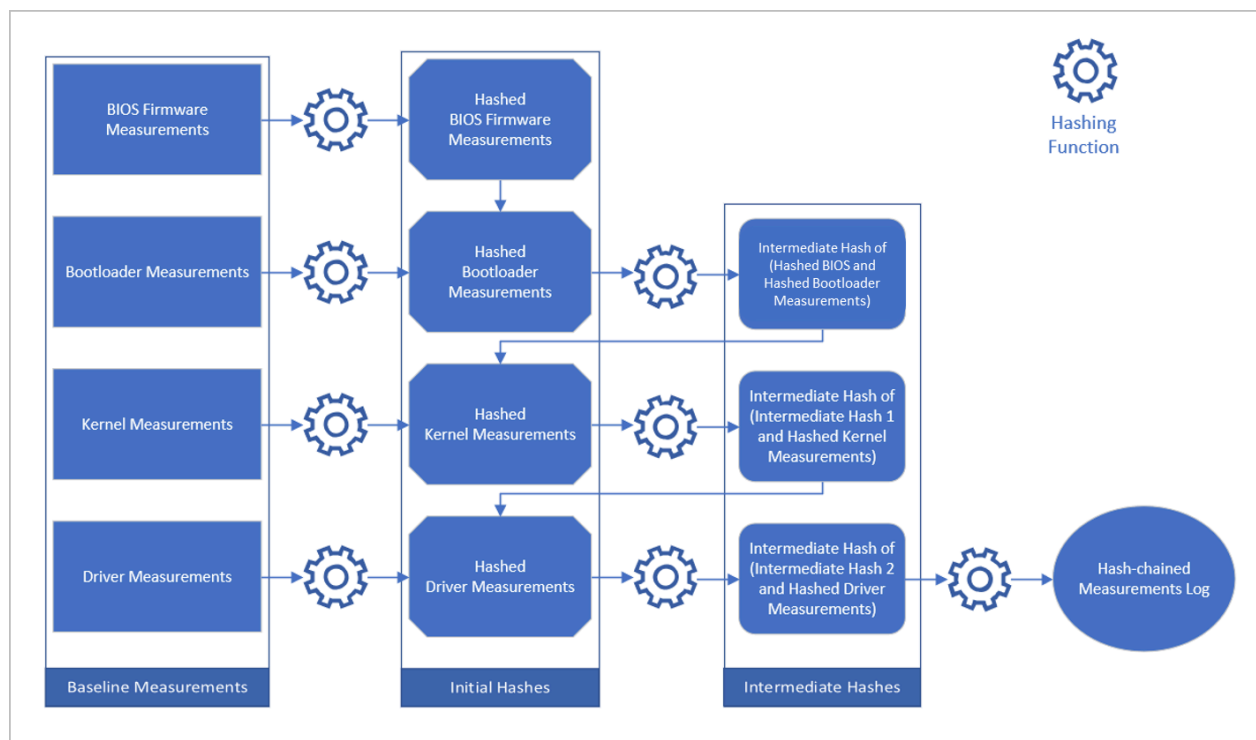
Measured boot and host attestation

Article • 09/10/2024

This article describes how Microsoft ensures integrity and security of hosts through measured boot and host attestation.

Measured boot

The [Trusted Platform Module \(TPM\)](#) is a tamper-proof, cryptographically secure auditing component with firmware supplied by a trusted third party. The boot configuration log contains hash-chained measurements recorded in its Platform Configuration Registers (PCR) when the host last underwent the bootstrapping sequence. The following figure shows this recording process. Incrementally adding a previously hashed measurement to the next measurement's hash and running the hashing algorithm on the union accomplishes hash-chaining.



Attestation is accomplished when a host furnishes proof of its configuration state using its boot configuration log (TCGLog). Forgery of a boot log is difficult because the TPM doesn't expose its PCR values other than the read and extend operations. Furthermore, the credentials supplied by the Host Attestation Service are sealed to specific PCR values. The use of hash-chaining makes it computationally infeasible to spoof or unseal the credentials out-of-band.

Host Attestation Service

Host Attestation Service is a preventative measure that checks if host machines are trustworthy before they're allowed to interact with customer data or workloads. Host Attestation Service checks by validating a compliance statement (verifiable proof of the host's compliance) sent by each host against an attestation policy (definition of the secure state). The integrity of this system is assured by a [root-of-trust](#) provided by a TPM.

Host Attestation Service is present in each Azure cluster within a specialized locked-down environment. The locked down environment includes other gatekeeper services that participate in the host machine bootstrapping protocol. A public key infrastructure (PKI) acts as an intermediary for validating the provenance of attestation requests and as an identity issuer (contingent upon successful host attestation). The post-attestation credentials issued to the attesting host are sealed to its identity. Only the requesting host can unseal the credentials and leverage them for obtaining incremental permissions. This prevents against man-in-the middle and spoofing attacks.

If an Azure host arrives from factory with a security misconfiguration or is tampered with in the datacenter, its TCGLog contains indicators of compromise flagged by the Host Attestation Service upon the next attestation, which causes an attestation failure. Attestation failures prevent the Azure fleet from trusting the offending host. This prevention effectively blocks all communications to and from the host and triggers an incident workflow. Investigation and a detailed post-mortem analysis are conducted to determine root causes and any potential indications of compromise. It's only after the analysis is complete that a host is remediated and has the opportunity to join the Azure fleet and take on customer workloads.

Following is a high-level architecture of the host attestation service:



Attestation measurements

Following are examples of the many measurements captured today.

Secure Boot and Secure Boot keys

By validating that the signature database and revoked signatures database digests are correct, the Host Attestation Service assures the client agent considers the right software to be trusted. By validating the signatures of the public key enrollment key database and public platform key, the Host Attestation Service confirms that only trusted parties have permission to modify the definitions of what software is considered trusted. Lastly, by ensuring that secure boot is active the Host Attestation Service validates these definitions are being enforced.

Debug controls

Debuggers are powerful tools for developers. However, the unfettered access to memory and other debug commands could weaken data protection and system integrity if given to a non-trusted party. Host Attestation Service ensures any kind of debugging is disabled on boot on production machines.

Code integrity

UEFI [Secure Boot](#) ensures that only trusted low-level software can run during the boot sequence. The same checks, though, must also be applied in the post-boot environment to drivers and other executables with kernel-mode access. To that end, a code integrity (CI) policy is used to define which drivers, binaries, and other executables are considered trusted by specifying valid and invalid signatures. These policies are enforced. Violations of policy generate alerts to the security incident response team for investigation.

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Firmware security](#)
- [Platform code integrity](#)
- [Secure boot](#)
- [Project Cerberus](#)
- [Encryption at rest](#)
- [Hypervisor security](#)

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

Project Cerberus

Article • 08/29/2024

Cerberus is a NIST 800-193 compliant hardware root-of-trust with an identity that cannot be cloned. Cerberus is designed to further raise the security posture of Azure infrastructure by providing a strong anchor of trust for firmware integrity.

Enabling an anchor of trust

Every Cerberus chip has a unique cryptographic identity that is established using a signed certificate chain rooted to a Microsoft certificate authority (CA). Measurements obtained from Cerberus can be used to validate integrity of components such as:

- Host
- Baseboard Management Controller (BMC)
- All peripherals, including network interface card and [system-on-a-chip](#) [↗] (SoC)

This anchor of trust helps defend platform firmware from:

- Compromised firmware binaries running on the platform
- Malware and hackers that exploit bugs in the operating system, application, or hypervisor
- Certain types of supply chain attacks (manufacturing, assembly, transit)
- Malicious insiders with administrative privileges or access to hardware

Cerberus attestation

Cerberus authenticates firmware integrity for server components using a Platform Firmware Manifest (PFM). PFM defines a list of authorized firmware versions and provides a platform measurement to the Azure [Host Attestation Service](#). The Host Attestation Service validates the measurements and makes a determination to only allow trusted hosts to join the Azure fleet and host customer workloads.

In conjunction with the Host Attestation Service, Cerberus' capabilities enhance and promote a highly secure Azure production infrastructure.

📌 Note

To learn more, see the [Project Cerberus](#) [↗] information on GitHub.

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Firmware security](#)
 - [Platform code integrity](#)
 - [Secure boot](#)
 - [Measured boot and host attestation](#)
 - [Encryption at rest](#)
 - [Hypervisor security](#)
-

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

Azure Data Encryption at rest

Microsoft Azure includes tools to safeguard data according to your company's security and compliance needs. This paper focuses on:

- How data is protected at rest across Microsoft Azure
- Discusses the various components taking part in the data protection implementation,
- Reviews pros and cons of the different key management protection approaches.

Encryption at Rest is a common security requirement. In Azure, organizations can encrypt data at rest without the risk or cost of a custom key management solution. Organizations have the option of letting Azure completely manage Encryption at Rest. Additionally, organizations have various options to closely manage encryption or encryption keys.

What is encryption at rest?

Encryption is the secure encoding of data used to protect confidentiality of data. The Encryption at Rest designs in Azure use symmetric encryption to encrypt and decrypt large amounts of data quickly according to a simple conceptual model:

- A symmetric encryption key is used to encrypt data as it is written to storage.
- The same encryption key is used to decrypt that data as it is readied for use in memory.
- Data may be partitioned, and different keys may be used for each partition.
- Keys must be stored in a secure location with identity-based access control and audit policies. Data encryption keys which are stored outside of secure locations are encrypted with a key encryption key kept in a secure location.

In practice, key management and control scenarios, as well as scale and availability assurances, require additional constructs. Microsoft Azure Encryption at Rest concepts and components are described below.

The purpose of encryption at rest

Encryption at rest provides data protection for stored data (at rest). Attacks against data at-rest include attempts to obtain physical access to the hardware on which the data is stored, and then compromise the contained data. In such an attack, a server's hard drive may have been mishandled during maintenance allowing an attacker to remove the hard drive. Later the attacker would put the hard drive into a computer under their control to attempt to access the data.

Encryption at rest is designed to prevent the attacker from accessing the unencrypted data by ensuring the data is encrypted when on disk. If an attacker obtains a hard drive with encrypted

data but not the encryption keys, the attacker must defeat the encryption to read the data. This attack is much more complex and resource consuming than accessing unencrypted data on a hard drive. For this reason, encryption at rest is highly recommended and is a high priority requirement for many organizations.

Encryption at rest may also be required by an organization's need for data governance and compliance efforts. Industry and government regulations such as HIPAA, PCI and FedRAMP, lay out specific safeguards regarding data protection and encryption requirements. Encryption at rest is a mandatory measure required for compliance with some of those regulations. For more information on Microsoft's approach to FIPS 140-2 validation, see [Federal Information Processing Standard \(FIPS\) Publication 140-2](#).

In addition to satisfying compliance and regulatory requirements, encryption at rest provides defense-in-depth protection. Microsoft Azure provides a compliant platform for services, applications, and data. It also provides comprehensive facility and physical security, data access control, and auditing. However, it's important to provide additional "overlapping" security measures in case one of the other security measures fails and encryption at rest provides such a security measure.

Microsoft is committed to encryption at rest options across cloud services and giving customers control of encryption keys and logs of key use. Additionally, Microsoft is working towards encrypting all customer data at rest by default.

Azure Encryption at Rest Components

As described previously, the goal of encryption at rest is that data that is persisted on disk is encrypted with a secret encryption key. To achieve that goal secure key creation, storage, access control, and management of the encryption keys must be provided. Though details may vary, Azure services Encryption at Rest implementations can be described in terms illustrated in the following diagram.



Azure Key Vault

The storage location of the encryption keys and access control to those keys is central to an encryption at rest model. The keys need to be highly secured but manageable by specified users and available to specific services. For Azure services, Azure Key Vault is the recommended key storage solution and provides a common management experience across services. Keys are stored and managed in key vaults, and access to a key vault can be given to users or services. Azure Key Vault supports customer creation of keys or import of customer keys for use in customer-managed encryption key scenarios.

Microsoft Entra ID

Permissions to use the keys stored in Azure Key Vault, either to manage or to access them for Encryption at Rest encryption and decryption, can be given to Microsoft Entra accounts.

Envelope Encryption with a Key Hierarchy

More than one encryption key is used in an encryption at rest implementation. Storing an encryption key in Azure Key Vault ensures secure key access and central management of keys. However, service local access to encryption keys is more efficient for bulk encryption and decryption than interacting with Key Vault for every data operation, allowing for stronger encryption and better performance. Limiting the use of a single encryption key decreases the risk that the key will be compromised and the cost of re-encryption when a key must be replaced. Azure encryption at rest models use envelope encryption, where a key encryption key encrypts a data encryption key. This model forms a key hierarchy which is better able to address performance and security requirements:

- **Data Encryption Key (DEK)** – A symmetric AES256 key used to encrypt a partition or block of data, sometimes also referred to as simply a Data Key. A single resource may have many partitions and many Data Encryption Keys. Encrypting each block of data with a different key makes crypto analysis attacks more difficult. And keeping DEKs local to the service encrypting and decrypting data maximizes performance.
- **Key Encryption Key (KEK)** – An encryption key used to encrypt the Data Encryption Keys using envelope encryption, also referred to as wrapping. Use of a Key Encryption Key that never leaves Key Vault allows the data encryption keys themselves to be encrypted and controlled. The entity that has access to the KEK may be different than the entity that requires the DEK. An entity may broker access to the DEK to limit the access of each DEK to a specific partition. Since the KEK is required to decrypt the DEKs, customers can cryptographically erase DEKs and data by disabling of the KEK.

Resource providers and application instances store the encrypted Data Encryption Keys as metadata. Only an entity with access to the Key Encryption Key can decrypt these Data Encryption Keys. Different models of key storage are supported. For more information, see [data encryption models](#).

Encryption at rest in Microsoft cloud services

Microsoft Cloud services are used in all three cloud models: IaaS, PaaS, SaaS. Below you have examples of how they fit on each model:

- Software services, referred to as Software as a Service or SaaS, which have applications provided by the cloud such as Microsoft 365.
- Platform services in which customers use the cloud for things like storage, analytics, and service bus functionality in their applications.
- Infrastructure services, or Infrastructure as a Service (IaaS) in which customer deploys operating systems and applications that are hosted in the cloud and possibly leveraging other cloud services.

Encryption at rest for SaaS customers

Software as a Service (SaaS) customers typically have encryption at rest enabled or available in each service. Microsoft 365 has several options for customers to verify or enable encryption at rest. For information about Microsoft 365 services, see [Encryption in Microsoft 365](#).

Encryption at rest for PaaS customers

Platform as a Service (PaaS) customer's data typically resides in a storage service such as Blob Storage but may also be cached or stored in the application execution environment, such as a

virtual machine. To see the encryption at rest options available to you, examine the [Data encryption models](#) for the storage and application platforms that you use.

Encryption at rest for IaaS customers

Infrastructure as a Service (IaaS) customers can have a variety of services and applications in use. IaaS services can enable encryption at rest in their Azure hosted virtual machines and VHDs using encryption at host.

Encrypted storage

Like PaaS, IaaS solutions can leverage other Azure services that store data encrypted at rest. In these cases, you can enable the Encryption at Rest support as provided by each consumed Azure service. The [Data encryption models](#) enumerates the major storage, services, and application platforms and the model of Encryption at Rest supported.

Encrypted compute

All Managed Disks, Snapshots, and Images are encrypted using Storage Service Encryption using a service-managed key. A more complete Encryption at Rest solution ensures that the data is never persisted in unencrypted form. While processing the data on a virtual machine, data can be persisted to the Windows page file or Linux swap file, a crash dump, or to an application log. To ensure this data is encrypted at rest, IaaS applications can use encryption at host on an Azure IaaS virtual machine.

Custom encryption at rest

It is recommended that whenever possible, IaaS applications leverage encryption at host and Encryption at Rest options provided by any consumed Azure services. In some cases, such as irregular encryption requirements or non-Azure based storage, a developer of an IaaS application may need to implement encryption at rest themselves. Developers of IaaS solutions can better integrate with Azure management and customer expectations by leveraging certain Azure components. Specifically, developers should use the Azure Key Vault service to provide secure key storage as well as provide their customers with key management options consistent with that of Azure platform services. Additionally, custom solutions should use Azure managed service identities to enable service accounts to access encryption keys. For developer information on Azure Key Vault and Managed Service Identities, see their respective SDKs.

Azure resource providers encryption model support

Microsoft Azure Services each support one or more of the encryption at rest models. For some services, however, one or more of the encryption models may not be applicable. For services that support customer-managed key scenarios, they may support only a subset of the key types that Azure Key Vault supports for key encryption keys. Additionally, services may release support for these scenarios and key types at different schedules. This section describes the encryption at rest support at the time of this writing for each of the major Azure data storage services.

Azure VM disk encryption

Any customer using Azure Infrastructure as a Service (IaaS) features can achieve encryption at rest for their IaaS VM disks through encryption at host. For more information, see [Encryption at host - End-to-end encryption for your VM](#).

Azure storage

All Azure Storage services (Blob storage, Queue storage, Table storage, and Azure Files) support server-side encryption at rest with customer-managed keys; some services additionally support client-side encryption.

- Server-side: All Azure Storage Services enable server-side encryption by default using service-managed keys, which is transparent to the application. For more information, see [Azure Storage Service Encryption for Data at Rest](#). They also support customer-managed keys in Azure Key Vault. For more information, see [Storage Service Encryption using customer-managed keys in Azure Key Vault](#).
- Client-side: Azure Blobs, Tables, and Queues support client-side encryption. When using client-side encryption, customers encrypt the data and upload the data as an encrypted blob. Key management is done by the customer. For more information, see [Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage](#).

Azure SQL Database

Azure SQL Database currently supports encryption at rest for Microsoft-managed service side and client-side encryption scenarios.

Support for server encryption is currently provided through the SQL feature called Transparent Data Encryption. Once an Azure SQL Database customer enables TDE, keys are automatically created and managed for them. Encryption at rest can be enabled at the database and server

levels. As of June 2017, [Transparent Data Encryption \(TDE\)](#) is enabled by default on newly created databases. Azure SQL Database supports RSA 2048-bit customer-managed keys in Azure Key Vault. For more information, see [Transparent Data Encryption with Bring Your Own Key support for Azure SQL Database and Data Warehouse](#).

Client-side encryption of Azure SQL Database data is supported through the [Always Encrypted](#) feature. Always Encrypted uses a key that is created and stored by the client. Customers can store the master key in a Windows certificate store, Azure Key Vault, or a local Hardware Security Module. Using SQL Server Management Studio, SQL users choose what key they'd like to use to encrypt which column.

Conclusion

Protection of customer data stored within Azure Services is of paramount importance to Microsoft. All Azure hosted services are committed to providing Encryption at Rest options. Azure services support either service-managed keys, customer-managed keys, or client-side encryption. Azure services are broadly enhancing Encryption at Rest availability and new options are planned for preview and general availability in the upcoming months.

Next steps

- See [data encryption models](#) to learn more about service-managed keys and customer-managed keys.
- Learn how Azure uses [double encryption](#) to mitigate threats that come with encrypting data.
- Learn what Microsoft does to ensure [platform integrity and security](#) of hosts traversing the hardware and firmware build-out, integration, operationalization, and repair pipelines.

Last updated on 10/25/2025

Hypervisor security on the Azure fleet

Article • 11/11/2022

The Azure hypervisor system is based on Windows Hyper-V. The hypervisor system enables the computer administrator to specify guest partitions that have separate address spaces. The separate address spaces allow you to load an operating system and applications operating in parallel of the (host) operating system that executes in the root partition of the computer. The host OS (also known as privileged root partition) has direct access to all the physical devices and peripherals on the system (storage controllers, networking adaptations). The host OS allows guest partitions to share the use of these physical devices by exposing “virtual devices” to each guest partition. Thus, an operating system executing in a guest partition has access to virtualized peripheral devices that are provided by virtualization services executing in the root partition.

The Azure hypervisor is built keeping the following security objectives in mind:

Objective	Source
Isolation	A security policy mandates no information transfer between VMs. This constraint requires capabilities in the Virtual Machine Manager (VMM) and hardware for isolation of memory, devices, the network, and managed resources such as persisted data.
VMM integrity	To achieve overall system integrity, the integrity of individual hypervisor components is established and maintained.
Platform integrity	The integrity of the hypervisor depends on the integrity of the hardware and software on which it relies. Although the hypervisor doesn't have direct control over the integrity of the platform, Azure relies on hardware and firmware mechanisms such as the Cerberus chip to protect and detect the underlying platform integrity. The VMM and guests are prevented from running if platform integrity is compromised.
Restricted access	Management functions are exercised only by authorized administrators connected over secure connections. A principle of least privilege is enforced by Azure role-based access control (Azure RBAC) mechanisms.
Audit	Azure enables audit capability to capture and protect data about what happens on a system so that it can later be inspected.

Microsoft’s approach to hardening the Azure hypervisor and the virtualization subsystem can be broken down into the following three categories.

Strongly defined security boundaries enforced by the hypervisor

The Azure hypervisor enforces multiple security boundaries between:

- Virtualized “guest” partitions and privileged partition (“host”)
- Multiple guests
- Itself and the host
- Itself and all guests

Confidentiality, integrity, and availability are assured for the hypervisor security boundaries. The boundaries defend against a range of attacks including side-channel information leaks, denial-of-service, and elevation of privilege.

The hypervisor security boundary also provides segmentation between tenants for network traffic, virtual devices, storage, compute resources, and all other VM resources.

Defense-in-depth exploit mitigations

In the unlikely event a security boundary has a vulnerability, the Azure hypervisor includes multiple layers of mitigations including:

- Isolation of host-based process hosting cross-VM components
- Virtualization-based security (VBS) for ensuring the integrity of user and kernel mode components from a secure world
- Multiple levels of exploit mitigations. Mitigations include address space layout randomization (ASLR), data execution prevention (DEP), arbitrary code guard, control flow integrity, and data corruption prevention
- Automatic initialization of stack variables at the compiler level
- Kernel APIs that automatically zero-initialize kernel heap allocations made by Hyper-V

These mitigations are designed to make the development of an exploit for a cross-VM vulnerability infeasible.

Strong security assurance processes

The attack surface related to the hypervisor includes software networking, virtual devices, and all cross-VM surfaces. The attack surface is tracked through automated build integration, which triggers periodic security reviews.

All VM attack surfaces are threat modeled, code reviewed, fuzzed, and tested by our RED team for security boundary violations. Microsoft has a [bug bounty program](#) that pays an award for relevant vulnerabilities in eligible product versions for Microsoft Hyper-V.

 Note

Learn more about **strong security assurance processes** in Hyper-V.

Next steps

To learn more about what we do to drive platform integrity and security, see:

- [Firmware security](#)
- [Platform code integrity](#)
- [Secure boot](#)
- [Measured boot and host attestation](#)
- [Project Cerberus](#)
- [Encryption at rest](#)

Isolation in the Azure Public Cloud

Azure allows you to run applications and virtual machines (VMs) on shared physical infrastructure. One of the prime economic motivations to running applications in a cloud environment is the ability to distribute the cost of shared resources among multiple customers. This practice of multi-tenancy improves efficiency by multiplexing resources among disparate customers at low costs. Unfortunately, it also introduces the risk of sharing physical servers and other infrastructure resources to run your sensitive applications and VMs that may belong to an arbitrary and potentially malicious user.

This article outlines how Azure provides isolation against both malicious and non-malicious users and serves as a guide for architecting cloud solutions by offering various isolation choices to architects.

Tenant Level Isolation

One of the primary benefits of cloud computing is concept of a shared, common infrastructure across numerous customers simultaneously, leading to economies of scale. This concept is called multi-tenancy. Microsoft works continuously to ensure that the multi-tenant architecture of Microsoft Cloud Azure supports security, confidentiality, privacy, integrity, and availability standards.

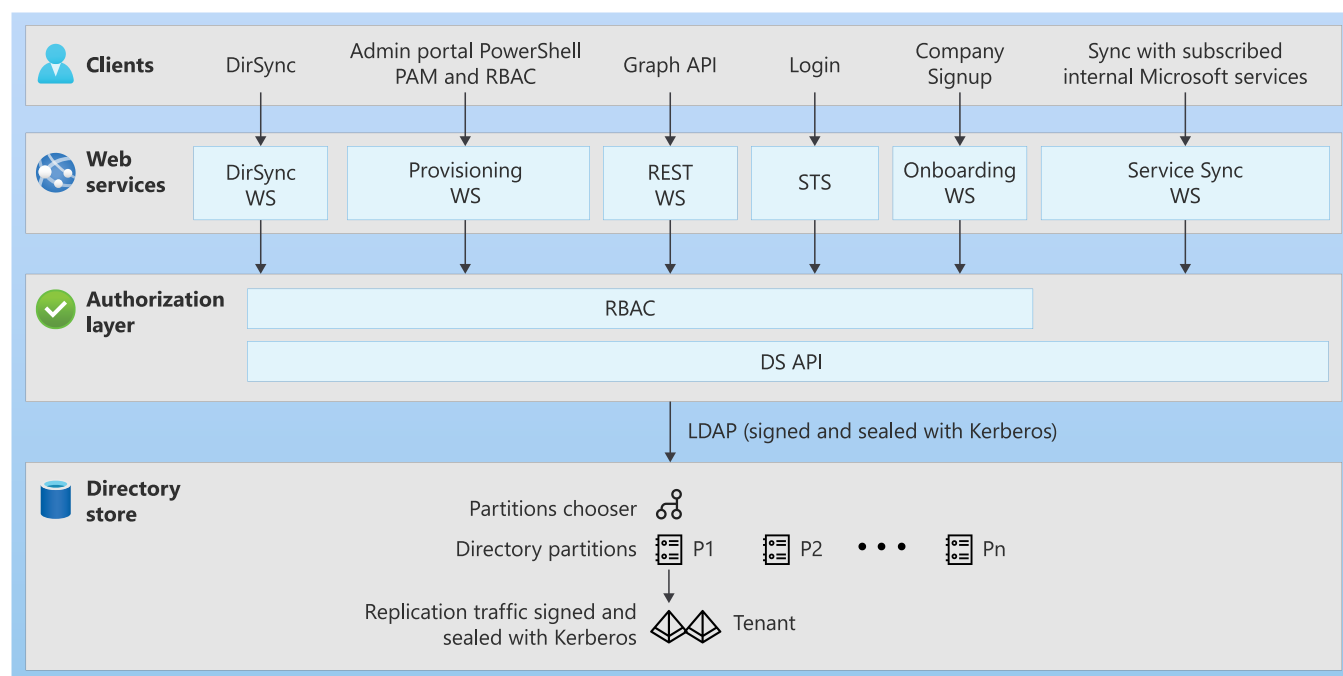
In the cloud-enabled workplace, a tenant can be defined as a client or organization that owns and manages a specific instance of that cloud service. With the identity platform provided by Microsoft Azure, a tenant is simply a dedicated instance of Microsoft Entra ID that your organization receives and owns when it signs up for a Microsoft cloud service.

Each Microsoft Entra directory is distinct and separate from other Microsoft Entra directories. Just like a corporate office building is a secure asset specific to only your organization, a Microsoft Entra directory was also designed to be a secure asset for use by only your organization. The Microsoft Entra architecture isolates customer data and identity information from co-mingling. This means that users and administrators of one Microsoft Entra directory can't accidentally or maliciously access data in another directory.

Azure Tenancy

Azure tenancy (Azure Subscription) refers to a "customer/billing" relationship and a unique [tenant](#) in [Microsoft Entra ID](#). Tenant level isolation in Microsoft Azure is achieved using Microsoft Entra ID and [Azure role-based access control](#) offered by it. Each Azure subscription is associated with one Microsoft Entra directory.

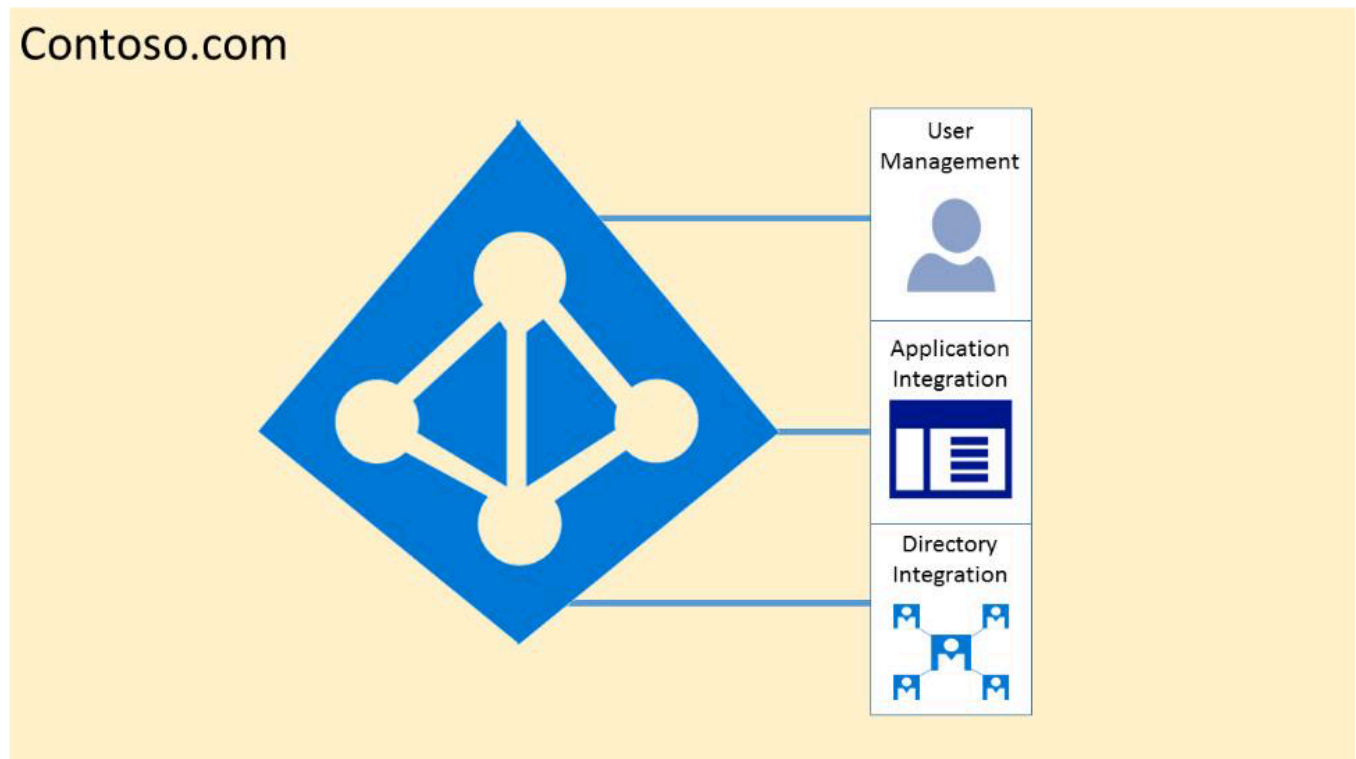
Users, groups, and applications from that directory can manage resources in the Azure subscription. You can assign these access rights using the Azure portal, Azure command-line tools, and Azure Management APIs. A Microsoft Entra tenant is logically isolated using security boundaries so that no customer can access or compromise co-tenants, either maliciously or accidentally. Microsoft Entra ID runs on “bare metal” servers isolated on a segregated network segment, where host-level packet filtering and Windows Firewall block unwanted connections and traffic.



- Access to data in Microsoft Entra ID requires user authentication via a security token service (STS). Information on the user's existence, enabled state, and role is used by the authorization system to determine whether the requested access to the target tenant is authorized for this user in this session.
- Tenants are discrete containers and there's no relationship between these.
- No access across tenants unless tenant admin grants it through federation or provisioning user accounts from other tenants.
- Physical access to servers that comprise the Microsoft Entra service, and direct access to Microsoft Entra ID's back-end systems, is restricted.
- Microsoft Entra users have no access to physical assets or locations, and therefore it isn't possible for them to bypass the logical Azure RBAC policy checks stated following.

For diagnostics and maintenance needs, an operational model that employs a just-in-time privilege elevation system is required and used. Microsoft Entra Privileged Identity Management (PIM) introduces the concept of an eligible admin. [Eligible admins](#) should be users that need privileged access now and then, but not every day. The role is inactive until the

user needs access, then they complete an activation process and become an active admin for a predetermined amount of time.



Microsoft Entra ID hosts each tenant in its own protected container, with policies and permissions to and within the container solely owned and managed by the tenant.

The concept of tenant containers is deeply ingrained in the directory service at all layers, from portals all the way to persistent storage.

Even when metadata from multiple Microsoft Entra tenants is stored on the same physical disk, there's no relationship between the containers other than what is defined by the directory service, which in turn is dictated by the tenant administrator.

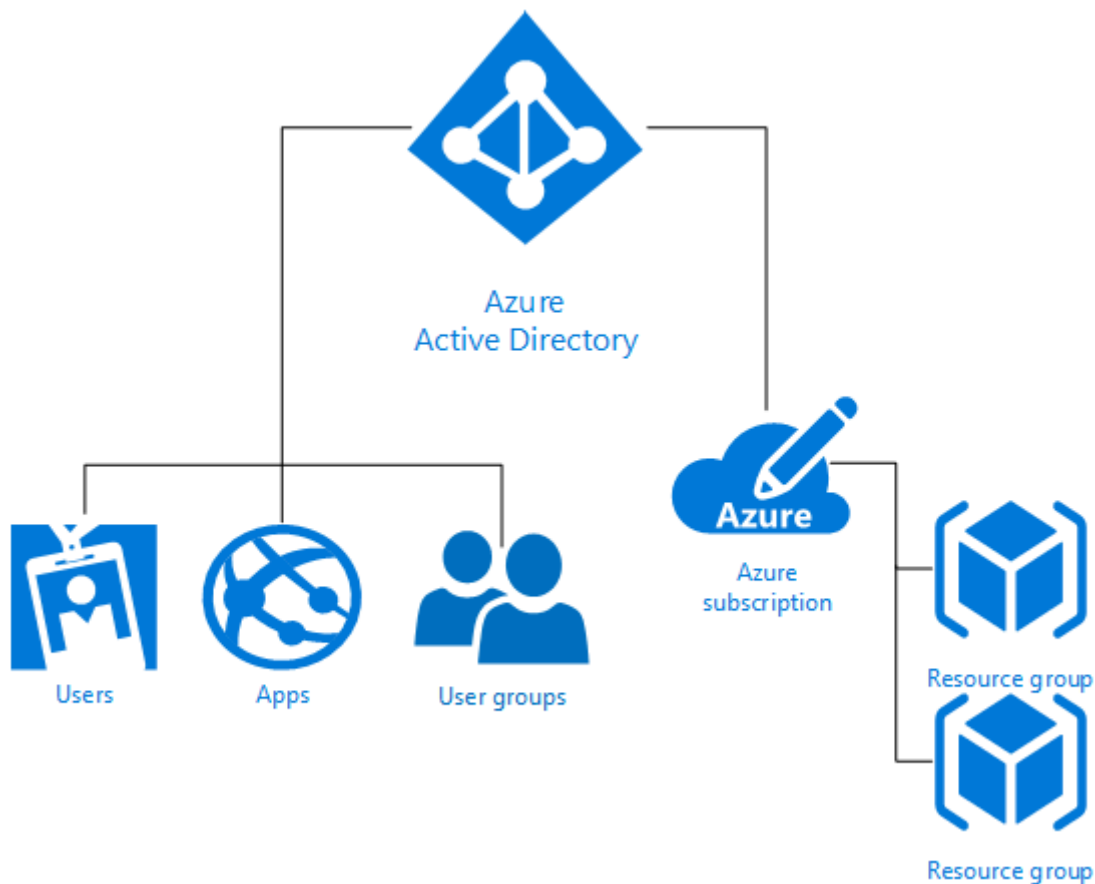
Azure role-based access control (Azure RBAC)

[Azure role-based access control \(Azure RBAC\)](#) helps you to share various components available within an Azure subscription by providing fine-grained access management for Azure. Azure RBAC enables you to segregate duties within your organization and grant access based on what users need to perform their jobs. Instead of giving everybody unrestricted permissions in Azure subscription or resources, you can allow only certain actions.

Azure RBAC has three basic roles that apply to all resource types:

- **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.



The rest of the Azure roles in Azure allow management of specific Azure resources. For example, the Virtual Machine Contributor role allows the user to create and manage virtual machines. It doesn't give them access to the Azure Virtual Network or the subnet that the virtual machine connects to.

[Azure built-in roles](#) list the roles available in Azure. It specifies the operations and scope that each built-in role grants to users. If you're looking to define your own roles for even more control, see how to build [Custom roles in Azure RBAC](#).

Some other capabilities for Microsoft Entra ID include:

- Microsoft Entra ID enables SSO to SaaS applications, regardless of where they're hosted. Some applications are federated with Microsoft Entra ID, and others use password SSO. Federated applications can also support user provisioning and [password vaulting](#).
- Access to data in [Azure Storage](#) [↗](#) is controlled via authentication. Each storage account has a primary key ([storage account key](#), or SAK) and a secondary secret key (the shared access signature, or SAS).
- Microsoft Entra ID provides Identity as a Service through federation by using [Active Directory Federation Services](#), synchronization, and replication with on-premises directories.

- [Microsoft Entra multifactor authentication](#) requires users to verify sign-ins by using a mobile app, phone call, or text message. It can be used with Microsoft Entra ID to help secure on-premises resources with the Multi-Factor Authentication Server, and also with custom applications and directories using the SDK.
- [Microsoft Entra Domain Services](#) [↗] lets you join Azure virtual machines to an Active Directory domain without deploying domain controllers. You can sign in to these virtual machines with your corporate Active Directory credentials and administer domain-joined virtual machines by using Group Policy to enforce security baselines on all your Azure virtual machines.
- [Microsoft Entra External ID](#) provides a highly available global-identity management service for consumer-facing applications that scales to hundreds of millions of identities. It can be integrated across mobile and web platforms. Your consumers can sign in to all your applications through customizable experiences by using their existing social accounts or by creating credentials.

Isolation from Microsoft Administrators & Data Deletion

Microsoft takes strong measures to protect your data from inappropriate access or use by unauthorized persons. These operational processes and controls are backed by the [Online Services Terms](#) [↗], which offer contractual commitments that govern access to your data.

- Microsoft engineers don't have default access to your data in the cloud. Instead, they're granted access, under management oversight, only when necessary. That access is carefully controlled and logged, and revoked when it's no longer needed.
- Microsoft may hire other companies to provide limited services on its behalf. Subcontractors may access customer data only to deliver the services for which, we have hired them to provide, and they're prohibited from using it for any other purpose. Further, they're contractually bound to maintain the confidentiality of our customers' information.

Business services with audited certifications such as ISO/IEC 27001 are regularly verified by Microsoft and accredited audit firms, which perform sample audits to attest that access, only for legitimate business purposes. You can always access your own customer data at any time and for any reason.

If you delete any data, Microsoft Azure deletes the data, including any cached or backup copies. For in-scope services, that deletion will occur within 90 days after the end of the retention period. (In-scope services are defined in the Data Processing Terms section of our [Online Services Terms](#) [↗].)

If a disk drive used for storage suffers a hardware failure, it's securely [erased or destroyed](#) before Microsoft returns it to the manufacturer for replacement or repair. The data on the drive is overwritten to ensure that the data can't be recovered by any means.

Compute Isolation

Microsoft Azure provides various cloud-based computing services that include a wide selection of compute instances & services that can scale up and down automatically to meet the needs of your application or enterprise. These compute instance and service offer isolation at multiple levels to secure data without sacrificing the flexibility in configuration that customers demand.

Isolated Virtual Machine Sizes

Azure Compute offers virtual machine sizes that are Isolated to a specific hardware type and dedicated to a single customer. The Isolated sizes live and operate on specific hardware generation and will be deprecated when the hardware generation is retired or new hardware generation is available.

Isolated virtual machine sizes are best suited for workloads that require a high degree of isolation from other customers' workloads. This is sometimes required to meet compliance and regulatory requirements. Utilizing an isolated size guarantees that your virtual machine is the only one running on that specific server instance.

Additionally, as the Isolated size VMs are large, customers may choose to subdivide the resources of these VMs by using [Azure support for nested virtual machines](#).

The current Isolated virtual machine offerings include:

- Standard_E80ids_v4
- Standard_E80is_v4
- Standard_E104i_v5
- Standard_E104is_v5
- Standard_E104id_v5
- Standard_E104ids_v5
- Standard_M192is_v2
- Standard_M192ims_v2
- Standard_M192ids_v2
- Standard_M192idms_v2
- Standard_F72s_v2
- Standard_M832ids_16_v3
- Standard_M832is_16_v3
- Standard_M896ixds_24_v3

- Standard_M896ixds_32_v3
- Standard_M1792ixds_32_v3

Note

Isolated VM Sizes have a limited lifespan due to hardware depreciation.

Deprecation of Isolated VM Sizes

Isolated VM sizes have a hardware limited lifespan. Azure issues reminders 12 months in advance of the official deprecation date of the sizes and provides an updated isolated offering for your consideration. The following sizes have retirement announced.

[Expand table](#)

Size	Isolation Retirement Date
Standard_DS15_v2	May 15, 2021
Standard_D15_v2	May 15, 2021
Standard_G5	February 15, 2022
Standard_GS5	February 15, 2022
Standard_E64i_v3	February 15, 2022
Standard_E64is_v3	February 15, 2022
Standard_M192is_v2	March 31, 2027
Standard_M192ims_v2	March 31, 2027
Standard_M192ids_v2	March 31, 2027
Standard_M192idms_v2	March 31, 2027

FAQ

Q: Is the size going to get retired or only its "isolation" feature?

A: Any size that is published as isolated but have no "i" in the name, the isolation feature of the VM sizes is being retired unless communicated differently. Sizes with "i" in the name will be

deprecated.

Q: Is there a downtime when my vm lands on a nonisolated hardware?

A: For VM sizes, where only isolation is deprecating but not the size, no action is needed and there will be no downtime. On contrary if isolation is required, announcement includes the recommended replacement size. Selecting the replacement size requires customers to resize their VMs.

Q: Is there any cost delta for moving to a nonisolated virtual machine?

A: No

Q: When are the other isolated sizes going to retire?

A: We provide reminders 12 months in advance of the official deprecation of the isolated size. Our latest announcement includes isolation feature retirement of Standard_G5, Standard_GS5, Standard_E64i_v3 and Standard_E64i_v3.

Q: I'm an Azure Service Fabric Customer relying on the Silver or Gold Durability Tiers. Does this change impact me?

A: No. The guarantees provided by Service Fabric's [Durability Tiers](#) will continue to function even after this change. If you require physical hardware isolation for other reasons, you may still need to take one of the actions described above.

Q: What are the milestones for D15_v2 or DS15_v2 isolation retirement?

A:

 Expand table

Date	Action
May 15, 2020 ¹	D/DS15_v2 isolation retirement announcement
May 15, 2021	D/DS15_v2 isolation guarantee removed

¹ Existing customer using these sizes will receive an announcement email with detailed instructions on the next steps.

Q: What are the milestones for G5, Gs5, E64i_v3 and E64is_v3 isolation retirement?

A:

 Expand table

Date	Action
Feb 15, 2021 ¹	G5/GS5/E64i_v3/E64is_v3 isolation retirement announcement
Feb 28, 2022	G5/GS5/E64i_v3/E64is_v3 isolation guarantee removed

¹ Existing customer using these sizes will receive an announcement email with detailed instructions on the next steps.

Next steps

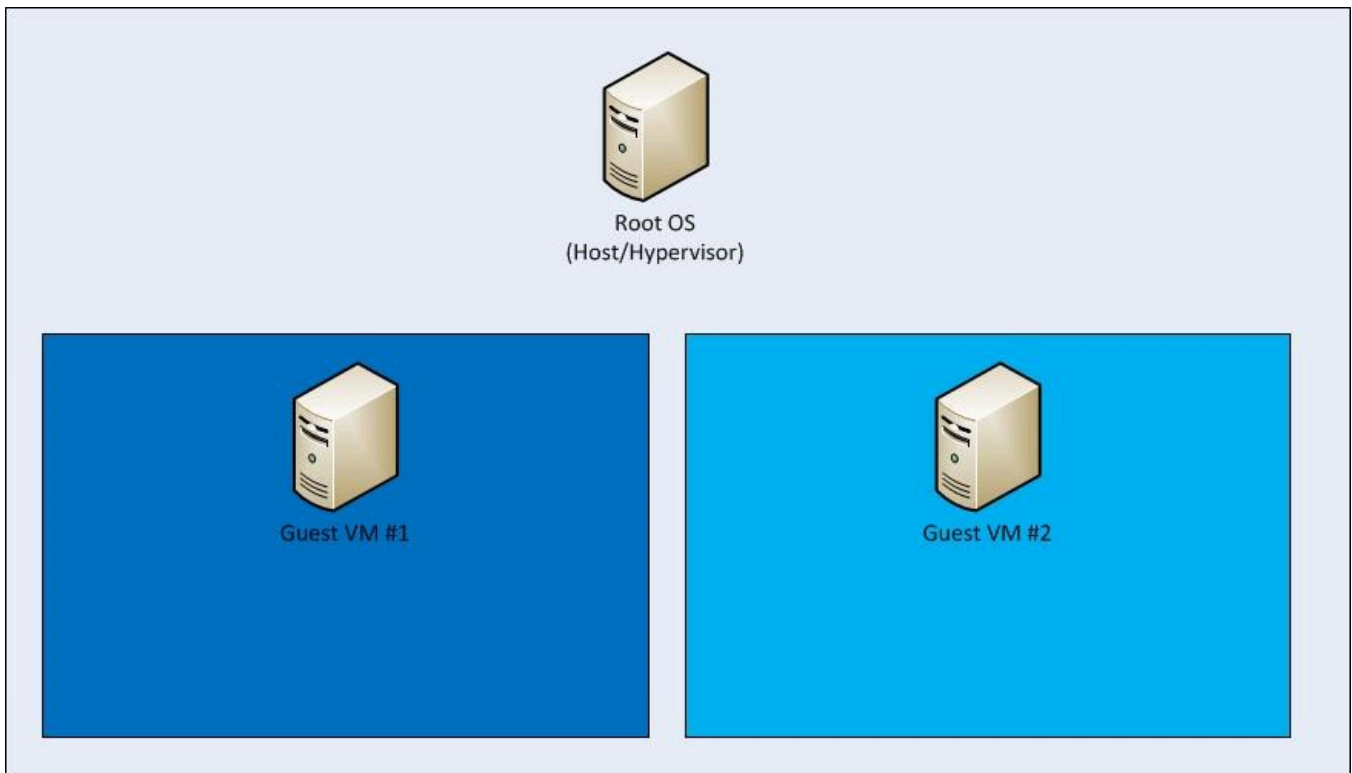
Customers can also choose to further subdivide the resources of these Isolated virtual machines by using [Azure support for nested virtual machines](#) .

Dedicated hosts

In addition to the isolated hosts described in the preceding section, Azure also offers dedicated hosts. Dedicated hosts in Azure is a service that provides physical servers that can host one or more virtual machines, and which are dedicated to a single Azure subscription. Dedicated hosts provide hardware isolation at the physical server level. No other VMs will be placed on your hosts. Dedicated hosts are deployed in the same datacenters and share the same network and underlying storage infrastructure as other, non-isolated hosts. For more information, see the detailed overview of [Azure dedicated hosts](#).

Hyper-V & Root OS Isolation Between Root VM & Guest VMs

Azure's compute platform is based on machine virtualization—meaning that all customer code executes in a Hyper-V virtual machine. On each Azure node (or network endpoint), there's a Hypervisor that runs directly over the hardware and divides a node into a variable number of Guest Virtual Machines (VMs).



Each node also has one special Root VM, which runs the Host OS. A critical boundary is the isolation of the root VM from the guest VMs and the guest VMs from one another, managed by the hypervisor and the root OS. The hypervisor/root OS pairing leverages Microsoft's decades of operating system security experience, and more recent learning from Microsoft's Hyper-V, to provide strong isolation of guest VMs.

The Azure platform uses a virtualized environment. User instances operate as standalone virtual machines that don't have access to a physical host server.

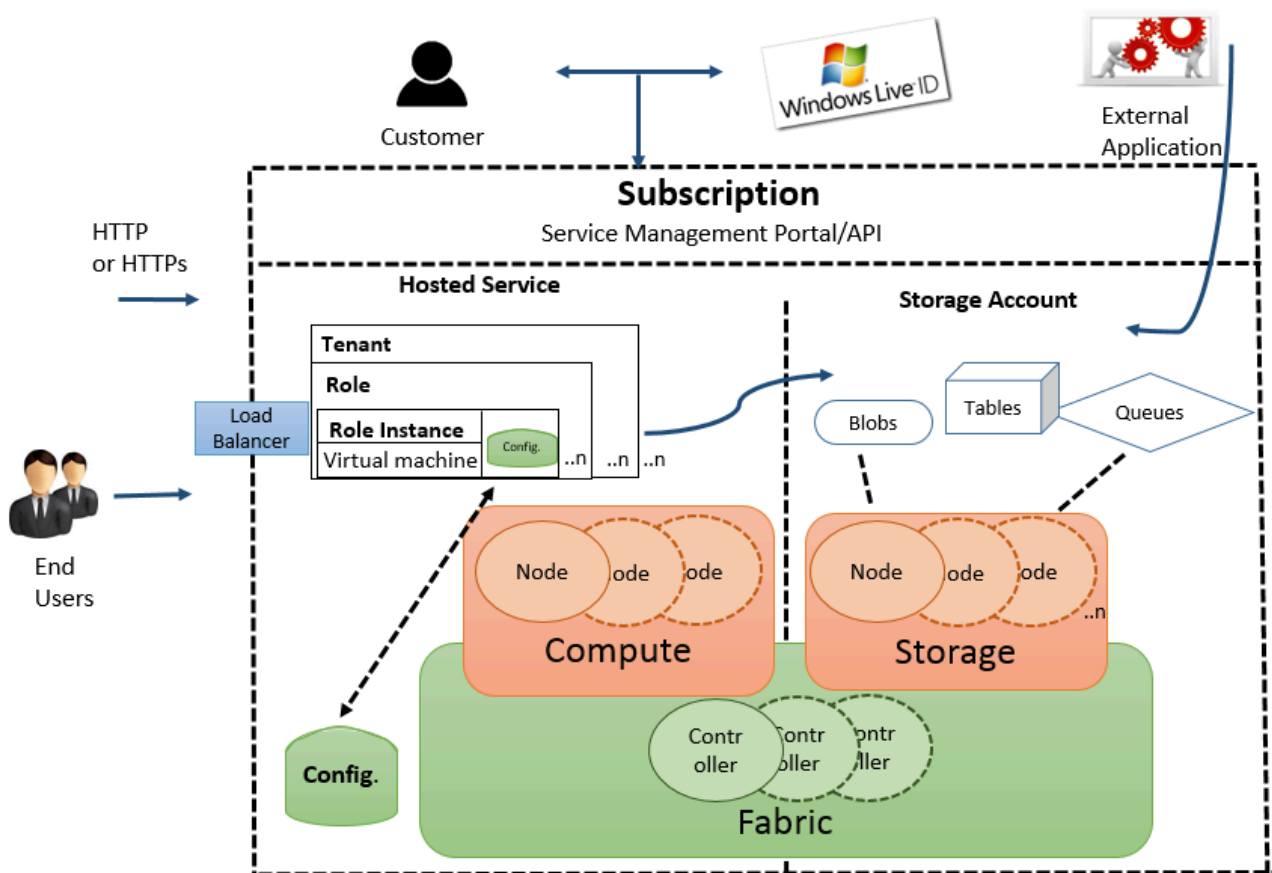
The Azure hypervisor acts like a micro-kernel and passes all hardware access requests from guest virtual machines to the host for processing by using a shared-memory interface called VM Bus. This prevents users from obtaining raw read/write/execute access to the system and mitigates the risk of sharing system resources.

Advanced VM placement algorithm & protection from side channel attacks

Any cross-VM attack involves two steps: placing an adversary-controlled VM on the same host as one of the victim VMs, and then breaching the isolation boundary to either steal sensitive victim information or affect its performance for greed or vandalism. Microsoft Azure provides protection at both steps by using an advanced VM placement algorithm and protection from all known side channel attacks including noisy neighbor VMs.

The Azure Fabric Controller

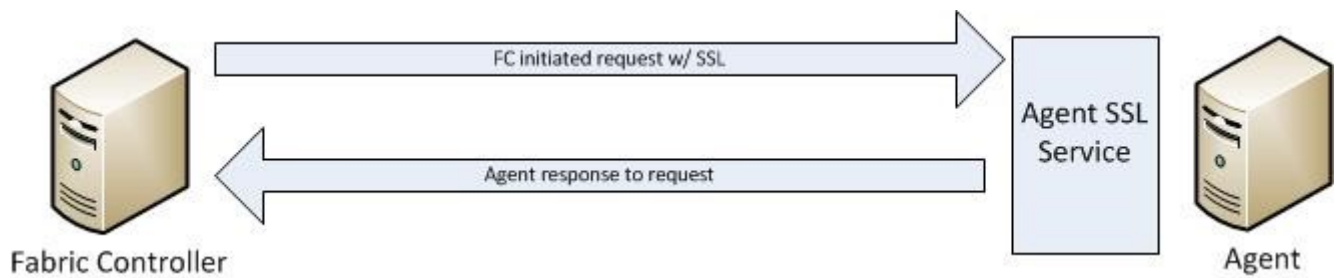
The Azure Fabric Controller is responsible for allocating infrastructure resources to tenant workloads, and it manages unidirectional communications from the host to virtual machines. The VM placing algorithm of the Azure fabric controller is highly sophisticated and nearly impossible to predict as physical host level.



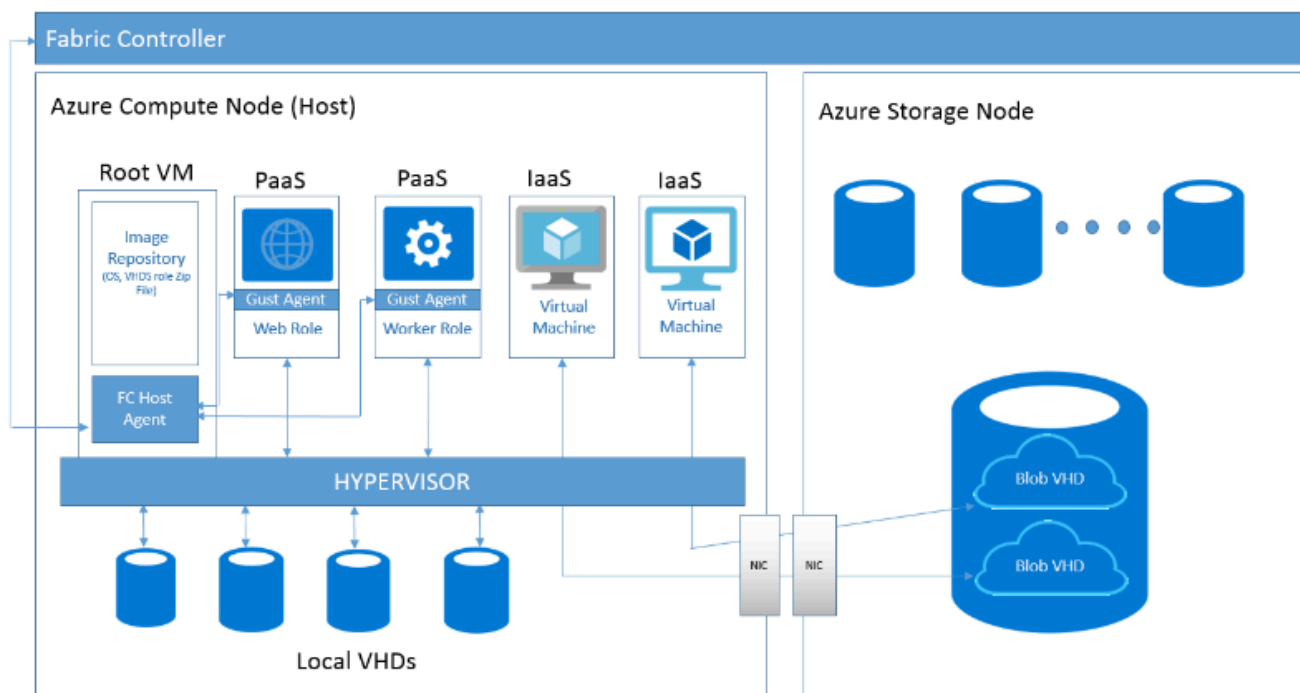
The Azure hypervisor enforces memory and process separation between virtual machines, and it securely routes network traffic to guest OS tenants. This eliminates possibility of and side channel attack at VM level.

In Azure, the root VM is special: it runs a hardened operating system called the root OS that hosts a fabric agent (FA). FAs are used in turn to manage guest agents (GA) within guest operating systems on customer VMs. FAs also manage storage nodes.

The collection of Azure hypervisor, root OS/FA, and customer VMs/GAs comprises a compute node. FAs are managed by a fabric controller (FC), which exists outside of compute and storage nodes (compute and storage clusters are managed by separate FCs). If a customer updates their application's configuration file while it's running, the FC communicates with the FA, which then contacts GAs, which notify the application of the configuration change. In the event of a hardware failure, the FC will automatically find available hardware and restart the VM there.



Communication from a Fabric Controller to an agent is unidirectional. The agent implements an SSL-protected service that only responds to requests from the controller. It cannot initiate connections to the controller or other privileged internal nodes. The FC treats all responses as if they were untrusted.



Isolation extends from the Root VM from Guest VMs, and the Guest VMs from one another. Compute nodes are also isolated from storage nodes for increased protection.

The hypervisor and the host OS provide network packet - filters to help assure that untrusted virtual machines cannot generate spoofed traffic or receive traffic not addressed to them, direct traffic to protected infrastructure endpoints, or send/receive inappropriate broadcast traffic.

Additional Rules Configured by Fabric Controller Agent to Isolate VM

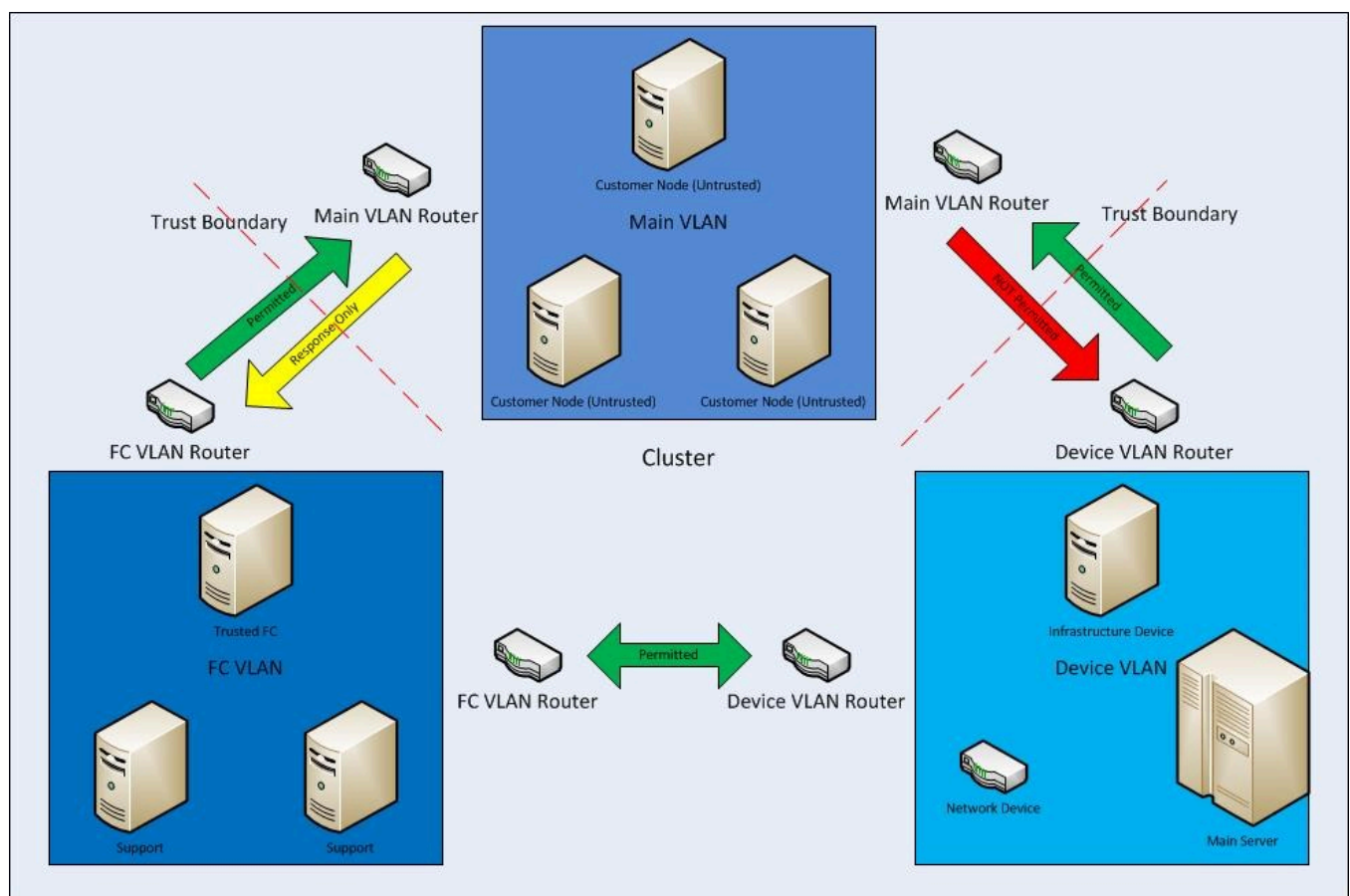
By default, all traffic is blocked when a virtual machine is created, and then the fabric controller agent configures the packet filter to add rules and exceptions to allow authorized traffic.

There are two categories of rules that are programmed:

- **Machine configuration or infrastructure rules:** By default, all communication is blocked. There are exceptions to allow a virtual machine to send and receive DHCP and DNS traffic. Virtual machines can also send traffic to the "public" internet and send traffic to other virtual machines within the same Azure Virtual Network and the OS activation server. The virtual machines' list of allowed outgoing destinations doesn't include Azure router subnets, Azure management, and other Microsoft properties.
- **Role configuration file:** This defines the inbound Access Control Lists (ACLs) based on the tenant's service model.

VLAN Isolation

There are three VLANs in each cluster:



- The main VLAN – interconnects untrusted customer nodes
- The FC VLAN – contains trusted FCs and supporting systems
- The device VLAN – contains trusted network and other infrastructure devices

Communication is permitted from the FC VLAN to the main VLAN, but cannot be initiated from the main VLAN to the FC VLAN. Communication is also blocked from the main VLAN to the device VLAN. This assures that even if a node running customer code is compromised, it cannot attack nodes on either the FC or device VLANs.

Storage Isolation

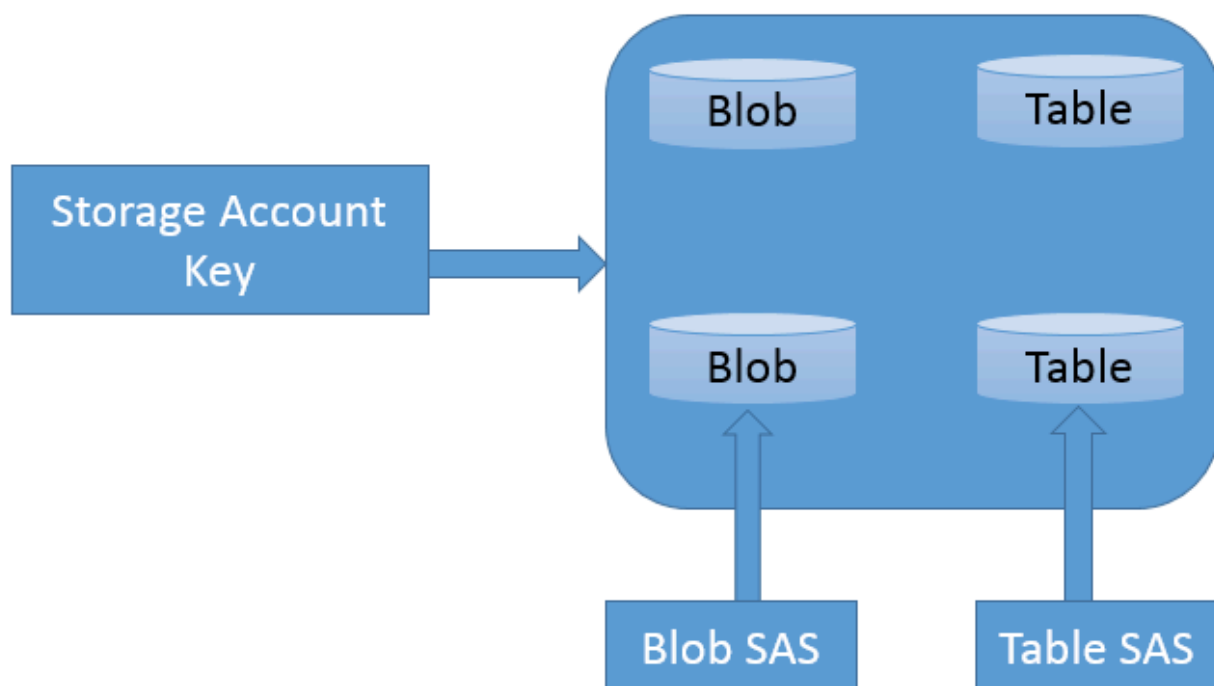
Logical Isolation Between Compute and Storage

As part of its fundamental design, Microsoft Azure separates VM-based computation from storage. This separation enables computation and storage to scale independently, making it easier to provide multi-tenancy and isolation.

Therefore, Azure Storage runs on separate hardware with no network connectivity to Azure Compute except logically. This means that when a virtual disk is created, disk space isn't allocated for its entire capacity. Instead, a table is created that maps addresses on the virtual disk to areas on the physical disk and that table is initially empty. **The first time a customer writes data on the virtual disk, space on the physical disk is allocated, and a pointer to it's placed in the table.**

Isolation Using Storage Access control

Access Control in Azure Storage has a simple access control model. Each Azure subscription can create one or more Storage Accounts. Each Storage Account has a single secret key that's used to control access to all data in that Storage Account.



Access to Azure Storage data (including Tables) can be controlled through a [SAS \(Shared Access Signature\)](#) token, which grants scoped access. The SAS is created through a query template (URL), signed with the [SAK \(Storage Account Key\)](#). That [signed URL](#) can be given to another process (that is, delegated), which can then fill in the details of the query and make the request of the storage service. A SAS enables you to grant time-based access to clients without revealing the storage account's secret key.

The SAS means that we can grant a client limited permissions, to objects in our storage account for a specified period of time and with a specified set of permissions. We can grant these limited permissions without having to share your account access keys.

IP Level Storage Isolation

You can establish firewalls and define an IP address range for your trusted clients. With an IP address range, only clients that have an IP address within the defined range can connect to [Azure Storage](#).

IP storage data can be protected from unauthorized users via a networking mechanism that's used to allocate a dedicated or dedicated tunnel of traffic to IP storage.

Encryption

Azure offers the following types of Encryption to protect data:

- Encryption in transit
- Encryption at rest

Encryption in Transit

Encryption in transit is a mechanism of protecting data when it's transmitted across networks. With Azure Storage, you can secure data using:

- [Transport-level encryption](#), such as HTTPS when you transfer data into or out of Azure Storage.
- [Wire encryption](#), such as SMB 3.0 encryption for Azure File shares.
- [Client-side encryption](#), to encrypt the data before it's transferred into storage and to decrypt the data after it's transferred out of storage.

Encryption at Rest

For many organizations, data encryption at rest is a mandatory step towards data privacy, compliance, and data sovereignty. There are three Azure features that provide encryption of

data that's "at rest":

- [Storage Service Encryption](#) allows you to request that the storage service automatically encrypt data when writing it to Azure Storage.
- [Client-side Encryption](#) also provides the feature of encryption at rest.
- [Azure Disk Encryption for Linux VMs](#) and [Azure Disk Encryption for Windows VMs](#).

For more information, see [Overview of managed disk encryption options](#).

Azure Disk Encryption

[Azure Disk Encryption for Linux VMs](#) and [Azure Disk Encryption for Windows VMs](#) help you address organizational security and compliance requirements by encrypting your VM disks (including boot and data disks) with keys and policies you control in [Azure Key Vault](#) [↗].

The Disk Encryption solution for Windows is based on [Microsoft BitLocker Drive Encryption](#), and the Linux solution is based on [dm-crypt](#) [↗].

The solution supports the following scenarios for IaaS VMs when they're enabled in Microsoft Azure:

- Integration with Azure Key Vault
- Standard tier VMs: A, D, DS, G, GS, and so forth, series IaaS VMs
- Enabling encryption on Windows and Linux IaaS VMs
- Disabling encryption on OS and data drives for Windows IaaS VMs
- Disabling encryption on data drives for Linux IaaS VMs
- Enabling encryption on IaaS VMs that are running Windows client OS
- Enabling encryption on volumes with mount paths
- Enabling encryption on Linux VMs that are configured with disk striping (RAID) by using [mdadm](#) [↗]
- Enabling encryption on Linux VMs by using [LVM\(Logical Volume Manager\)](#) for data disks
- Enabling encryption on Windows VMs that are configured by using storage spaces
- All Azure public regions are supported

The solution doesn't support the following scenarios, features, and technology in the release:

- Basic tier IaaS VMs
- Disabling encryption on an OS drive for Linux IaaS VMs
- IaaS VMs that are created by using the classic VM creation method
- Integration with your on-premises Key Management Service
- Azure Files (shared file system), Network File System (NFS), dynamic volumes, and Windows VMs that are configured with software-based RAID systems

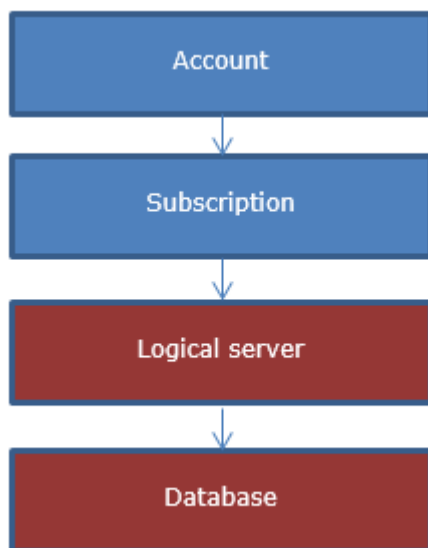
SQL Database Isolation

SQL Database is a relational database service in the Microsoft cloud based on the market-leading Microsoft SQL Server engine and capable of handling mission-critical workloads. SQL Database offers predictable data isolation at account level, geography / region based and based on networking— all with near-zero administration.

SQL Database Application Model

[Microsoft SQL Database](#) is a cloud-based relational database service built on SQL Server technologies. It provides a highly available, scalable, multi-tenant database service hosted by Microsoft in cloud.

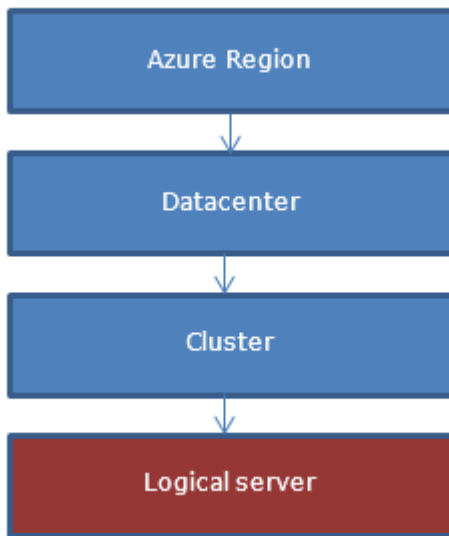
From an application perspective, SQL Database provides the following hierarchy: Each level has one-to-many containment of levels below.



The account and subscription are Microsoft Azure platform concepts to associate billing and management.

Logical SQL servers and databases are SQL Database-specific concepts and are managed by using SQL Database, provided OData and TSQL interfaces or via the Azure portal.

Servers in SQL Database aren't physical or VM instances, instead they 're collections of databases, sharing management and security policies, which are stored in so called "logical master" database.



Logical master databases include:

- SQL logins used to connect to the server
- Firewall rules

Billing and usage-related information for databases from the same server aren't guaranteed to be on the same physical instance in the cluster, instead applications must provide the target database name when connecting.

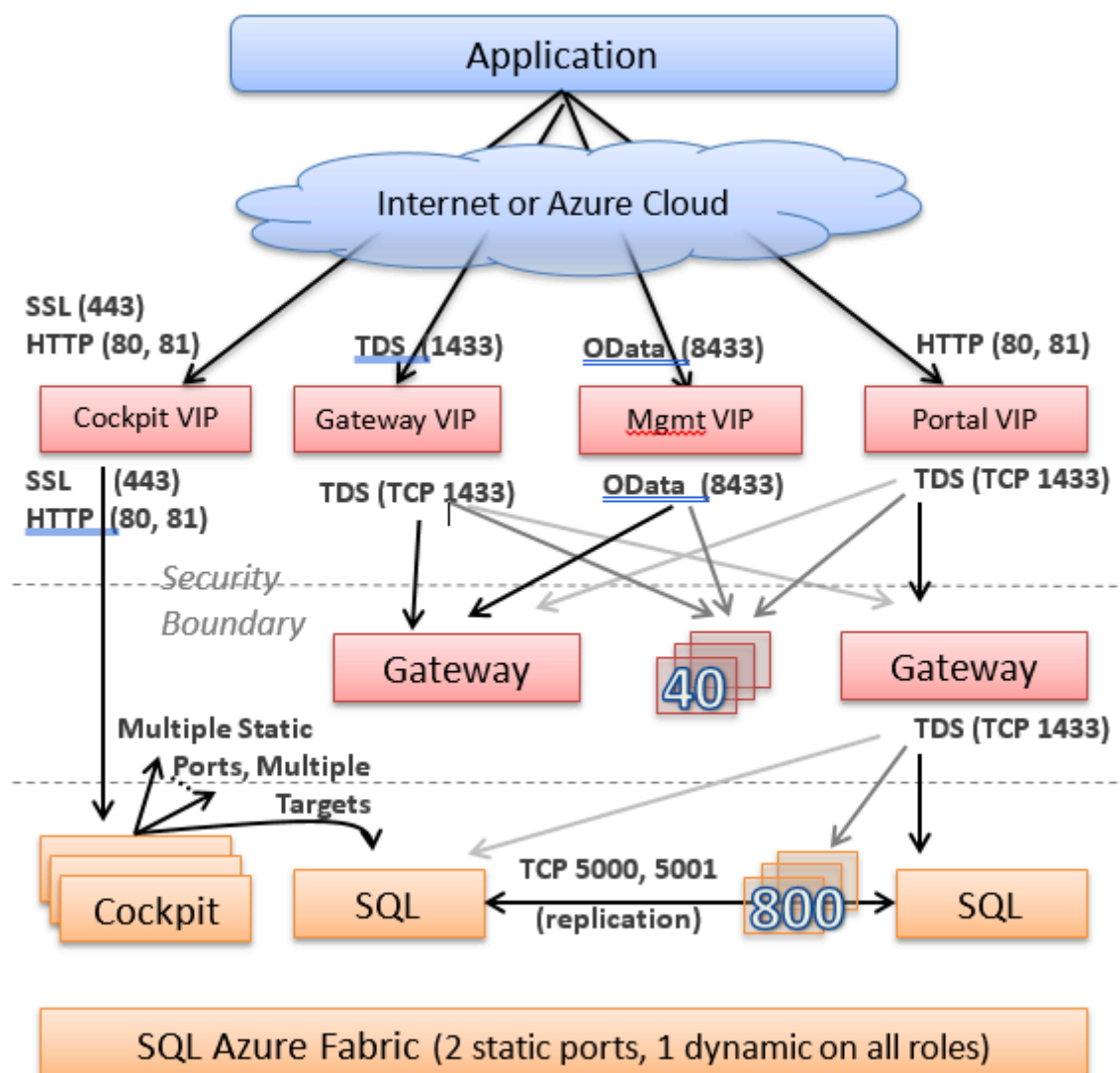
From a customer perspective, a server is created in a geo-graphical region while the actual creation of the server happens in one of the clusters in the region.

Isolation through Network Topology

When a server is created and its DNS name is registered, the DNS name points to the so called "Gateway VIP" address in the specific data center where the server was placed.

Behind the VIP (virtual IP address), we have a collection of stateless gateway services. In general, gateways get involved when there's coordination needed between multiple data sources (master database, user database, etc.). Gateway services implement the following:

- **TDS connection proxying.** This includes locating user database in the backend cluster, implementing the login sequence and then forwarding the TDS packets to the backend and back.
- **Database management.** This includes implementing a collection of workflows to do CREATE/ALTER/DROP database operations. The database operations can be invoked by either sniffing TDS packets or explicit OData APIs.
- CREATE/ALTER/DROP login/user operations
- Server management operations via OData API



The tier behind the gateways is called “back-end”. This is where all the data is stored in a highly available fashion. Each piece of data is said to belong to a “partition” or “failover unit”, each of them having at least three replicas. Replicas are stored and replicated by SQL Server engine and managed by a failover system often referred to as “fabric”.

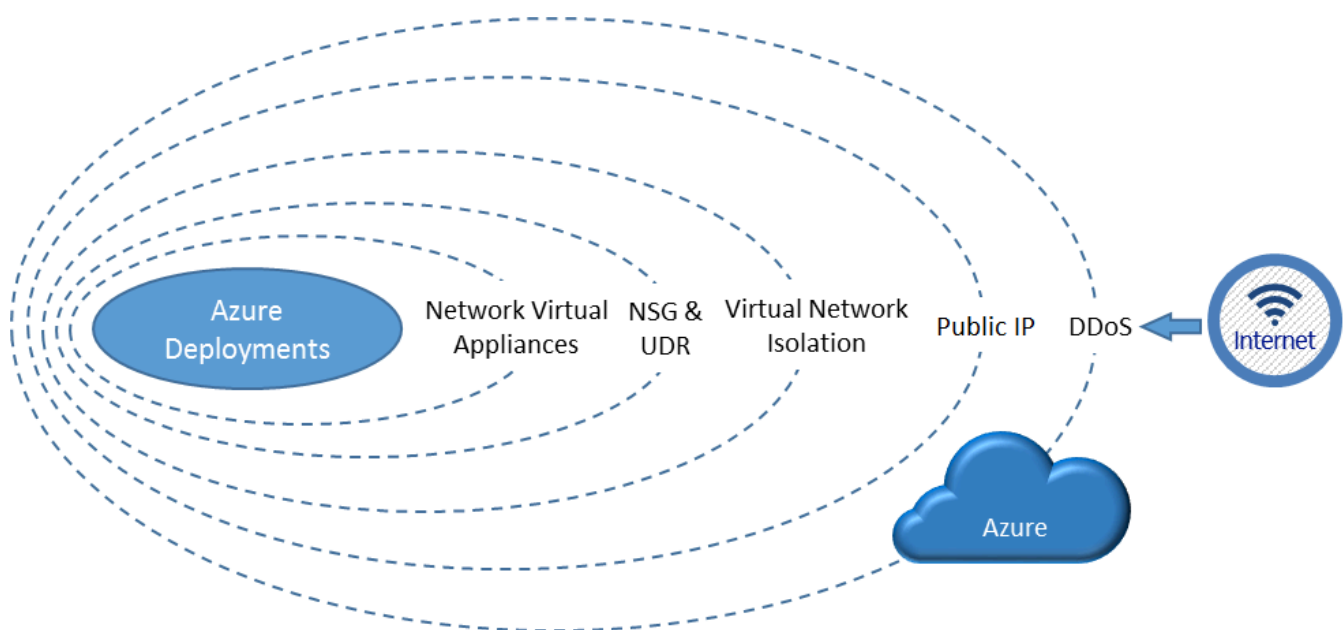
Generally, the back-end system doesn't communicate outbound to other systems as a security precaution. This is reserved to the systems in the front-end (gateway) tier. The gateway tier machines have limited privileges on the back-end machines to minimize the attack surface as a defense-in-depth mechanism.

Isolation by Machine Function and Access

SQL Database is composed of services running on different machine functions. SQL Database is divided into “backend” Cloud Database and “front-end” (Gateway/Management) environments, with the general principle of traffic only going into back-end and not out. The front-end environment can communicate to the outside world of other services and in general, has only limited permissions in the back-end (enough to call the entry points it needs to invoke).

Networking Isolation

Azure deployment has multiple layers of network isolation. The following diagram shows various layers of network isolation Azure provides to customers. These layers are both native in the Azure platform itself and customer-defined features. Inbound from the Internet, Azure DDoS provides isolation against large-scale attacks against Azure. The next layer of isolation is customer-defined public IP addresses (endpoints), which are used to determine which traffic can pass through the cloud service to the virtual network. Native Azure virtual network isolation ensures complete isolation from all other networks, and that traffic only flows through user configured paths and methods. These paths and methods are the next layer, where NSGs, UDR, and network virtual appliances can be used to create isolation boundaries to protect the application deployments in the protected network.



Traffic isolation: A [virtual network](#) is the traffic isolation boundary on the Azure platform. Virtual machines (VMs) in one virtual network cannot communicate directly to VMs in a different virtual network, even if both virtual networks are created by the same customer. Isolation is a critical property that ensures customer VMs and communication remains private within a virtual network.

[Subnet](#) offers an additional layer of isolation within a virtual network based on IP range. IP addresses in the virtual network, you can divide a virtual network into multiple subnets for organization and security. VMs and PaaS role instances deployed to subnets (same or different) within a VNet can communicate with each other without any extra configuration. You can also configure [network security groups \(NSGs\)](#) to allow or deny network traffic to a VM instance based on security rules. NSGs can be associated with either subnets or individual network interfaces attached to VMs. When an NSG is associated with a subnet, the security rules apply to all the VM instances in that subnet.

Next Steps

- Learn about [network security groups](#). Network security groups filter network traffic between Azure resources in a virtual network, allowing you to restrict traffic to subnets or virtual machines based on source, destination, port, and protocol using security rules.
- Learn about [virtual machine isolation in Azure](#). Azure Compute offers virtual machine sizes that are isolated to a specific hardware type and dedicated to a single customer.

Last updated on 01/06/2026

Azure identity management security overview

Identity management is the process of authenticating and authorizing [security principals](#).

Microsoft Entra ID provides comprehensive identity and access management for applications and resources across your organization. This article covers core Azure identity management features that help protect access to resources.

Single sign-on

Single sign-on (SSO) enables users to access multiple applications and resources with a single user account and password. Users sign in once and can access all their applications without repeated authentication. Microsoft Entra ID supports SSO for thousands of SaaS applications and on-premises web applications.

Microsoft Entra ID extends on-premises Active Directory into the cloud, enabling users to sign in with their organizational account to domain-joined devices, company resources, and integrated applications. SSO reduces password fatigue and improves security by minimizing exposed credentials.

Learn more:

- [What is single sign-on in Microsoft Entra ID?](#)
- [Plan a single sign-on deployment](#)

Multifactor authentication

Microsoft Entra multifactor authentication (MFA) adds a critical second layer of security by requiring two or more verification methods. MFA helps protect against unauthorized access while maintaining a simple sign-in experience for users.

Verification methods include:

- Microsoft Authenticator app
- Windows Hello for Business
- FIDO2 security keys
- Certificate-based authentication
- OATH tokens (hardware and software)
- SMS and voice call

Microsoft Entra ID P1 and P2 licenses support Conditional Access policies that enforce MFA based on user, location, device, and application context.

Learn more:

- [How Microsoft Entra multifactor authentication works](#)
- [Plan a Microsoft Entra multifactor authentication deployment](#)

Azure role-based access control

Azure role-based access control (Azure RBAC) provides fine-grained access management for Azure resources. With Azure RBAC, you can grant users the minimum permissions needed to perform their jobs.

Azure RBAC includes built-in roles:

- **Owner:** Full access to all resources, including the right to delegate access
- **Contributor:** Create and manage all types of Azure resources, but can't grant access
- **Reader:** View existing Azure resources
- **User Access Administrator:** Manage user access to Azure resources

You can also create custom roles tailored to your specific needs.

Learn more:

- [What is Azure role-based access control \(Azure RBAC\)?](#)
- [Azure built-in roles](#)
- [Assign Azure roles using the Azure portal](#)

Application Proxy

Microsoft Entra application proxy enables secure remote access to on-premises web applications without requiring VPN connections. Application Proxy publishes applications like SharePoint sites, Outlook Web App, and IIS-based apps to external users while maintaining security through Microsoft Entra ID authentication and Conditional Access policies.

Application Proxy supports SSO and can integrate with existing on-premises authentication methods.

Learn more:

- [Microsoft Entra application proxy overview](#)
- [Publish on-premises apps with Microsoft Entra application proxy](#)

Privileged Identity Management

Microsoft Entra Privileged Identity Management (PIM) helps you manage, control, and monitor privileged access to important resources. PIM provides just-in-time (JIT) privileged access, reducing the risk of excessive or unnecessary permissions.

With PIM, you can:

- Provide time-bound access to Azure and Microsoft Entra roles
- Require approval to activate privileged roles
- Enforce multifactor authentication for role activation
- Require justification for role activation
- Receive notifications for privileged role activations
- Conduct access reviews to ensure users still need privileged roles
- Generate audit reports for compliance

Learn more:

- [What is Microsoft Entra Privileged Identity Management?](#)
- [Plan a Privileged Identity Management deployment](#)

Identity Protection

Microsoft Entra ID Protection detects potential vulnerabilities and risky activities affecting your organization's identities. It uses machine learning to identify anomalous sign-in behaviors and user activities.

Identity Protection provides:

- **Risk-based Conditional Access:** Policies that respond to detected risks in real-time
- **Risk detection:** Identification of suspicious activities, including anonymous IP address usage, atypical travel, and malware-linked IP addresses
- **Investigation tools:** Reports and dashboards for analyzing risks
- **Automated remediation:** Risk-based policies that can automatically require password changes or block access

Learn more:

- [What is Microsoft Entra ID Protection?](#)
- [Investigate risk with Identity Protection](#)

Microsoft Entra access reviews

Microsoft Entra access reviews enable efficient management of group memberships, access to enterprise applications, and privileged role assignments. Regular access reviews help ensure users have only the access they need.

Access reviews support:

- **Automated reviews:** Scheduled recurring reviews with customizable frequency
- **Delegated reviews:** Business owners and managers can review access for their teams
- **Self-attestation:** Users can confirm they still need access
- **Recommendations:** Machine learning suggests which users should lose access based on sign-in activity
- **Automated actions:** Remove access automatically when reviews complete

Learn more:

- [What are Microsoft Entra access reviews?](#)
- [Plan a Microsoft Entra access reviews deployment](#)

Hybrid identity management

For organizations with on-premises Active Directory, Microsoft provides hybrid identity solutions to synchronize identities between on-premises and cloud environments.

Microsoft Entra Connect (maintenance mode) synchronizes on-premises AD DS identities to Microsoft Entra ID. It runs on an on-premises server and provides:

- Directory synchronization for users, groups, and contacts
- Password hash synchronization or pass-through authentication
- Federation integration with AD FS
- Health monitoring

Microsoft Entra Cloud Sync is the modern, cloud-based synchronization solution that uses lightweight provisioning agents:

- Simplified deployment with lightweight agents
- Support for multi-forest disconnected environments
- High availability with multiple agents
- Cloud-based configuration and management

Microsoft recommends Cloud Sync for new hybrid identity deployments.

Learn more:

- [What is Microsoft Entra Cloud Sync?](#)

- [Choose the right sync client for Microsoft Entra ID](#)

Device registration

Microsoft Entra device registration enables device-based Conditional Access policies.

Registered devices receive an identity that authenticates the device during user sign-in. Device attributes can enforce Conditional Access policies for cloud and on-premises applications.

When combined with mobile device management (MDM) solutions like Microsoft Intune, device attributes are enriched with configuration and compliance information. This enables Conditional Access rules based on device security and compliance posture.

Learn more:

- [Plan your Microsoft Entra device deployment](#)
- [Microsoft Entra joined devices](#)
- [Microsoft Entra hybrid joined devices](#)

External identities

Microsoft Entra External ID provides identity management for customer-facing applications and B2B collaboration. External ID supports consumer sign-up and sign-in with social accounts (Facebook, Google, LinkedIn) or email-based credentials.

For B2B collaboration, External ID enables secure sharing of applications and resources with external partners while maintaining control over your corporate data. External users authenticate with their home organization or supported identity providers.

Learn more:

- [Microsoft Entra External ID overview](#)
- [B2B collaboration overview](#)

Next steps

- [Azure security best practices and patterns](#)
- [Network security overview](#)
- [Threat detection and protection](#)
- [Key management in Azure](#)

Azure Identity Management and access control security best practices

05/28/2025

In this article, we discuss a collection of Azure identity management and access control security best practices. These best practices are derived from our experience with [Microsoft Entra ID](#) and the experiences of customers like yourself.

For each best practice, we explain:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

This Azure identity management and access control security best practices article is based on a consensus opinion and Azure platform capabilities and feature sets, as they exist at the time this article was written.

The intention in writing this article is to provide a general roadmap to a more robust security posture after deployment guided by our "[5 steps to securing your identity infrastructure](#)" checklist, which walks you through some of our core features and services.

Opinions and technologies change over time and this article will be updated on a regular basis to reflect those changes.

Azure identity management and access control security best practices discussed in this article include:

- Treat identity as the primary security perimeter
- Centralize identity management
- Manage connected tenants
- Enable single sign-on
- Turn on Conditional Access
- Plan for routine security improvements
- Enable password management
- Enforce multifactor verification for users
- Use role-based access control
- Lower exposure of privileged accounts
- Control locations where resources are located
- Use Microsoft Entra ID for storage authentication

Treat identity as the primary security perimeter

Many consider identity to be the primary perimeter for security. This is a shift from the traditional focus on network security. Network perimeters keep getting more porous, and that perimeter defense can't be as effective as it was before the explosion of [BYOD](#) devices and cloud applications.

[Microsoft Entra ID](#) is the Azure solution for identity and access management. Microsoft Entra ID is a multitenant, cloud-based directory and identity management service from Microsoft. It combines core directory services, application access management, and identity protection into a single solution.

The following sections list best practices for identity and access security using Microsoft Entra ID.

Best practice: Center security controls and detections around user and service identities. **Detail:** Use Microsoft Entra ID to collocate controls and identities.

Centralize identity management

In a hybrid identity scenario we recommend that you integrate your on-premises and cloud directories. Integration enables your IT team to manage accounts from one location, regardless of where an account is created. Integration also helps your users be more productive by providing a common identity for accessing both cloud and on-premises resources.

Best practice: Establish a single Microsoft Entra instance. Consistency and a single authoritative source will increase clarity and reduce security risks from human errors and configuration complexity.

Detail: Designate a single Microsoft Entra directory as the authoritative source for corporate and organizational accounts.

Best practice: Integrate your on-premises directories with Microsoft Entra ID.

Detail: Use [Microsoft Entra Connect](#) to synchronize your on-premises directory with your cloud directory.

ⓘ Note

There are [factors that affect the performance of Microsoft Entra Connect](#). Ensure Microsoft Entra Connect has enough capacity to keep underperforming systems from impeding security and productivity. Large or complex organizations (organizations

provisioning more than 100,000 objects) should follow the [recommendations](#) to optimize their Microsoft Entra Connect implementation.

Best practice: Don't synchronize accounts to Microsoft Entra ID that have high privileges in your existing Active Directory instance.

Detail: Don't change the default [Microsoft Entra Connect configuration](#) that filters out these accounts. This configuration mitigates the risk of adversaries pivoting from cloud to on-premises assets (which could create a major incident).

Best practice: Turn on password hash synchronization.

Detail: Password hash synchronization is a feature used to synch user password hashes from an on-premises Active Directory instance to a cloud-based Microsoft Entra instance. This sync helps to protect against leaked credentials being replayed from previous attacks.

Even if you decide to use federation with Active Directory Federation Services (AD FS) or other identity providers, you can optionally set up password hash synchronization as a backup in case your on-premises servers fail or become temporarily unavailable. This sync enables users to sign in to the service by using the same password that they use to sign in to their on-premises Active Directory instance. It also allows Identity Protection to detect compromised credentials by comparing synchronized password hashes with passwords known to be compromised, if a user has used the same email address and password on other services that aren't connected to Microsoft Entra ID.

For more information, see [Implement password hash synchronization with Microsoft Entra Connect Sync](#).

Best practice: For new application development, use Microsoft Entra ID for authentication.

Detail: Use the correct capabilities to support authentication:

- Microsoft Entra ID for employees
- [Microsoft Entra B2B](#) for guest users and external partners
- [Microsoft Entra External ID](#) to control how customers sign up, sign in, and manage their profiles when they use your applications

Organizations that don't integrate their on-premises identity with their cloud identity can have more overhead in managing accounts. This overhead increases the likelihood of mistakes and security breaches.

Note

You need to choose which directories critical accounts will reside in and whether the admin workstation used is managed by new cloud services or existing processes. Using existing management and identity provisioning processes can decrease some risks but can

also create the risk of an attacker compromising an on-premises account and pivoting to the cloud. You might want to use a different strategy for different roles (for example, IT admins vs. business unit admins). You have two options. First option is to create Microsoft Entra accounts that aren't synchronized with your on-premises Active Directory instance. Join your admin workstation to Microsoft Entra ID, which you can manage and patch by using Microsoft Intune. Second option is to use existing admin accounts by synchronizing to your on-premises Active Directory instance. Use existing workstations in your Active Directory domain for management and security.

Manage connected tenants

Your security organization needs visibility to assess risk and to determine whether the policies of your organization, and any regulatory requirements, are being followed. You should ensure that your security organization has visibility into all subscriptions connected to your production environment and network (via [Azure ExpressRoute](#) or [site-to-site VPN](#)). A [Global Administrator](#) in Microsoft Entra ID can elevate their access to the [User Access Administrator](#) role and see all subscriptions and managed groups connected to your environment.

See [elevate access to manage all Azure subscriptions and management groups](#) to ensure that you and your security group can view all subscriptions or management groups connected to your environment. You should remove this elevated access after you've assessed risks.

Enable single sign-on

In a mobile-first, cloud-first world, you want to enable single sign-on (SSO) to devices, apps, and services from anywhere so your users can be productive wherever and whenever. When you have multiple identity solutions to manage, this becomes an administrative problem not only for IT but also for users who have to remember multiple passwords.

By using the same identity solution for all your apps and resources, you can achieve SSO. And your users can use the same set of credentials to sign in and access the resources that they need, whether the resources are located on-premises or in the cloud.

Best practice: Enable SSO.

Detail: Microsoft Entra ID [extends on-premises Active Directory](#) to the cloud. Users can use their primary work or school account for their domain-joined devices, company resources, and all of the web and SaaS applications that they need to get their jobs done. Users don't have to remember multiple sets of usernames and passwords, and their application access can be automatically provisioned (or deprovisioned) based on their organization group memberships and their status as an employee. And you can control that access for gallery apps or for your

own on-premises apps that you've developed and published through the [Microsoft Entra application proxy](#).

Use SSO to enable users to access their [SaaS applications](#) based on their work or school account in Microsoft Entra ID. This is applicable not only for Microsoft SaaS apps, but also other apps, such as [Google Apps](#) and [Salesforce](#). You can configure your application to use Microsoft Entra ID as a [SAML-based identity](#) provider. As a security control, Microsoft Entra ID does not issue a token that allows users to sign in to the application unless they have been granted access through Microsoft Entra ID. You can grant access directly, or through a group that users are a member of.

Organizations that don't create a common identity to establish SSO for their users and applications are more exposed to scenarios where users have multiple passwords. These scenarios increase the likelihood of users reusing passwords or using weak passwords.

Turn on Conditional Access

Users can access your organization's resources by using a variety of devices and apps from anywhere. As an IT admin, you want to make sure that these devices meet your standards for security and compliance. Just focusing on who can access a resource is not sufficient anymore.

To balance security and productivity, you need to think about how a resource is accessed before you can make a decision about access control. With Microsoft Entra Conditional Access, you can address this requirement. With Conditional Access, you can make automated access control decisions based on conditions for accessing your cloud apps.

Best practice: Manage and control access to corporate resources.

Detail: Configure common Microsoft Entra [Conditional Access policies](#) based on a group, location, and application sensitivity for SaaS apps and Microsoft Entra ID-connected apps.

Best practice: Block legacy authentication protocols.

Detail: Attackers exploit weaknesses in older protocols every day, particularly for password spray attacks. Configure Conditional Access to [block legacy protocols](#).

Plan for routine security improvements

Security is always evolving, and it is important to build into your cloud and identity management framework a way to regularly show growth and discover new ways to secure your environment.

Identity Secure Score is a set of recommended security controls that Microsoft publishes that works to provide you a numerical score to objectively measure your security posture and help

plan future security improvements. You can also view your score in comparison to those in other industries as well as your own trends over time.

Best practice: Plan routine security reviews and improvements based on best practices in your industry.

Detail: Use the Identity Secure Score feature to rank your improvements over time.

Enable password management

If you have multiple tenants or you want to enable users to [reset their own passwords](#), it's important that you use appropriate security policies to prevent abuse.

Best practice: Set up self-service password reset (SSPR) for your users.

Detail: Use the Microsoft Entra ID [self-service password reset](#) feature.

Best practice: Monitor how or if SSPR is really being used.

Detail: Monitor the users who are registering by using the Microsoft Entra ID [Password Reset Registration Activity report](#). The reporting feature that Microsoft Entra ID provides helps you answer questions by using prebuilt reports. If you're appropriately licensed, you can also create custom queries.

Best practice: Extend cloud-based password policies to your on-premises infrastructure.

Detail: Enhance password policies in your organization by performing the same checks for on-premises password changes as you do for cloud-based password changes. Install [Microsoft Entra password protection](#) for Windows Server Active Directory agents on-premises to extend banned password lists to your existing infrastructure. Users and admins who change, set, or reset passwords on-premises are required to comply with the same password policy as cloud-only users.

Enforce multifactor verification for users

We recommend that you require two-step verification for all of your users. This includes administrators and others in your organization who can have a significant impact if their account is compromised (for example, financial officers).

There are multiple options for requiring two-step verification. The best option for you depends on your goals, the Microsoft Entra edition you're running, and your licensing program. See [How to require two-step verification for a user](#) to determine the best option for you. See the [Microsoft Entra ID](#) and [Microsoft Entra multifactor authentication](#) pricing pages for more information about licenses and pricing.

Following are options and benefits for enabling two-step verification:

Option 1: Enable MFA for all users and login methods with Microsoft Entra Security Defaults

Benefit: This option enables you to easily and quickly enforce MFA for all users in your environment with a stringent policy to:

- Challenge administrative accounts and administrative logon mechanisms
- Require MFA challenge via Microsoft Authenticator for all users
- Restrict legacy authentication protocols.

This method is available to all licensing tiers but is not able to be mixed with existing Conditional Access policies. You can find more information in [Microsoft Entra Security Defaults](#)

Option 2: [Enable multifactor authentication by changing user state.](#)

Benefit: This is the traditional method for requiring two-step verification. It works with both [Microsoft Entra multifactor authentication in the cloud](#) and [Azure Multi-Factor Authentication Server](#). Using this method requires users to perform two-step verification every time they sign in and overrides Conditional Access policies.

To determine where multifactor authentication needs to be enabled, see [Which version of Microsoft Entra multifactor authentication is right for my organization?](#).

Option 3: [Enable multifactor authentication with Conditional Access policy.](#)

Benefit: This option allows you to prompt for two-step verification under specific conditions by using [Conditional Access](#). Specific conditions can be user sign-in from different locations, untrusted devices, or applications that you consider risky. Defining specific conditions where you require two-step verification enables you to avoid constant prompting for your users, which can be an unpleasant user experience.

This is the most flexible way to enable two-step verification for your users. Enabling a Conditional Access policy works only for Microsoft Entra multifactor authentication in the cloud and is a premium feature of Microsoft Entra ID. You can find more information on this method in [Deploy cloud-based Microsoft Entra multifactor authentication](#).

Option 4: Enable multifactor authentication with Conditional Access policies by evaluating [Risk-based Conditional Access policies](#).

Benefit: This option enables you to:

- Detect potential vulnerabilities that affect your organization's identities.
- Configure automated responses to detected suspicious actions that are related to your organization's identities.
- Investigate suspicious incidents and take appropriate action to resolve them.

This method uses the Microsoft Entra ID Protection risk evaluation to determine if two-step verification is required based on user and sign-in risk for all cloud applications. This method

requires Microsoft Entra ID P2 licensing. You can find more information on this method in [Microsoft Entra ID Protection](#).

ⓘ Note

Option 2, enabling multifactor authentication by changing the user state, overrides Conditional Access policies. Because options 3 and 4 use Conditional Access policies, you cannot use option 2 with them.

Organizations that don't add extra layers of identity protection, such as two-step verification, are more susceptible for credential theft attack. A credential theft attack can lead to data compromise.

Use role-based access control

Access management for cloud resources is critical for any organization that uses the cloud. [Azure role-based access control \(Azure RBAC\)](#) helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to.

Designating groups or individual roles responsible for specific functions in Azure helps avoid confusion that can lead to human and automation errors that create security risks. Restricting access based on the [need to know](#) and [least privilege](#) security principles is imperative for organizations that want to enforce security policies for data access.

Your security team needs visibility into your Azure resources in order to assess and remediate risk. If the security team has operational responsibilities, they need additional permissions to do their jobs.

You can use [Azure RBAC](#) to assign permissions to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

Best practice: Segregate duties within your team and grant only the amount of access to users that they need to perform their jobs. Instead of giving everybody unrestricted permissions in your Azure subscription or resources, allow only certain actions at a particular scope.

Detail: Use [Azure built-in roles](#) in Azure to assign privileges to users.

ⓘ Note

Specific permissions create unneeded complexity and confusion, accumulating into a "legacy" configuration that's difficult to fix without fear of breaking something. Avoid resource-specific permissions. Instead, use management groups for enterprise-wide

permissions and resource groups for permissions within subscriptions. Avoid user-specific permissions. Instead, assign access to groups in Microsoft Entra ID.

Best practice: Grant security teams with Azure responsibilities access to see Azure resources so they can assess and remediate risk.

Detail: Grant security teams the Azure RBAC [Security Reader](#) role. You can use the root management group or the segment management group, depending on the scope of responsibilities:

- **Root management group** for teams responsible for all enterprise resources
- **Segment management group** for teams with limited scope (commonly because of regulatory or other organizational boundaries)

Best practice: Grant the appropriate permissions to security teams that have direct operational responsibilities.

Detail: Review the Azure built-in roles for the appropriate role assignment. If the built-in roles don't meet the specific needs of your organization, you can create [Azure custom roles](#). As with built-in roles, you can assign custom roles to users, groups, and service principals at subscription, resource group, and resource scopes.

Best practices: Grant Microsoft Defender for Cloud access to security roles that need it. Defender for Cloud allows security teams to quickly identify and remediate risks.

Detail: Add security teams with these needs to the Azure RBAC [Security Admin](#) role so they can view security policies, view security states, edit security policies, view alerts and recommendations, and dismiss alerts and recommendations. You can do this by using the root management group or the segment management group, depending on the scope of responsibilities.

Organizations that don't enforce data access control by using capabilities like Azure RBAC might be giving more privileges than necessary to their users. This can lead to data compromise by allowing users to access types of data (for example, high business impact) that they shouldn't have.

Lower exposure of privileged accounts

Securing privileged access is a critical first step to protecting business assets. Minimizing the number of people who have access to secure information or resources reduces the chance of a malicious user getting access, or an authorized user inadvertently affecting a sensitive resource.

Privileged accounts are accounts that administer and manage IT systems. Cyber attackers target these accounts to gain access to an organization's data and systems. To secure

privileged access, you should isolate the accounts and systems from the risk of being exposed to a malicious user.

We recommend that you develop and follow a roadmap to secure privileged access against cyber attackers. For information about creating a detailed roadmap to secure identities and access that are managed or reported in Microsoft Entra ID, Microsoft Azure, Microsoft 365, and other cloud services, review [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#).

The following summarizes the best practices found in [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#):

Best practice: Manage, control, and monitor access to privileged accounts.

Detail: Turn on [Microsoft Entra Privileged Identity Management](#). After you turn on Privileged Identity Management, you'll receive notification email messages for privileged access role changes. These notifications provide early warning when additional users are added to highly privileged roles in your directory.

Best practice: Ensure all critical admin accounts are managed Microsoft Entra accounts. **Detail:** Remove any consumer accounts from critical admin roles (for example, Microsoft accounts like hotmail.com, live.com, and outlook.com).

Best practice: Ensure all critical admin roles have a separate account for administrative tasks in order to avoid phishing and other attacks to compromise administrative privileges.

Detail: Create a separate admin account that's assigned the privileges needed to perform the administrative tasks. Block the use of these administrative accounts for daily productivity tools like Microsoft 365 email or arbitrary web browsing.

Best practice: Identify and categorize accounts that are in highly privileged roles.

Detail: After turning on Microsoft Entra Privileged Identity Management, view the users who are in the global administrator, privileged role administrator, and other highly privileged roles. Remove any accounts that are no longer needed in those roles, and categorize the remaining accounts that are assigned to admin roles:

- Individually assigned to administrative users and can be used for non-administrative purposes (for example, personal email)
- Individually assigned to administrative users and designated for administrative purposes only
- Shared across multiple users
- For emergency access scenarios
- For automated scripts
- For external users

Best practice: Implement "just in time" (JIT) access to further lower the exposure time of privileges and increase your visibility into the use of privileged accounts.

Detail: Microsoft Entra Privileged Identity Management lets you:

- Limit users to only taking on their privileges JIT.
- Assign roles for a shortened duration with confidence that the privileges are revoked automatically.

Best practice: Define at least two emergency access accounts.

Detail: Emergency access accounts help organizations restrict privileged access in an existing Microsoft Entra environment. These accounts are highly privileged and are not assigned to specific individuals. Emergency access accounts are limited to scenarios where normal administrative accounts can't be used. Organizations must limit the emergency account's usage to only the necessary amount of time.

Evaluate the accounts that are assigned or eligible for the global admin role. If you don't see any cloud-only accounts by using the *.onmicrosoft.com domain (intended for emergency access), create them. For more information, see [Managing emergency access administrative accounts in Microsoft Entra ID](#).

Best practice: Have a "break glass" process in place in case of an emergency.

Detail: Follow the steps in [Securing privileged access for hybrid and cloud deployments in Microsoft Entra ID](#).

Best practice: Require all critical admin accounts to be password-less (preferred), or require multifactor authentication.

Detail: Use the [Microsoft Authenticator app](#) to sign in to any Microsoft Entra account without using a password. Like [Windows Hello for Business](#), the Microsoft Authenticator uses key-based authentication to enable a user credential that's tied to a device and uses biometric authentication or a PIN.

Require Microsoft Entra multifactor authentication at sign-in for all individual users who are permanently assigned to one or more of the Microsoft Entra admin roles: Global Administrator, Privileged Role Administrator, Exchange Online Administrator, and SharePoint Online Administrator. Enable [multifactor authentication for your admin accounts](#) and ensure that admin account users have registered.

Best practice: For critical admin accounts, have an admin workstation where production tasks aren't allowed (for example, browsing and email). This will protect your admin accounts from attack vectors that use browsing and email and significantly lower your risk of a major incident.

Detail: Use an admin workstation. Choose a level of workstation security:

- Highly secure productivity devices provide advanced security for browsing and other productivity tasks.
- [Privileged Access Workstations \(PAWs\)](#) provide a dedicated operating system that's protected from internet attacks and threat vectors for sensitive tasks.

Best practice: Deprovision admin accounts when employees leave your organization.

Detail: Have a process in place that disables or deletes admin accounts when employees leave your organization.

Best practice: Regularly test admin accounts by using current attack techniques.

Detail: Use Microsoft 365 Attack Simulator or a third-party offering to run realistic attack scenarios in your organization. This can help you find vulnerable users before a real attack occurs.

Best practice: Take steps to mitigate the most frequently used attacked techniques.

Detail: [Identify Microsoft accounts in administrative roles that need to be switched to work or school accounts](#)

[Ensure separate user accounts and mail forwarding for global administrator accounts](#)

[Ensure that the passwords of administrative accounts have recently changed](#)

[Turn on password hash synchronization](#)

[Require multifactor authentication for users in all privileged roles as well as exposed users](#)

[Obtain your Microsoft 365 Secure Score \(if using Microsoft 365\)](#)

[Review the Microsoft 365 security guidance \(if using Microsoft 365\)](#)

[Configure Microsoft 365 Activity Monitoring \(if using Microsoft 365\)](#)

[Establish incident/emergency response plan owners](#)

[Secure on-premises privileged administrative accounts](#)

If you don't secure privileged access, you might find that you have too many users in highly privileged roles and are more vulnerable to attacks. Malicious actors, including cyber attackers, often target admin accounts and other elements of privileged access to gain access to sensitive data and systems by using credential theft.

Control locations where resources are created

Enabling cloud operators to perform tasks while preventing them from breaking conventions that are needed to manage your organization's resources is very important. Organizations that

want to control the locations where resources are created should hard code these locations.

You can use [Azure Resource Manager](#) to create security policies whose definitions describe the actions or resources that are specifically denied. You assign those policy definitions at the desired scope, such as the subscription, the resource group, or an individual resource.

📌 **Note**

Security policies are not the same as Azure RBAC. They actually use Azure RBAC to authorize users to create those resources.

Organizations that are not controlling how resources are created are more susceptible to users who might abuse the service by creating more resources than they need. Hardening the resource creation process is an important step to securing a multitenant scenario.

Actively monitor for suspicious activities

An active identity monitoring system can quickly detect suspicious behavior and trigger an alert for further investigation. The following table lists Microsoft Entra capabilities that can help organizations monitor their identities:

Best practice: Have a method to identify:

- Attempts to sign in [without being traced](#).
- [Brute force](#) attacks against a particular account.
- Attempts to sign in from multiple locations.
- Sign-ins from [infected devices](#).
- Suspicious IP addresses.

Detail: Use Microsoft Entra ID P1 or P2 [anomaly reports](#). Have processes and procedures in place for IT admins to run these reports on a daily basis or on demand (usually in an incident response scenario).

Best practice: Have an active monitoring system that notifies you of risks and can adjust risk level (high, medium, or low) to your business requirements.

Detail: Use [Microsoft Entra ID Protection](#), which flags the current risks on its own dashboard and sends daily summary notifications via email. To help protect your organization's identities, you can configure risk-based policies that automatically respond to detected issues when a specified risk level is reached.

Organizations that don't actively monitor their identity systems are at risk of having user credentials compromised. Without knowledge that suspicious activities are taking place

through these credentials, organizations can't mitigate this type of threat.

Use Microsoft Entra ID for storage authentication

[Azure Storage](#) supports authentication and authorization with Microsoft Entra ID for Blob storage and Queue storage. With Microsoft Entra authentication, you can use the Azure role-based access control to grant specific permissions to users, groups, and applications down to the scope of an individual blob container or queue.

We recommend that you use [Microsoft Entra ID for authenticating access to storage](#) .

Next step

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

Five steps to securing your identity infrastructure

Article • 09/06/2024

If you're reading this document, you're aware of the significance of security. You likely already carry the responsibility for securing your organization. If you need to convince others of the importance of security, send them to read the latest [Microsoft Digital Defense Report](#) .

This document helps you get a more secure posture using the capabilities of Microsoft Entra ID by using a five-step checklist to improve your organization's protection against cyber-attacks.

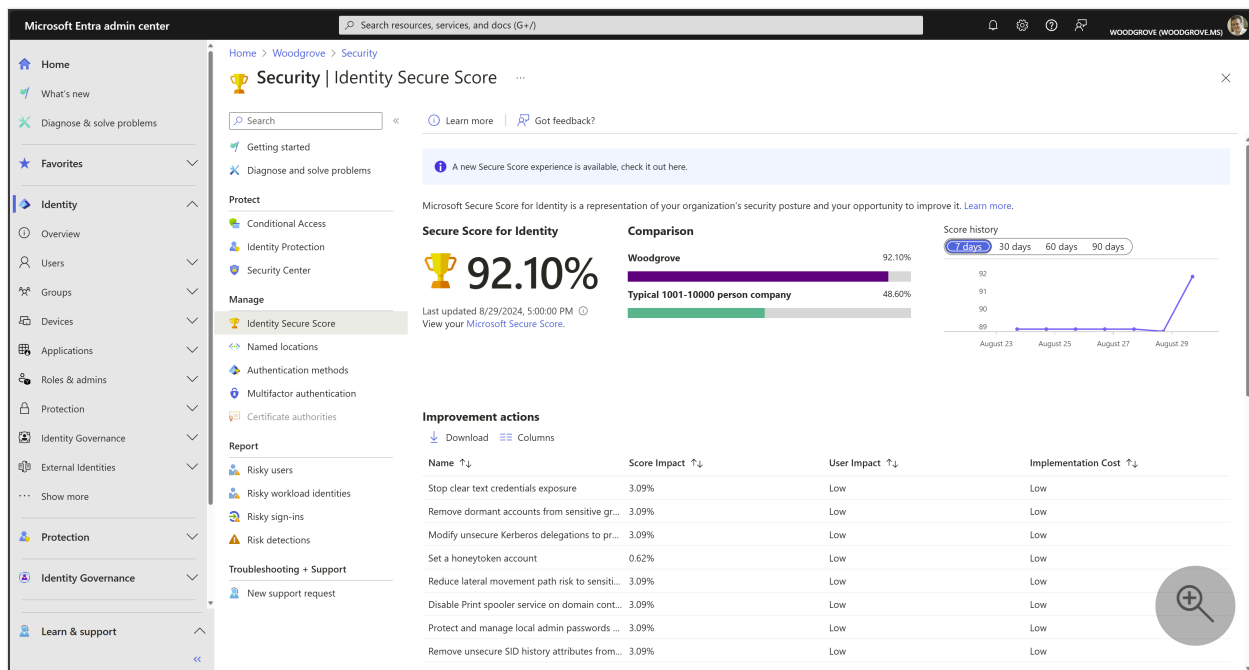
This checklist helps you quickly deploy critical recommended actions to protect your organization immediately by explaining how to:

- Strengthen your credentials
- Reduce your attack surface area
- Automate threat response
- Utilize cloud intelligence
- Enable end-user self-service

Note

Many of the recommendations in this document apply only to applications that are configured to use Microsoft Entra ID as their identity provider. Configuring apps for Single Sign-On assures the benefits of credential policies, threat detection, auditing, logging, and other features add to those applications. [Microsoft Entra Application Management](#) is the foundation on which all these recommendations are based.

The recommendations in this document are aligned with the [Identity Secure Score](#), an automated assessment of your Microsoft Entra tenant's identity security configuration. Organizations can use the Identity Secure Score page in the Microsoft Entra admin center to find gaps in their current security configuration to ensure they follow current Microsoft best practices for security. Implementing each recommendation in the Secure Score page increases your score and allow you to track your progress, plus help you compare your implementation against other similar size organizations.



Note

Some of the functionality recommended here is available to all customers, while others require a Microsoft Entra ID P1 or P2 subscription. Please review [Microsoft Entra pricing](#) and [Microsoft Entra Deployment checklist](#) for more information.

Before you begin: Protect privileged accounts with MFA

Before you begin this checklist, make sure you don't get compromised while you're reading this checklist. In Microsoft Entra we observe 50 million password attacks daily, yet only 20% of users and 30% of global admins are using strong authentications such as multifactor authentication (MFA). These statistics are based on data as of August 2021. In Microsoft Entra ID, users who have privileged roles, such as administrators, are the root of trust to build and manage the rest of the environment. Implement the following practices to minimize the effects of a compromise.

Attackers who get control of privileged accounts can do tremendous damage, so it's critical to [protect these accounts before proceeding](#). Enable and require [Microsoft Entra multifactor authentication \(MFA\)](#) for all administrators in your organization using [Microsoft Entra Security Defaults](#) or [Conditional Access](#). It's critical.

All set? Let's get started on the checklist.

Step 1: Strengthen your credentials

Although other types of attacks are emerging, including consent phishing and attacks on nonhuman identities, password-based attacks on user identities are still the most prevalent vector of identity compromise. Well-established spear phishing and password spray campaigns by adversaries continue to be successful against organizations that don't implement multifactor authentication (MFA) or other protections against this common tactic.

As an organization you need to make sure that your identities are validated and secured with MFA everywhere. In 2020, the [Federal Bureau of Investigation \(FBI\) Internet Crime Complaint Center \(IC3\) Report](#) [↗] identified phishing as the top crime type for victim complaints. The number of reports doubled compared to the previous year. Phishing poses a significant threat to both businesses and individuals, and credential phishing was used in many of the most damaging attacks last year. Microsoft Entra multifactor authentication (MFA) helps safeguard access to data and applications, providing another layer of security by using a second form of authentication. Organizations can enable multifactor authentication with Conditional Access to make the solution fit their specific needs. Take a look at this deployment guide to see how you how to [plan, implement, and roll-out Microsoft Entra multifactor authentication](#).

Make sure your organization uses strong authentication

To easily enable the basic level of identity security, you can use the one-select enablement with [Microsoft Entra security defaults](#). Security defaults enforce Microsoft Entra multifactor authentication for all users in a tenant and blocks sign-ins from legacy protocols tenant-wide.

If your organization has Microsoft Entra ID P1 or P2 licenses, then you can also use the [Conditional Access insights and reporting workbook](#) to help you discover gaps in your configuration and coverage. From these recommendations, you can easily close this gap by creating a policy using the new Conditional Access templates experience. [Conditional Access templates](#) are designed to provide an easy method to deploy new policies that align with Microsoft recommended [best practices](#), making it easy to deploy common policies to protect your identities and devices.

Start banning commonly attacked passwords and turn off traditional complexity, and expiration rules.

Many organizations use traditional complexity and password expiration rules. [Microsoft's research](#) [↗] shows, and [National Institute of Standards and Technology \(NIST\) Special Publication 800-63B Digital Identity Guidelines](#) [↗] state, that these policies cause users to choose passwords that are easier to guess. We recommend you use [Microsoft](#)

[Entra password protection](#) a dynamic banned password feature using current attacker behavior to prevent users from setting passwords that can easily be guessed. This capability is always on when users are created in the cloud, but is now also available for hybrid organizations when they deploy [Microsoft Entra password protection for Windows Server Active Directory](#). In addition, we recommend you remove expiration policies. Password change offers no containment benefits as cyber criminals almost always use credentials as soon as they compromise them. Refer to the following article to [Set the password expiration policy for your organization](#).

Protect against leaked credentials and add resilience against outages

The simplest and recommended method for enabling cloud authentication for on-premises directory objects in Microsoft Entra ID is to enable [password hash synchronization \(PHS\)](#). If your organization uses a hybrid identity solution with pass-through authentication or federation, then you should enable password hash sync for the following two reasons:

- The [Users with leaked credentials report](#) in Microsoft Entra ID warns of publicly exposed username and password pairs. An incredible volume of passwords is leaked via phishing, malware, and password reuse on third-party sites that are later breached. Microsoft finds many of these leaked credentials and tells you, in this report, if they match credentials in your organization – but only if you enable [password hash sync](#) or have cloud-only identities.
- If an on-premises outage happens, like a ransomware attack, you can [switch over to using cloud authentication using password hash sync](#). This backup authentication method allows you to continue accessing apps configured for authentication with Microsoft Entra ID, including Microsoft 365. In this case, IT staff doesn't need to resort to shadow IT or personal email accounts to share data until the on-premises outage is resolved.

Passwords are never stored in clear text or encrypted with a reversible algorithm in Microsoft Entra ID. For more information on the actual process of password hash synchronization, see [Detailed description of how password hash synchronization works](#).

Implement AD FS extranet smart logout

Smart logout helps lock out bad actors that try to guess your users' passwords or use brute-force methods to get in. Smart logout can recognize sign-ins that come from valid users and treat them differently than ones of attackers and other unknown sources. Attackers get locked out, while your users continue to access their accounts and be

productive. Organizations, which configure applications to authenticate directly to Microsoft Entra ID benefit from Microsoft Entra smart lockout. Federated deployments that use AD FS 2016 and AD FS 2019 can enable similar benefits using [AD FS Extranet Lockout](#) and [Extranet Smart Lockout](#).

Step 2: Reduce your attack surface area

Given the pervasiveness of password compromise, minimizing the attack surface in your organization is critical. Disable the use of older, less secure protocols, limit access entry points, moving to cloud authentication, exercise more significant control of administrative access to resources, and embrace Zero Trust security principles.

Use Cloud Authentication

Credentials are a primary attack vector. The practices in this blog can reduce the attack surface by using cloud authentication, deploy MFA, and use passwordless authentication methods. You can deploy passwordless methods such as Windows Hello for Business, Phone Sign-in with the Microsoft Authenticator App or FIDO.

Block legacy authentication

Apps using their own legacy methods to authenticate with Microsoft Entra ID and access company data, pose another risk for organizations. Examples of apps using legacy authentication are POP3, IMAP4, or SMTP clients. Legacy authentication apps authenticate on behalf of the user and prevent Microsoft Entra ID from doing advanced security evaluations. The alternative, modern authentication, reduces your security risk, because it supports multifactor authentication and Conditional Access.

We recommend the following actions:

1. Discover legacy authentication in your organization with Microsoft Entra sign-in logs and Log Analytics workbooks.
2. Setup SharePoint Online and Exchange Online to use modern authentication.
3. If you have Microsoft Entra ID P1 or P2 licenses, use Conditional Access policies to block legacy authentication. For Microsoft Entra ID Free tier, use Microsoft Entra Security Defaults.
4. Block legacy authentication if you use AD FS.
5. Block Legacy Authentication with Exchange Server 2019.
6. Disable legacy authentication in Exchange Online.

For more information, see the article [Blocking legacy authentication protocols in Microsoft Entra ID](#).

Block invalid authentication entry points

Using the *verify explicitly principle*, you should reduce the impact of compromised user credentials when they happen. For each app in your environment, consider the valid use cases: which groups, which networks, which devices and other elements are authorized – then block the rest. With Microsoft Entra Conditional Access, you can control how authorized users access their apps and resources based on specific conditions you define.

For more information on how to use Conditional Access for your Cloud Apps and user actions, see [Conditional Access Cloud apps, actions, and authentication context](#).

Review and govern admin roles

Another Zero Trust pillar is the need to minimize the likelihood a compromised account can operate with a privileged role. This control can be accomplished by assigning the least amount of privilege to an identity. If you're new to Microsoft Entra roles, this article helps you understand Microsoft Entra roles.

Privileged roles in Microsoft Entra ID should be cloud only accounts in order to isolate them from any on-premises environments and don't use on-premises password vaults to store the credentials.

Implement Privilege Access Management

Privileged Identity Management (PIM) provides a time-based and approval-based role activation to mitigate the risks of excessive, unnecessary, or misused access permissions to important resources. These resources include resources in Microsoft Entra ID, Azure, and other Microsoft Online Services such as Microsoft 365 or Microsoft Intune.

Microsoft Entra Privileged Identity Management (PIM) helps you minimize account privileges by helping you:

- Identify and manage users assigned to administrative roles.
- Understand unused or excessive privilege roles you should remove.
- Establish rules to make sure privileged roles are protected by multifactor authentication.
- Establish rules to make sure privileged roles are granted only long enough to accomplish the privileged task.

Enable Microsoft Entra PIM, then view the users who are assigned administrative roles and remove unnecessary accounts in those roles. For remaining privileged users, move them from permanent to eligible. Finally, establish appropriate policies to make sure when they need to gain access to those privileged roles, they can do so securely, with the necessary change control.

Microsoft Entra built-in and custom roles operate on concepts similar to roles found in the role-based access control system for Azure resources (Azure roles). The difference between these two role-based access control systems is:

- Microsoft Entra roles control access to Microsoft Entra resources such as users, groups, and applications using the Microsoft Graph API
- Azure roles control access to Azure resources such as virtual machines or storage using Azure Resource Management

Both systems contain similarly used role definitions and role assignments. However, Microsoft Entra role permissions can't be used in Azure custom roles and vice versa. As part of deploying your privileged account process, follow the best practice to create at least two emergency accounts to make sure you still have access to Microsoft Entra ID if you lock yourself out.

For more information, see the article [Plan a Privileged Identity Management deployment and securing privileged access](#).

Restrict user consent operations

It's important to understand the various Microsoft Entra application consent experiences, the types of permissions and consent, and their implications on your organization's security posture. While allowing users to consent by themselves does allow users to easily acquire useful applications that integrate with Microsoft 365, Azure, and other services, it can represent a risk if not used and monitored carefully.

Microsoft recommends restricting user consent to allow end-user consent only for apps from verified publishers and only for permissions you select. If end-user consent is restricted, previous consent grants will still be honored but all future consent operations that an administrator must perform. For restricted cases, users can request admin consent through an integrated admin consent request workflow or through your own support processes. Before restricting end-user consent, use our recommendations to plan this change in your organization. For applications you wish to allow all users to access, consider granting consent on behalf of all users, making sure users who didn't yet individually consent can access the app. If you don't want these applications to be

available to all users in all scenarios, use application assignment and Conditional Access to restrict user access to specific apps.

Make sure users can request admin approval for new applications to reduce user friction, minimize support volume, and prevent users from signing up for applications using non-Microsoft Entra credentials. Once you regulate your consent operations, administrators should audit app and consent permissions regularly.

For more information, see the article [Microsoft Entra consent framework](#).

Step 3: Automate threat response

Microsoft Entra ID has many capabilities that automatically intercept attacks, to remove the latency between detection and response. You can reduce the costs and risks, when you reduce the time criminals use to embed themselves into your environment. Here are the concrete steps you can take.

For more information, see the article [How To: Configure and enable risk policies](#).

Implement sign-in risk policy

A sign-in risk represents the probability that a given that the identity owner didn't authorize the authentication request. A sign-in risk-based policy can be implemented through adding a sign-in risk condition to your Conditional Access policies that evaluates the risk level to a specific user or group. Based on the risk level (high/medium/low), a policy can be configured to block access or force multifactor authentication. We recommend that you force multifactor authentication on Medium or above risky sign-ins.

... > Conditional Access | Policies >

CA05 - Require Multi-factor authentication for high risk

Conditional Access policy

Delete
 View policy information

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies. [Learn more](#)

Name *

Assignments

Users ¹

[All users included and specific users excluded](#)

Target resources ¹

[No target resources selected](#)

Network NEW ¹

[Not configured](#)

Conditions ¹

[1 condition selected](#)

Access controls

Grant ¹

Enable policy

Report-only ☐ ☒ On ☐ Off

Control access based on signals from conditional access policy like risk, device platform, location, client application, and device state. [Learn more](#)

User risk ¹

User risk level is the likelihood that the user's account is compromised.

[Not configured](#)

Sign-in risk ¹

Sign-in risk level is the likelihood that the user's session is compromised.

[2 included](#)

Insider risk ¹

Insider risk assesses the user's risky data-related activity in Microsoft Purview Insider Management.

[Not configured](#)

Device platforms ¹

[Not configured](#)

Locations ¹

[Not configured](#)

Sign-in risk

Control user access to respond to specific sign-in risk levels. [Learn more](#)

Configure ¹

☒ Yes ☐ No

Sign-in risk level is generated based on all real-time risk detections.

Select the sign-in risk level this policy will apply to

☒ High

☒ Medium

☐ Low

☐ No risk

Implement user risk security policy

User risk indicates the likelihood of user identity compromise and is calculated based on the user risk detections that are associated with a user's identity. A user risk-based policy can be implemented through adding a user risk condition to your Conditional Access policies that evaluates the risk level to a specific user. Based on Low, Medium, High risk-level, a policy can be configured to block access or require a secure password change using multifactor authentication. Microsoft's recommendation is to require a secure password change for users on high risk.

[Home](#) > [Identity Protection | Dashboard](#) > [Conditional Access | Policies](#) >

CA06 - Require password change for high-risk users

Conditional Access policy

Delete
 View policy information

Control access based on Conditional Access policy to bring signals together, to make decisions, and enforce organizational policies.
[Learn more](#)

Name *

CA06 - Require password change for high-risk users

Assignments

Users ①

[All users included and specific users excluded](#)

Target resources ①

[All cloud apps](#)

Network NEW ①

[Not available](#)

Conditions ①

[1 condition selected](#)

Access controls

Grant ①

[1 control selected](#)

Session ①

[Sign in frequency: Every time](#)

Enable policy

Report-only ☒ On ☐ Off

[Save](#)

Grant

☐ Require authentication strength ①

"Require authentication strength" cannot be used with "Require multifactor authentication". [Learn more](#)

☐ Require device to be marked as compliant ①

☐ Require Microsoft Entra hybrid joined device ①

☐ Require approved client app ①
[See list of approved client apps](#)

☐ Require app protection policy ①
[See list of policy protected client apps](#)

☒ Require password change ①

"Require password change" can only be used when policy is assigned to "All cloud apps". [Learn more](#)

For multiple controls

☒ Require all the selected controls

☐ Require one of the selected controls

[Select](#)

Included in the user risk detection is a check whether the user's credentials match to credentials leaked by cybercriminals. To function optimally, it's important to implement password hash synchronization with Microsoft Entra Connect Sync.

Integrate Microsoft Defender XDR with Microsoft Entra ID Protection

For Identity Protection to be able to perform the best risk detection possible, it needs to get as many signals as possible. It's therefore important to integrate the complete suite of Microsoft Defender XDR services:

- Microsoft Defender for Endpoint
- Microsoft Defender for Office 365
- Microsoft Defender for Identity

- Microsoft Defender for Cloud Apps

Learn more about Microsoft Threat Protection and the importance of integrating different domains, in the following short video.

<https://www.microsoft.com/en-us/videoplayer/embed/RE4Bzww?postJsllMsg=true> 

Set up monitoring and alerting

Monitoring and auditing your logs is important to detect suspicious behavior. The Azure portal has several ways to integrate Microsoft Entra logs with other tools, like Microsoft Sentinel, Azure Monitor, and other SIEM tools. For more information, see the [Microsoft Entra security operations guide](#).

Step 4: Utilize cloud intelligence

Auditing and logging of security-related events and related alerts are essential components of an efficient protection strategy. Security logs and reports provide you with an electronic record of suspicious activities and help you detect patterns that might indicate attempted or successful external penetration of the network, and internal attacks. You can use auditing to monitor user activity, document regulatory compliance, do forensic analysis, and more. Alerts provide notifications of security events. Make sure you have a log retention policy in place for both your sign-in logs and audit logs for Microsoft Entra ID by exporting into Azure Monitor or a SIEM tool.

Monitor Microsoft Entra ID

Microsoft Azure services and features provide you with configurable security auditing and logging options to help you identify gaps in your security policies and mechanisms and address those gaps to help prevent breaches. You can use [Azure Logging and Auditing](#) and use [Audit activity reports in the Microsoft Entra admin center](#). See the [Microsoft Entra Security Operations guide](#) for more details on monitoring user accounts, Privileged accounts, apps, and devices.

Monitor Microsoft Entra Connect Health in hybrid environments

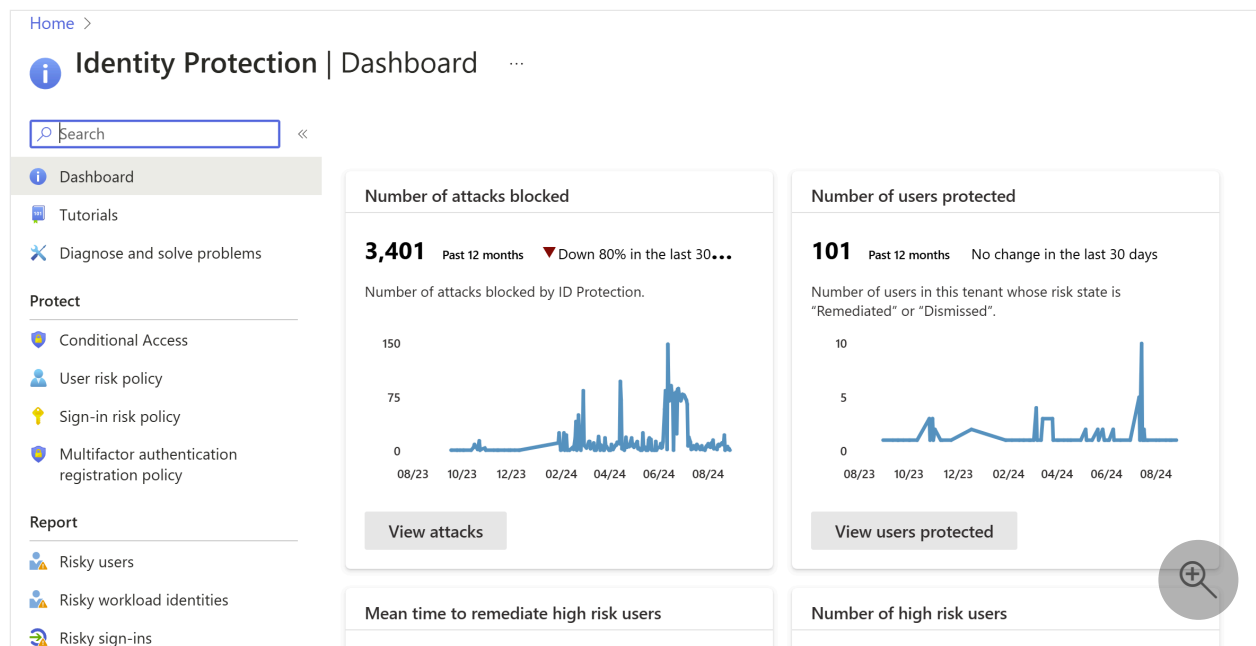
[Monitoring AD FS with Microsoft Entra Connect Health](#) provides you with greater insight into potential issues and visibility of attacks on your AD FS infrastructure. You can now view [ADFS sign-ins](#) to give greater depth for your monitoring. Microsoft Entra Connect Health delivers alerts with details, resolution steps, and links to related documentation;

usage analytics for several metrics related to authentication traffic; performance monitoring and reports. Utilize the [Risky IP WorkBook for ADFS](#) that can help identify the norm for your environment and alert when there's a change. All Hybrid Infrastructure should be monitored as a Tier 0 asset. Detailed monitoring guidance for these assets can be found in the [Security Operations guide for Infrastructure](#).

Monitor Microsoft Entra ID Protection events

[Microsoft Entra ID Protection](#) provides two important reports you should monitor daily:

1. Risky sign-in reports surface user sign-in activities you should investigate whether the legitimate owner performed the sign-in.
2. Risky user reports surface user accounts that might be compromised, such as leaked credential that was detected or the user signed in from different locations, causing an impossible travel event.



Audit apps and consented permissions

Users can be tricked into navigating to a compromised web site or apps that gain access to their profile information and user data, such as their email. A malicious actor can use the consented permissions it received to encrypt their mailbox content and demand a ransom to regain your mailbox data. [Administrators should review and audit](#) the permissions given by users. In addition to auditing the permissions given by users, you can [locate risky or unwanted OAuth applications](#) in premium environments.

Step 5: Enable end-user self-service

As much as possible you want to balance security with productivity. Approaching your journey with the mindset that you're setting a foundation for security, you can remove friction from your organization by empowering your users while remaining vigilant and reducing your operational overheads.

Implement self-service password reset

Microsoft Entra ID's [self-service password reset \(SSPR\)](#) offers a simple means for IT administrators to allow users to reset or unlock their passwords or accounts without helpdesk or administrator intervention. The system includes detailed reporting that tracks when users reset their passwords, along with notifications to alert you to misuse or abuse.

Implement self-service group and application access

Microsoft Entra ID can allow nonadministrators to manage access to resources, using security groups, Microsoft 365 groups, application roles, and access package catalogs. [Self-service group management](#) enables group owners to manage their own groups, without needing to be assigned an administrative role. Users can also create and manage Microsoft 365 groups without relying on administrators to handle their requests, and unused groups expire automatically. [Microsoft Entra entitlement management](#) further enables delegation and visibility, with comprehensive access request workflows and automatic expiration. You can delegate to nonadministrators the ability to configure their own access packages for groups, Teams, applications, and SharePoint Online sites they own, with custom policies for who is required to approve access, including configuring employee's managers and business partner sponsors as approvers.

Implement Microsoft Entra access reviews

With [Microsoft Entra access reviews](#), you can manage access package and group memberships, access to enterprise applications, and privileged role assignments to make sure you maintain a security standard. Regular oversight by the users themselves, resource owners, and other reviewers ensure that users don't retain access for extended periods of time when they no longer need it.

Implement automatic user provisioning

Provisioning and deprovisioning are the processes that ensure consistency of digital identities across multiple systems. These processes are typically applied as part of

[identity lifecycle management](#).

Provisioning is the processes of creating an identity in a target system based on certain conditions. Deprovisioning is the process of removing the identity from the target system, when conditions are no longer met. Synchronization is the process of keeping the provisioned object, up to date, so that the source object and target object are similar.

Microsoft Entra ID currently provides three areas of automated provisioning. They are:

- Provisioning from an external nondirectory authoritative system of record to Microsoft Entra ID, via [HR-driven provisioning](#)
- Provisioning from Microsoft Entra ID to applications, via [App provisioning](#)
- Provisioning between Microsoft Entra ID and Active Directory Domain Services, via [inter-directory provisioning](#)

Find out more here: [What is provisioning with Microsoft Entra ID?](#)

Summary

There are many aspects to a secure Identity infrastructure, but this five-step checklist helps you to quickly accomplish a safer and secure identity infrastructure:

- Strengthen your credentials
- Reduce your attack surface area
- Automate threat response
- Utilize cloud intelligence
- Enable end-user self-service

We appreciate how seriously you take security and hope this document is a useful roadmap to a more secure posture for your organization.

Next steps

If you need assistance to plan and deploy the recommendations, refer to the [Microsoft Entra ID project deployment plans](#) for help.

If you're confident all these steps are complete, use Microsoft's [Identity Secure Score](#), which keeps you up to date with the [latest best practices](#) and security threats.

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

Authentication methods in Microsoft Entra ID - passkeys (FIDO2)

Remote phishing attacks are on the rise. These attacks aim to steal or relay identity proofs—such as passwords, SMS codes, or email one-time passcodes—without physical access to the user's device. Attackers often use social engineering, credential harvesting, or downgrade techniques to bypass stronger protections like passkeys or security keys. With AI-driven attack toolkits, these threats are becoming more sophisticated and scalable.

Passkeys help prevent remote phishing by replacing phishable methods like passwords, SMS, and email codes. Built on **FIDO (Fast Identity Online) standards**, passkeys use origin-bound public key cryptography, ensuring credentials can't be replayed or shared with malicious actors. In addition to stronger security, passkeys (FIDO2) offer a frictionless sign-in experience by eliminating passwords, reducing prompts, and enabling fast, secure authentication across devices.

Passkeys (FIDO2) can also be used to sign in to Microsoft Entra ID or Microsoft Entra hybrid joined Windows 11 devices and get single-sign on to cloud and on-premises resources.

What are passkeys?

Passkeys are phishing-resistant credentials that provide **strong authentication** and can serve as a **multifactor authentication (MFA)** method when combined with device biometrics or PIN. They also provide verifier impersonation resistance, which ensures an authenticator only releases secrets to the Relying Party (RP) the passkey was registered with and not an attacker pretending to be that RP. Passkeys (FIDO2) follow FIDO2 standards, using WebAuthn for browsers and CTAP for authenticator communication.

The following process is used when a user signs in to Microsoft Entra ID with a passkey (FIDO2):

1. The user initiates sign-in to Microsoft Entra ID.
2. The user selects a passkey:
 - Same device (stored on the device)
 - Cross-device (via QR code) or a FIDO2 security key
3. Microsoft Entra ID sends a challenge (nonce) to the authenticator.
4. The authenticator locates the key pair using the hashed RP ID and credential ID.
5. The user performs a biometric or PIN gesture to unlock the private key.
6. The authenticator signs the challenge with the private key and returns the signature.
7. Microsoft Entra ID verifies the signature using the public key and issues a token.

Types of passkeys

- **Device-bound passkeys:** The private key is created and stored on a single physical device and never leaves it. Examples:
 - Microsoft Authenticator
 - FIDO2 Security keys
- **Synced passkeys:** The private key is stored in a passkey provider's cloud and synced across devices signed-in to the same passkey provider account. Synced passkeys do not support attestation. Examples:
 - Apple iCloud Keychain
 - Google Password Manager

Synced passkeys offer a seamless and convenient user experience where users can use a device's native unlock mechanism like face, fingerprint or PIN to authenticate. Based on the learnings from hundreds of millions of consumer users of Microsoft accounts that have registered and are using synced passkeys, we have learned:

- **99% of users successfully register synced passkeys**
- Synced passkeys are **14x faster compared to password and a traditional MFA combination: 3 seconds instead of 69 seconds**
- Users are **3x more successful signing-in with synced passkey than legacy authentication methods (95% vs 30%)**
- Synced passkeys in Microsoft Entra ID bring MFA simplicity at scale for all enterprise users. They're a convenient and low-cost alternative to traditional MFA options like SMS and authenticator apps.

For more information about how to deploy passkeys in your organization, see [How to enable synced passkeys](#).

Attestation verifies the authenticity of the passkey provider or device during registration. When enforced:

- It provides cryptographically verifiable device identity through FIDO Metadata Service (MDS). When attestation is enforced, relying parties can validate the authenticator model and apply policy decisions for certified devices.
- Unattested passkeys, including synced passkeys and unattested device-bound passkeys, don't provide device provenance.

In Microsoft Entra ID:

- Attestation can be enforced at the **passkey profile** level.
- If attestation is enabled, only device-bound passkeys are allowed; synced passkeys are excluded.

Choose the right passkey option

FIDO2 security keys are recommended for highly regulated industries or users with elevated privileges. They provide strong security, but can increase costs for equipment, training, and helpdesk support—especially when users lose their physical keys and need account recovery. Passkeys in the Microsoft Authenticator app are another option for these user groups.

For most users—those outside highly regulated environments or without access to sensitive systems—**synced passkeys** offer a convenient, low-cost alternative to traditional MFA. Apple and Google have implemented advanced protections for passkeys stored in their clouds.

Regardless of type—device-bound or synced—passkeys represent a significant security upgrade over phishable MFA methods.

For more details, see [Get started with phishing-resistant MFA deployment in Microsoft Entra ID](#).

Related content

- [Enable passkeys \(FIDO2\) in Microsoft Entra ID](#)
- [Enable passkey profiles in Microsoft Entra ID](#)
- [Enable synced passkeys in Microsoft Entra ID](#)
- [Passkeys in Authenticator App](#)
- [Attestation requirements](#)

Last updated on 11/18/2025

Architecture strategies for identity and access management

11/15/2023

Applies to this Azure Well-Architected Framework Security checklist recommendation:

 Expand table

SE:05 **Implement strict, conditional, and auditable identity and access management (IAM) across all workload users, team members, and system components. Limit access exclusively to *as necessary*. Use modern industry standards for all authentication and authorization implementations. Restrict and rigorously audit access that's not based on identity.**

This guide describes the recommendations for authenticating and authorizing identities that are attempting to access your workload resources.

From a technical control perspective, **identity is always the primary perimeter**. This scope doesn't just include the edges of your workload. It also includes individual components that are inside your workload. Typical identities include:

- **Humans.** Application users, admins, operators, auditors, and bad actors.
- **Systems.** Workload identities, managed identities, API keys, service principals, and Azure resources.
- **Anonymous.** Entities who haven't provided any evidence about who they are.

Definitions

 Expand table

Terms	Definition
Authentication (AuthN)	A process that verifies that an identity is who or what it says it is.
Authorization (AuthZ)	A process that verifies whether an identity has permission to perform a requested action.
Conditional access	A set of rules that allows actions based on specified criteria.
IdP	An identity provider, like Microsoft Entra ID.
Persona	A job function or a title that has a set of responsibilities and actions.

Terms	Definition
Preshared keys	A type of secret that's shared between a provider and consumer and used through a secure and agreed upon mechanism.
Resource identity	An identity defined for cloud resources that's managed by the platform.
Role	A set of permissions that define what a user or group can do.
Scope	Different levels of organizational hierarchy where a role is permitted to operate. Also a group of features in a system.
Security principal	An identity that provides permissions. It can be a user, a group, or a service principal. Any group members get the same level of access.
User identity	An identity for a person, like an employee or an external user.
Workload identity	A system identity for an application, service, script, container, or other component of a workload that's used to authenticate itself to other services and resources.

⚠ Note

An identity can be grouped with other, similar identities under a parent called a *security principal*. A security group is an example of a security principal. This hierarchical relationship simplifies maintenance and improves consistency. Because identity attributes aren't handled at the individual level, chances of errors are also reduced. In this article, the term *identity* is inclusive of security principals.

The role of an identity provider

An identity provider (IdP) is a cloud-hosted service that stores and manages users as digital identities.

Take advantage of the capabilities provided by a trusted IdP for your identity and access management. Don't implement custom systems to replace an IdP. IdP systems are improved frequently based on the latest attack vectors by capturing billions of signals across multiple tenants each day. Microsoft Entra ID is the IdP for Azure cloud platform.

Authentication

Authentication is a process that verifies identities. The requesting identity is required to provide some form of verifiable identification. For example:

- A user name and password.

- A preshared secret, like an API key that grants access.
- A shared access signature (SAS) token.
- A certificate that's used in TLS mutual authentication.

As much as possible, the verification process should be handled by your IdP.

Authorization

Authorization is a process that allows or denies actions that are requested by the verified identity. The action might be operational or related to resource management.

Authorization requires that you assign permissions to the identities, which you need to do by using the functionality provided by your IdP.

To get a holistic view of the identity needs for a workload, you need to catalog the flows, workload assets, and personas, and the actions the assets and personas will perform. Your strategy must cover all use cases that handle **the flows that reach the workload or its components (outside-in access) and flows that reach out from the workload to other sources (inside-out access)**.

Each use case will probably have its own set of controls that you need to design with an assume-breach mindset. Based on the identity requirements of the use case or the personas, identify the conditional choices. Avoid using one solution for all use cases. Conversely, the controls shouldn't be so granular that you introduce unnecessary management overhead.

You need to log the identity access trail. Doing so helps validate the controls, and you can use the logs for compliance audits.

Determine all identities for authentication

- **Outside-in access.** Your identity design must authenticate all users that access the workload for various purposes. For example, an end user who accesses the application by calling APIs.

At a granular level, components of the workload might also need access from outside. For example, an operator who needs access through the portal or access to the compute to run commands.

Both are examples of **user identities** that have different personas.

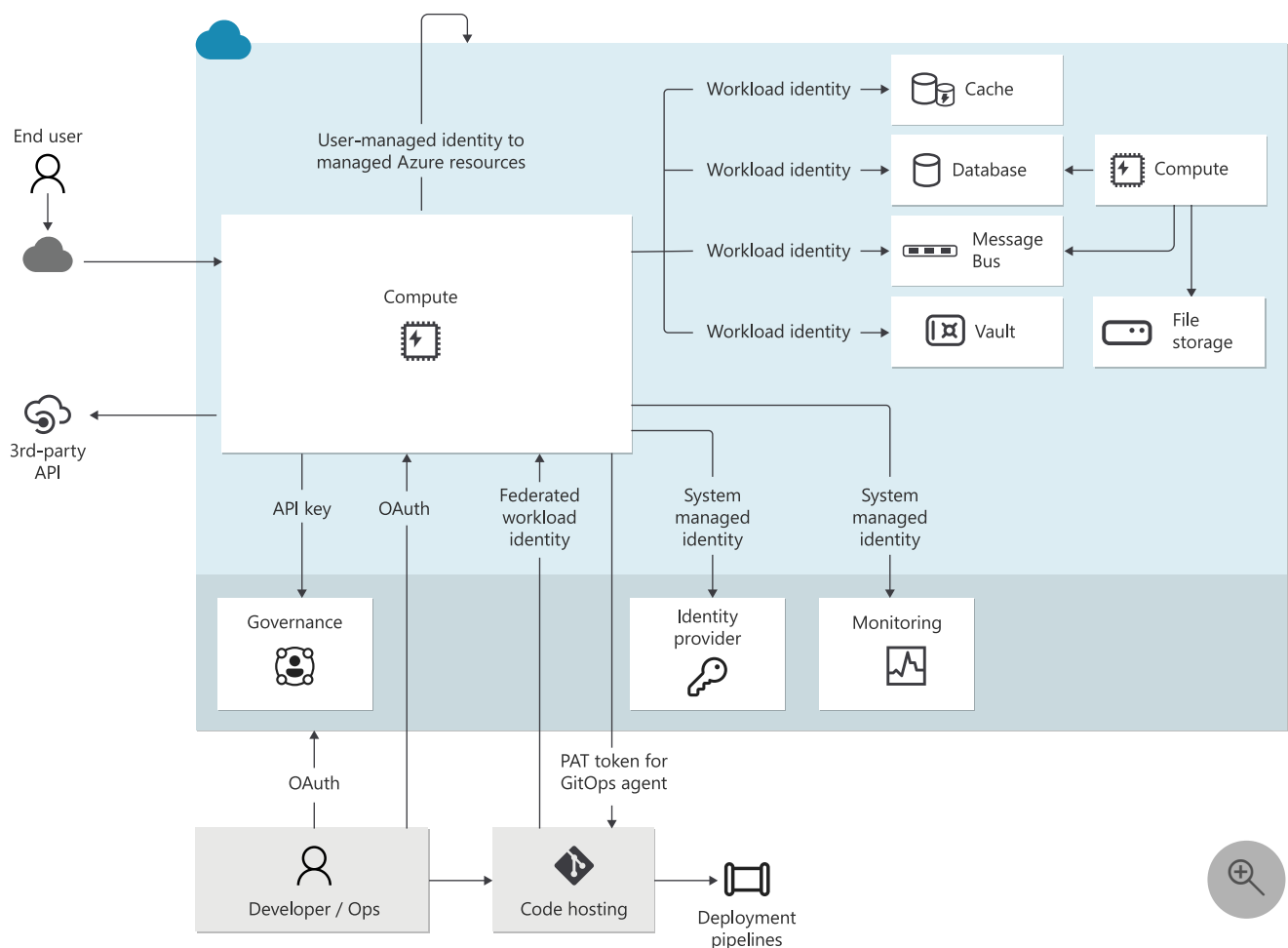
- **Inside-out access.** Your application will need to access other resources. For example, reading from or writing to the data platform, retrieving secrets from the secret store, and

logging telemetry to monitoring services. It might even need to access third-party services. These access needs require **workload identity**, which enables the application to authenticate itself against the other resources.

The concept applies at the component level. In the following example, the container might need access to deployment pipelines to get its configuration. These access needs require **resource identity**.

All these identities should be authenticated by your IdP.

Here's an example of how identity can be implemented in an architecture:



Determine actions for authorization

Next, you need to know what each authenticated identity is trying to do so that those actions can be authorized. The actions can be divided by the type of access that they require:

- **Data plane access.** Actions that take place in the data plane cause data transfer for inside-out or outside-in access. For example, an application reading data from a database and writing data to a database, fetching secrets, or writing logs to a monitoring sink. At the component level, compute that's pulling or pushing images to or from a registry are considered data plane operations.

- **Control plane access.** Actions that take place in the control plane cause an Azure resource to be created, modified, or deleted. For example, changes to resource properties.

Applications typically target data plane operations, while operations often access both control and data planes. To identify authorization needs, note the operational actions that can be performed on the resource. For information about the permitted actions for each resource, see [Azure resource provider operations](#).

Provide role-based authorization

Based on the responsibility of each identity, authorize actions that should be permitted. **An identity must not be allowed to do more than it needs to do.** Before you set authorization rules, you need to have a clear understanding of who or what is making requests, what that role is allowed to do, and to what extent it can do it. Those factors lead to choices that combine identity, role, and scope.

Consider a workload identity as an example. The application must have data plane access to the database, so read and write actions to the data resource must be allowed. However, does the application need control plane access to the secret store? If the workload identity is compromised by a bad actor, what would the impact to the system be, in terms of confidentiality, integrity, and availability?

Role assignment

A role is a *set of permissions* that's assigned to an identity. Assign roles that only allow the identity to complete the task, and no more. When user's permissions are restricted to their job requirements, it's easier to identify suspicious or unauthorized behavior in the system.

Ask questions like these:

- Is read-only access enough?
- Does the identity need permissions to delete resources?

Limiting the level of access that users, applications, or services have to Azure resources reduces the potential attack surface. If you grant only the minimum permissions that are required to perform specific tasks, the risk of a successful attack or unauthorized access is significantly reduced. For example, security teams only need read-only access to security attributes for all technical environments. That level is enough to assess risk factors, identify potential mitigations, and report on the risks.

There are scenarios in which users need more access because of the organizational structure and team organization. There might be an overlap between various roles, or single users might perform multiple standard roles. In this case, use multiple role assignments that are based on the business function instead of creating a custom role for each of these users. Doing so makes the roles easier to manage.

Avoid permissions that specifically reference individual resources or users. Granular and custom permissions create complexity and confusion because they don't pass on the intention to new resources that are similar. This can create a complex legacy configuration that's difficult to maintain and negatively impact both security and reliability.



Tradeoff: A granular access control approach enables better auditing and monitoring of user activities.

A role also has an *associated scope*. The role can operate at the allowed management group, subscription, resource group, or resource scope, or at another custom scope. Even if the identity has a limited set of permissions, widening the scope to include resources that are outside the identity's job function is risky. For example, read access to all source code and data can be dangerous and must be controlled.

You assign roles to identities by using role-based access control (RBAC). **Always use IdP-provided RBAC** to take advantage of features that enable you to apply access control consistently and revoke it rigorously.

Use built-in roles. They're designed to cover most use cases. Custom roles are powerful and sometimes useful, but you should reserve them for scenarios in which built-in roles won't work. Customization leads to complexity that increases confusion and makes automation more complex, challenging, and fragile. These factors all negatively impact security.

Grant roles that start with least privilege and add more based your operational or data access needs. Your technical teams must have clear guidance to implement permissions.

If you want fine-grained control on RBAC, add conditions on the role assignment based on context, such as actions and attributes.

Make conditional access choices

Don't give all identities the same level of access. Base your decisions on two main factors:

- **Time.** How long the identity can access your environment.
- **Privilege.** The level of permissions.

Those factors aren't mutually exclusive. A compromised identity that has more privileges and unlimited duration of access can gain more control over the system and data or use that access to continue to change the environment. Constrain those access factors both as a preventive measure and to control the blast radius.

- *Just in Time (JIT)* approaches provide the required privileges only when they're needed.
- *Just Enough Access (JEA)* provides only the required privileges.

Although time and privilege are the primary factors, there are other conditions that apply. For example, you can also use the device, network, and location from which the access originated to set policies.

Use strong controls that filter, detect, and block unauthorized access, including parameters like user identity and location, device health, workload context, data classification, and anomalies.

For example, your workload might need to be accessed by third-party identities like vendors, partners, and customers. They need the appropriate level of access rather than the default permissions that you provide to full-time employees. Clear differentiation of external accounts makes it easier to prevent and detect attacks that come from these vectors.

Your choice of IdP must be able to provide that differentiation, provide built-in features that grant permissions based on the least privilege, and provide built-in threat intelligence. This includes monitoring of access requests and sign-ins. The Azure IdP is Microsoft Entra ID. For more information, see the [Azure facilitation section](#) of this article.

Protect critical impact accounts

Administrative identities introduce some of the highest impact security risks because the tasks they perform require privileged access to a broad set of these systems and applications. Compromise or misuse can have a detrimental effect on your business and its information systems. Security of administration is one of the most critical security areas.

Protecting privileged access against determined adversaries requires you to take a complete and thoughtful approach to isolate these systems from risks. Here are some strategies:

- **Minimize the number of critical impact accounts.**
- **Use separate roles** instead of elevating privileges for existing identities.
- **Avoid permanent or standing access** by using the JIT features of your IdP. For break glass situations, follow an emergency access process.

- **Use modern access protocols** like passwordless authentication or multifactor authentication. Externalize those mechanisms to your IdP.
- Enforce key security attributes by using **conditional access policies**.
- **Decommission administrative accounts** that aren't being used.

Use a single identity across environments and associate a single identity with the user or principal. Consistency of identities across cloud and on-premises environments reduces human errors and the resulting security risks. Teams in both environments that manage resources need a consistent, authoritative source in order to meet security assurances. Work with your central identity team to ensure that identities in hybrid environments are synchronized.



Risk: There's a risk associated with synchronizing high privilege identities. An attacker can get full control of on-premises assets, and this can lead to a successful compromise of a cloud account. Evaluate your synchronization strategy by filtering out accounts that can add to the attack surface.

Establish processes to manage the identity lifecycle

Access to identities must not last longer than the resources that the identities access. Ensure that you have a process for disabling or deleting identities when there are changes in team structure or software components.

This guidance applies to source control, data, control planes, workload users, infrastructure, tooling, the monitoring of data, logs, metrics, and other entities.

Establish an identity governance process to manage the lifecycle of digital identities, high-privileged users, external/guest users, and workload users. Implement access reviews to ensure that when identities leave the organization or the team, their workload permissions are removed.

Protect nonidentity based secrets

Application secrets like preshared keys should be considered vulnerable points in the system. In the two-way communication, if the provider or consumer is compromised, significant security risks can be introduced. Those keys can also be burdensome because they introduce operational processes.

When you can, avoid using secrets and consider using identity-based authentication for user access to the application itself, not just to its resources.

The following list provides a summary of guidance. For more information, see [Recommendations for application secrets](#).

- Treat these secrets as entities that can be dynamically pulled from a secret store. They shouldn't be hard coded in your application code, IaC scripts, deployment pipelines, or in any other artifact.
- Be sure that you have the **ability to revoke secrets**.
- Apply operational practices that handle tasks like **key rotation and expiration**.

For information about rotation policies, see [Automate the rotation of a secret for resources that have two sets of authentication credentials](#) and [Tutorial: Updating certificate auto-rotation frequency in Key Vault](#).

Keep development environments safe

All code and scripts, pipeline tooling, and source control systems should be considered workload assets. **Access to writes should be gated** with automation and peer review. **Read access to source code should be limited** to roles on a need-to-know basis. Code repositories must have versioning, and **security code reviews** by peers must be a regular practice that's integrated with the development lifecycle. You need to have a process in place that **scans resources regularly** and identifies the latest vulnerabilities.

Use workload identities to grant access to resources from deployment environments, such as GitHub.

Maintain an audit trail

One aspect of identity management is ensuring that the system is auditable. Audits validate whether assume-breach strategies are effective. Maintaining an audit trail helps you:

- Verify that identity is authenticated with strong authentication. **Any action must be traceable** to prevent repudiation attacks.
- **Detect weak or missing authentication protocols** and get visibility into and insights about user and application sign-ins.
- Evaluate access from identities to the workload based on security and **compliance requirements** and consider user account risk, device status, and other criteria and policies that you set.
- **Track progress or deviation** from compliance requirements.

Most resources have data plane access. You need to know the identities that access resources and the actions that they perform. You can use that information for security diagnostics.

For more information, see [Recommendations on security monitoring and threat analysis](#).

Azure facilitation

We recommend that you always use modern authentication protocols that take into account all available data points and use conditional access. **Microsoft Entra ID provides identity and access management in Azure.** It covers the management plane of Azure and is integrated with the data planes of most Azure services. Microsoft Entra ID is the tenant that's associated with the workload subscription. It tracks and manages identities and their allowed permissions and simplifies overall management to minimize the risk of oversight or human error.

These capabilities natively integrate into the same Microsoft Entra identity and permission model for user segments:

- [Microsoft Entra ID](#). Employees and enterprise resources.
- [Microsoft Entra External ID](#). Partners.
- [Microsoft Entra federation compatibility list](#). Third-party federation solutions.

You can use Microsoft Entra ID for authentication and authorization of custom applications via Microsoft Authentication Library (MSAL) or platform features, like authentication for web apps. It covers the management plane of Azure, the data planes of most of Azure services, and integration capabilities for your applications.

You can stay current by visiting [What's new in Microsoft Entra ID](#).



Tradeoff: Microsoft Entra ID is a single point of failure just like any other foundational service. There's no workaround until the outage is fixed by Microsoft. However, the rich feature set of Microsoft Entra outweighs the risk of using custom solutions.

Azure supports open protocols like OAuth2 and OpenID Connect. We recommend that you use these standard authentication and authorization mechanisms instead of designing your own flows.

Azure RBAC

Azure RBAC represents security principals in Microsoft Entra ID. All role assignments are done via Azure RBAC. Take advantage of built-in roles that provide most of the permissions that you

need. For more information, see [Microsoft Entra built-in roles](#).

Here are some use cases:

- By assigning users to roles, you can control access to Azure resources. For more information, see [Overview of role-based access control in Microsoft Entra ID](#).
- You can use Privileged Identity Management to provide time-based and approval-based role activation for roles that are associated with high-impact identities. For more information, see [What is Privileged Identity Management?](#).

For more information about RBAC, see [Best practices for Azure RBAC](#).

For information about attribute-based controls, see [What is Azure ABAC?](#).

Workload identity

Microsoft Entra ID can handle your application's identity. The service principal that's associated with the application can dictate its access scope.

For more information, see [What are workload identities?](#).

The service principal is also abstracted when you use a managed identity. The advantage is that Azure manages all credentials for the application.

Not all services support managed identities. If you can't use managed identities, you can use service principals. However, using service principals increases your management overhead. For more information, see [What are managed identities for Azure resources?](#).

Resource identity

The concept of **managed identities can be extended to Azure resources**. Azure resources can use managed identities to authenticate themselves to other services that support Microsoft Entra authentication. For more information, see [Azure services that can use managed identities to access other services](#).

Conditional access policies

Conditional access describes your policy for an access decision. To use conditional access, you need to understand the restrictions that are required for the use case. Configure Microsoft Entra Conditional Access by setting up an access policy for that's based on your operational needs.

For more information, see [Conditional access: Users, groups, and workload identities](#).

Group access management

Instead of granting permissions to specific users, assign access to groups in Microsoft Entra ID. If a group doesn't exist, work with your identity team to create one. You can then add and remove group members outside of Azure and make sure that permissions are current. You can also use the group for other purposes, like mailing lists.

For more information, see [Secure access control using groups in Microsoft Entra ID](#).

Threat detection

Microsoft Entra ID Protection can help you detect, investigate, and remediate identity-based risks. For more information, see [What is Identity Protection?](#).

Threat detection can take the form of reacting to an alert of suspicious activity or proactively searching for anomalous events in activity logs. User and Entity Behavior Analytics (UEBA) in Microsoft Sentinel makes it easy to detect suspicious activities. For more information, see [Identify advanced threats with UEBA](#).

Hybrid systems

On Azure, **don't synchronize accounts to Microsoft Entra ID that have high privileges in your existing Active Directory**. This synchronization is blocked in the default Microsoft Entra Connect Sync configuration, so you only need to confirm that you haven't customized this configuration.

For information about filtering in Microsoft Entra ID, see [Microsoft Entra Connect Sync: Configure filtering](#).

Identity logging

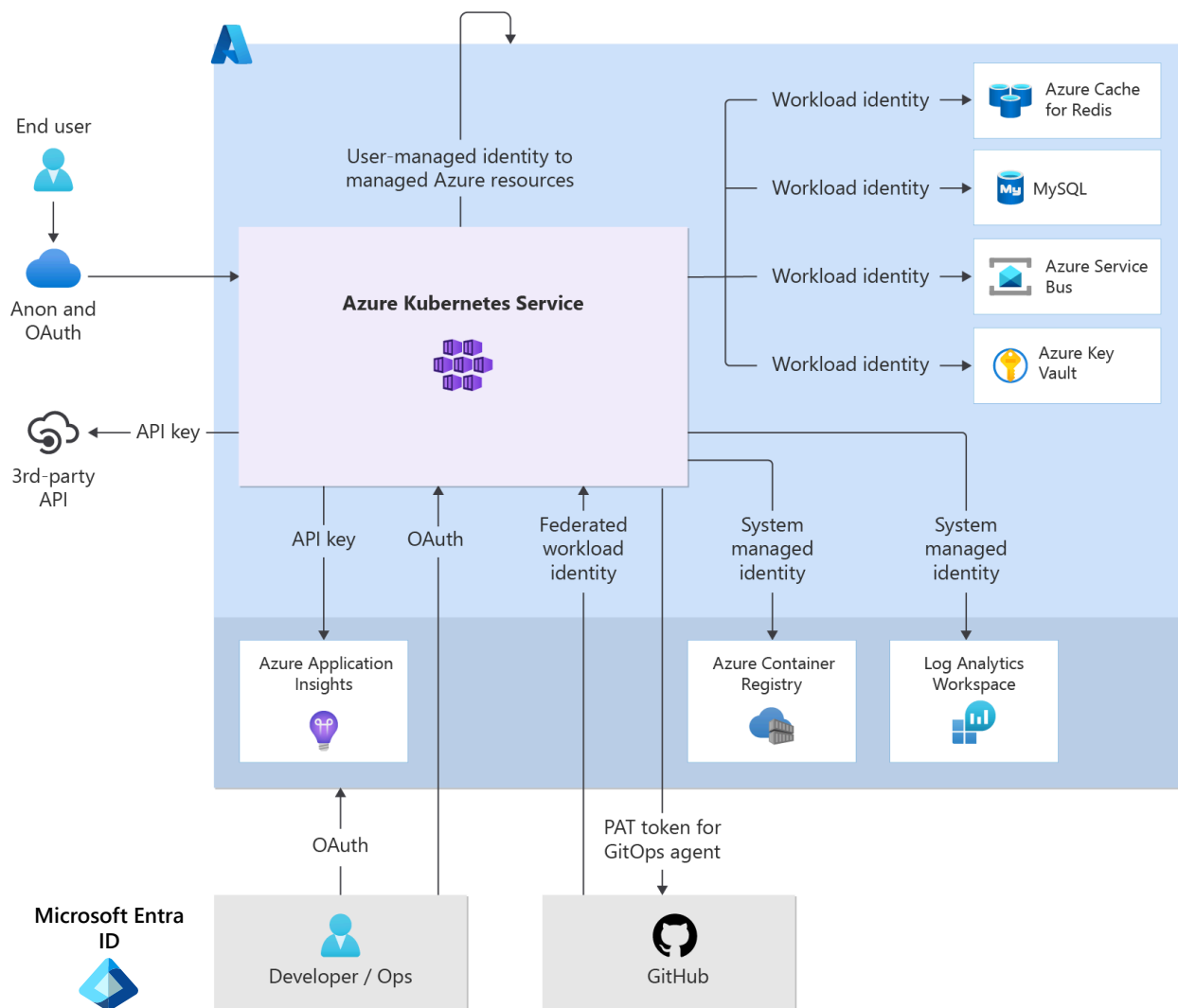
Enable diagnostic settings on Azure resources to emit information that you can use as an audit trail. The diagnostic information shows which identities attempt to access which resources and the outcome of those attempts. The collected logs are sent to Azure Monitor.



Tradeoff: Logging incurs costs because of the data storage that's used to store the logs. It also might cause a performance impact, especially on the code and on logging solutions that you add to the application.

Example

The following example shows an identity implementation. Different types of identities are used together to provide the required levels of access.



Identity components

- **System-managed identities.** Microsoft Entra ID provides access to service data planes that don't face users, like Azure Key Vault and data stores. These identities also control access, via RBAC, to the Azure management plane for workload components, deployment agents, and team members.
- **Workload identities.** The application services in the Azure Kubernetes Service (AKS) cluster use workload identities to authenticate themselves to other components in the solution.

- **Managed identities.** System components in the client role use system-managed identities, including build agents.
- **Human identities.** User and operator authentication is delegated to Microsoft Entra ID or Microsoft Entra ID (native, B2B, or B2C).

The security of preshared secrets is critical for any application. Azure Key Vault provides a secure storage mechanism for these secrets, including Redis and third-party secrets.

A rotation mechanism is used to help ensure that secrets aren't compromised. Tokens for the Microsoft identity platform implementation of OAuth 2 and OpenID Connect are used to authenticate users.

Azure Policy is used to ensure that identity components like Key Vault use RBAC instead of access policies. JIT and JEA provide traditional standing permissions for human operators.

Access logs are enabled across all components via Azure Diagnostics, or via code for code components.

Related links

- [Tutorial: Automate the rotation of a secret for resources that have two sets of authentication credentials](#)
- [Tutorial: Updating certificate auto-rotation frequency in Key Vault](#)
- [What's new in Microsoft Entra ID?](#)
- [Microsoft Entra built-in roles](#)
- [Overview of role-based access control in Microsoft Entra ID](#)
- [What are workload identities?](#)
- [What are managed identities for Azure resources?](#)
- [Conditional access: Users, groups, and workload identities](#)
- [Microsoft Entra Connect Sync: Configure filtering](#)

Security checklist

Refer to the complete set of recommendations.

Security checklist

Azure network security overview

Network security protects resources from unauthorized access or attack by controlling network traffic. Azure provides a robust networking infrastructure to support your application and service connectivity requirements, with security controls at every layer.

This article covers key network security capabilities in Azure:

- Network access control
- Azure Firewall
- Secure remote access and cross-premises connectivity
- Availability and load balancing
- Name resolution
- DDoS protection
- Azure Front Door
- Monitoring and threat detection

ⓘ Note

For web workloads, we recommend using [Azure DDoS Protection](#) and a [web application firewall](#) to protect against DDoS attacks. [Azure Front Door](#) with a web application firewall provides platform-level protection against network-level DDoS attacks.

Azure Virtual Network

Azure Virtual Network is the fundamental building block for your private network in Azure. Each virtual network is isolated from other virtual networks, helping ensure that network traffic in your deployments isn't accessible to other Azure customers. Virtual networks enable Azure resources to securely communicate with each other, the internet, and on-premises networks.

Learn more:

- [Azure Virtual Network overview](#)
- [Plan virtual networks](#)

Network access control

Network access control limits connectivity to and from specific devices or subnets within a virtual network. The goal is to restrict access to your virtual machines and services to approved users and devices.

Network Security Groups

Network Security Groups (NSGs) provide basic, stateful packet filtering based on IP address and TCP/UDP protocols. NSGs control access using a 5-tuple (source IP, source port, destination IP, destination port, protocol).

NSGs include features to simplify management:

- **Augmented security rules:** Create complex rules instead of multiple simple rules to achieve the same result
- **Service tags:** Microsoft-managed labels representing groups of IP addresses that update dynamically
- **Application security groups:** Organize resources into application groups and control access based on those groups

Learn more:

- [Network Security Groups](#)
- [Network security best practices](#)

Service endpoints

Virtual Network service endpoints extend your virtual network private address space to Azure services over a direct connection. Service endpoints keep traffic on the Azure backbone network and restrict communication with supported services to your virtual networks only.

Learn more:

- [Virtual Network service endpoints](#)

Azure Private Link

Azure Private Link provides private connectivity from a virtual network to Azure PaaS services, customer-owned services, or Microsoft partner services. Private Link traffic remains on the Microsoft Azure backbone network, eliminating exposure to the public internet.

Learn more:

- [What is Azure Private Link?](#)
- [Private endpoints](#)

Azure Firewall

Azure Firewall is a cloud-native, intelligent network firewall security service providing threat protection for cloud workloads. It's a fully stateful firewall as a service with built-in high availability and unrestricted cloud scalability.

Azure Firewall is available in three SKUs:

- **Azure Firewall Basic:** Simplified security for small and medium-sized businesses
- **Azure Firewall Standard:** L3-L7 filtering and threat intelligence from Microsoft Cyber Security
- **Azure Firewall Premium:** Advanced capabilities including signature-based IDPS for rapid attack detection

Learn more:

- [What is Azure Firewall?](#)
- [Choose the right Azure Firewall SKU](#)
- [Threat detection and protection overview](#)

Secure remote access and cross-premises connectivity

Azure supports several secure remote access scenarios for managing Azure resources and deploying hybrid IT solutions.

Point-to-site VPN

Point-to-site VPN connections enable individual users to establish private, secure connections to a virtual network. Users can access virtual machines and services in Azure after authenticating. Point-to-site VPN supports:

- **Secure Socket Tunneling Protocol (SSTP):** Proprietary SSL-based VPN protocol (Windows devices)
- **IKEv2 VPN:** Standards-based IPsec VPN solution (Mac devices)
- **OpenVPN Protocol:** SSL/TLS-based VPN protocol (Android, iOS, Windows, Linux, and Mac devices)

Learn more:

- [About point-to-site VPN](#)
- [Configure a point-to-site VPN connection](#)

Site-to-site VPN

Site-to-site VPN Gateway connections establish secure cross-premises connectivity between your on-premises network and Azure virtual networks. Site-to-site VPNs use the highly secure IPsec tunnel mode VPN protocol.

VPN Gateway is essential for hybrid IT scenarios where parts of a service are hosted both in Azure and on-premises.

Learn more:

- [About VPN Gateway](#)
- [Create a site-to-site connection](#)

ExpressRoute

ExpressRoute provides dedicated WAN links between your on-premises network and Microsoft cloud services. ExpressRoute connections don't traverse the public internet, offering enhanced security, reliability, speed, and lower latency compared to internet connections.

ExpressRoute supports:

- **ExpressRoute Direct:** Direct connectivity to Microsoft global network
- **ExpressRoute Global Reach:** Connectivity between your on-premises sites through ExpressRoute circuits

Learn more:

- [ExpressRoute overview](#)
- [ExpressRoute connectivity models](#)

VNet peering

Virtual Network peering connects two Azure virtual networks, enabling resources in either network to communicate with each other. VNet peering uses the Microsoft backbone infrastructure, bypassing the public internet. Peering supports connections within the same Azure region or across different regions (global VNet peering).

Learn more:

- [Virtual network peering](#)
- [Create a VNet peering](#)

Availability and load balancing

Load balancing distributes connections across multiple devices to increase availability and performance. Azure provides several load balancing options.

Azure Load Balancer

Azure Load Balancer provides high-performance, low-latency Layer 4 load balancing for all UDP and TCP protocols. Load Balancer distributes inbound traffic to backend instances according to configured rules and health probes.

Load Balancer features include:

- Support for internal and external load-balancing scenarios
- Zone redundancy and zonal deployments
- TCP and UDP application support
- Health probes to determine backend instance availability

Learn more:

- [What is Azure Load Balancer?](#)
- [Standard Load Balancer and availability zones](#)

Azure Application Gateway

Azure Application Gateway is a web traffic load balancer (Layer 7) that manages traffic to your web applications. Application Gateway makes routing decisions based on HTTP request attributes such as URI path or host headers.

Application Gateway features include:

- Web Application Firewall (WAF) for centralized protection
- TLS termination to reduce encryption overhead on web servers
- Cookie-based session affinity
- URL-based content routing
- Autoscaling and zone redundancy

Learn more:

- [What is Azure Application Gateway?](#)
- [Application Gateway components](#)

Azure Traffic Manager

Azure Traffic Manager is a DNS-based traffic load balancer that distributes traffic optimally to services across global Azure regions. Traffic Manager provides high availability and responsiveness by routing client requests to the most appropriate service endpoint based on traffic-routing method and endpoint health.

Traffic Manager supports multiple routing methods including priority, weighted, performance, geographic, multivalue, and subnet routing.

Learn more:

- [What is Traffic Manager?](#)
- [Traffic Manager routing methods](#)

Name resolution

Secure name resolution is crucial for all cloud-hosted services. Compromised name resolution functions can redirect requests to malicious sites.

Azure DNS

Azure DNS provides highly available and performant name resolution using Microsoft Azure infrastructure. Azure DNS supports:

- Public DNS domains hosted on Azure global infrastructure
- Private DNS zones for name resolution within and across virtual networks
- Split-horizon DNS scenarios where the same domain name resolves differently for private and public queries

Learn more:

- [Azure DNS overview](#)
- [Azure Private DNS zones](#)

DDoS protection

Distributed denial of service (DDoS) attacks are among the largest availability and security concerns for customers moving applications to the cloud. Azure DDoS Protection safeguards Azure resources from DDoS attacks.

Azure DDoS Protection SKUs:

- **DDoS Infrastructure Protection:** Basic protection enabled by default on all Azure properties at no additional cost

- **DDoS Network Protection:** Advanced protection for resources in virtual networks with adaptive tuning, mitigation policies, and monitoring

DDoS Network Protection features include:

- Native platform integration with configuration through the Azure portal
- Always-on traffic monitoring and real-time mitigation
- Attack analytics including mitigation reports and flow logs
- Adaptive tuning based on application traffic patterns
- Cost guarantee including data transfer and application scale-out service credits

Learn more:

- [Azure DDoS Protection overview](#)
- [Manage DDoS Protection](#)

Azure Front Door

Azure Front Door is a global, scalable entry point that uses the Microsoft global edge network to create fast, secure, and widely scalable web applications. Front Door provides Layer 7 load balancing, TLS termination, URL-based routing, and integrated security.

Front Door capabilities include:

- Global HTTP load balancing with instant failover
- TLS termination at the edge
- URL path-based routing
- Cookie-based session affinity
- Web Application Firewall protection
- Platform-level DDoS protection
- Private Link integration to protect backend origins

Learn more:

- [What is Azure Front Door?](#)
- [Front Door routing architecture](#)

Monitoring and threat detection

Azure provides tools to monitor network security and detect threats.

Azure Network Watcher

Azure Network Watcher provides tools to monitor, diagnose, and gain insights into your network in Azure.

Network Watcher capabilities include:

- **Connection Monitor:** Monitors connectivity between Azure resources and endpoints
- **NSG flow logs:** Logs information about IP traffic flowing through Network Security Groups
- **Packet capture:** Captures network traffic to and from virtual machines
- **VPN troubleshoot:** Diagnoses issues with VPN Gateways and connections
- **Network diagnostics:** Validates network configuration and identifies security issues

Learn more:

- [What is Azure Network Watcher?](#)
- [Network Watcher monitoring overview](#)

Microsoft Defender for Cloud

Microsoft Defender for Cloud helps you prevent, detect, and respond to threats with increased visibility and control over the security of your Azure resources. Defender for Cloud provides network security recommendations, monitors network security configuration, and alerts you to network-based threats.

Learn more:

- [Introduction to Microsoft Defender for Cloud](#)
- [Protect your network resources](#)
- [Threat detection overview](#)

Next steps

- [Azure security best practices and patterns](#)
- [Network security best practices](#)
- [Identity management security overview](#)
- [Threat detection and protection](#)

Azure best practices for network security

This article discusses a collection of Azure best practices to enhance your network security. These best practices are derived from our experience with Azure networking, and the experiences of customers like yourself.

For each best practice, this article explains:

- What the best practice is
- Why you want to enable that best practice
- What might be the result if you fail to enable the best practice
- Possible alternatives to the best practice
- How you can learn to enable the best practice

These best practices are based on a consensus opinion, and Azure platform capabilities and feature sets, as they exist at the time this article was written. Opinions and technologies change over time and this article will be updated regularly to reflect those changes.

Use strong network controls

You can connect [Azure virtual machines \(VMs\)](#) [↗] and appliances to other networked devices by placing them on [Azure virtual networks](#). That is, you can connect virtual network interface cards to a virtual network to allow TCP/IP-based communications between network-enabled devices. Virtual machines connected to an Azure virtual network can connect to devices on the same virtual network, different virtual networks, the internet, or your own on-premises networks.

As you plan your network and the security of your network, we recommend that you centralize:

- Management of core network functions like ExpressRoute, virtual network and subnet provisioning, and IP addressing.
- Governance of network security elements, such as network virtual appliance functions like ExpressRoute, virtual network and subnet provisioning, and IP addressing.

If you use a common set of management tools to monitor your network and the security of your network, you get clear visibility into both. A straightforward, unified security strategy reduces errors because it increases human understanding and the reliability of automation.

Logically segment subnets

Azure virtual networks are similar to LANs on your on-premises network. The idea behind an Azure virtual network is that you create a network, based on a single private IP address space,

on which you can place all your Azure virtual machines. The private IP address spaces available are in the Class A (10.0.0.0/8), Class B (172.16.0.0/12), and Class C (192.168.0.0/16) ranges.

Best practices for logically segmenting subnets include:

Best practice: Don't assign allow rules with broad ranges (for example, allow 0.0.0.0 through 255.255.255.255).

Detail: Ensure troubleshooting procedures discourage or ban setting up these types of rules. These allow rules lead to a false sense of security and are frequently found and exploited by red teams.

Best practice: Segment the larger address space into subnets.

Detail: Use [CIDR](#) -based subnetting principles to create your subnets.

Best practice: Create network access controls between subnets. Routing between subnets happens automatically, and you don't need to manually configure routing tables. By default, there are no network access controls between the subnets that you create on an Azure virtual network.

Detail: Use a [network security group](#) to protect against unsolicited traffic into Azure subnets. Network security groups (NSGs) are simple, stateful packet inspection devices. NSGs use the 5-tuple approach (source IP, source port, destination IP, destination port and protocol) to create allow/deny rules for network traffic. You allow or deny traffic to and from a single IP address, to and from multiple IP addresses, or to and from entire subnets.

When you use network security groups for network access control between subnets, you can put resources that belong to the same security zone or role in their own subnets.

Best practice: Avoid small virtual networks and subnets to ensure simplicity and flexibility.

Detail: Most organizations add more resources than initially planned, and reallocating addresses is labor intensive. Using small subnets adds limited security value, and mapping a network security group to each subnet adds overhead. Define subnets broadly to ensure that you have flexibility for growth.

Best practice: Simplify network security group rule management by defining [Application Security Groups](#).

Detail: Define an Application Security Group for lists of IP addresses that you think might change in the future or be used across many network security groups. Be sure to name Application Security Groups clearly so others can understand their content and purpose.

Adopt a Zero Trust approach

Perimeter-based networks operate on the assumption that all systems within a network can be trusted. But today's employees access their organization's resources from anywhere on various

devices and apps, which makes perimeter security controls irrelevant. Access control policies that focus only on who can access a resource aren't enough. To master the balance between security and productivity, security admins also need to factor in *how* a resource is being accessed.

Networks need to evolve from traditional defenses because networks might be vulnerable to breaches: an attacker can compromise a single endpoint within the trusted boundary and then quickly expand a foothold across the entire network. [Zero Trust](#) networks eliminate the concept of trust based on network location within a perimeter. Instead, Zero Trust architectures use device and user trust claims to gate access to organizational data and resources. For new initiatives, adopt Zero Trust approaches that validate trust at the time of access.

Best practices are:

Best practice: Give Conditional Access to resources based on device, identity, assurance, network location, and more.

Detail: [Microsoft Entra Conditional Access](#) lets you apply the right access controls by implementing automated access control decisions based on the required conditions. For more information, see [Manage access to Azure management with Conditional Access](#).

Best practice: Enable port access only after workflow approval.

Detail: You can use [just-in-time VM access in Microsoft Defender for Cloud](#) to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy access to connect to VMs when needed.

Best practice: Grant temporary permissions to perform privileged tasks, which prevents malicious or unauthorized users from gaining access after the permissions have expired. Access is granted only when users need it.

Detail: Use just-in-time access in Microsoft Entra Privileged Identity Management or in a third-party solution to grant permissions to perform privileged tasks.

Zero Trust is the next evolution in network security. The state of cyberattacks drives organizations to take the "assume breach" mindset, but this approach shouldn't be limiting. Zero Trust networks protect corporate data and resources while ensuring that organizations can build a modern workplace by using technologies that empower employees to be productive anytime, anywhere, in any way.

Control routing behavior

When you put a virtual machine on an Azure virtual network, the VM can connect to any other VM on the same virtual network, even if the other VMs are on different subnets. This is possible because a collection of system routes enabled by default allows this type of communication.

These default routes allow VMs on the same virtual network to initiate connections with each other, and with the internet (for outbound communications to the internet only).

Although the default system routes are useful for many deployment scenarios, there are times when you want to customize the routing configuration for your deployments. You can configure the next-hop address to reach specific destinations.

We recommend that you configure [user-defined routes](#) when you deploy a security appliance for a virtual network. We talk about this recommendation in a later section titled [secure your critical Azure service resources to only your virtual networks](#).

 Note

User-defined routes aren't required, and the default system routes usually work.

Use virtual network appliances

Network security groups and user-defined routing can provide a certain measure of network security at the network and transport layers of the [OSI model](#)[↗]. But in some situations, you want or need to enable security at high levels of the stack. In such situations, we recommend that you deploy virtual network security appliances provided by Azure partners.

Azure network security appliances can deliver better security than what network-level controls provide. Network security capabilities of virtual network security appliances include:

- Firewalling
- Intrusion detection/intrusion prevention
- Vulnerability management
- Application control
- Network-based anomaly detection
- Web filtering
- Antivirus
- Botnet protection

To find available Azure virtual network security appliances, go to the [Azure Marketplace](#)[↗] and search for "security" and "network security."

Deploy perimeter networks for security zones

A [perimeter network](#) (also known as a DMZ) is a physical or logical network segment that provides an extra layer of security between your assets and the internet. Specialized network

access control devices on the edge of a perimeter network allow only desired traffic into your virtual network.

Perimeter networks are useful because you can focus your network access control management, monitoring, logging, and reporting on the devices at the edge of your Azure virtual network. A perimeter network is where you typically enable [distributed denial of service \(DDoS\) protection](#), intrusion detection/intrusion prevention systems (IDS/IPS), firewall rules and policies, web filtering, network antimalware, and more. The network security devices sit between the internet and your Azure virtual network and have an interface on both networks.

Although this is the basic design of a perimeter network, there are many different designs, like back-to-back, tri-homed, and multi-homed.

Based on the Zero Trust concept mentioned earlier, we recommend that you consider using a perimeter network for all high security deployments to enhance the level of network security and access control for your Azure resources. You can use Azure or a third-party solution to provide an extra layer of security between your assets and the internet:

- Azure native controls. [Azure Firewall](#) and [Azure Web Application Firewall](#) offer basic security advantages. Advantages are a fully stateful firewall as a service, built-in high availability, unrestricted cloud scalability, FQDN filtering, support for OWASP core rule sets, and simple setup and configuration.
- Third-party offerings. Search the [Azure Marketplace](#) [↗](#) for next-generation firewall (NGFW) and other third-party offerings that provide familiar security tools and enhanced levels of network security. Configuration might be more complex, but a third-party offering might allow you to use existing capabilities and skillsets.

Avoid exposure to the internet with dedicated WAN links

Many organizations have chosen the hybrid IT route. With hybrid IT, some of the company's information assets are in Azure, and others remain on-premises. In many cases, some components of a service are running in Azure while other components remain on-premises.

In a hybrid IT scenario, there's usually some type of cross-premises connectivity. Cross-premises connectivity allows the company to connect its on-premises networks to Azure virtual networks. Two cross-premises connectivity solutions are available:

- [Site-to-site VPN](#). It's a trusted, reliable, and established technology, but the connection takes place over the internet. Bandwidth is constrained to a maximum of about 1.25 Gbps. Site-to-site VPN is a desirable option in some scenarios.

- **Azure ExpressRoute.** We recommend that you use [ExpressRoute](#) for your cross-premises connectivity. ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services like Azure, Microsoft 365, and Dynamics 365. ExpressRoute is a dedicated WAN link between your on-premises location or a Microsoft Exchange hosting provider. Because this is a telco connection, your data doesn't travel over the internet, so it isn't exposed to the potential risks of internet communications.

The location of your ExpressRoute connection can affect firewall capacity, scalability, reliability, and network traffic visibility. You'll need to identify where to terminate ExpressRoute in existing (on-premises) networks. You can:

- Terminate outside the firewall (the perimeter network paradigm). Use this recommendation if you require visibility into the traffic, if you need to continue an existing practice of isolating datacenters, or if you're solely putting extranet resources on Azure.
- Terminate inside the firewall (the network extension paradigm). This is the default recommendation. In all other cases, we recommend treating Azure as another datacenter.

Optimize uptime and performance

If a service is down, information can't be accessed. If performance is so poor that the data is unusable, you can consider the data to be inaccessible. From a security perspective, you need to do whatever you can to make sure that your services have optimal uptime and performance.

A popular and effective method for enhancing availability and performance is load balancing. Load balancing is a method of distributing network traffic across servers that are part of a service. For example, if you have front-end web servers as part of your service, you can use load balancing to distribute the traffic across your multiple front-end web servers.

This distribution of traffic increases availability because if one of the web servers becomes unavailable, the load balancer stops sending traffic to that server and redirects it to the servers that are still online. Load balancing also helps performance, because the processor, network, and memory overhead for serving requests is distributed across all the load-balanced servers.

We recommend that you employ load balancing whenever you can, and as appropriate for your services. Following are scenarios at both the Azure virtual network level and the global level, along with load-balancing options for each.

Scenario: You have an application that:

- Requires requests from the same user/client session to reach the same back-end virtual machine. Examples of this are shopping cart apps and web mail servers.
- Accepts only a secure connection, so unencrypted communication to the server isn't an acceptable option.
- Requires multiple HTTP requests on the same long-running TCP connection to be routed or load balanced to different back-end servers.

Load-balancing option: Use [Azure Application Gateway](#), an HTTP web traffic load balancer. Application Gateway supports end-to-end TLS encryption and [TLS termination](#) at the gateway. Web servers can then be unburdened from encryption and decryption overhead and traffic flowing unencrypted to the back-end servers.

Scenario: You need to load balance incoming connections from the internet among your servers located in an Azure virtual network. Scenarios are when you:

- Have stateless applications that accept incoming requests from the internet.
- Don't require sticky sessions or TLS offload. Sticky sessions is a method used with Application Load Balancing, to achieve server-affinity.

Load-balancing option: Use the Azure portal to [create an external load balancer](#) that spreads incoming requests across multiple VMs to provide a higher level of availability.

Scenario: You need to load balance connections from VMs that are not on the internet. In most cases, the connections that are accepted for load balancing are initiated by devices on an Azure virtual network, such as SQL Server instances or internal web servers.

Load-balancing option: Use the Azure portal to [create an internal load balancer](#) that spreads incoming requests across multiple VMs to provide a higher level of availability.

Scenario: You need global load balancing because you:

- Have a cloud solution that is widely distributed across multiple regions and requires the highest level of uptime (availability) possible.
- Need the highest level of uptime possible to make sure that your service is available even if an entire datacenter becomes unavailable.

Load-balancing option: Use Azure Traffic Manager. Traffic Manager makes it possible to load balance connections to your services based on the location of the user.

For example, if the user makes a request to your service from the EU, the connection is directed to your services located in an EU datacenter. This part of Traffic Manager global load balancing helps to improve performance because connecting to the nearest datacenter is faster than connecting to datacenters that are far away.

Disable RDP/SSH Access to virtual machines

It's possible to reach Azure virtual machines by using [Remote Desktop Protocol \(RDP\)](#) and the [Secure Shell](#) (SSH) protocol. These protocols enable the management VMs from remote locations and are standard in datacenter computing.

The potential security problem with using these protocols over the internet is that attackers can use [brute force](#) techniques to gain access to Azure virtual machines. After the attackers gain access, they can use your VM as a launch point for compromising other machines on your virtual network or even attack networked devices outside Azure.

We recommend that you disable direct RDP and SSH access to your Azure virtual machines from the internet. After direct RDP and SSH access from the internet is disabled, you have other options that you can use to access these VMs for remote management.

Scenario: Enable a single user to connect to an Azure virtual network over the internet.

Option: [Point-to-site VPN](#) is another term for a remote access VPN client/server connection. After the point-to-site connection is established, the user can use RDP or SSH to connect to any VMs located on the Azure virtual network that the user connected to via point-to-site VPN. This assumes that the user is authorized to reach those VMs.

Point-to-site VPN is more secure than direct RDP or SSH connections because the user has to authenticate twice before connecting to a VM. First, the user needs to authenticate (and be authorized) to establish the point-to-site VPN connection. Second, the user needs to authenticate (and be authorized) to establish the RDP or SSH session.

Scenario: Enable users on your on-premises network to connect to VMs on your Azure virtual network.

Option: A [site-to-site VPN](#) connects an entire network to another network over the internet. You can use a site-to-site VPN to connect your on-premises network to an Azure virtual network. Users on your on-premises network connect by using the RDP or SSH protocol over the site-to-site VPN connection. You don't have to allow direct RDP or SSH access over the internet.

Scenario: Use a dedicated WAN link to provide functionality similar to the site-to-site VPN.

Option: Use [ExpressRoute](#). It provides functionality similar to the site-to-site VPN. The main differences are:

- The dedicated WAN link doesn't traverse the internet.
- Dedicated WAN links are typically more stable and perform better.

Secure your critical Azure service resources to only your virtual networks

Use Azure Private Link to access Azure PaaS Services (for example, Azure Storage and SQL Database) over a private endpoint in your virtual network. Private Endpoints allow you to secure your critical Azure service resources to only your virtual networks. Traffic from your virtual network to the Azure service always remains on the Microsoft Azure backbone network. Exposing your virtual network to the public internet is no longer necessary to consume Azure PaaS Services.

Azure Private Link provides the following benefits:

- **Improved security for your Azure service resources:** With Azure Private Link, Azure service resources can be secured to your virtual network using private endpoint. Securing service resources to a private endpoint in virtual network provides improved security by fully removing public internet access to resources, and allowing traffic only from private endpoint in your virtual network.
- **Privately access Azure service resources on the Azure platform:** Connect your virtual network to services in Azure using private endpoints. There's no need for a public IP address. The Private Link platform will handle the connectivity between the consumer and services over the Azure backbone network.
- **Access from On-premises and peered networks:** Access services running in Azure from on-premises over ExpressRoute private peering, VPN tunnels, and peered virtual networks using private endpoints. There's no need to configure ExpressRoute Microsoft peering or traverse the internet to reach the service. Private Link provides a secure way to migrate workloads to Azure.
- **Protection against data leakage:** A private endpoint is mapped to an instance of a PaaS resource instead of the entire service. Consumers can only connect to the specific resource. Access to any other resource in the service is blocked. This mechanism provides protection against data leakage risks.
- **Global reach:** Connect privately to services running in other regions. The consumer's virtual network could be in region A and it can connect to services in region B.
- **Simple to set up and manage:** You no longer need reserved, public IP addresses in your virtual networks to secure Azure resources through an IP firewall. There are no NAT or gateway devices required to set up the private endpoints. Private endpoints are configured through a simple workflow. On service side, you can also manage the connection requests on your Azure service resource with ease. Azure Private Link works for consumers and services belonging to different Microsoft Entra tenants too.

To learn more about private endpoints and the Azure services and regions that private endpoints are available for, see [Azure Private Link](#).

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

Last updated on 11/10/2025

Azure DDoS Protection fundamental best practices

The following sections give prescriptive guidance to build DDoS-resilient services on Azure.

Design for security

Ensure that security is a priority throughout the entire lifecycle of an application, from design and implementation to deployment and operations. Applications can have bugs that allow a relatively low volume of requests to use an inordinate amount of resources, resulting in a service outage.

To help protect a service running on Microsoft Azure, you should have a good understanding of your application architecture and focus on the [five pillars of software quality](#). You should know typical traffic volumes, the connectivity model between the application and other applications, and the service endpoints that are exposed to the public internet.

Ensuring that an application is resilient enough to handle a denial of service that's targeted at the application itself is most important. Security and privacy are built into the Azure platform, beginning with the [Security Development Lifecycle \(SDL\)](#) [↗](#). The SDL addresses security at every development phase and ensures that Azure is continually updated to make it even more secure. To learn more about maximizing your effectiveness using DDoS Protection, see [Maximizing Effectiveness: Best Practices for Azure DDoS Protection and Application Resilience](#) [↗](#).

Design for scalability

Scalability is how well a system can handle increased load. Design your applications to [scale horizontally](#) to meet the demand of an amplified load, specifically in the event of a DDoS attack. If your application depends on a single instance of a service, it creates a single point of failure. Provisioning multiple instances makes your system more resilient and more scalable.

For [Azure App Service](#), select an [App Service plan](#) that offers multiple instances. For Azure Cloud Services, configure each of your roles to use [multiple instances](#). For [Azure Virtual Machines](#), ensure that your virtual machine (VM) architecture includes more than one VM and that each VM is included in an [availability set](#). We recommend using [virtual machine scale sets](#) for autoscaling capabilities.

Defense in depth

The idea behind defense in depth is to manage risk by using diverse defensive strategies. Layering security defenses in an application reduces the chance of a successful attack. We recommend that you implement secure designs for your applications by using the built-in capabilities of the Azure platform.

For example, the risk of attack increases with the size (*surface area*) of the application. You can reduce the surface area by using an approval list to close down the exposed IP address space and listening ports that aren't needed on the load balancers ([Azure Load Balancer](#) and [Azure Application Gateway](#)). [Network security groups \(NSGs\)](#) are another way to reduce the attack surface. You can use [service tags](#) and [application security groups](#) to minimize complexity for creating security rules and configuring network security, as a natural extension of an application's structure. Additionally, you can use [Azure DDoS Solution for Microsoft Sentinel](#) [↗](#) to pinpoint offending DDoS sources and to block them from launching other, sophisticated attacks, such as data theft.

You should deploy Azure services in a [virtual network](#) whenever possible. This practice allows service resources to communicate through private IP addresses. Azure service traffic from a virtual network uses public IP addresses as source IP addresses by default. Using [service endpoints](#) will switch service traffic to use virtual network private addresses as the source IP addresses when they're accessing the Azure service from a virtual network.

We often see customers' on-premises resources getting attacked along with their resources in Azure. If you're connecting an on-premises environment to Azure, we recommend that you minimize exposure of on-premises resources to the public internet. You can use the scale and advanced DDoS protection capabilities of Azure by deploying your well-known public entities in Azure. Because these publicly accessible entities are often a target for DDoS attacks, putting them in Azure reduces the impact on your on-premises resources.

Azure security baseline for Azure DDoS Protection

Article • 02/25/2025

This security baseline applies guidance from the [Microsoft cloud security benchmark version 1.0](#) to Azure DDoS Protection. The Microsoft cloud security benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Microsoft cloud security benchmark and the related guidance applicable to Azure DDoS Protection.

You can monitor this security baseline and its recommendations using Microsoft Defender for Cloud. Azure Policy definitions will be listed in the Regulatory Compliance section of the Microsoft Defender for Cloud portal page.

When a feature has relevant Azure Policy Definitions, they are listed in this baseline to help you measure compliance with the Microsoft cloud security benchmark controls and recommendations. Some recommendations may require a paid Microsoft Defender plan to enable certain security scenarios.

 Note

Features not applicable to Azure DDoS Protection have been excluded. To see how Azure DDoS Protection completely maps to the Microsoft cloud security benchmark, see the [full Azure DDoS Protection security baseline mapping file](#).

Security profile

The security profile summarizes high-impact behaviors of Azure DDoS Protection, which may result in increased security considerations.

 Expand table

Service Behavior Attribute	Value
Product Category	Networking, Security
Customer can access HOST / OS	No Access
Service can be deployed into customer's virtual network	False
Stores customer content at rest	False

Asset management

For more information, see the [Microsoft cloud security benchmark: Asset management](#).

AM-2: Use only approved services

Features

Azure Policy Support

Description: Service configurations can be monitored and enforced via Azure Policy. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Use Microsoft Defender for Cloud to configure Azure Policy to audit and enforce configurations of your Azure resources. Use Azure Monitor to create alerts when there is a configuration deviation detected on the resources.

Reference: [DDOS Protection Policy](#)

Logging and threat detection

For more information, see the [Microsoft cloud security benchmark: Logging and threat detection](#).

LT-4: Enable logging for security investigation

Features

Azure Resource Logs

Description: Service produces resource logs that can provide enhanced service-specific metrics and logging. The customer can configure these resource logs and send them to their own data sink like a storage account or log analytics workspace. [Learn more](#).

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Configuration Guidance: Configure DDoS diagnostic logs, including notifications, mitigation reports and mitigation flow logs.

Reference: [View and configure DDoS diagnostic logging](#)

Next steps

- See the [Microsoft cloud security benchmark overview](#)
- Learn more about [Azure security baselines](#)

Prevent dangling DNS entries and avoid subdomain takeover

Article • 04/16/2025

This article describes the common security threat of subdomain takeover and the steps you can take to mitigate against it.

What is a subdomain takeover?

Subdomain takeovers are a common, high-severity threat for organizations that regularly create, and delete many resources. A subdomain takeover can occur when you have a [DNS record](#) that points to a deprovisioned Azure resource. Such DNS records are also known as "dangling DNS" entries. CNAME records are especially vulnerable to this threat. Subdomain takeovers enable malicious actors to redirect traffic intended for an organization's domain to a site performing malicious activity.

A common scenario for a subdomain takeover:

1. CREATION:

- a. You provision an Azure resource with a fully qualified domain name (FQDN) of `app-contogreat-dev-001.azurewebsites.net`.
- b. You assign a CNAME record in your DNS zone with the subdomain `greatapp.contoso.com` that routes traffic to your Azure resource.

2. DEPROVISIONING:

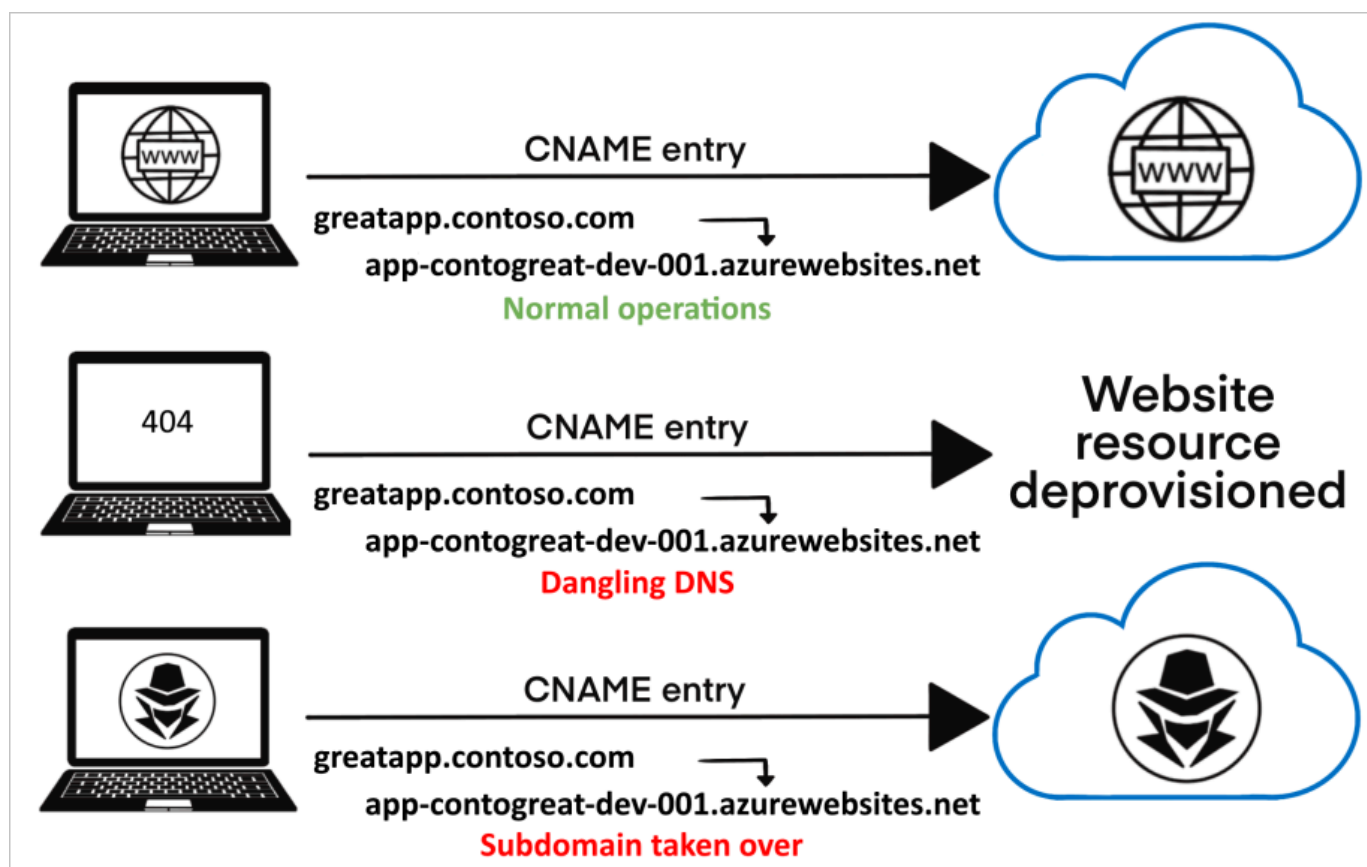
- a. The Azure resource is deprovisioned or deleted after it is no longer needed.

At this point, the CNAME record `greatapp.contoso.com` *should* be removed from your DNS zone. If the CNAME record isn't removed, it's advertised as an active domain but doesn't route traffic to an active Azure resource. You now have a "dangling" DNS record.

- b. The dangling subdomain, `greatapp.contoso.com`, is now vulnerable and can be taken over by being assigned to another Azure subscription's resource.

3. TAKEOVER:

- a. Using commonly available methods and tools, a threat actor discovers the dangling subdomain.
- b. The threat actor provisions an Azure resource with the same FQDN of the resource you previously controlled. In this example, `app-contogreat-dev-001.azurewebsites.net`.
- c. Traffic being sent to the subdomain `greatapp.contoso.com` is now routed to the malicious actor's resource where they control the content.



The risks of subdomain takeover

When a DNS record points to a resource that isn't available, the record itself should be removed from your DNS zone. If it isn't deleted, it's a "dangling DNS" record and creates the possibility for subdomain takeover.

Dangling DNS entries make it possible for threat actors to take control of the associated DNS name to host a malicious website or service. Malicious pages and services on an organization's subdomain might result in:

- **Loss of control over the content of the subdomain** - Negative press about your organization's inability to secure its content, brand damage, and loss of trust.
- **Cookie harvesting from unsuspecting visitors** - It's common for web apps to expose session cookies to subdomains (*.contoso.com). Any subdomain can access them. Threat actors can use subdomain takeover to build an authentic looking page, trick unsuspecting users to visit it, and harvest their cookies (even secure cookies). A common misconception is that using SSL certificates protects your site, and your users' cookies, from a takeover. However, a threat actor can use the hijacked subdomain to apply for and receive a valid SSL certificate. Valid SSL certificates grant them access to secure cookies and can further increase the perceived legitimacy of the malicious site.
- **Phishing campaigns** - Malicious actors often exploit authentic-looking subdomains in phishing campaigns. The risk extends to both malicious websites and MX records, which could enable threat actors to receive emails directed to legitimate subdomains associated with trusted brands.
- **Further risks** - Malicious sites might be used to escalate into other classic attacks such as XSS, CSRF, CORS bypass, and more.

Identify dangling DNS entries

To identify DNS entries within your organization that might be dangling, use Microsoft's GitHub-hosted PowerShell tools "[Get-DanglingDnsRecords](#)" [↗](#).

This tool helps Azure customers list all domains with a CNAME associated to an existing Azure resource that was created on their subscriptions or tenants.

If your CNAMEs are in other DNS services and point to Azure resources, provide the CNAMEs in an input file to the tool.

The tool supports the Azure resources listed in the following table. The tool extracts, or takes as inputs, all the tenant's CNAMEs.

 Expand table

Service	Type	FQDNproperty	Example
Azure Front Door	microsoft.network/frontdoors	properties.cName	abc.azurefd.net
Azure Blob Storage	microsoft.storage/storageaccounts	properties.primaryEndpoints.blob	abc.blob.core.windows.net
Azure CDN	microsoft.cdn/profiles/endpoints	properties.hostName	abc.azureedge.net
Public IP addresses	microsoft.network/publicipaddresses	properties.dnsSettings.fqdn	abc.EastUs.cloudapp.azure.com
Azure Traffic Manager	microsoft.network/trafficmanagerprofiles	properties.dnsConfig.fqdn	abc.trafficmanager.net
Azure Container Instance	microsoft.containerinstance/containergroups	properties.ipAddress.fqdn	abc.EastUs.azurecontainer.io
Azure API Management	microsoft.apimanagement/service	properties.hostnameConfigurations.hostName	abc.azure-api.net
Azure App Service	microsoft.web/sites	properties.defaultHostName	abc.azurewebsites.net
Azure App Service - Slots	microsoft.web/sites/slots	properties.defaultHostName	abc-def.azurewebsites.net

Prerequisites

Run the query as a user who has:

- at least reader level access to the Azure subscriptions
- read access to Azure resource graph

If you're a Global Administrator of your organization's tenant, follow the guidance in [Elevate access to manage all Azure subscriptions and management groups](#) to gain access to all your organization's subscriptions

Tip

Azure Resource Graph has throttling and paging limits that you should consider if you have a large Azure environment.

[Learn more about working with large Azure resource data sets.](#)

The tool uses subscription batching to avoid these limitations.

Run the script

Learn more about the PowerShell script, `Get-DanglingDnsRecords.ps1`, and download it from GitHub: <https://aka.ms/Get-DanglingDnsRecords> .

Remediate dangling DNS entries

Review your DNS zones and identify CNAME records that are dangling or taken over. If subdomains are found to be dangling or have been taken over, remove the vulnerable subdomains and mitigate the risks with the following steps:

1. From your DNS zone, remove all CNAME records that point to FQDNs of resources no longer provisioned.
2. To enable traffic to be routed to resources in your control, provision more resources with the FQDNs specified in the CNAME records of the dangling subdomains.
3. Review your application code for references to specific subdomains and update any incorrect or outdated subdomain references.
4. Investigate whether any compromise occurred and take action per your organization's incident response procedures.
Tips and best practices for investigating:

If your application logic results in secrets, such as OAuth credentials, being sent to dangling subdomains or if privacy-sensitive information is transmitted to those subdomains, there is a possibility for this data to be exposed to third parties.

5. Understand why the CNAME record was not removed from your DNS zone when the resource was deprovisioned and take steps to ensure that DNS records are updated appropriately when Azure resources are deprovisioned in the future.

Prevent dangling DNS entries

Ensuring that your organization has implemented processes to prevent dangling DNS entries and the resulting subdomain takeovers is a crucial part of your security program.

Some Azure services offer features to aid in creating preventative measures and are detailed below. Other methods to prevent this issue must be established through your organization's best practices or standard operating procedures.

Enable Microsoft Defender for App Service

Microsoft Defender for Cloud's integrated cloud workload protection platform (CWPP) offers a range of plans to protect your Azure, hybrid, and multicloud resources and workloads.

The **Microsoft Defender for App Service** plan includes dangling DNS detection. With this plan enabled, you'll get security alerts if you decommission an App Service website but don't remove its custom domain from your DNS registrar.

Microsoft Defender for Cloud's dangling DNS protection is available whether your domains are managed with Azure DNS or an external domain registrar and applies to App Service on both Windows and Linux.

Learn more about this and other benefits of this Microsoft Defender plans in [Introduction to Microsoft Defender for App Service](#).

Use Azure DNS alias records

Azure DNS's [alias records](#) can prevent dangling references by coupling the lifecycle of a DNS record with an Azure resource. For example, consider a DNS record that's qualified as an alias record to point to a public IP address or a Traffic Manager profile. If you delete those underlying resources, the DNS alias record becomes an empty record set. It no longer references the deleted resource. It's important to note that there are limits to what you can protect with alias records. Today, the list is limited to:

- Azure Front Door
- Traffic Manager profiles
- Azure Content Delivery Network (CDN) endpoints
- Public IPs

Despite the limited service offerings today, we recommend using alias records to defend against subdomain takeover whenever possible.

[Learn more about the capabilities of Azure DNS's alias records.](#)

Use Azure App Service's custom domain verification

When creating DNS entries for Azure App Service, create an `asuid.{subdomain}` TXT record with the Domain Verification ID. When such a TXT record exists, no other Azure Subscription can validate the Custom Domain that is, take it over.

These records don't prevent someone from creating the Azure App Service with the same name that's in your CNAME entry. Without the ability to prove ownership of the domain name, threat actors can't receive traffic or control the content.

[Learn more about how to map an existing custom DNS name to Azure App Service.](#)

Build and automate processes to mitigate the threat

It's often up to developers and operations teams to run cleanup processes to avoid dangling DNS threats. The practices below will help ensure your organization avoids suffering from this threat.

- **Create procedures for prevention:**
 - Educate your application developers to reroute addresses whenever they delete resources.
 - Put "Remove DNS entry" on the list of required checks when decommissioning a service.
 - Put [delete locks](#) on any resources that have a custom DNS entry. A delete lock serves as an indicator that the mapping must be removed before the resource is deprovisioned. Measures like this can only work when combined with internal education programs.
- **Create procedures for discovery:**
 - Review your DNS records regularly to ensure that your subdomains are all mapped to Azure resources that:
 - Exist - Query your DNS zones for resources pointing to Azure subdomains such as `*.azurewebsites.net` or `*.cloudapp.azure.com` (see the [Reference list of Azure domains](#)).
 - You own - Confirm that you own all resources that your DNS subdomains are targeting.
 - Maintain a service catalog of your Azure fully qualified domain name (FQDN) endpoints and the application owners. To build your service catalog, run the following Azure Resource Graph query script. This script projects the FQDN endpoint information of the resources you have access to and outputs them in a CSV file. If you have access to all the subscriptions for your tenant, the script considers all those subscriptions as shown in the following sample script. To limit the results to a specific set of subscriptions, edit the script as shown.
- **Create procedures for remediation:**
 - When dangling DNS entries are found, your team needs to investigate whether any compromise has occurred.
 - Investigate why the address wasn't rerouted when the resource was decommissioned.
 - Delete the DNS record if it's no longer in use, or point it to the correct Azure resource (FQDN) owned by your organization.

Clean up DNS pointers or Re-claim the DNS

Upon deletion of the classic cloud service resource, the corresponding DNS is reserved as per Azure DNS policies. During the reservation period, re-use of the DNS will be forbidden EXCEPT for subscriptions belonging to the Microsoft Entra tenant of the subscription originally owning the DNS. After the reservation expires, the DNS is free to be claimed by any subscription. By taking DNS reservations, the customer is afforded some time to either 1) clean up any associations/pointers to said DNS or 2) re-claim the DNS in Azure. The recommendation would be to delete unwanted DNS entries at the earliest. The DNS name being reserved can be derived by appending the cloud service name to the DNS zone for that cloud.

- Public - cloudapp.net
- Mooncake - chinacloudapp.cn
- Fairfax - usgovcloudapp.net
- BlackForest - azurecloudapp.de

For example, a hosted service in Public named "test" would have DNS "test.cloudapp.net"

Example: Subscription 'A' and subscription 'B' are the only subscriptions belonging to Microsoft Entra tenant 'AB'. Subscription 'A' contains a classic cloud service 'test' with DNS name 'test.cloudapp.net'. Upon deletion of the cloud service, a reservation is taken on DNS name 'test.cloudapp.net'. During the reservation period, only subscription 'A' or subscription 'B' will be able to claim the DNS name 'test.cloudapp.net' by creating a classic cloud service named 'test'. No other subscriptions will be allowed to claim it. After the reservation period, any subscription in Azure can now claim 'test.cloudapp.net'.

Next steps

To learn more about related services and Azure features you can use to defend against subdomain takeover, see the following pages.

- [Enable Microsoft Defender for App Service](#) - to receive alerts when dangling DNS entries are detected
- [Prevent dangling DNS records with Azure DNS](#)
- [Use a domain verification ID when adding custom domains in Azure App Service](#)
- [Quickstart: Run your first Resource Graph query using Azure PowerShell](#)

Implement a secure hybrid network

Azure Firewall

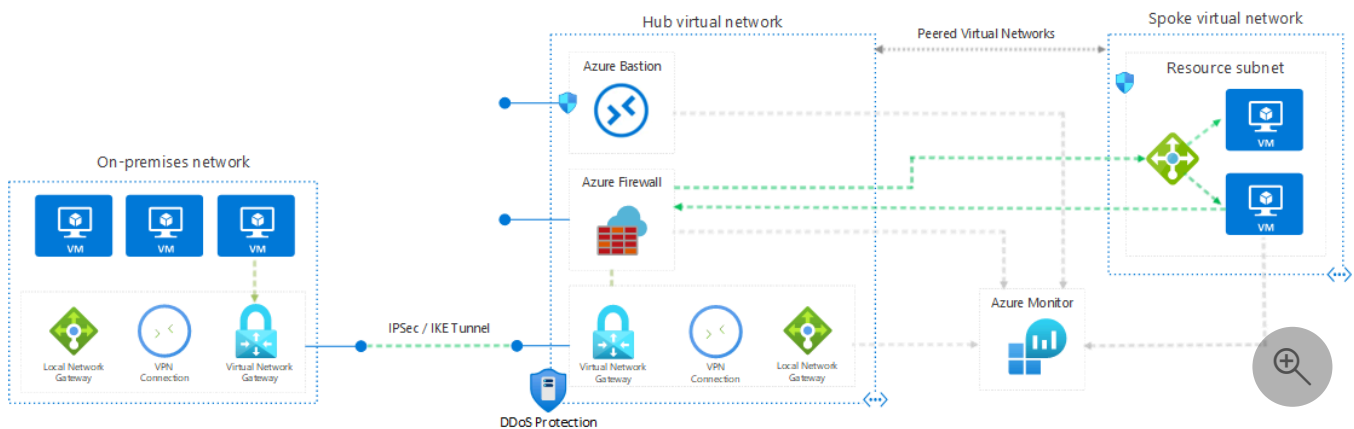
Azure Load Balancer

Azure Virtual Machines

Azure Virtual Network

This reference architecture shows a secure hybrid network that extends an on-premises network to Azure. The architecture implements a *perimeter network*, also called a *DMZ*, between the on-premises network and an Azure virtual network. All inbound and outbound traffic passes through Azure Firewall.

Architecture



Download a [Visio file](#) of this architecture.

Components

The architecture consists of the following aspects:

- **On-premises network.** A private local-area network implemented in an organization.
- **Azure virtual network.** The virtual network hosts the solution components and other resources running in Azure.

Virtual network routes define the flow of IP traffic within the Azure virtual network. In the diagram, there are two user-defined route tables.

In the gateway subnet, traffic is routed through the Azure Firewall instance.

! Note

Depending on the requirements of your VPN connection, you can configure Border Gateway Protocol (BGP) routes to implement the forwarding rules that direct traffic back through the on-premises network.

- **Gateway.** The gateway provides connectivity between the routers in the on-premises network and the virtual network. The gateway is placed in its own subnet.
- **Azure Firewall.** [Azure Firewall](#) is a managed firewall as a service. The Firewall instance is placed in its own subnet.
- **Network security groups.** Use [security groups](#) to restrict network traffic within the virtual network.
- **Azure Bastion.** [Azure Bastion](#) allows you to log into virtual machines (VMs) in the virtual network through SSH or remote desktop protocol (RDP) without exposing the VMs directly to the internet. Use Bastion to manage the VMs in the virtual network.

Bastion [requires a dedicated subnet named AzureBastionSubnet](#).

Potential use cases

This architecture requires a connection to your on-premises datacenter, using either a [VPN gateway](#) or an ExpressRoute connection. Typical uses for this architecture include:

- Hybrid applications where workloads run partly on-premises and partly in Azure.
- Infrastructure that requires granular control over traffic entering an Azure virtual network from an on-premises datacenter.
- Applications that must audit outgoing traffic. Auditing is often a regulatory requirement of many commercial systems and can help to prevent public disclosure of private information.

Recommendations

The following recommendations apply for most scenarios. Follow these recommendations unless you have a specific requirement that overrides them.

Access control recommendations

Use [Azure role-based access control \(Azure RBAC\)](#) to manage the resources in your application. Consider creating the following [custom roles](#):

- A DevOps role with permissions to administer the infrastructure for the application, deploy the application components, and monitor and restart VMs.
- A centralized IT administrator role to manage and monitor network resources.
- A security IT administrator role to manage secure network resources such as the firewall.

The IT administrator role shouldn't have access to the firewall resources. Access should be restricted to the security IT administrator role.

Resource group recommendations

Azure resources such as VMs, virtual networks, and load balancers can be managed by grouping them together into resource groups. Assign Azure roles to each resource group to restrict access.

We recommend creating the following resource groups:

- A resource group containing the virtual network (excluding the VMs), NSGs, and the gateway resources for connecting to the on-premises network. Assign the centralized IT administrator role to this resource group.
- A resource group containing the VMs for the Azure Firewall instance and the user-defined routes for the gateway subnet. Assign the security IT administrator role to this resource group.
- Separate resource groups for each spoke virtual network that contains the load balancer and VMs.

Networking recommendations

To accept inbound traffic from the internet, add a [Destination Network Address Translation \(DNAT\)](#) rule to Azure Firewall.

- Destination address = Public IP address of the firewall instance.
- Translated address = Private IP address within the virtual network.

[Force-tunnel](#) all outbound internet traffic through your on-premises network using the site-to-site VPN tunnel, and route to the internet using network address translation (NAT). This design prevents accidental leakage of any confidential information and allows inspection and auditing of all outgoing traffic.

Don't completely block internet traffic from the resources in the spoke network subnets. Blocking traffic will prevent these resources from using Azure PaaS services that rely on public IP addresses, such as VM diagnostics logging, downloading of VM extensions, and other functionality. Azure diagnostics also requires that components can read and write to an Azure Storage account.

Verify that outbound internet traffic is force-tunneled correctly. If you're using a VPN connection with the [routing and remote access service](#) on an on-premises server, use a tool such as [WireShark](#) .

Consider using Application Gateway or Azure Front Door for SSL termination.

Considerations

These considerations implement the pillars of the Azure Well-Architected Framework, which is a set of guiding tenets that can be used to improve the quality of a workload. For more information, see [Microsoft Azure Well-Architected Framework](#).

Reliability

Reliability ensures your application can meet the commitments you make to your customers. For more information, see [Design review checklist for Reliability](#).

If you're using Azure ExpressRoute to provide connectivity between the virtual network and on-premises network, [configure a VPN gateway to provide failover](#) if the ExpressRoute connection becomes unavailable.

For information on maintaining availability for VPN and ExpressRoute connections, see the availability considerations in:

- [Implementing a hybrid network architecture with Azure and on-premises VPN](#)
- [Implementing a hybrid network architecture with Azure ExpressRoute](#)

Security

Security provides assurances against deliberate attacks and the abuse of your valuable data and systems. For more information, see [Design review checklist for Security](#).

This reference architecture implements multiple levels of security.

Routing all on-premises user requests through Azure Firewall

The user-defined route in the gateway subnet blocks all user requests other than those received from on-premises. The route passes allowed requests to the firewall. The requests are passed on to the resources in the spoke virtual networks if they're allowed by the firewall rules. You can add other routes, but make sure they don't inadvertently bypass the firewall or block administrative traffic intended for the management subnet.

Using NSGs to block/pass traffic to spoke virtual network subnets

Traffic to and from resource subnets in spoke virtual networks is restricted by using NSGs. If you have a requirement to expand the NSG rules to allow broader access to these resources, weigh these requirements against the security risks. Each new inbound pathway represents an opportunity for accidental or purposeful data leakage or application damage.

DDoS protection

[Azure DDoS Protection](#), combined with application-design best practices, provides enhanced DDoS mitigation features to provide more defense against DDoS attacks. You should enable [Azure DDOS Protection](#) on any perimeter virtual network.

Use AVNM to create baseline Security Admin rules

AVNM allows you to create baselines of security rules, which can take priority over network security group rules. [Security admin rules](#) are evaluated before NSG rules and have the same nature of NSGs, with support for prioritization, service tags, and L3-L4 protocols. AVNM allows central IT to enforce a baseline of security rules, while allowing an independency of additional NSG rules by the spoke virtual network owners. To facilitate a controlled rollout of security rules changes, AVNM's [deployments](#) feature allows you to safely release of these configurations' breaking changes to the hub-and-spoke environments.

DevOps access

Use [Azure RBAC](#) to restrict the operations that DevOps can perform on each tier. When granting permissions, use the [principle of least privilege](#). Log all administrative operations and perform regular audits to ensure any configuration changes were planned.

Cost Optimization

Cost Optimization is about looking at ways to reduce unnecessary expenses and improve operational efficiencies. For more information, see [Design review checklist for Cost Optimization](#).

Use the [Azure pricing calculator](#) [↗] to estimate costs. Other considerations are described in the Cost optimization section in [Microsoft Azure Well-Architected Framework](#).

Here are cost considerations for the services used in this architecture.

Azure Firewall

In this architecture, Azure Firewall is deployed in the virtual network to control traffic between the gateway's subnet and the resources in the spoke virtual networks. Using Azure Firewall as a shared solution across multiple workloads can help reduce duplicate infrastructure. Here are the Azure Firewall pricing models:

- Fixed rate per deployment hour.
- Data processed per GB to support auto scaling.

When compared to network virtual appliances (NVAs), with Azure Firewall you can save up to 30-50%. For more information, see [Azure Firewall vs NVA](#).

Azure Bastion

Azure Bastion securely connects to your virtual machine over RDP and SSH without having the need to configure a public IP on the virtual machine.

Bastion billing is comparable to a basic, low-level virtual machine configured as a jump box. Bastion is more cost effective than a jump box as it has built-in security features, and doesn't incur extra costs for storage and managing a separate server.

Azure Virtual Network

Azure Virtual Network is free. Every subscription is allowed to create up to 1,000 virtual networks across all regions. All traffic that occurs within the boundaries of a virtual network is free. For example, VMs in the same virtual network that talk to each other don't incur network traffic charges.

Internal load balancer

Basic load balancing between virtual machines that reside in the same virtual network is free.

In this architecture, internal load balancers are used to load balance traffic inside a virtual network.

Operational Excellence

Operational Excellence covers the operations processes that deploy an application and keep it running in production. For more information, see [Design review checklist for Operational Excellence](#).

If gateway connectivity from your on-premises network to Azure is down, you can still reach the VMs in the Azure virtual network through Azure Bastion.

Each tier's subnet in the reference architecture is protected by NSG rules. You might need to create a rule to open port 3389 for remote desktop protocol (RDP) access on Windows VMs or port 22 for secure shell (SSH) access on Linux VMs. Other management and monitoring tools might require rules to open additional ports.

If you're using ExpressRoute to provide the connectivity between your on-premises datacenter and Azure, use the [Azure Connectivity Toolkit \(AzureCT\)](#) to monitor and troubleshoot connection issues.

You can find additional information about monitoring and managing VPN and ExpressRoute connections in the article [Implementing a hybrid network architecture with Azure and on-premises VPN](#).

Performance Efficiency

Performance Efficiency is the ability of your workload to scale to meet the demands placed on it by users in an efficient manner. For more information, see [Design review checklist for Performance Efficiency](#).

For more information about the bandwidth limits of VPN Gateway, see [Gateway SKUs](#). For higher bandwidths, consider upgrading to an ExpressRoute gateway. ExpressRoute provides up to 10-Gbps bandwidth with lower latency than a VPN connection.

For more information about the scalability of Azure gateways, see the scalability consideration sections in:

- [Implementing a hybrid network architecture with Azure and on-premises VPN](#)
- [Implementing a hybrid network architecture with Azure ExpressRoute](#)

For more information about managing virtual networks and NSGs at scale, see [Azure Virtual Network Manager \(AVNM\): Create a secured hub and spoke network](#) to create new (and onboard existing) hub and spoke virtual network topologies for central management of connectivity and NSG rules.

Deploy this scenario

This deployment creates two resource groups; the first holds a mock on-premises network, the second a set of hub and spoke networks. The mock on-premises network and the hub network are connected using Azure Virtual Network gateways to form a site-to-site connection. This configuration is very similar to how you would connect your on-premises datacenter to Azure.

This deployment can take up to 45 minutes to complete. The recommended deployment method is using the following portal option.

Use the following button to deploy the reference using the Azure portal.

[Deploy to Azure](#)

After the deployment finishes, verify site-to-site connectivity by looking at the newly created connection resources. In the Azure portal, search for *connections* and check the status of each connection.

Connections						
Microsoft						
+ Add Edit columns Refresh Assign tags						
Subscriptions: 1 of 41 selected – Don't see a subscription? Open Directory + Subscription settings						
Filter by name... All resource groups All locations All tags						
2 items						
☐	Name ↑↓	Status	Peer 1	Peer 2	Resource group ↑↓	Location ↑↓
☐	hub-to-mock-prem	Connected	vpn-azure-network	local-gateway-azure-netw...	site-to-site-azure-network	East US
☐	mock-prem-to-hub	Connected	vpn-mock-prem	local-gateway-moc-prem	site-to-site-mock-prem	East US

The IIS instance found in the spoke network can be accessed from the virtual machine located in the mock on-premises network. Create a connection to the virtual machine using the included Azure Bastion host, open a web browser, and navigate to the address of the application's network load balancer.

For more information and other deployment options, see the Azure Resource Manager templates (ARM templates) used to deploy this solution: [Secure Hybrid Network](#).

Next steps

- [The virtual datacenter: A network perspective](#).
- [Azure security documentation](#).

Related resources

- [Connect an on-premises network to Azure using ExpressRoute](#).
- [Configure ExpressRoute and Site-to-Site coexisting connections using PowerShell](#)
- [Extend an on-premises network using ExpressRoute](#).

Microsoft Antimalware for Azure Cloud Services and Virtual Machines (VMs)

Article • 05/02/2025

Microsoft Antimalware for Azure is a free real-time protection that helps identify and remove viruses, spyware, and other malicious software. It generates alerts when known malicious or unwanted software tries to install itself or run on your Azure systems.

The solution is built on the same antimalware platform as Microsoft Security Essentials (MSE), Microsoft Forefront Endpoint Protection, Microsoft System Center Endpoint Protection, Microsoft Intune, and Microsoft Defender for Cloud. Microsoft Antimalware for Azure is a single-agent solution for applications and tenant environments, designed to run in the background without human intervention. Protection can be deployed based on the needs of application workloads, with either basic secure-by-default or advanced custom configuration, including antimalware monitoring.

When you deploy and enable Microsoft Antimalware for Azure for your applications, the following core features are available:

- **Real-time protection** - monitors activity in Cloud Services and on Virtual Machines to detect and block malware execution.
- **Scheduled scanning** - Scans periodically to detect malware, including actively running programs.
- **Malware remediation** - automatically takes action on detected malware, such as deleting or quarantining malicious files and cleaning up malicious registry entries.
- **Signature updates** - automatically installs the latest protection signatures (virus definitions) to ensure protection is up-to-date on a predetermined frequency.
- **Antimalware Engine updates** - automatically updates the Microsoft Antimalware engine.
- **Antimalware Platform updates** - automatically updates the Microsoft Antimalware platform.
- **Active protection** - reports telemetry metadata about detected threats and suspicious resources to Microsoft Azure to ensure rapid response to the evolving threat landscape and enables real-time synchronous signature delivery through the Microsoft Active Protection System (MAPS).
- **Samples reporting** - provides and reports samples to the Microsoft Antimalware service to help refine the service and enable troubleshooting.
- **Exclusions** - allows application and service administrators to configure exclusions for files, processes, and drives.
- **Antimalware event collection** - records the antimalware service health, suspicious activities, and remediation actions taken in the operating system event log and collects them into the customer's Azure Storage account.

ⓘ Note

Microsoft Antimalware can also be deployed using Microsoft Defender for Cloud. Read [Install Endpoint Protection in Microsoft Defender for Cloud](#) for more information.

Architecture

Microsoft Antimalware for Azure consists of several components:

- The Microsoft Antimalware Client and Service
- Antimalware classic deployment model
- Antimalware PowerShell cmdlets
- Azure Diagnostics Extension

Platform Support and Deployment

Virtual Machines:

- Not installed by default
- Available as an optional security extension through the Azure portal or Visual Studio Virtual Machine configuration
- Supported on Windows Server 2008 R2, Windows Server 2012, and Windows Server 2012 R2
- Not supported on Windows Server 2008 and Linux operating systems

Cloud Services:

- Installed by default in a disabled state on all supported Azure guest operating systems
- Requires explicit activation to protect your cloud service

Azure App Service:

- Enabled on the underlying service hosting Windows-based web apps
- Limited to protecting Azure App Service infrastructure only, not customer content
- Not sufficient for complete web application security (implement more security controls as outlined in the [Azure Web Application Security Best Practices](#))

ⓘ Note

Microsoft Defender Antivirus is the built-in Antimalware enabled in Windows Server 2016 and above. The Azure VM Antimalware extension can still be added to a Windows Server

2016 and above Azure VM with Microsoft Defender Antivirus. In this scenario, the extension applies any optional [configuration policies](#) to be used by Microsoft Defender Antivirus. The extension does not deploy any other antimalware services. For more information about Microsoft Defender Antivirus, see [Code samples to enable and configure Microsoft Antimalware for Azure](#).

Microsoft antimalware workflow

The Azure service administrator can enable Antimalware for Azure with a default or custom configuration for your Virtual Machines and Cloud Services using the following options:

- Virtual Machines - In the Azure portal, under **Security Extensions**
- Virtual Machines - Using the Visual Studio virtual machines configuration in Server Explorer
- Virtual Machines and Cloud Services - Using the Antimalware [classic deployment model](#)
- Virtual Machines and Cloud Services - Using Antimalware PowerShell cmdlets

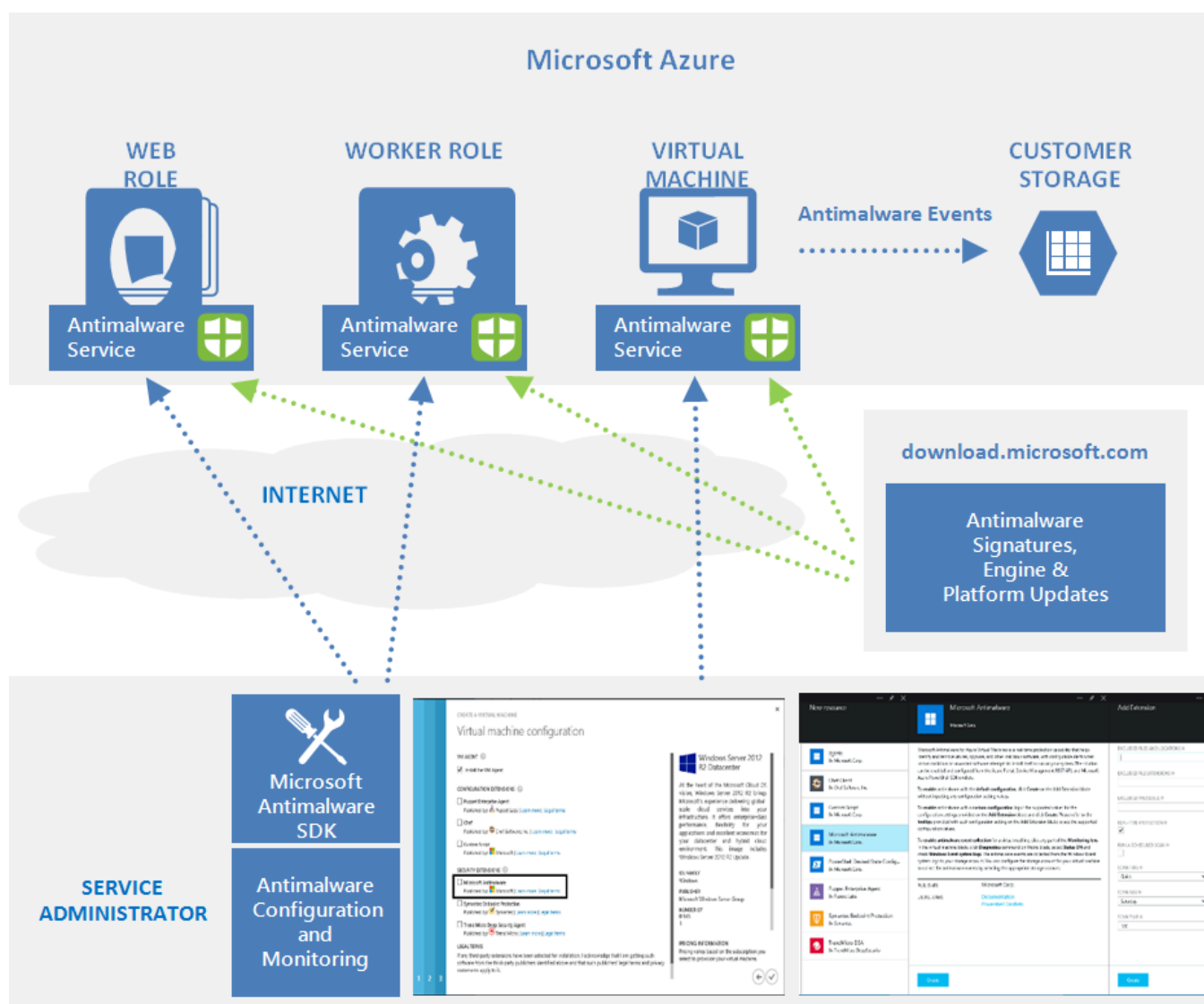
The Azure portal or PowerShell cmdlets push the Antimalware extension package file to the Azure system at a predetermined fixed location. The Azure Guest Agent (or the Fabric Agent) launches the Antimalware Extension, applying the Antimalware configuration settings supplied as input. This step enables the Antimalware service with either default or custom configuration settings. If no custom configuration is provided, then the antimalware service is enabled with the default configuration settings. For more information about Antimalware configuration, see [Code samples to enable and configure Microsoft Antimalware for Azure](#).

After initialization, the Microsoft Antimalware client automatically retrieves the latest protection engine and signature definitions from the Internet and applies them to your Azure system. The service logs all activity to the operating system event log under the "Microsoft Antimalware" event source. These logs include information about:

- Client health status
- Protection and remediation activities
- Configuration changes
- Engine and signature definition updates
- Other operational events

You can enable Antimalware monitoring for your Cloud Service or Virtual Machine to have the Antimalware event log events written as they're produced to your Azure storage account. The Antimalware Service uses the Azure Diagnostics extension to collect Antimalware events from the Azure system into tables in the customer's Azure Storage account.

The deployment workflow including configuration steps and options supported for the above scenarios are documented in [Antimalware deployment scenarios](#) section of this document.



Note

You can however use PowerShell/APIs and Azure Resource Manager templates to deploy Virtual Machine Scale Sets with the Microsoft Anti-Malware extension. For installing an extension on an already running Virtual Machine, you can use the sample Python script [vmssextn.py](#)[↗]. This script gets the existing extension config on the Scale Set and adds an extension to the list of existing extensions on the Azure Virtual Machines Scale Sets.

Default and Custom Antimalware Configuration

The default configuration settings are applied to enable Antimalware for Azure Cloud Services or Virtual Machines when you don't provide custom configuration settings. The default configuration settings are preoptimized for running in the Azure environment. Optionally, you

can customize these default configuration settings as required for your Azure application or service deployment and apply them for other deployment scenarios.

The following table summarizes the configuration settings available for the Antimalware service. The default configuration settings are marked under the column labeled "Default."

Setting	Options	Default	Description
Enable Antimalware	true (lower case sensitive)	None	true - Enables the Antimalware service false – not supported Note – This is a required configuration setting to enable the Antimalware service
Exclusions Extensions	extension1, extension2, ...	None	List of file extensions to exclude from scanning. Example: gif, log, txt excludes files with the .gif, .log, or .txt extension from being scanned. Each excluded file extension should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
Exclusions Paths	path1, path2 	None	List of paths to files or folders to exclude from scanning. Example: e:\approot\worker.dll, e:\approot\temp excludes the file worker.dll in the e:\approot folder and anything under the folder e:\approot\temp from being scanned. Note: For antimalware JSON configuration for virtual machines, use two backslashes (\\) instead of one to escape properly. For example: e:\\approot\\worker.dll Each excluded path should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
Exclusions Processes	process1, process2, ...	None	List of process exclusions. Any file opened by an excluded process will not be scanned (the process itself will still be scanned – to exclude the process itself, use the ExcludedPaths option). Example: C:\Program Files\MyApp.exe excludes any files opened by MyApp.exe from being scanned. Each excluded process should be added as a separate row element value in your antimalware XML configuration or semicolon separated in antimalware JSON configuration
RealtimeProtectionEnabled	true false (lower case sensitive)	true	true – Enables real-time protection false – Disables real-time protection Default = true when AntimalwareEnabled = true
ScheduledScanSettings isEnabled	true false (lower case sensitive)	false	Enables or disables a periodic scan for active malware on the system Default = false
ScheduledScanSettings Day	0 – 8	7	0 – scan daily, 1 – Sunday, 2 – Monday, 3- Tuesday..., 7 – Saturday, 8 – disabled Default = 7 if only ScheduledScanSettings

			Default = 120 mins if ScheduledScanSettings.isEnabled = true
ScheduledScanSettings Time	0 – 1440	120	Hour at which to begin the scheduled scan. Measured in 60 minute increments from midnight 60 mins = 1:00 AM 120 mins = 2:00 AM ... 1380 mins = 11:00 PM Default = 120 mins if ScheduledScanSettings.isEnabled = true
ScheduledScanSettings Scan Type	Quick/Full	Quick	Default = Quick if ScheduledScanSettings.isEnabled = true
StorageAccountName	Storage Account Name	None	Storage account name for your Azure store table to collect antimalware events in storage Note - Storage account name is required if monitoring is specified as ON

Antimalware Deployment Scenarios

The scenarios to enable and configure antimalware, including monitoring for Azure Cloud Services and Virtual Machines, are discussed in this section.

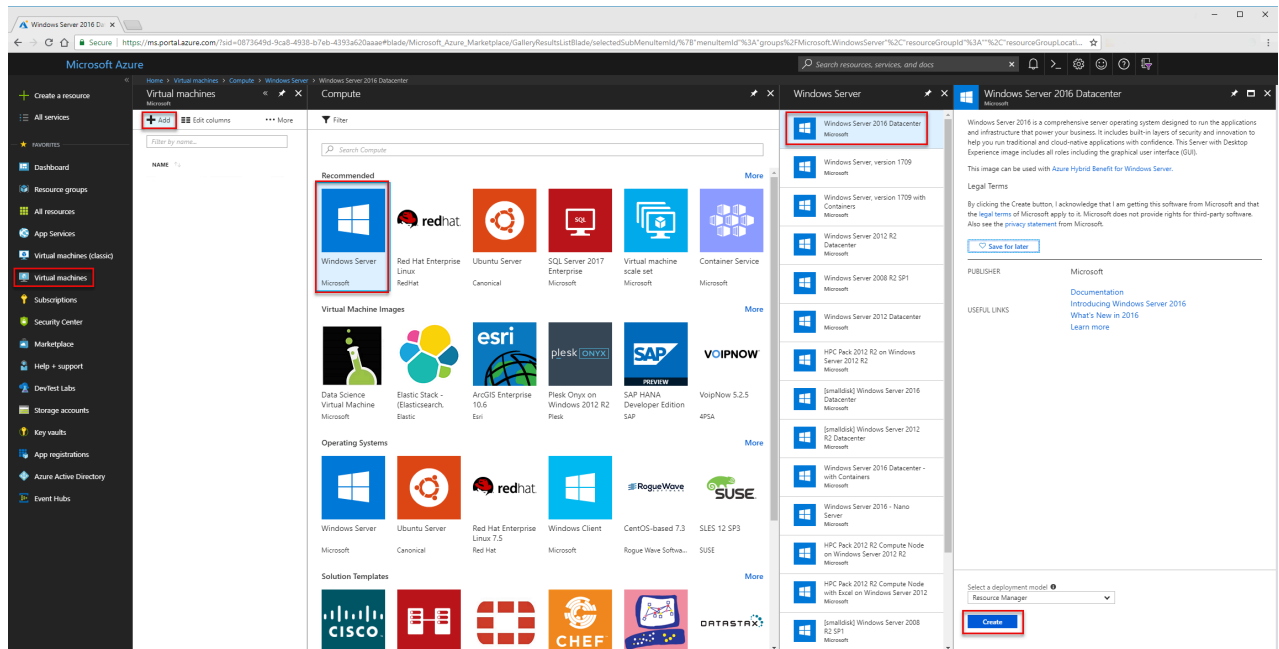
Virtual machines - enable and configure antimalware

Deployment While creating a VM using the Azure portal

Follow these steps to enable and configure Microsoft Antimalware for Azure Virtual Machines using the Azure portal while provisioning a Virtual Machine:

1. Sign in to the [Azure portal](#).
2. To create a new virtual machine, navigate to **Virtual machines**, select **Add**, and choose **Windows Server**.
3. Select the version of Windows server that you would like to use.

4. Select Create.



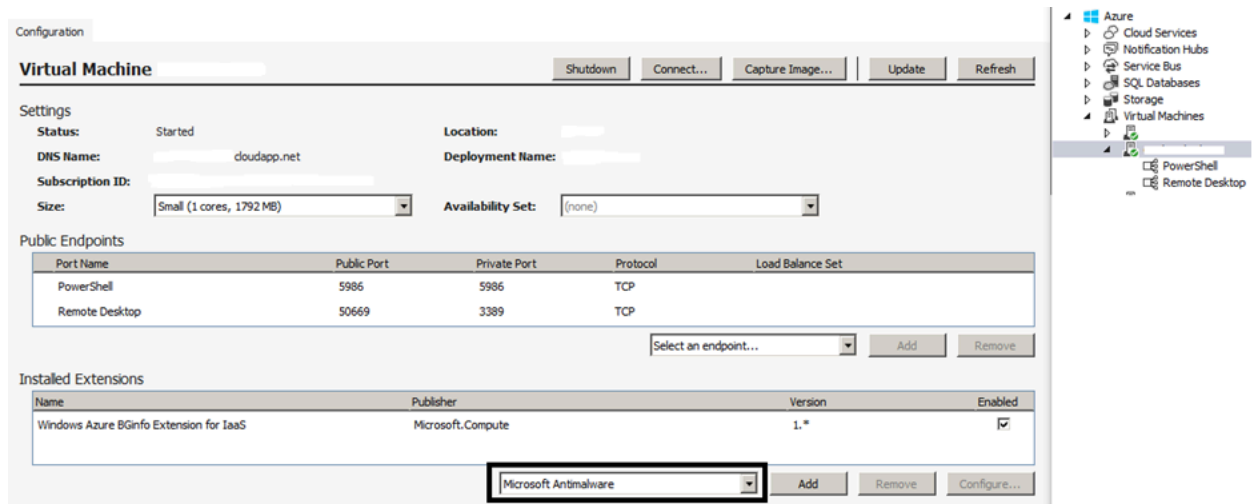
5. Provide a **Name**, **Username**, **Password**, and create a new resource group or choose an existing resource group.
6. Select **Ok**.
7. Choose a vm size.
8. In the next section, make the appropriate choices for your needs select the **Extensions** section.
9. Select **Add extension**
10. Under **New resource**, choose **Microsoft Antimalware**.
11. Select **Create**
12. In the **Install extension** section file, locations, and process exclusions can be configured as well as other scan options. Choose **Ok**.
13. Choose **Ok**.
14. Back in the **Settings** section, choose **Ok**.
15. In the **Create** screen, choose **Ok**.

See this [Azure Resource Manager template](#) for deployment of Antimalware VM extension for Windows.

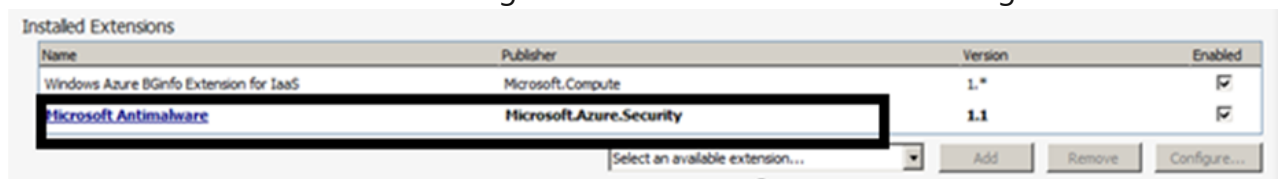
Deployment using the Visual Studio virtual machine configuration

To enable and configure the Microsoft Antimalware service using Visual Studio:

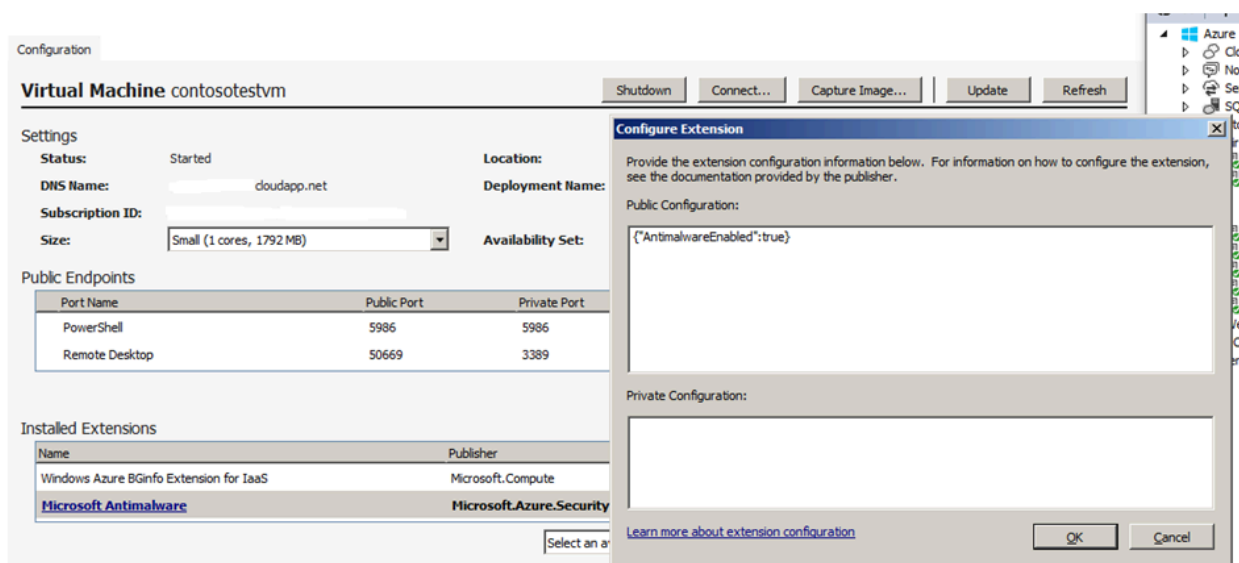
1. Connect to Microsoft Azure in Visual Studio.
2. Choose your Virtual Machine in the **Virtual Machines** node in **Server Explorer**



3. Right-click **configure** to view the Virtual Machine configuration page
4. Select **Microsoft Antimalware** extension from the dropdown list under **Installed Extensions** and select **Add** to configure with default antimalware configuration.



5. To customize the default Antimalware configuration, select (highlight) the Antimalware extension in the installed extensions list and select **Configure**.
6. Replace the default Antimalware configuration with your custom configuration in the **public configuration** textbox and select OK.
7. Select the **Update** button to push the configuration updates to your Virtual Machine.



ⓘ **Note**

The Visual Studio Virtual Machines configuration for Antimalware supports only JSON format configuration. For more information about sample configurations, see [Code samples to enable and configure Microsoft Antimalware for Azure](#).

Deployment Using PowerShell cmdlets

An Azure application or service can enable and configure Microsoft Antimalware for Azure Virtual Machines using PowerShell cmdlets.

To enable and configure Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell> ↗
2. Use the [Set-AzureVMMicrosoftAntimalwareExtension](#) cmdlet to enable and configure Microsoft Antimalware for your Virtual Machine.

ⓘ Note

The Azure Virtual Machines configuration for Antimalware supports only JSON format configuration. For more information about sample configurations, see [Code samples to enable and configure Microsoft Antimalware for Azure](#).

Enable and Configure Antimalware Using PowerShell cmdlets

An Azure application or service can enable and configure Microsoft Antimalware for Azure Cloud Services using PowerShell cmdlets. Microsoft Antimalware is installed in a disabled state in the Cloud Services platform and requires an action by an Azure application to enable it.

To enable and configure Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell> ↗
2. Use the [Set-AzureServiceExtension](#) cmdlet to enable and configure Microsoft Antimalware for your Cloud Service.

For more information about sample PowerShell commands, see [Code samples to enable and configure Microsoft Antimalware for Azure](#).

Cloud Services and Virtual Machines - Configuration Using PowerShell cmdlets

An Azure application or service can retrieve the Microsoft Antimalware configuration for Cloud Services and Virtual Machines using PowerShell cmdlets.

To retrieve the Microsoft Antimalware configuration using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell> ↗
2. **For Virtual Machines:** Use the [Get-AzureVMMicrosoftAntimalwareExtension](#) cmdlet to get the antimalware configuration.
3. **For Cloud Services:** Use the [Get-AzureServiceExtension](#) cmdlet to get the Antimalware configuration.

Samples

Remove Antimalware Configuration Using PowerShell cmdlets

An Azure application or service can completely remove Microsoft Antimalware protection by uninstalling the relevant extensions from your Cloud Services or Virtual Machines. This process removes both the antimalware protection and associated monitoring settings, completely discontinuing malware protection and event collection for the specified resources.

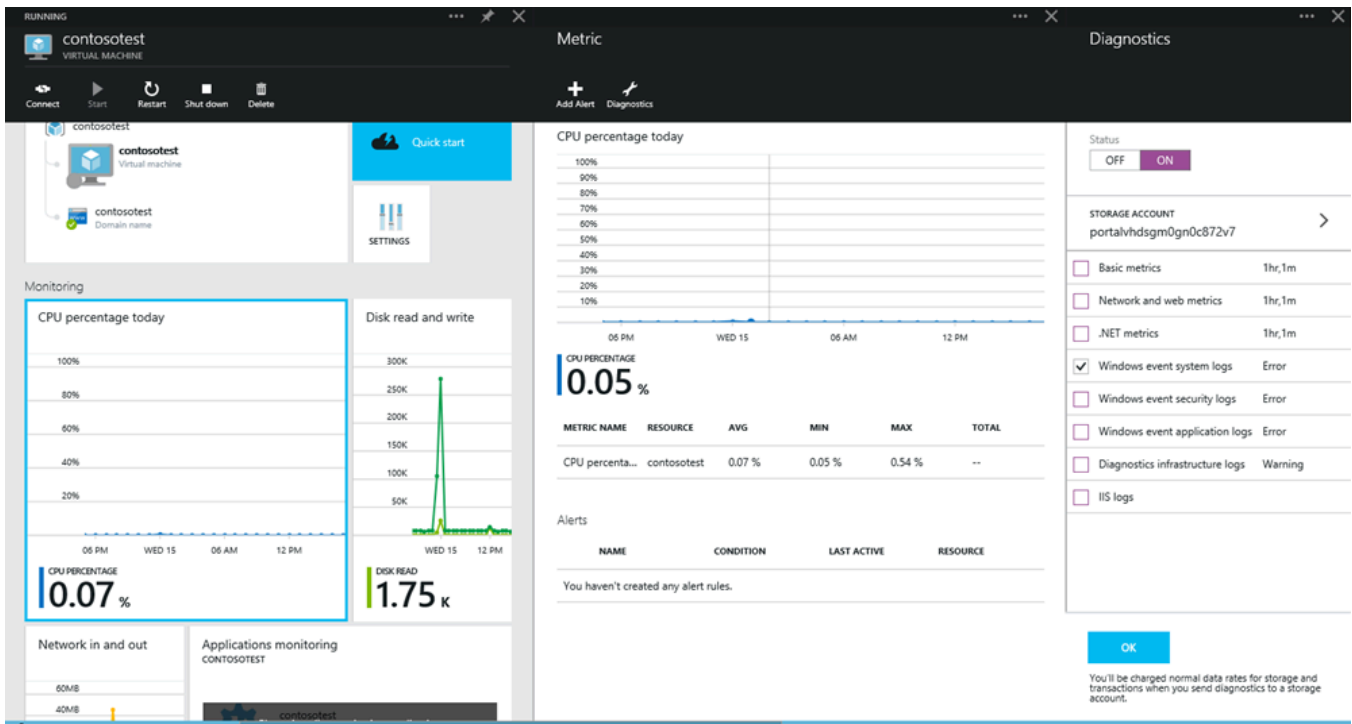
To remove Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell> ↗
2. **For Virtual Machines:** Use the [Remove-AzureVMMicrosoftAntimalwareExtension](#) cmdlet.
3. **For Cloud Services:** Use the [Remove-AzureServiceExtension](#) cmdlet.

To **enable** antimalware event collection for a virtual machine using the Azure Preview Portal:

1. Select any part of the Monitoring section in the Virtual Machine details page.
2. Select the Diagnostics command in the Metrics section.
3. Select **Status** ON and check the option for Windows event system.
4. You can choose to uncheck all other options in the list, or leave them enabled per your application service needs.
5. The Antimalware event categories "Error", "Warning", "Informational", etc., are captured in your Azure Storage account.

Antimalware events are collected from the Windows event system logs to your Azure Storage account. You can configure the Storage Account for your Virtual Machine to collect Antimalware events by selecting the appropriate storage account.



Enable and configure Antimalware using PowerShell cmdlets for Azure Resource Manager VMs

To enable and configure Microsoft Antimalware for Azure Resource Manager VMs using PowerShell cmdlets:

1. Set up your PowerShell environment using this [documentation](#) on GitHub.
2. Use the `Set-AzVMExtension` cmdlet to enable and configure Microsoft Antimalware for your VM.

The following code samples are available:

- [Deploy Microsoft Antimalware on ARM template VMs](#)
- [Add Microsoft Antimalware to Azure Service Fabric Clusters](#)

Enable and configure Antimalware to Azure Cloud Service Extended Support (CS-ES) using PowerShell cmdlets

To enable and configure Microsoft Antimalware using PowerShell cmdlets:

1. Set up your PowerShell environment - Refer to the documentation at <https://github.com/Azure/azure-powershell>
2. Use the `New-AzCloudServiceExtensionObject` cmdlet to enable and configure Microsoft Antimalware for your Cloud Service VM.

The following code sample is available:

- [Add Microsoft Antimalware to Azure Cloud Service using Extended Support\(CS-ES\)](#)

Enable and configure Antimalware using PowerShell cmdlets for Azure Arc-enabled servers

To enable and configure Microsoft Antimalware for Azure Arc-enabled servers using PowerShell cmdlets:

1. Set up your PowerShell environment using this [documentation](#)  on GitHub.
2. Use the [New-AzConnectedMachineExtension](#) cmdlet to enable and configure Microsoft Antimalware for your Arc-enabled servers.

The following code samples are available:

- [Add Microsoft Antimalware for Azure Arc-enabled servers](#)

Next steps

- [Code samples to enable and configure Microsoft Antimalware for Azure](#)
- [Microsoft Defender for Cloud](#)

Code samples to enable and configure Microsoft Antimalware for Azure

This article provides PowerShell code samples to enable and configure Microsoft Antimalware for different Azure services including:

- Azure Resource Manager VMs
- Azure Service Fabric Clusters
- Azure Cloud Services using Extended Support
- Azure Arc-enabled servers

You can use these samples to deploy and configure the Microsoft Antimalware extension across your Azure environments.

Deploy Microsoft Antimalware on Azure Resource Manager VMs

⚠ Note

Before executing this code sample, you must uncomment the variables and provide appropriate values.

PowerShell

```
# Script to add Microsoft Antimalware extension to Azure Resource Manager VMs
# Specify your subscription ID
$subscriptionId= " SUBSCRIPTION ID HERE "
# specify location, resource group, and VM for the extension
$location = " LOCATION HERE " # eg., "Southeast Asia" or "Central US"
$resourceGroupName = " RESOURCE GROUP NAME HERE "
$vmName = " VM NAME HERE "

# Enable Antimalware with default policies
$settingString = '{"AntimalwareEnabled": true}';
# Enable Antimalware with custom policies
# $settingString = '{
# "AntimalwareEnabled": true,
# "RealtimeProtectionEnabled": true,
# "ScheduledScanSettings": {
#     "isEnabled": true,
#     "day": 0,
#     "time": 120,
#     "scanType": "Quick"
# },
# "Exclusions": {
```

```
#         "Extensions": ".ext1,.ext2",
#         "Paths": "",
#         "Processes": "sample1.exe, sample2.exe"
#     },
# "SignatureUpdates": {
#         "FileSharesSources": "",
#         "FallbackOrder": "",
#         "ScheduleDay": 0,
#         "UpdateInterval": 0,
#     },
# "CloudProtection": true
# }';
# Login to your Azure Resource Manager Account and select the Subscription to use
Login-AzureRmAccount

Select-AzureRmSubscription -SubscriptionId $subscriptionId
# retrieve the most recent version number of the extension
$allVersions= (Get-AzureRmVMExtensionImage -Location $location -PublisherName
"Microsoft.Azure.Security" -Type "IaaSAntimalware").Version
$versionString = $allVersions[($allVersions.count)-1].Split(".")[0] + "." +
$allVersions[($allVersions.count)-1].Split(".")[1]
# set the extension using prepared values
# ****--Use this script till cmdlets address the -SettingsString format issue we
observed ****--
Set-AzureRmVMExtension -ResourceGroupName $resourceGroupName -Location $location -
VMName $vmName -Name "IaaSAntimalware" -Publisher "Microsoft.Azure.Security" -
ExtensionType "IaaSAntimalware" -TypeHandlerVersion $versionString -SettingString
$settingString
```

Add Microsoft Antimalware to Azure Service Fabric Clusters

Azure Service Fabric uses Azure virtual machine scale sets to create the Service Fabric Clusters. Presently the virtual machine scale sets template used for creating the Service Fabric Clusters isn't enabled with the Antimalware extension. As such, Antimalware needs to be enabled separately on the scale sets. As you enable it on scale sets, all the nodes created under the virtual machine scale sets inherit and get the extension automatically.

The code sample below shows how you can enable IaaS Antimalware extension using the AzureRmVmss PowerShell cmdlets.

⚠ Note

Before executing this code sample, you must uncomment the variables and provide appropriate values.

PowerShell

```
# Script to add Microsoft Antimalware extension to VM Scale Set(VMSS) and Service
Fabric Cluster(in turn it used VMSS)
# Login to your Azure Resource Manager Account and select the Subscription to use
Login-AzureRmAccount
# Specify your subscription ID
$subscriptionId="SUBSCRIPTION ID HERE"
Select-AzureRmSubscription -SubscriptionId $subscriptionId
# Specify location, resource group, and VM Scaleset for the extension
$location = "LOCATION HERE" # eg., "West US or Southeast Asia" or "Central US"
$resourceGroupName = "RESOURCE GROUP NAME HERE"
$vmScaleSetName = "YOUR VM SCALE SET NAME"

# Configuration.JSON configuration file can be customized as per MSDN documentation:
https://msdn.microsoft.com/en-us/library/dn771716.aspx
$settingString = '{"AntimalwareEnabled": true}';
# Enable Antimalware with custom policies
# $settingString = '{
# "AntimalwareEnabled": true,
# "RealtimeProtectionEnabled": true,
# "ScheduledScanSettings": {
#                               "isEnabled": true,
#                               "day": 0,
#                               "time": 120,
#                               "scanType": "Quick"
#                               },
# "Exclusions": {
#     "Extensions": ".ext1,.ext2",
#     "Paths": "",
#     "Processes": "sample1.exe, sample2.exe"
#   },
# "SignatureUpdates": {
#                               "FileSharesSources": "",
#                               "FallbackOrder": "",
#                               "ScheduleDay": 0,
#                               "UpdateInterval": 0,
#                               },
# "CloudProtection": true
# }';

# retrieve the most recent version number of the extension
$allVersions= (Get-AzureRmVMExtensionImage -Location $location -PublisherName
"Microsoft.Azure.Security" -Type "IaaSAntimalware").Version
$versionString = $allVersions[(($allVersions.count)-1).Split(".")[0] + "." +
$allVersions[(($allVersions.count)-1).Split(".")[1]
$VMSS = Get-AzureRmVmss -ResourceGroupName $resourceGroupName -VMScaleSetName
$vmScaleSetName
Add-AzureRmVmssExtension -VirtualMachineScaleSet $VMSS -Name "IaaSAntimalware" -
Publisher "Microsoft.Azure.Security" -Type "IaaSAntimalware" -TypeHandlerVersion
$versionString
Update-AzureRmVmss -ResourceGroupName $resourceGroupName -Name $vmScaleSetName -
VirtualMachineScaleSet $VMSS
```

Add Microsoft Antimalware to Azure Cloud Service using Extended Support

The code sample below shows how you can add or configure Microsoft Antimalware to Azure Cloud Service using extended support(CS-ES) via PowerShell cmdlets.

ⓘ Note

Before executing this code sample, you must uncomment the variables and provide appropriate values.

PowerShell

```
# Create Antimalware extension object, where file is the AntimalwareSettings
$xmlconfig = [IO.File]::ReadAllText("C:\path\to\file.xml")
$extension = New-AzCloudServiceExtensionObject -Name "AntimalwareExtension" -Type
"PaaSAntimalware" -Publisher "Microsoft.Azure.Security" -Setting $xmlconfig -
TypeHandlerVersion "1.5" -AutoUpgradeMinorVersion $true

# Get existing Cloud Service
$cloudService = Get-AzCloudService -ResourceGroup "ContosoOrg" -CloudServiceName
"ContosoCS"

# Add Antimalware extension to existing Cloud Service extension object
$cloudService.ExtensionProfile.Extension = $cloudService.ExtensionProfile.Extension
+ $extension

# Update Cloud Service
$cloudService | Update-AzCloudService
```

Here's an example of the private configuration XML file

```
<?xml version="1.0" encoding="utf-8"?>
<AntimalwareConfig
  xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <AntimalwareEnabled>true</AntimalwareEnabled>
  <RealtimeProtectionEnabled>true</RealtimeProtectionEnabled>
  <ScheduledScanSettings isEnabled="true" day="1" time="120" scanType="Full" />
  <Exclusions>
    <Extensions>
      <Extension>.ext1</Extension>
      <Extension>.ext2</Extension>
    </Extensions>
    <Paths>
      <Path>c:\excluded-path-1</Path>
      <Path>c:\excluded-path-2</Path>
    </Paths>
  </Exclusions>
</AntimalwareConfig>
```

```
</Paths>
<Processes>
  <Process>excludedproc1.exe</Process>
  <Process>excludedproc2.exe</Process>
</Processes>
</Exclusions>
</AntimalwareConfig>
```

Add Microsoft Antimalware for Azure Arc-enabled servers

The code sample below shows how you can add Microsoft Antimalware for Azure Arc-enabled servers via PowerShell cmdlets.

⚠ Note

Before executing this code sample, you must uncomment the variables and provide appropriate values.

PowerShell

```
#Before using Azure PowerShell to manage VM extensions on your hybrid server managed
by Azure Arc-enabled servers, you need to install the Az.ConnectedMachine module.
Run the following command on your Azure Arc-enabled server:
#If you have Az.ConnectedMachine installed, please make sure the version is at least
0.4.0
install-module -Name Az.ConnectedMachine
Import-Module -name Az.ConnectedMachine

# specify location, resource group, and VM for the extension
$subscriptionid =" SUBSCRIPTION ID HERE "
$location = " LOCATION HERE " # eg., "Southeast Asia" or "Central US"
$resourceGroupName = " RESOURCE GROUP NAME HERE "
$machineName = "MACHINE NAME HERE "

# Enable Antimalware with default policies
$setting = @{"AntimalwareEnabled"=$true}
# Enable Antimalware with custom policies
$setting2 = @{
"AntimalwareEnabled"=$true;
"RealtimeProtectionEnabled"=$true;
"ScheduledScanSettings"= @{
    "isEnabled"=$true;
    "day"=0;
    "time"=120;
    "scanType"="Quick"
};
"Exclusions"= @{"
```

```

        "Extensions"=".ext1, .ext2";
        "Paths"="";
        "Processes"="sample1.exe, sample2.exe"
    };
"SignatureUpdates"= @{
                                "FileSharesSources"="";
                                "FallbackOrder"="";
                                "ScheduleDay"=0;
                                "UpdateInterval"=0;
                                };
"CloudProtection"=$true
}
# Will be prompted to login
Connect-AzAccount
# Enable Antimalware with the policies
New-AzConnectedMachineExtension -Name "IaaSAntimalware" -ResourceGroupName
$resourceGroupName -MachineName $machineName -Location $location -SubscriptionId
$subscriptionid -Publisher "Microsoft.Azure.Security" -Settings $setting -
ExtensionType "IaaSAntimalware"

```

Next steps

- [Microsoft Antimalware for Azure Cloud Services and Virtual Machines](#)
- [Microsoft Defender for Cloud](#)
-

Last updated on 12/03/2025

Azure Virtual Machines security overview

This article provides an overview of core Azure security features for virtual machines.

Azure Virtual Machines lets you deploy a wide range of computing solutions in an agile way. The service supports Microsoft Windows, Linux, Microsoft SQL Server, Oracle, IBM, SAP, and Azure BizTalk Services. You can deploy any workload and any language on nearly any operating system.

Azure VMs provide the flexibility of virtualization without buying and maintaining physical hardware. You can build and deploy applications with the assurance that your data is protected in highly secure datacenters.

With Azure, you can build security-enhanced, compliant solutions that:

- Protect virtual machines from viruses and malware
- Encrypt sensitive data
- Secure network traffic
- Identify and detect threats
- Meet compliance requirements

Trusted launch

[Trusted launch](#) is the default for newly created Generation 2 Azure VMs and Virtual Machine Scale Sets. Trusted launch protects against advanced and persistent attack techniques including boot kits, rootkits, and kernel-level malware.

Trusted launch provides:

- **Secure Boot:** Protects against installation of malware-based rootkits and boot kits by ensuring only signed operating systems and drivers can boot
- **vTPM (virtual Trusted Platform Module):** A dedicated secure vault for keys and measurements that enables attestation and boot integrity verification
- **Boot Integrity Monitoring:** Uses attestation through Microsoft Defender for Cloud to verify boot chain integrity and alert on failures

Trusted launch can be enabled on existing VMs and Virtual Machine Scale Sets. For more information, see [Trusted launch for Azure virtual machines](#).

Confidential computing

[Azure confidential computing](#) protects data while in use through hardware-based trusted execution environments. Confidential VMs use AMD SEV-SNP technology to create a

hardware-enforced boundary between your application and the virtualization stack.

Confidential VMs provide:

- **Hardware-based isolation:** Between virtual machines, hypervisor, and host management code
- **Confidential OS disk encryption:** Binds disk encryption keys to the VM's TPM, making disk content accessible only to the VM
- **Secure key release:** Cryptographic binding between platform attestation and VM encryption keys
- **Attestation:** Customizable policies to ensure host compliance before deployment

For more information, see [Azure confidential VMs](#).

Azure Backup

[Azure Backup](#) is a scalable solution that protects your application data with zero capital investment and minimal operating costs. Application errors can corrupt your data, and human errors can introduce bugs into your applications. With Azure Backup, your virtual machines running Windows and Linux are protected.

Azure Backup provides independent and isolated backups to guard against accidental destruction of data. Backups are stored in a Recovery Services vault with built-in management of recovery points.

For more information, see [What is Azure Backup?](#) and [Azure Backup service FAQ](#).

Azure Site Recovery

[Azure Site Recovery](#) helps orchestrate replication, failover, and recovery of workloads and apps so they're available from a secondary location if your primary location goes down.

Site Recovery:

- **Simplifies BCDR strategy:** Makes it easy to handle replication, failover, and recovery of multiple business workloads and apps from a single location
- **Provides flexible replication:** Replicate workloads running on Hyper-V VMs, VMware VMs, and Windows/Linux physical servers
- **Supports failover and recovery:** Provides test failovers for disaster recovery drills without affecting production environments
- **Eliminates secondary datacenters:** Replicate to Azure, eliminating the cost and complexity of maintaining a secondary site

For more information, see [What is Azure Site Recovery?](#), [How does Azure Site Recovery work?](#), and [What workloads are protected by Azure Site Recovery?](#).

Virtual networking

Virtual machines require network connectivity. Azure requires virtual machines to be connected to an Azure virtual network.

An Azure virtual network is a logical construct built on top of the physical Azure network fabric. Each logical Azure virtual network is isolated from all other Azure virtual networks. This isolation helps ensure that network traffic in your deployments is not accessible to other Microsoft Azure customers.

For more information, see [Azure network security overview](#) and [Virtual Network overview](#).

Security policy management

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats. Defender for Cloud gives you increased visibility into, and control over, the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions.

Defender for Cloud helps you optimize and monitor VM security by:

- Providing security recommendations for virtual machines
- Monitoring the state of your virtual machines
- Providing Microsoft Defender for Servers with advanced threat protection

Microsoft Defender for Servers includes:

- Microsoft Defender for Endpoint integration for endpoint detection and response
- Vulnerability assessment for identifying security weaknesses
- Just-in-time VM access to reduce attack surface
- File integrity monitoring to detect changes to critical files
- Adaptive application controls for approved applications

For more information, see [Introduction to Microsoft Defender for Cloud](#), [Microsoft Defender for Servers](#), and [Microsoft Defender for Cloud frequently asked questions](#).

Compliance

Azure Virtual Machines is certified for FISMA, FedRAMP, HIPAA, PCI DSS Level 1, and other key compliance programs. This certification makes it easier for your Azure applications to meet compliance requirements and for your business to address domestic and international regulatory requirements.

For more information, see [Microsoft Trust Center: Compliance](#) and [Azure compliance documentation](#).

Hardware security module

[Azure Key Vault](#) provides secure storage for keys and secrets. Key Vault offers the option to store your keys in hardware security modules (HSMs) certified to FIPS 140 validated standards.

Your SQL Server encryption keys for backup or transparent data encryption can all be stored in Key Vault with any keys or secrets from your applications. Permissions and access to these protected items are managed through [Microsoft Entra ID](#).

For more information about Azure key management, see [Key management in Azure](#).

Next steps

Learn about [security best practices](#) for VMs and operating systems.

Last updated on 11/07/2025

Automatic Guest Patching for Azure Virtual Machines and Scale Sets

Applies to: ✓ Linux VMs ✓ Windows VMs ✓ Flexible scale sets

Enabling automatic guest patching for your Azure Virtual Machines (VMs) and Scale Sets (VMSS) helps ease update management by safely and automatically patching virtual machines to maintain security compliance, while limiting the blast radius of VMs.

Automatic VM guest patching has the following characteristics:

- Patches classified as *Critical* or *Security* are automatically downloaded and applied on the VM.
- Patches are applied during off-peak hours for IaaS VMs in the VM's time zone.
- Patches are applied during all hours for VMSS Flex.
- Azure manages the patch orchestration and follows [availability-first principles](#).
- Virtual machine health, as determined through platform health signals, is monitored to detect patching failures.
- Application health can be monitored through the [Application Health extension](#).
- Works for all VM sizes.

How does automatic VM guest patching work?

If automatic VM guest patching is enabled on a VM, then the available *Critical* and *Security* patches are downloaded and applied automatically on the VM. This process kicks off automatically every month when new patches are released. Patch assessment and installation are automatic, and the process includes rebooting the VM as configured. The `rebootSetting` parameter on the VM Model takes precedence over settings in another system, such as [Maintenance Configuration](#).

The VM is assessed periodically every few days and multiple times within any 30-day period to determine the applicable patches for that VM. The patches can be installed any day on the VM during off-peak hours for the VM. This automatic assessment ensures that any missing patches are discovered at the earliest possible opportunity.

Patches are installed within 30 days of the monthly patch releases, following availability-first orchestration. Patches are installed only during off-peak hours for the VM, depending on the time zone of the VM. The VM must be running during the off-peak hours for patches to be automatically installed. If a VM is powered off during a periodic assessment, the platform automatically assess and apply patches (if necessary) during the next periodic assessment (usually within a few days) when the VM is powered on.

Definition updates and other patches not classified as *Critical* or *Security* won't be installed through automatic VM guest patching. To install patches with other patch classifications or schedule patch installation within your own custom maintenance window, you can use [Azure Update Manager](#).

Enabling Automatic Guest Patching on single-instance VMs or Virtual Machine Scale Sets in Flexible orchestration mode allows the Azure platform to update your fleet in phases. Phased deployment follows Azure's [Safe Deployment Practices](#) and reduces the impact radius if any issues are identified with the latest update. [Health monitoring](#) is recommended for single instance VMs and required for Virtual Machine Scale Sets in Flexible orchestration mode to detect issues any issues with the update.

Availability-first Updates

Azure orchestrates the patch installation process across all public and private clouds for VMs that have enabled Automatic Guest Patching. The orchestration follows availability-first principles across different levels of availability provided by Azure.

For a group of virtual machines undergoing an update, the Azure platform orchestrates updates:

Across regions:

- A monthly update is orchestrated across Azure globally in a phased manner to prevent global deployment failures.
- A phase can have one or more regions, and an update moves to the next phases only if eligible VMs in a phase update successfully.
- Geo-paired regions aren't updated concurrently and can't be in the same regional phase.
- The success of an update is measured by tracking the VM's health post update. VM Health is tracked through platform health indicators for the VM.

Within a region:

- VMs in different Availability Zones aren't updated concurrently with the same update.
- VMs that aren't part of an availability set are batched on a best effort basis to avoid concurrent updates for all VMs in a subscription.

Within an availability set:

- All VMs in a common availability set aren't updated concurrently.
- VMs in a common availability set are updated within Update Domain boundaries and VMs across multiple Update Domains aren't updated concurrently.

- In an Update Domain, no more than 20% of the VMs within an availability set are be updated at a time. For availability sets with fewer than 10 VMs, VMs update one at a time within an Update Domain.

Restricting the number of concurrently patched VMs across regions, within a region, or within an availability set limits the impact of a faulty patch on a given set of VMs. With health monitoring, any potential issues are flagged before they impact the entire workload.

The patch installation date for a given VM may vary month-to-month, as a specific VM may be picked up in a different batch between monthly patching cycles.

Which patches are installed?

The patches installed depend on the rollout stage for the VM. Every month, a new global rollout is started where all security and critical patches assessed for an individual VM are installed for that VM. The rollout is orchestrated across all Azure regions in batches.

The exact set of patches to be installed vary based on the VM configuration, including OS type, and assessment timing. It's possible for two identical VMs in different regions to get different patches installed if there are more or less patches available when the patch orchestration reaches different regions at different times. Similarly, but less frequently, VMs within the same region but assessed at different times (due to different Availability Zone or Availability Set batches) might get different patches.

As the Automatic VM Guest Patching doesn't configure the patch source, two similar VMs configured to different patch sources, such as public repository vs private repository, may also see a difference in the exact set of patches installed.

For OS types that release patches on a fixed cadence, VMs configured to the public repository for the OS can expect to receive the same set of patches across the different rollout phases in a month. For example, Windows VMs configured to the public Windows Update repository.

As a new rollout is triggered every month, a VM receives at least one patch rollout every month if the VM is powered on during off-peak hours. This process ensures that the VM is patched with the latest available security and critical patches on a monthly basis. To ensure consistency in the set of patches installed, you can configure your VMs to assess and download patches from your own private repositories.

Supported OS images

 **Important**

Automatic VM guest patching, on-demand patch assessment and on-demand patch installation are supported only on VMs created from images with the exact combination of publisher, offer and sku from the below supported OS images list. Custom images or any other publisher, offer, sku combinations aren't supported. More images are added periodically. Don't see your SKU in the list? Request support by filing out [Image Support Request](#) [↗].

Supported Windows Images (Hotpatchable)

 Expand table

Publisher	OS Offer	Skus
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-core
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-smalldisk
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-core-smalldisk
MicrosoftWindowsServer	MicrosoftServerOperatingSystems-previews	windows-server-2025-azure-edition-hotpatch
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core
MicrosoftWindowsServer	MicrosoftServerOperatingSystems-previews	windows-server-2022-azure-edition-core-preview
MicrosoftWindowsServer	WindowsServerHotpatch-previews	windows-server-2022-azure-edition-hotpatch
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-hotpatch
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-hotpatch-smalldisk

Supported Windows Images (non-Hotpatchable)

Publisher	OS Offer	Sku
Center-for-Internet-Security-Inc	cis-windows-server-2019-v1-0-0-l2	cis-ws2019-l2
Center-for-Internet-Security-Inc	cis-windows-server	cis-windows-server2019-l1-gen1
Center-for-Internet-Security-Inc	cis-windows-server	cis-windows-server2022-l1-gen1
Center-for-Internet-Security-Inc	cis-windows-server	cis-windows-server2022-l1-gen2
Center-for-Internet-Security-Inc	cis-windows-server	cis-windows-server2022-l2-gen2
Center-for-Internet-Security-Inc	cis-windows-server-2022-l2	cis-windows-server-2022-l2-gen2
MicrosoftWindowsServer	WindowsServer	2008-R2-SP1
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2012-R2-Datacenter-smalldisk-g2
MicrosoftWindowsServer	WindowsServer	2016-Datacenter
MicrosoftWindowsServer	WindowsServer	2016-datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2016-Datacenter-Server-Core
MicrosoftWindowsServer	WindowsServer	2016-datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2016-datacenter-with-containers
MicrosoftWindowsServer	WindowsServer	2019-Datacenter
MicrosoftWindowsServer	WindowsServer	2019-Datacenter-Core
MicrosoftWindowsServer	WindowsServer	2019-datacenter-gensecond
MicrosoftWindowsServer	WindowsServer	2019-datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2019-datacenter-smalldisk-g2

Publisher	OS Offer	Sku
MicrosoftWindowsServer	WindowsServer	2019-datacenter-with-containers
MicrosoftWindowsServer	WindowsServer	2022-datacenter
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-core-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-hotpatch
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-hotpatch-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-azure-edition-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-core
MicrosoftWindowsServer	WindowsServer	2022-datacenter-core-g2
MicrosoftWindowsServer	WindowsServer	2022-datacenter-g2
MicrosoftWindowsServer	WindowsServer	2022-datacenter-smalldisk
MicrosoftWindowsServer	WindowsServer	2022-datacenter-smalldisk-g2
MicrosoftWindowsServer	WindowsServerHotpatch-previews	windows-server-2022-azure-edition-hotpatch
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-core
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-core-smalldisk
MicrosoftWindowsServer	WindowsServer	2025-datacenter-azure-edition-smalldisk
MicrosoftWindowsServer	WindowsServer	2025-datacenter
MicrosoftWindowsServer	WindowsServer	2025-datacenter-core
MicrosoftWindowsServer	WindowsServer	2025-datacenter-core-smalldisk

Publisher	OS Offer	Skus
MicrosoftWindowsServer	WindowsServer	2025-datacenter-smalldisk
MicrosoftWindowsServer	MicrosoftServerOperatingSystems-previews	windows-server-2025-azure-edition-hotpatch

Supported Linux Images

 Expand table

Publisher	OS Offer	Skus
AlmaLinux	almalinux-hpc	8-hpc-gen2
AlmaLinux	almalinux-hpc	8_5-hpc
AlmaLinux	almalinux-hpc	8_5-hpc-gen2
AlmaLinux	almalinux-hpc	8_6-hpc
AlmaLinux	almalinux-hpc	8_6-hpc-gen2
AlmaLinux	almalinux-hpc	8_7-hpc-gen1
AlmaLinux	almalinux-hpc	8_7-hpc-gen2
AlmaLinux	almalinux-hpc	8_10-hpc-gen1
AlmaLinux	almalinux-hpc	8_10-hpc-gen2
Canonical	UbuntuServer	16.04-LTS
Canonical	UbuntuServer	16.04.0-LTS
Canonical	UbuntuServer	18.04-LTS
Canonical	UbuntuServer	18_04-LTS-arm64
Canonical	UbuntuServer	18_04-LTS-gen2
Canonical	ubuntu-24_04-lts	cvm
Canonical	ubuntu-24_04-lts	server
Canonical	ubuntu-24_04-lts	server-arm64
Canonical	ubuntu-24_04-lts	server-gen1
Canonical	ubuntu-24_04-lts	ubuntu-pro

Publisher	OS Offer	Sku
Canonical	ubuntu-24_04-lts	ubuntu-pro-cvm
Canonical	0001-com-ubuntu-confidential-vm-focal	20_04-lts-cvm
Canonical	0001-com-ubuntu-confidential-vm-jammy	22_04-lts-cvm
Canonical	0001-com-ubuntu-pro-bionic	pro-18_04-lts
Canonical	0001-com-ubuntu-pro-focal	pro-20_04-lts
Canonical	0001-com-ubuntu-pro-focal	pro-20_04-lts-gen2
Canonical	0001-com-ubuntu-pro-jammy	pro-22_04-lts-gen2
Canonical	0001-com-ubuntu-server-focal	20_04-lts
Canonical	0001-com-ubuntu-server-focal	20_04-lts-arm64
Canonical	0001-com-ubuntu-server-focal	20_04-lts-gen2
Canonical	0001-com-ubuntu-server-jammy	22_04-lts
Canonical	0001-com-ubuntu-server-jammy	22_04-lts-arm64
Canonical	0001-com-ubuntu-server-jammy	22_04-lts-gen2
Center-for-Internet-Security-Inc	cis-rhel-7-v2-2-0-l1	cis-redhat7-l1
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat7-l2-gen1
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat8-l1-gen1
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat8-l1-gen2
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat8-l2-gen1
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat9-l1-gen2
Center-for-Internet-Security-Inc	cis-rhel	cis-redhat9-l2-gen2
Center-for-Internet-Security-Inc	cis-ubuntu-linux-2004-l1	cis-ubuntu-linux-2004-l1

Publisher	OS Offer	Sku
Center-for-Internet-Security-Inc	cis-ubuntu-linux-2204-l1	cis-ubuntu-linux-2204-l1
Center-for-Internet-Security-Inc	cis-ubuntu	cis-ubuntulinux2004-l1-gen1
Center-for-Internet-Security-Inc	cis-ubuntu	cis-ubuntulinux2204-l1-gen2
Center-for-Internet-Security-Inc	cis-ubuntu	cis-ubuntulinux2404-l1-gen1
Center-for-Internet-Security-Inc	cis-ubuntu	cis-ubuntulinux2404-l1-gen2
MicrosoftCblMariner	azure-linux-3	azure-linux-3-gen2
MicrosoftCblMariner	cbl-mariner	cbl-mariner-1
MicrosoftCblMariner	cbl-mariner	1-gen2
MicrosoftCblMariner	cbl-mariner	cbl-mariner-2
MicrosoftCblMariner	cbl-mariner	cbl-mariner-2-arm64
MicrosoftCblMariner	cbl-mariner	cbl-mariner-2-gen2
MicrosoftCblMariner	cbl-mariner	cbl-mariner-2-gen2-fips
Microsoft-AKS	aks	aks-engine-ubuntu-1804-202112
Microsoft-DSVM	aml-workstation	ubuntu-20
Microsoft-DSVM	aml-workstation	ubuntu-20-gen2
Microsoft-DSVM	ubuntu-hpc	1804
Microsoft-DSVM	ubuntu-hpc	1804-ncv4
Microsoft-DSVM	ubuntu-hpc	2004
Microsoft-DSVM	ubuntu-hpc	2004-preview-ndv5
Microsoft-DSVM	ubuntu-hpc	2204
Microsoft-DSVM	ubuntu-hpc	2204-preview-ndv5
Redhat	RHEL	7.2
Redhat	RHEL	7.3

Publisher	OS Offer	Sku
Redhat	RHEL	7.4
Redhat	RHEL	7.5
Redhat	RHEL	7.6
Redhat	RHEL	7.7
Redhat	RHEL	7.8
Redhat	RHEL	7_9
Redhat	RHEL	7-lvm
Redhat	RHEL	7-raw
Redhat	RHEL	8
Redhat	RHEL	8.1
Redhat	RHEL	81gen2
Redhat	RHEL	8.2
Redhat	RHEL	82gen2
Redhat	RHEL	8_3
Redhat	RHEL	83-gen2
Redhat	RHEL	8_4
Redhat	RHEL	84-gen2
Redhat	RHEL	8_5
Redhat	RHEL	85-gen2
Redhat	RHEL	8_6
Redhat	RHEL	86-gen2
Redhat	RHEL	8_7
Redhat	RHEL	8_8
Redhat	RHEL	8_10
Redhat	RHEL	8-lvm
Redhat	RHEL	8-lvm-gen2

Publisher	OS Offer	Sku
Redhat	RHEL-RAW	8-raw
Redhat	RHEL-RAW	8-raw-gen2
Redhat	RHEL	9_0
Redhat	RHEL	9_1
Redhat	RHEL	9_2
Redhat	RHEL	9_3
Redhat	RHEL	9_4
Redhat	RHEL	9-lvm
Redhat	RHEL	9-lvm-gen2
Redhat	RHEL-ARM64	8_6-arm64
Redhat	RHEL-ARM64	9_0-arm64
Redhat	RHEL-ARM64	9_1-arm64
Redhat	RHEL-SAP-APPS	86sapapps-gen2
Redhat	RHEL-SAP-APPS	88sapapps-gen2
Redhat	RHEL-SAP-HA	86sapha-gen2
Redhat	RHEL-SAP-HA	88sapha-gen2
SUSE	sles-12-sp5	gen1
SUSE	sles-12-sp5	gen2
SUSE	sles-15-sp2	gen1
SUSE	sles-15-sp2	gen2
SUSE	sles-sap-15-sp4	gen2
SUSE	sles-sap-15-sp5	gen2

Patch orchestration modes

VMs on Azure now support the following patch orchestration modes:

AutomaticByPlatform (Azure-orchestrated patching):

- This mode is supported for both Linux and Windows VMs.
- This mode enables automatic VM guest patching for the virtual machine and subsequent patch installation is orchestrated by Azure.
- During the installation process, this mode [assesses the VM](#) for available patches and save the details in [Azure Resource Graph](#).
- This mode is required for availability-first patching.
- This mode is only supported for VMs that are created using the supported OS platform images above.
- For Windows VMs, setting this mode also disables the native Automatic Updates on the Windows virtual machine to avoid duplication.
- To use this mode on Linux VMs, set the property `osProfile.linuxConfiguration.patchSettings.patchMode=AutomaticByPlatform` in the VM template.
- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByPlatform` in the VM template.
- Enabling this mode sets the Registry Key `SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU\NoAutoUpdate` to 1

AutomaticByOS:

- This mode is supported only for Windows VMs.
- This mode enables Automatic Updates on the Windows virtual machine, and patches are installed on the VM through Automatic Updates.
- This mode doesn't support availability-first patching.
- This mode is set by default if no other patch mode is specified for a Windows VM.
- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.enableAutomaticUpdates=true`, and set the property `osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByOS` in the VM template.
- Enabling this mode sets the Registry Key `SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU\NoAutoUpdate` to 0

Manual:

- This mode is supported only for Windows VMs.
- This mode disables Automatic Updates on the Windows virtual machine. When deploying a VM using CLI or PowerShell, setting `--enable-auto-updates` to `false` also sets `patchMode` to `manual` and disables Automatic Updates.
- This mode doesn't support availability-first patching.
- This mode should be set when using custom patching solutions.

- To use this mode on Windows VMs, set the property `osProfile.windowsConfiguration.enableAutomaticUpdates=false`, and set the property `osProfile.windowsConfiguration.patchSettings.patchMode=Manual` in the VM template.
- Enabling this mode sets the Registry Key `SOFTWARE\Policies\Microsoft\Windows\WindowsUpdate\AU\NoAutoUpdate` to 1

ImageDefault:

- This mode is supported only for Linux VMs.
- This mode doesn't support availability-first patching.
- This mode honors the default patching configuration in the image used to create the VM.
- This mode is set by default if no other patch mode is specified for a Linux VM.
- To use this mode on Linux VMs, set the property `osProfile.linuxConfiguration.patchSettings.patchMode=ImageDefault` in the VM template.

ⓘ Note

For Windows VMs, the property `osProfile.windowsConfiguration.enableAutomaticUpdates` can only be set when the VM is first created. This impacts certain patch mode transitions. Switching between AutomaticByPlatform and Manual modes is supported on VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=false`. Similarly switching between AutomaticByPlatform and AutomaticByOS modes is supported on VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=true`. Switching between AutomaticByOS and Manual modes isn't supported. Azure recommends that [Assessment Mode](#) be enabled on a VM even if Azure Orchestration isn't enabled for patching. This allows the platform to assess the VM every 24 hours for any pending updates, and save the details in [Azure Resource Graph](#). The platform performs assessment to report consolidated results when the machine's desired patch configuration state is applied or confirmed. This is reported as a 'Platform'-initiated assessment.

Requirements for enabling automatic VM guest patching

- The virtual machine must have the Azure VM Agent for [Windows](#) or [Linux](#) installed.
- For Linux VMs, the Azure Linux agent must be version 2.2.53.1 or higher. [Update the Linux agent](#) if the current version is lower than the required version.
- For Windows VMs, the Windows Update service must be running on the virtual machine.

- The virtual machine must be able to access the configured update endpoints. If your virtual machine is configured to use private repositories for Linux or Windows Server Update Services (WSUS) for Windows VMs, the relevant update endpoints must be accessible.
- Use Compute API version 2021-03-01 or higher to access all functionality including on-demand assessment and on-demand patching.
- Custom images aren't currently supported.
- VMSS Flexible Orchestration requires the installation of [Application Health extension](#). This is optional for IaaS VMs.

Enable automatic VM guest patching

Automatic VM guest patching can be enabled on any Windows or Linux VM that is created from a supported platform image.

REST API for Linux VMs

The following example describes how to enable automatic VM guest patching:

```
PUT on
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.C
ompute/virtualMachines/myVirtualMachine?api-version=2020-12-01`
```

JSON

```
{
  "location": "<location>",
  "properties": {
    "osProfile": {
      "linuxConfiguration": {
        "provisionVMAgent": true,
        "patchSettings": {
          "patchMode": "AutomaticByPlatform"
        }
      }
    }
  }
}
```

REST API for Windows VMs

The following example describes how to enable automatic VM guest patching:

PUT on

```
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.Compute/virtualMachines/myVirtualMachine?api-version=2020-12-01`
```

JSON

```
{
  "location": "<location>",
  "properties": {
    "osProfile": {
      "windowsConfiguration": {
        "provisionVMAgent": true,
        "enableAutomaticUpdates": true,
        "patchSettings": {
          "patchMode": "AutomaticByPlatform"
        }
      }
    }
  }
}
```

Azure PowerShell when creating a Windows VM

Use the [Set-AzVMOperatingSystem](#) cmdlet to enable automatic VM guest patching when creating a VM.

Azure PowerShell

```
Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -
Credential $Credential -ProvisionVMAgent -EnableAutoUpdate -PatchMode
"AutomaticByPlatform"
```

Azure PowerShell when updating a Windows VM

Use the [Set-AzVMOperatingSystem](#) and [Update-AzVM](#) cmdlet to enable automatic VM guest patching on an existing VM.

Azure PowerShell

```
$VirtualMachine = Get-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM"
Set-AzVMOperatingSystem -VM $VirtualMachine -PatchMode "AutomaticByPlatform"
Update-AzVM -ResourceGroupName "myResourceGroup" -VM $VirtualMachine
```

Azure CLI for Windows VMs

Use [az vm create](#) to enable automatic VM guest patching when creating a new VM. The following example configures automatic VM guest patching for a VM named *myVM* in the resource group named *myResourceGroup*:

Azure CLI

```
az vm create --resource-group myResourceGroup --name myVM --image Win2019Datacenter
--enable-agent --enable-auto-update --patch-mode AutomaticByPlatform
```

To modify an existing VM, use [az vm update](#)

Azure CLI

```
az vm update --resource-group myResourceGroup --name myVM --set
osProfile.windowsConfiguration.enableAutomaticUpdates=true
osProfile.windowsConfiguration.patchSettings.patchMode=AutomaticByPlatform
```

Azure portal

When creating a VM using the Azure portal, patch orchestration modes can be set under the **Management** tab for both Linux and Windows.

Auto-shutdown

Enable auto-shutdown ? ☒

Shutdown time ?

Time zone ?

Notification before shutdown ?

Email * ?

Backup

Enable backup ?

Guest OS updates

Patch orchestration options ?

? Some patch orchestration options are not available for this image. [Learn more](#)

[Review + create](#) [< Previous](#) [Next : Advanced >](#)

Enablement and assessment

? Note

It can take more than three hours to enable automatic VM guest updates on a VM, as the enablement is completed during the VM's off-peak hours. As assessment and patch installation occur only during off-peak hours, your VM must also be running during off-peak hours to apply patches.

When automatic VM guest patching is enabled for a VM, a VM extension of type `Microsoft.CPlat.Core.LinuxPatchExtension` is installed on a Linux VM or a VM extension of type `Microsoft.CPlat.Core.WindowsPatchExtension` is installed on a Windows VM. This extension doesn't need to be manually installed or updated, as this extension is managed by the Azure platform as part of the automatic VM guest patching process.

It can take more than three hours to enable automatic VM guest updates on a VM, as the enablement is completed during the VM's off-peak hours. The extension is also installed and updated during off-peak hours for the VM. If the VM's off-peak hours end before enablement can be completed, the enablement process resumes during the next available off-peak time.

The platform makes periodic patching configuration calls to ensure alignment when model changes are detected on IaaS VMs or scale sets in Flexible orchestration. Certain model changes such as, but not limited to, updating assessment mode, patch mode, and extension update may trigger a patching configuration call.

Automatic updates are disabled in most scenarios, and patch installation is done through the extension going forward. The following conditions apply.

- If a Windows VM previously had Automatic Windows Update turned on through the AutomaticByOS patch mode, then Automatic Windows Update is turned off for the VM when the extension is installed.
- For Ubuntu VMs, the default automatic updates are disabled automatically when Automatic VM Guest Patching completes enablement.
- For RHEL, automatic updates need to be manually disabled. Execute:

Bash

```
sudo systemctl stop packagekit
```

Bash

```
sudo systemctl mask packagekit
```

To verify whether automatic VM guest patching has completed and the patching extension is installed on the VM, you can review the VM's instance view. If the enablement process is complete, the extension is installed and the assessment results for the VM are available under

`patchStatus`. The VM's instance view can be accessed through multiple ways as described below.

REST API

```
GET on
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.C
ompute/virtualMachines/myVirtualMachine/instanceView?api-version=2020-12-01`
```

Azure PowerShell

Use the `Get-AzVM` cmdlet with the `-Status` parameter to access the instance view for your VM.

Azure PowerShell

```
Get-AzVM -ResourceGroupName "myResourceGroup" -Name "myVM" -Status
```

PowerShell currently only provides information on the patch extension. Information about `patchStatus` will also be available soon through PowerShell.

Azure CLI

Use `az vm get-instance-view` to access the instance view for your VM.

Azure CLI

```
az vm get-instance-view --resource-group myResourceGroup --name myVM
```

Understanding the patch status for your VM

The `patchStatus` section of the instance view response provides details on the latest assessment and the last patch installation for your VM.

The assessment results for your VM can be reviewed under the `availablePatchSummary` section. An assessment is periodically conducted for a VM that has automatic VM guest patching enabled. The count of available patches after an assessment is provided under `criticalAndSecurityPatchCount` and `otherPatchCount` results. Automatic VM guest patching installs all patches assessed under the *Critical* and *Security* patch classifications. Any other assessed patch is skipped.

The patch installation results for your VM can be reviewed under the `lastPatchInstallationSummary` section. This section provides details on the last patch installation attempt on the VM, including the number of patches that were installed, pending, failed, or skipped. Patches are installed only during the off-peak hours maintenance window for the VM. Pending and failed patches are automatically retried during the next off-peak hours maintenance window.

Disable automatic VM guest patching

Automatic VM guest patching can be disabled by changing the [patch orchestration mode](#) for the VM.

To disable automatic VM guest patching on a Linux VM, change the patch mode to `ImageDefault`.

To enable automatic VM guest patching on a Windows VM, the property `osProfile.windowsConfiguration.enableAutomaticUpdates` determines which patch modes can be set on the VM and this property can only be set when the VM is first created. This impacts certain patch mode transitions:

- For VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=false`, disable automatic VM guest patching by changing the patch mode to `Manual`.
- For VMs that have `osProfile.windowsConfiguration.enableAutomaticUpdates=true`, disable automatic VM guest patching by changing the patch mode to `AutomaticByOS`.
- Switching between `AutomaticByOS` and `Manual` modes isn't supported.

Use the examples from the [enablement](#) section above in this article for API, PowerShell and CLI usage examples to set the required patch mode.

On-demand patch assessment

If automatic VM guest patching is already enabled for your VM, a periodic patch assessment is performed on the VM during the VM's off-peak hours. This process is automatic and the results of the latest assessment can be reviewed through the VM's instance view as described earlier in this document. You can also trigger an on-demand patch assessment for your VM at any time. Patch assessment can take a few minutes to complete and the status of the latest assessment is updated on the VM's instance view.

Note

On-demand patch assessment doesn't automatically trigger patch installation. If you have enabled automatic VM guest patching, then the assessed and applicable patches for the VM are installed during the VM's off-peak hours, following the availability-first patching process described earlier in this document.

REST API

Use the [Assess Patches](#) API to assess available patches for your virtual machine.

```
POST on  
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.C  
ompute/virtualMachines/myVirtualMachine/assessPatches?api-version=2020-12-01`
```

Azure PowerShell

Use the [Invoke-AzVmPatchAssessment](#) cmdlet to assess available patches for your virtual machine.

Azure PowerShell

```
Invoke-AzVmPatchAssessment -ResourceGroupName "myResourceGroup" -VMName "myVM"
```

Azure CLI

Use [az vm assess-patches](#) to assess available patches for your virtual machine.

Azure CLI

```
az vm assess-patches --resource-group myResourceGroup --name myVM
```

On-demand patch installation

If automatic VM guest patching is already enabled for your VM, a periodic patch installation of Security and Critical patches is performed on the VM during the VM's off-peak hours. This process is automatic and the results of the latest installation can be reviewed through the VM's instance view as described earlier in this document.

You can also trigger an on-demand patch installation for your VM at any time. Patch installation can take a few minutes to complete and the status of the latest installation is

updated on the VM's instance view.

You can use on-demand patch installation to install all patches of one or more patch classifications. You can also choose to include or exclude specific packages for Linux or specific KB IDs for Windows. When triggering an on-demand patch installation, ensure that you specify at least one patch classification or at least one patch (package for Linux, KB ID for Windows) in the inclusion list.

REST API

Use the [Install Patches](#) API to install patches on your virtual machine.

```
POST on
`/subscriptions/subscription_id/resourceGroups/myResourceGroup/providers/Microsoft.C
ompute/virtualMachines/myVirtualMachine/installPatches?api-version=2020-12-01`
```

Example request body for Linux:

JSON

```
{
  "maximumDuration": "PT1H",
  "Setting": "IfRequired",
  "linuxParameters": {
    "classificationsToInclude": [
      "Critical",
      "Security"
    ]
  }
}
```

Example request body for Windows:

JSON

```
{
  "maximumDuration": "PT1H",
  "rebootSetting": "IfRequired",
  "windowsParameters": {
    "classificationsToInclude": [
      "Critical",
      "Security"
    ]
  }
}
```

Azure PowerShell

Use the [Invoke-AzVmInstallPatch](#) cmdlet to install patches on your virtual machine.

Example to install certain packages on a Linux VM:

Azure PowerShell

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -
MaximumDuration "PT90M" -RebootSetting "Always" -Linux -
ClassificationToIncludeForLinux "Security" -PackageNameMaskToInclude ["package123"]
-PackageNameMaskToExclude ["package567"]
```

Example to install all Critical patches on a Windows VM:

Azure PowerShell

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -
MaximumDuration "PT2H" -RebootSetting "Never" -Windows -
ClassificationToIncludeForWindows Critical
```

Example to install all Security patches on a Windows VM, while including and excluding patches with specific KB IDs and excluding any patch that requires a reboot:

Azure PowerShell

```
Invoke-AzVmInstallPatch -ResourceGroupName "myResourceGroup" -VMName "myVM" -
MaximumDuration "PT90M" -RebootSetting "Always" -Windows -
ClassificationToIncludeForWindows "Security" -KBNumberToInclude ["KB1234567",
"KB123567"] -KBNumberToExclude ["KB1234702", "KB1234802"] -ExcludeKBsRequiringReboot
```

Azure CLI

Use [az vm install-patches](#) to install patches on your virtual machine.

Example to install all Critical patches on a Linux VM:

Azure CLI

```
az vm install-patches --resource-group myResourceGroup --name myVM --maximum-
duration PT2H --reboot-setting IfRequired --classifications-to-include-linux
Critical
```

Example to install all Critical and Security patches on a Windows VM, while excluding any patch that requires a reboot:

```
az vm install-patches --resource-group myResourceGroup --name myVM --maximum-duration PT2H --reboot-setting IfRequired --classifications-to-include-win Critical Security --exclude-kbs-requiring-reboot true
```

Strict Safe Deployment on Canonical Images

[Microsoft and Canonical have partnered](#) to make it easier for our customers to stay current with Linux OS updates and increase the security and resiliency of their Ubuntu workloads on Azure. By using Canonical's snapshot service, Azure now applies the same set of Ubuntu updates consistently to your fleet across regions.

Azure stores the package related updates within the customer repository for up to 90 days, depending on the available space. This allows customers to update their fleet leveraging Strict Safe Deployment for VMs that are up to three months behind on updates.

There's no action required for customers that have enabled Auto Patching. The platform installs a package that is snapped to a point-in-time by default. In the event a snapshot-based update can't be installed, Azure applies the latest package on the VM to ensure the VM remains secure. The point-in-time updates are consistent on all VMs across regions to ensure homogeneity. Customers can view the published date information related to the applied update in [Azure Resource Graph](#) and the [Instance View](#) of the VM.

Image End-of-Life (EOL)

Publishers may no longer support generating new updates for their images after a certain date. This is commonly referred to as End-of-life (EOL) for the image. Azure doesn't recommend using images after their EOL date, since it exposes the service to security vulnerabilities or performance issues. The Azure Guest Patching Service (AzGPS) communicates necessary steps for customers and impacted partners. AzGPS removes the image from the support list after the EOL date. VMs that use an end of life image on Azure might continue to work beyond their date. However, any issues experienced by these VMs aren't eligible for support.

Next steps

[Learn more about creating and managing Windows virtual machines](#)

Security best practices for IaaS workloads in Azure

This article describes security best practices for VMs and operating systems.

The best practices are based on a consensus of opinion, and they work with current Azure platform capabilities and feature sets. Because opinions and technologies can change over time, this article will be updated to reflect those changes.

In most infrastructure as a service (IaaS) scenarios, [Azure virtual machines \(VMs\)](#) are the main workload for organizations that use cloud computing. This fact is evident in [hybrid scenarios](#) where organizations want to slowly migrate workloads to the cloud. In such scenarios, follow the [general security considerations for IaaS](#), and apply security best practices to all your VMs.

Protect VMs by using authentication and access control

The first step in protecting your VMs is to ensure that only authorized users can set up new VMs and access VMs.

ⓘ Note

To improve the security of Linux VMs on Azure, you can integrate with Microsoft Entra authentication. When you use [Microsoft Entra authentication for Linux VMs](#), you centrally control and enforce policies that allow or deny access to the VMs.

Best practice: Control VM access. **Detail:** Use [Azure policies](#) to establish conventions for resources in your organization and create customized policies. Apply these policies to resources, such as [resource groups](#). VMs that belong to a resource group inherit its policies.

If your organization has many subscriptions, you might need a way to efficiently manage access, policies, and compliance for those subscriptions. [Azure management groups](#) provide a level of scope above subscriptions. You organize subscriptions into management groups (containers) and apply your governance conditions to those groups. All subscriptions within a management group automatically inherit the conditions applied to the group. Management groups give you enterprise-grade management at a large scale no matter what type of subscriptions you might have.

Best practice: Reduce variability in your setup and deployment of VMs. **Detail:** Use [Azure Resource Manager](#) templates to strengthen your deployment choices and make it easier to

understand and inventory the VMs in your environment.

Best practice: Secure privileged access. **Detail:** Use a [least privilege approach](#) and built-in Azure roles to enable users to access and set up VMs:

- [Virtual Machine Contributor](#): Can manage VMs, but not the virtual network or storage account to which they are connected.
- [Classic Virtual Machine Contributor](#): Can manage VMs created by using the classic deployment model, but not the virtual network or storage account to which the VMs are connected.
- [Security Admin](#): In Defender for Cloud only: Can view security policies, view security states, edit security policies, view alerts and recommendations, dismiss alerts and recommendations.
- [DevTest Labs User](#): Can view everything and connect, start, restart, and shut down VMs.

Your subscription admins and coadmins can change this setting, making them administrators of all the VMs in a subscription. Be sure that you trust all of your subscription admins and coadmins to log in to any of your machines.

Note

We recommend that you consolidate VMs with the same lifecycle into the same resource group. By using resource groups, you can deploy, monitor, and roll up billing costs for your resources.

Organizations that control VM access and setup improve their overall VM security.

Use Virtual Machine Scale Sets for high availability

If your VM runs critical applications that need to have high availability, we strongly recommend that you use [Virtual Machine Scale Sets](#).

Virtual Machine Scale Sets let you create and manage a group of load balanced VMs. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. Scale sets provide high availability to your applications, and allow you to centrally manage, configure, and update many VMs. There is no cost for the scale set itself, you only pay for each VM instance that you create.

Virtual machines in a scale set can also be deployed into multiple availability zones, a single availability zone, or regionally.

Protect against malware

You should install antimalware protection to help identify and remove viruses, spyware, and other malicious software. You can install [Microsoft Antimalware](#) or a Microsoft partner's endpoint protection solution ([Trend Micro](#) [↗], [Broadcom](#) [↗], [McAfee](#) [↗], [Windows Defender](#) [↗], and [System Center Endpoint Protection](#)).

Microsoft Antimalware includes features like real-time protection, scheduled scanning, malware remediation, signature updates, engine updates, samples reporting, and exclusion event collection. For environments that are hosted separately from your production environment, you can use an antimalware extension to help protect your VMs and cloud services.

You can integrate Microsoft Antimalware and partner solutions with [Microsoft Defender for Cloud](#) for ease of deployment and built-in detections (alerts and incidents).

Best practice: Install an antimalware solution to protect against malware.

Detail: [Install a Microsoft partner solution or Microsoft Antimalware](#)

Best practice: Integrate your antimalware solution with Defender for Cloud to monitor the status of your protection.

Detail: [Manage endpoint protection issues with Defender for Cloud](#)

Manage your VM updates

Azure VMs, like all on-premises VMs, are meant to be user managed. Azure doesn't push Windows updates to them. You need to manage your VM updates.

Best practice: Keep your VMs current.

Detail: Use the [Update Management](#) solution in Azure Automation to manage operating system updates for your Windows and Linux computers that are deployed in Azure, in on-premises environments, or in other cloud providers. You can quickly assess the status of available updates on all agent computers and manage the process of installing required updates for servers.

Computers that are managed by Update Management use the following configurations to perform assessment and update deployments:

- Microsoft Monitoring Agent (MMA) for Windows or Linux
- PowerShell Desired State Configuration (DSC) for Linux
- Automation Hybrid Runbook Worker
- Microsoft Update or Windows Server Update Services (WSUS) for Windows computers

If you use Windows Update, leave the automatic Windows Update setting enabled.

Best practice: Ensure at deployment that images you built include the most recent round of Windows updates.

Detail: Check for and install all Windows updates as a first step of every deployment. This measure is especially important to apply when you deploy images that come from either you or your own library. Although images from the Azure Marketplace are updated automatically by default, there can be a lag time (up to a few weeks) after a public release.

Best practice: Periodically redeploy your VMs to force a fresh version of the OS.

Detail: Define your VM with an [Azure Resource Manager template](#) so you can easily redeploy it. Using a template gives you a patched and secure VM when you need it.

Best practice: Rapidly apply security updates to VMs.

Detail: Enable Microsoft Defender for Cloud (Free tier or Standard tier) to [identify missing security updates and apply them](#).

Best practice: Install the latest security updates.

Detail: Some of the first workloads that customers move to Azure are labs and external-facing systems. If your Azure VMs host applications or services that need to be accessible to the internet, be vigilant about patching. Patch beyond the operating system. Unpatched vulnerabilities on partner applications can also lead to problems that can be avoided if good patch management is in place.

Best practice: Deploy and test a backup solution.

Detail: A backup needs to be handled the same way that you handle any other operation. This is true of systems that are part of your production environment extending to the cloud.

Test and dev systems must follow backup strategies that provide restore capabilities that are similar to what users have grown accustomed to, based on their experience with on-premises environments. Production workloads moved to Azure should integrate with existing backup solutions when possible. Or, you can use [Azure Backup](#) to help address your backup requirements.

Organizations that don't enforce software-update policies are more exposed to threats that exploit known, previously fixed vulnerabilities. To comply with industry regulations, companies must prove that they are diligent and using correct security controls to help ensure the security of their workloads located in the cloud.

Software-update best practices for a traditional datacenter and Azure IaaS have many similarities. We recommend that you evaluate your current software update policies to include VMs located in Azure.

Manage your VM security posture

Cyberthreats are evolving. Safeguarding your VMs requires a monitoring capability that can quickly detect threats, prevent unauthorized access to your resources, trigger alerts, and reduce false positives.

To monitor the security posture of your [Windows](#) and [Linux VMs](#), use [Microsoft Defender for Cloud](#). In Defender for Cloud, safeguard your VMs by taking advantage of the following capabilities:

- Apply OS security settings with recommended configuration rules.
- Identify and download system security and critical updates that might be missing.
- Deploy recommendations for endpoint antimalware protection.
- Validate disk encryption.
- Assess and remediate vulnerabilities.
- Detect threats.

Defender for Cloud can actively monitor for threats, and potential threats are exposed in security alerts. Correlated threats are aggregated in a single view called a security incident.

Defender for Cloud stores data in [Azure Monitor logs](#). Azure Monitor logs provides a query language and analytics engine that gives you insights into the operation of your applications and resources. Data is also collected from [Azure Monitor](#), management solutions, and agents installed on virtual machines in the cloud or on-premises. This shared functionality helps you form a complete picture of your environment.

Organizations that don't enforce strong security for their VMs remain unaware of potential attempts by unauthorized users to circumvent security controls.

Monitor VM performance

Resource abuse can be a problem when VM processes consume more resources than they should. Performance issues with a VM can lead to service disruption, which violates the security principle of availability. This is particularly important for VMs that are hosting IIS or other web servers, because high CPU or memory usage might indicate a denial of service (DoS) attack. It's imperative to monitor VM access not only reactively while an issue is occurring, but also proactively against baseline performance as measured during normal operation.

We recommend that you use [Azure Monitor](#) to gain visibility into your resource's health. Azure Monitor features:

- [Resource diagnostic log files](#): Monitors your VM resources and identifies potential issues that might compromise performance and availability.
- [Azure Diagnostics extension](#): Provides monitoring and diagnostics capabilities on Windows VMs. You can enable these capabilities by including the extension as part of the

Organizations that don't monitor VM performance can't determine whether certain changes in performance patterns are normal or abnormal. A VM that's consuming more resources than normal might indicate an attack from an external resource or a compromised process running in the VM.

Encrypt your virtual hard disk files

We recommend that you encrypt your virtual hard disks (VHDs) to help protect your boot volume and data volumes at rest in storage, along with your encryption keys and secrets.

[Azure Disk Encryption for Linux VMs](#) and [Azure Disk Encryption for Windows VMs](#) helps you encrypt your Linux and Windows IaaS virtual machine disks. Azure Disk Encryption uses the industry-standard [DM-Crypt](#) feature of Linux and the [BitLocker](#) feature of Windows to provide volume encryption for the OS and the data disks. The solution is integrated with [Azure Key Vault](#) to help you control and manage the disk-encryption keys and secrets in your key vault subscription. The solution also ensures that all data on the virtual machine disks are encrypted at rest in Azure Storage.

Following are best practices for using Azure Disk Encryption:

Best practice: Enable encryption on VMs.

Detail: Azure Disk Encryption generates and writes the encryption keys to your key vault. Managing encryption keys in your key vault requires Microsoft Entra authentication. Create a Microsoft Entra application for this purpose. For authentication purposes, you can use either client secret-based authentication or [client certificate-based Microsoft Entra authentication](#).

Best practice: Use a key encryption key (KEK) for an additional layer of security for encryption keys. Add a KEK to your key vault.

Detail: Use the [Add-AzKeyVaultKey](#) cmdlet to create a key encryption key in the key vault. You can also import a KEK from your on-premises hardware security module (HSM) for key management. For more information, see the [Key Vault documentation](#). When a key encryption key is specified, Azure Disk Encryption uses that key to wrap the encryption secrets before writing to Key Vault. Keeping an escrow copy of this key in an on-premises key management HSM offers additional protection against accidental deletion of keys.

Best practice: Take a [snapshot](#) and/or backup before disks are encrypted. Backups provide a recovery option if an unexpected failure happens during encryption.

Detail: VMs with managed disks require a backup before encryption occurs. After a backup is made, you can use the [Set-AzVMDiskEncryptionExtension](#) cmdlet to encrypt managed disks

by specifying the `-skipVmBackup` parameter. For more information about how to back up and restore encrypted VMs, see the [Azure Backup](#) article.

Best practice: To make sure the encryption secrets don't cross regional boundaries, Azure Disk Encryption needs the key vault and the VMs to be located in the same region.

Detail: Create and use a key vault that is in the same region as the VM to be encrypted.

When you apply Azure Disk Encryption, you can satisfy the following business needs:

- IaaS VMs are secured at rest through industry-standard encryption technology to address organizational security and compliance requirements.
- IaaS VMs start under customer-controlled keys and policies, and you can audit their usage in your key vault.

Restrict direct internet connectivity

Monitor and restrict VM direct internet connectivity. Attackers constantly scan public cloud IP ranges for open management ports and attempt "easy" attacks like common passwords and known unpatched vulnerabilities. The following table lists best practices to help protect against these attacks:

Best practice: Prevent inadvertent exposure to network routing and security.

Detail: Use Azure RBAC to ensure that only the central networking group has permission to networking resources.

Best practice: Identify and remediate exposed VMs that allow access from "any" source IP address.

Detail: Use Microsoft Defender for Cloud. Defender for Cloud will recommend that you restrict access through internet-facing endpoints if any of your network security groups has one or more inbound rules that allow access from "any" source IP address. Defender for Cloud will recommend that you edit these inbound rules to [restrict access](#) to source IP addresses that actually need access.

Best practice: Restrict management ports (RDP, SSH).

Detail: [Just-in-time \(JIT\) VM access](#) can be used to lock down inbound traffic to your Azure VMs, reducing exposure to attacks while providing easy access to connect to VMs when needed. When JIT is enabled, Defender for Cloud locks down inbound traffic to your Azure VMs by creating a network security group rule. You select the ports on the VM to which inbound traffic will be locked down. These ports are controlled by the JIT solution.

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

The following resources are available to provide more general information about Azure security and related Microsoft services:

- [Azure Security Team Blog](#) - for up to date information on the latest in Azure Security
- [Microsoft Security Response Center](#)  - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com

Last updated on 12/03/2025

Security Recommendations for Azure Marketplace Images

Prior to uploading images to the Azure Marketplace, your image must be updated with several security configuration requirements. These requirements help maintain a high level of security for partner solution images across the Azure Marketplace.

Make sure to run a security vulnerability detection on your image Prior to submitting it to the Azure Marketplace. If you detect a security vulnerability in your own already published image, you must inform your customers in a timely manner both of the vulnerability's details and how to correct it in current deployments.

Linux and open source OS images

 Expand table

Category	Check
Security	Install all the latest security patches for the Linux distribution.
Security	Follow industry guidelines to secure the VM image for the specific Linux distribution.
Security	Limit the attack surface by keeping minimal footprint with only necessary Windows Server roles, features, services, and networking ports.
Security	Scan source code and resulting VM image for malware.
Security	The VHD image only includes necessary locked accounts that do not have default passwords that would allow interactive login; no back doors.
Security	Disable firewall rules unless application functionally relies on them, such as a firewall appliance.
Security	Remove all sensitive information from the VHD image, such as test SSH keys, known hosts file, log files, and unnecessary certificates.
Security	Avoid using LVM. LVM is Vulnerable to write caching issues with VM hypervisors and also increases data recovery complexity for users of your image.
Security	Include the latest versions of required libraries: - OpenSSL v1.0 or greater - Python 2.5 or above (Python 2.6+ is highly recommended) - Python pyasn1 package if not already installed - d.OpenSSL v 1.0 or greater
Security	Clear Bash/Shell history entries. This could include private information or plain-text

Category	Check
	credentials for other systems.
Networking	Include the SSH server by default. Set SSH keep alive to sshd config with the following option: ClientAliveInterval 180.
Networking	Remove any custom network configuration from the image. Delete the resolv.conf: <code>rm /etc/resolv.conf</code> .
Deployment	Install the latest Azure Linux Agent. - Install using the RPM or Deb package. - You may also use the manual install process, but the installer packages are recommended and preferred. - If installing the agent manually from the GitHub repository, first copy the <code>waagent</code> file to <code>/usr/sbin</code> and run (as root): <pre># chmod 755 /usr/sbin/waagent # /usr/sbin/waagent -install</pre> The agent configuration file is placed at <code>/etc/waagent.conf</code>.
Deployment	Ensure Azure Support can provide our partners with serial console output when needed and provide adequate timeout for OS disk mounting from cloud storage. Add the following parameters to the image Kernel Boot Line: <code>console=ttyS0 earlyprintk=ttyS0 rootdelay=300</code> .
Deployment	No swap partition on the OS disk. Swap can be requested for creation on the local resource disk by the Linux Agent.
Deployment	Create a single root partition for the OS disk.
Deployment	64-bit operating system only.

Windows Server images

 Expand table

Category	Check
Security	Use a secure OS base image. The VHD used for the source of any image based on Windows Server must be from the Windows Server OS images provided through Microsoft Azure.
Security	Install all latest security updates.
Security	Applications should not depend on restricted user names like administrator, root, or admin.
Security	Enable BitLocker Drive Encryption for both OS hard drives and data hard drives.

Category	Check
Security	Limit the attack surface by keeping minimal footprint with only necessary Windows Server roles, features, services, and networking ports enabled.
Security	Scan source code and resulting VM image for malware.
Security	Set Windows Server images security update to auto-update.
Security	The VHD image only includes necessary locked accounts that do not have default passwords that would allow interactive login; no back doors.
Security	Disable firewall rules unless application functionally relies on them, such as a firewall appliance.
Security	Remove all sensitive information from the VHD image, including HOSTS files, log files, and unnecessary certificates.
Deployment	64-bit operating system only.

Even if your organization does not have images in the Azure marketplace, consider checking your Windows and Linux image configurations against these recommendations.

Last updated on 12/03/2025

Best practices for protecting secrets

This article provides guidance on protecting your secrets and reducing the risk of unauthorized access. Follow this guidance to help ensure you do not store sensitive information such as credentials in code, GitHub repositories, logs, continuous integration/continuous deployment (CI/CD) pipelines, and so forth. The guidance in this article is compiled from recommendations of individual services, as well as from the [Microsoft Cloud Security Benchmark](#) (MCSB).

General best practices

In today's digital landscape, securing sensitive information such as application credentials and secrets is paramount. A breach can lead to severe consequences, including data loss, financial penalties, and damage to reputation. Implementing a comprehensive secret management strategy is essential to mitigating these risks.

Conduct an audit to identify secrets

Before you can secure your secrets, you need to know where they are. Conducting a thorough audit of your systems and applications helps identify all the sensitive information that needs protection. This includes passwords, API keys, connection strings, and other credentials. Regular audits ensure that new secrets are accounted for and existing ones are properly managed.

It's important to note that even dynamically created secrets, such as OAuth tokens, which may be temporary, still need to be safeguarded with the same rigor as long-term secrets.

Avoid hardcoding secrets

Embedding secrets directly into your code or configuration files is a significant security risk. If your codebase is compromised, so are your secrets. Instead, use environment variables or configuration management tools that keep secrets out of your source code. This practice minimizes the risk of accidental exposure and simplifies the process of updating secrets.

Additionally, integrating secret retrieval into your automated deployment pipeline and using secret injection patterns can prevent secrets from being accidentally exposed in logs or version control, further enhancing the security of your deployment process.

See [Recommendations for protecting application secrets](#)

Use secure key stores

Leveraging secure key stores ensures that your secrets are stored in a secure, encrypted location. Services like [Azure Key Vault](#) and [Azure Managed HSM](#) provide robust security features, including access control, logging, and automatic rotation. This approach centralizes the management of your secrets and reduces the risk of unauthorized access.

For even greater security, particularly for highly sensitive or critical secrets, consider encrypting the secret with a key stores in a Hardware Security Model (HSM), which offer enhanced protection compared to software-based secret stores. For an overview of all the key management offering in Azure and guidance on which to choose, see [Key management in Azure](#) and [How to choose the right key management solution](#).

Implement secret scanning tools

Regularly scanning your codebase for embedded secrets can prevent accidental exposure. Tools like [Azure DevOps Credential Scanner](#) and [GitHub secret scanning](#) [↗](#) feature can automatically detect and alert you to any secrets found in your repositories. Integrating these tools into your CI/CD pipeline ensures continuous monitoring. It is crucial to treat any secret found by these scanning tools as compromised, which means it should be immediately revoked and replaced to maintain the integrity of your security posture.

Leverage managed identities

[Managed identities](#) in Azure provide a secure way for applications to authenticate to Azure services without storing credentials in the code. By enabling managed identities for Azure resources, you can securely access Azure Key Vault and other services, reducing the need to handle secrets manually. This approach not only minimizes the creation of secrets but also reduces the surface area for potential breaches, as the responsibility for managing credentials is delegated to the platform.

Apply granular access control

Follow the principle of least privilege by applying granular access control to your secrets. Use [Azure role-based access control](#) (RBAC) to ensure that only authorized entities have access to specific secrets. Regularly review and update access permissions to prevent unauthorized access. It's also advisable to implement distinct roles such as user, administrator, and auditor to manage access to secrets, ensuring that only trusted identities have the appropriate level of permission.

See the [Azure Key Vault RBAC guide](#).

Rotate secrets regularly

Secrets are susceptible to leakage or exposure over time. Regularly rotating your secrets reduces the risk of unauthorized access. You can [rotate secrets in Azure Key Vault](#) for certain secrets; for those that cannot be automatically rotated, establish a manual rotation process and ensure they are purged when no longer in use.

Automating the secret rotation process and building redundancy into your secret management can ensure that rotation does not disrupt service availability. Implementing retry logic and concurrent access patterns in your code can help minimize issues during the rotation window.

Monitor and log access

Enable logging and monitoring for your secret management system to track access and usage. Use [Key Vault logging](#) and/or services like [Azure Monitor](#) and [Azure Event Grid](#), to monitor all activities related to your secrets. This provides visibility into who accessed your secrets and helps detect any suspicious behavior or potential security incidents. Maintaining detailed audit trails is critical for inspecting and validating access to secrets, which can help prevent identity theft, avoid repudiation, and reduce unnecessary exposure.

Implement network isolation

Reduce the exposure of your secrets by implementing network isolation. Configure [firewalls and network security groups](#) to restrict access to your key vaults. Only allow trusted applications and services to access your secrets, minimizing the attack surface and preventing unauthorized access. Additionally, consider using multiple key vaults to create isolation boundaries for different components, ensuring that if one component is compromised, it cannot gain control of other secrets or the entire workload.

Encrypt secrets at rest and in transit

Ensure that your secrets are encrypted both at rest and in transit. [Azure Key Vault](#) securely stores secrets using envelope encryption, where Data Encryption Keys (DEKs) are encrypted by Key Encryption Keys (KEKs), providing an additional layer of security. This approach enhances protection against unauthorized access. Additionally, use secure communication protocols like HTTPS to encrypt data in transit between your applications and the key vault, ensuring that your secrets are safeguarded during both storage and transmission.

In Azure, encryption at rest is implemented across various services using AES 256 encryption, while data in transit is secured through TLS and MACsec to prevent unauthorized access during transmission. These encryption practices provide comprehensive protection for your data, whether it's being stored or transmitted between systems. For more details, see [Encryption at rest and in transit](#).

Safe distribution of secrets

When distributing secrets, ensure they are shared securely within and outside the organization. Use tools designed for secure sharing and include secret recovery procedures in your disaster recovery plans. If a key is compromised or leaked, it should be regenerated immediately. To further enhance security, use distinct keys for each consumer rather than sharing keys, even if they have similar access patterns. This practice simplifies key management and revocation, ensuring that compromised keys can be revoked without affecting other consumers.

Service-specific best practices

Individual services may have additional best practices and guidance for protecting secrets. Here are some examples:

- API Management: [Use named values in Azure API Management policies with Key Vault Integration](#)
- App Service: [Use Key Vault references for App Service and Azure Functions](#)
- Application Gateway: [Configure an Application Gateway with TLS termination using the Azure portal](#)
- Automation: [Manage credentials in Azure Automation](#)
- Azure App Configuration: [Tutorial: Use Key Vault references in an ASP.NET Core app](#)
- Azure Bot Service: [Azure Bot Service encryption for data at rest](#)
- Azure Center for SAP solutions: [Azure Center for SAP Solutions - Deployment - Prepare network for deployment](#)
- Azure Communications Gateway: [Create and store secrets](#)
- Azure Communications Service: [Create and manage access tokens](#)
- Azure Database for PostgreSQL - Flexible Server: [Azure Database for PostgreSQL - Flexible Server Data Encryption with a Customer-managed Key](#)
- Azure Databricks: [Key Vault Integration in Databricks](#)
- Azure DevTest Labs: [Enable user-assigned managed identities on lab virtual machines in Azure DevTest Labs](#)
- Azure Front Door: [Azure Front Door Secrets](#)
- Azure HDInsight on AKS: [Resource prerequisites - Create Azure Key Vault](#)
- Azure Information Protection: [Details for Azure Information Protection Key Vault Support](#)
- Azure Kubernetes Service (AKS): [CSI Secret Store](#)
- Azure Managed Applications: [Access Key Vault secret when deploying Azure Managed Applications](#)
- Azure OpenAI: [Develop Azure AI services applications with Key Vault](#)
- Azure Pipelines: [Protecting secrets in Azure Pipelines](#)
- Azure Purview: [Credentials for source authentication in Microsoft Purview](#)
- Azure SignalR Service: [Key Vault secret reference in URL template settings](#)

- Azure Service Bus: [Authenticate and authorize an application with Microsoft Entra ID to access Azure Service Bus entities](#)
- Azure Stack Edge: [Manage Azure Stack Edge secrets using Azure Key Vault](#)
- Azure Stack Hub: [Rotate secrets](#)
- Azure Web PubSub: [Add a custom certificate](#)
- Backup: [Configure a vault to encrypt using customer-managed keys](#)
- Cognitive Services: [Develop Azure Cognitive Services applications with Key Vault](#)
- Data Factory: [Store credentials in Azure Key Vault](#)
- ExpressRoute: [Configure MACsec encryption for ExpressRoute Direct.](#)
- Functions: [Use Key Vault references for App Service and Azure Functions](#)
- Key Vault: [About Azure Key Vault secrets](#)
- Logic Apps: [Logic Apps Standard App Settings](#)
- Machine Learning Service: [Use authentication credential secrets in Azure Machine Learning jobs](#)
- SQL IaaS: [Configure Azure Key Vault integration for SQL Server on Azure VMs \(Resource Manager\)](#)
- Storage: [Manage storage account keys with Key Vault and the Azure CLI](#)

Next steps

Minimizing security risk is a shared responsibility. You need to be proactive in taking steps to secure your workloads. [Learn more about shared responsibility in the cloud.](#)

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

 **Note:** The author created this article with assistance from AI. [Learn more](#)

Last updated on 12/03/2025

Azure encryption overview

This article provides an overview of how encryption is used in Microsoft Azure. It covers the major areas of encryption, including encryption at rest, encryption in flight, and key management with Azure Key Vault.

Encryption of data at rest

Data at rest includes information that resides in persistent storage on physical media, in any digital format. Microsoft Azure offers a variety of data storage solutions to meet different needs, including file, disk, blob, and table storage. Microsoft also provides encryption to protect [Azure SQL Database](#), [Azure Cosmos DB](#), and Azure Data Lake.

Data encryption at rest using AES 256 encryption is available for services across the software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS) cloud models.

For a detailed discussion of how data at rest is encrypted in Azure, see [Azure Data Encryption-at-Rest](#).

Azure encryption models

Azure supports various encryption models, including server-side encryption that uses service-managed keys, customer-managed keys in Key Vault, or customer-managed keys on customer-controlled hardware. With client-side encryption, you can manage and store keys on-premises or in another secure location.

Client-side encryption

Client-side encryption is performed outside of Azure. It includes:

- Data encrypted by an application that's running in the customer's datacenter or by a service application
- Data that is already encrypted when it is received by Azure

With client-side encryption, cloud service providers don't have access to the encryption keys and cannot decrypt this data. You maintain complete control of the keys.

Server-side encryption

The three server-side encryption models offer different key management characteristics:

- **Service-managed keys:** Provides a combination of control and convenience with low overhead
- **Customer-managed keys:** Gives you control over the keys, including Bring Your Own Keys (BYOK) support, or allows you to generate new ones
- **Service-managed keys in customer-controlled hardware:** Enables you to manage keys in your proprietary repository, outside of Microsoft control (also called Host Your Own Key or HYOK)

Azure Disk Encryption

Important

Azure Disk Encryption for Virtual Machines and Virtual Machine Scale Sets will be retired on **September 15, 2028**. New customers should use [encryption at host](#) for all new VMs. Existing customers should plan to migrate current ADE-enabled VMs to encryption at host before the retirement date to avoid service disruption -- see [Migrate from Azure Disk Encryption to encryption at host](#).

All Managed Disks, Snapshots, and Images are encrypted using Storage Service Encryption using a service-managed key. Azure also offers options to protect temp disks, caches, and manage keys in Azure Key Vault. For more information, see [Overview of managed disk encryption options](#).

Azure Storage Service Encryption

Data at rest in Azure Blob storage and Azure file shares can be encrypted in both server-side and client-side scenarios.

[Azure Storage Service Encryption \(SSE\)](#) can automatically encrypt data before it is stored, and it automatically decrypts the data when you retrieve it. Storage Service Encryption uses 256-bit AES encryption, one of the strongest block ciphers available.

Azure SQL Database encryption

[Azure SQL Database](#) is a general-purpose relational database service that supports structures such as relational data, JSON, spatial, and XML. SQL Database supports both server-side encryption via the Transparent Data Encryption (TDE) feature and client-side encryption via the Always Encrypted feature.

Transparent Data Encryption

TDE encrypts [SQL Server](#), [Azure SQL Database](#), and [Azure Synapse Analytics](#) data files in real time using a Database Encryption Key (DEK). TDE is enabled by default on newly created Azure SQL databases.

Always Encrypted

The [Always Encrypted](#) feature in Azure SQL allows you to encrypt data within client applications prior to storing it in Azure SQL Database. You can enable delegation of on-premises database administration to third parties and maintain separation between those who own and can view the data and those who manage it.

Cell-level or column-level encryption

With Azure SQL Database, you can apply symmetric encryption to a column of data by using Transact-SQL. This approach is called [cell-level encryption or column-level encryption \(CLE\)](#), because you can use it to encrypt specific columns or cells with different encryption keys, giving you more granular encryption capability than TDE.

Azure Cosmos DB database encryption

[Azure Cosmos DB](#) is Microsoft's globally distributed, multi-model database. User data stored in Azure Cosmos DB in non-volatile storage (solid-state drives) is encrypted by default using service-managed keys. You can add a second layer of encryption with your own keys using the [customer-managed keys \(CMK\)](#) feature.

Azure Data Lake encryption

[Azure Data Lake](#) is an enterprise-wide repository of data. Data Lake Store supports "on by default" transparent encryption of data at rest, which is set up during account creation. By default, Azure Data Lake Store manages the keys for you, but you have the option to manage them yourself.

Encryption of data in transit

Azure offers many mechanisms for keeping data private as it moves from one location to another.

Data-link layer encryption

Whenever Azure customer traffic moves between datacenters—outside physical boundaries not controlled by Microsoft—a data-link layer encryption method using the [IEEE 802.1AE MAC Security Standards](#) (also known as MACsec) is applied from point-to-point across the underlying network hardware. The packets are encrypted on the devices before being sent, preventing physical "man-in-the-middle" or snooping/wiretapping attacks. This MACsec encryption is on by default for all Azure traffic traveling within a region or between regions.

TLS encryption

Microsoft gives customers the ability to use [Transport Layer Security \(TLS\)](#) protocol to protect data when it's traveling between cloud services and customers. Microsoft datacenters negotiate a TLS connection with client systems that connect to Azure services. TLS provides strong authentication, message privacy, and integrity.

Important

Azure is transitioning to require TLS 1.2 or later for all connections to Azure services. Most Azure services completed this transition by August 31, 2025. Ensure your applications use TLS 1.2 or later.

[Perfect Forward Secrecy \(PFS\)](#) protects connections between customers' client systems and Microsoft cloud services by unique keys. Connections support RSA-based 2,048-bit key lengths, ECC 256-bit key lengths, SHA-384 message authentication, and AES-256 data encryption.

Azure Storage transactions

When you interact with Azure Storage through the Azure portal, all transactions take place over HTTPS. You can also use the Storage REST API over HTTPS to interact with Azure Storage. You can enforce the use of HTTPS when you call the REST APIs by enabling the secure transfer requirement for the storage account.

[Shared Access Signatures \(SAS\)](#), which can be used to delegate access to Azure Storage objects, include an option to specify that only the HTTPS protocol can be used.

SMB encryption

[SMB 3.0](#), used to access Azure Files shares, supports encryption and is available in Windows Server 2012 R2, Windows 8, Windows 8.1, and Windows 10. It allows cross-region access and access on the desktop.

VPN encryption

You can connect to Azure through a virtual private network that creates a secure tunnel to protect the privacy of the data being sent across the network.

Azure VPN gateways

[Azure VPN gateway](#) can send encrypted traffic between your virtual network and your on-premises location across a public connection, or between virtual networks. Site-to-site VPNs use [IPsec](#) for transport encryption.

Point-to-site VPNs

Point-to-site VPNs allow individual client computers access to an Azure virtual network. The Secure Socket Tunneling Protocol (SSTP) is used to create the VPN tunnel. For more information, see [Configure a point-to-site connection to a virtual network](#).

Site-to-site VPNs

A site-to-site VPN gateway connection connects your on-premises network to an Azure virtual network over an IPsec/IKE VPN tunnel. For more information, see [Create a site-to-site connection](#).

Key management with Key Vault

Without proper protection and management of keys, encryption is rendered useless. Azure Key Vault is the Microsoft-recommended solution for managing and controlling access to encryption keys used by cloud services.

Key Vault relieves organizations of the need to configure, patch, and maintain hardware security modules (HSMs) and key management software. With Key Vault, you maintain control—Microsoft never sees your keys, and applications don't have direct access to them. You can also import or generate keys in HSMs.

For more information about key management in Azure, see [Key management in Azure](#).

Next steps

- [Azure security overview](#)
- [Azure network security overview](#)

- [Azure database security overview](#)
 - [Azure virtual machines security overview](#)
 - [Data encryption at rest](#)
 - [Data security and encryption best practices](#)
 - [Key management in Azure](#)
-

Last updated on 11/07/2025

Key management in Azure

ⓘ Note

[Zero Trust](#) is a security strategy comprising three principles: "Verify explicitly", "Use least privilege access", and "Assume breach". Data protection, including key management, supports the "use least privilege access" principle. For more information, see [What is Zero Trust?](#)

In Azure, encryption keys can be either platform managed or customer managed.

Platform-managed keys (PMKs) are encryption keys generated, stored, and managed entirely by Azure. Customers do not interact with PMKs. The keys used for [Azure Data Encryption-at-Rest](#), for instance, are PMKs by default.

Customer-managed keys (CMK), on the other hand, are keys read, created, deleted, updated, and/or administered by one or more customers. Keys stored in a customer-owned key vault or hardware security module (HSM) are CMKs. Bring Your Own Key (BYOK) is a CMK scenario in which a customer imports (brings) keys from an outside storage location into an Azure key management service (see the [Azure Key Vault: Bring your own key specification](#)).

A specific type of customer-managed key is the "key encryption key" (KEK). A KEK is a primary key that controls access to one or more encryption keys that are themselves encrypted.

Customer-managed keys can be stored on-premises or, more commonly, in a cloud key management service.

Azure key management services

Azure offers several options for storing and managing your keys in the cloud, including Azure Key Vault, Azure Key Vault Managed HSM, Azure Cloud HSM, and Azure Payment HSM. These options differ in terms of their FIPS compliance level, management overhead, and intended applications.

For a comprehensive guide to choosing the right key management solution for your specific needs, see [How to Choose the Right Key Management Solution](#).

Azure Key Vault (Standard Tier)

A FIPS 140-2 Level 1 validated multitenant cloud key management service that can be used to store asymmetric keys, secrets, and certificates. Keys stored in Azure Key Vault are software-

protected and can be used for encryption-at-rest and custom applications. Azure Key Vault Standard provides a modern API and a breadth of regional deployments and integrations with Azure Services. For more information, see [About Azure Key Vault](#).

Azure Key Vault (Premium Tier)

A FIPS 140-3 Level 3 validated, PCI compliant, multitenant HSM offering that can be used to store asymmetric keys, secrets, and certificates. Keys are stored in a secure hardware boundary using Marvell LiquidSecurity HSMs*. Microsoft manages and operates the underlying HSM, and keys stored in Azure Key Vault Premium can be used for encryption-at-rest and custom applications. Azure Key Vault Premium also provides a modern API and a breadth of regional deployments and integrations with Azure Services.

Important

Azure Integrated HSM: Starting with new Azure server hardware (AMD D and E Series v7 Preview), Microsoft-designed HSM chips are being embedded directly on servers, meeting FIPS 140-3 Level 3 standards. These tamper-resistant chips keep encryption keys within secure hardware boundaries, eliminating latency and exposure risks. The integrated HSM operates transparently by default for supported services like Azure Key Vault and Azure Storage encryption, providing hardware-enforced trust without additional configuration. This integration ensures that cryptographic operations benefit from hardware-level security isolation while maintaining the performance and scalability of cloud services.

If you are an Azure Key Vault Premium customer looking for key sovereignty, single tenancy, and/or higher crypto operations per second, you may want to consider Azure Key Vault Managed HSM instead. For more information, see [About Azure Key Vault](#).

Azure Key Vault Managed HSM

A FIPS 140-2 Level 3 validated, single-tenant HSM offering that gives customers full control of an HSM for encryption-at-rest, Keyless SSL/TLS offload, and custom applications. Azure Key Vault Managed HSM is the only key management solution offering confidential keys. Customers receive a pool of three HSM partitions—together acting as one logical, highly available HSM appliance—fronted by a service that exposes crypto functionality through the Key Vault API. Microsoft handles the provisioning, patching, maintenance, and hardware failover of the HSMs, but doesn't have access to the keys themselves, because the service executes within Azure's Confidential Compute Infrastructure. Azure Key Vault Managed HSM is integrated with the Azure SQL, Azure Storage, and Azure Information Protection PaaS services

and offers support for Keyless TLS with F5 and Nginx. For more information, see [What is Azure Key Vault Managed HSM?](#).

Azure Dedicated HSM

A FIPS 140-2 Level 3 validated single-tenant HSM offering that gives customers full control of an HSM for PKCS#11, offload SSL/TLS processing, certificate authority private key protection, transparent data encryption, including document and code signing, and custom applications. Customer has full administrative control of their HSM cluster. While customers own deployment and initialization of their HSM, Microsoft handles the service provisioning and hosting of the HSM. Azure Dedicated HSM supports existing use cases, including using lift-and-shift workloads, PKI, SSL Offloading and Keyless TLS, OpenSSL applications, Oracle TDE, and Azure SQL TDE IaaS. Azure Dedicated HSM is not integrated with any Azure PaaS offerings. For more information, see [What is Azure Dedicated HSM?](#).

Azure Payment HSM

A FIPS 140-2 Level 3, PCI HSM v3, validated single-tenant bare metal HSM offering that lets customers lease a payment HSM appliance in Microsoft datacenters for payments operations, including payment PIN processing, payment credential issuing, securing keys and authentication data, and sensitive data protection. The service is PCI DSS, PCI 3DS, and PCI PIN compliant. Azure Payment HSM offers single-tenant HSMs for customers to have complete administrative control and exclusive access to the HSM. Once the HSM is allocated to a customer, Microsoft has no access to customer data. Likewise, when the HSM is no longer required, customer data is zeroized and erased as soon as the HSM is released, to ensure complete privacy and security is maintained. For more information, see [What Is Azure Payment HSM?](#).

ⓘ Note

* Azure Key Vault Premium allows the creation of both software-protected and HSM protected keys. If using Azure Key Vault Premium, check to ensure that the key created is HSM protected.

Pricing

The Azure Key Vault Standard and Premium tiers are billed on a transactional basis, with an extra monthly per-key charge for premium hardware-backed keys. Azure Key Vault Managed HSM, Azure Dedicated HSM, and Azure Payment HSM don't charge on a transactional basis;

instead they are always-in-use devices that are billed at a fixed hourly rate. For detailed pricing information, see [Key Vault pricing](#) and [Payment HSM pricing](#).

Service Limits

Azure Key Vault Managed HSM, Azure Dedicated HSM, and Azure Payment HSM offer dedicated capacity. Azure Key Vault Standard and Premium are multitenant offerings and have throttling limits. For service limits, see [Key Vault service limits](#).

Encryption-At-Rest

Azure Key Vault and Azure Key Vault Managed HSM have integrations with Azure Services and Microsoft 365 for Customer Managed Keys, meaning customers may use their own keys in Azure Key Vault and Azure Key Vault Managed HSM for encryption-at-rest of data stored in these services. Azure Dedicated HSM and Azure Payment HSM are Infrastructure-as-Service offerings and do not offer integrations with Azure Services. For an overview of encryption-at-rest with Azure Key Vault and Azure Key Vault Managed HSM, see [Azure Data Encryption-at-Rest](#).

APIs

Azure Dedicated HSM and Azure Payment HSM support the PKCS#11, JCE/JCA, and KSP/CNG APIs, but Azure Key Vault and Azure Key Vault Managed HSM do not. Azure Key Vault and Azure Key Vault Managed HSM use the Azure Key Vault REST API and offer SDK support. For more information on the Azure Key Vault API, see [Azure Key Vault REST API Reference](#).

What's next

- [How to Choose the Right Key Management Solution](#)
- [Azure Key Vault](#)
- [Azure Key Vault Managed HSM](#)
- [Azure Dedicated HSM](#)
- [Azure Payment HSM](#)
- [What is Zero Trust?](#)

How to choose the right Azure key management solution

Azure offers several solutions for cryptographic key storage and management in the cloud: Azure Key Vault (standard and premium offerings), Azure Key Vault Managed HSM, Azure Dedicated HSM, and Azure Payment HSM. It might be overwhelming for customers to decide which key management solution is right for them. This article helps customers navigate this decision-making process by presenting the range of solutions based on three considerations: scenarios, requirements, and industry.

For an overview of key management concepts and detailed descriptions of each solution, see [Key management in Azure](#).

To narrow down a key management solution, follow the flowchart based on common high-level requirements and key management scenarios. Alternatively, use the table based on specific customer requirements that follows it. If either provides multiple products as solutions, or if you want reassurance about choosing the right product, use a combination of the flowchart and table to make a final decision. If you're curious about what other customers in the same industry use, read the table of common key management solutions by industry segment.

Choose an Azure key management solution by scenario

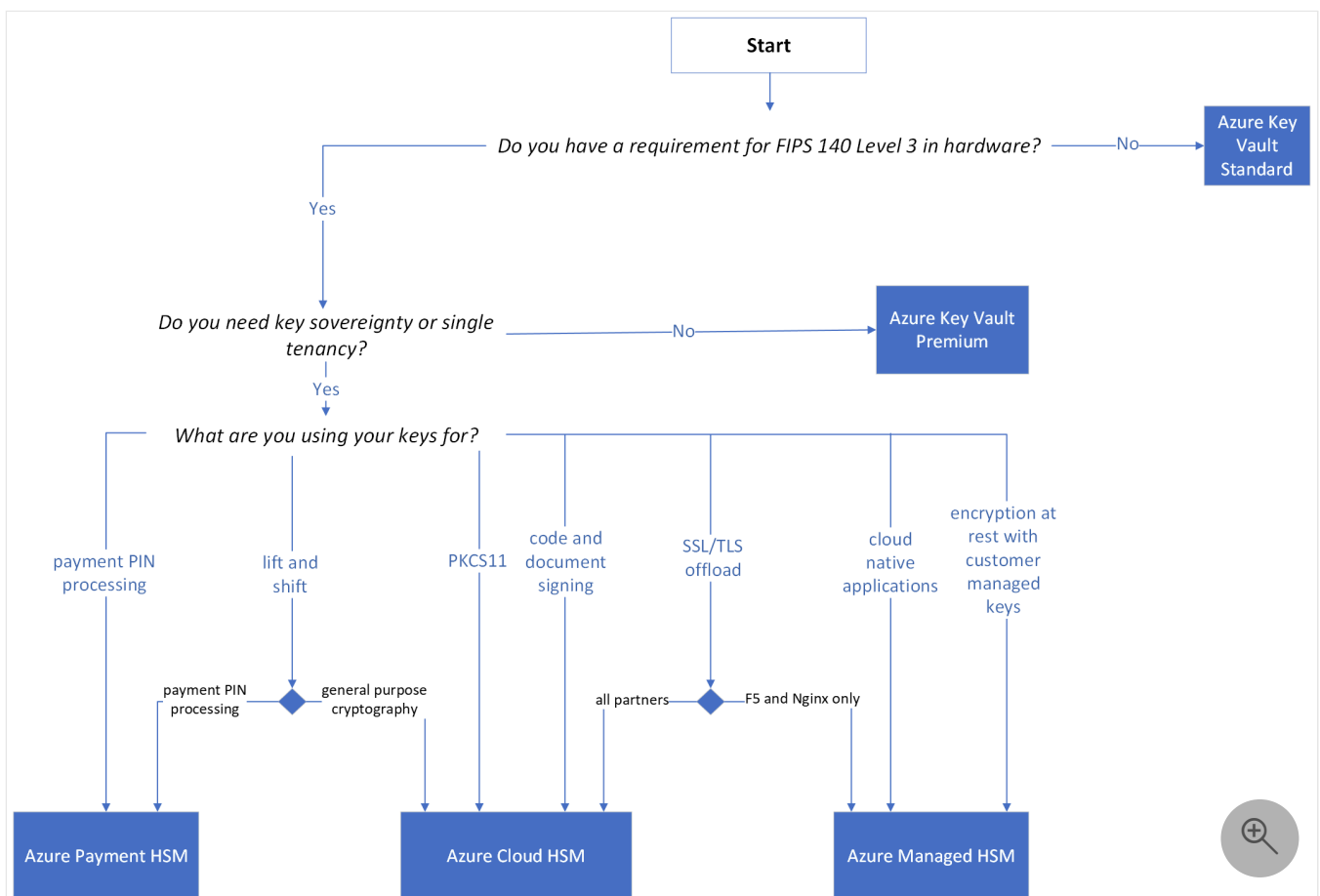
The following chart describes common requirements and use case scenarios and the recommended Azure key management solution.

The chart refers to these common requirements:

- *FIPS-140* is a US government standard with different levels of security requirements. For more information, see [Federal Information Processing Standard \(FIPS\) 140](#).
- *Key sovereignty* is when the customer's organization has full and exclusive control of their keys, including control over what users and services can access the keys and key management policies.
- *Single tenancy* refers to a single dedicated instance of an application deployed for each customer, rather than a shared instance among multiple customers. The need for single tenant products is often found as an internal compliance requirement in financial service industries.

It also refers to these various key management use cases:

- *Encryption at rest* is typically enabled for Azure IaaS, PaaS, and SaaS models. Applications such as Microsoft 365; Microsoft Purview Information Protection; platform services in which the cloud is used for storage, analytics, and service bus functionality; and infrastructure services in which operating systems and applications are hosted and deployed in the cloud use encryption at rest. *Customer managed keys for encryption at rest* is used with Azure Storage and Microsoft Entra. For highest security, keys should be HSM-backed, 3k or 4k RSA keys. For more information about encryption at rest, see [Azure Data Encryption at Rest](#).
- *SSL/TLS Offload* is supported on Azure Managed HSM and Azure Cloud HSM. Customers have improved high availability, security, and the best price point on Azure Managed HSM for F5 and Nginx.
- *Lift and shift* refer to scenarios where a PKCS11 application on-premises is migrated to Azure Virtual Machines and running software such as Oracle TDE in Azure Virtual Machines. Lift and shift requiring payment PIN processing is supported by Azure Payment HSM. All other scenarios are supported by Azure Cloud HSM. Legacy APIs and libraries such as PKCS11, JCA/JCE, and CNG/KSP are only supported by Azure Cloud HSM.
- *Payment PIN processing* includes allowing card and mobile payment authorization and 3D-Secure authentication; PIN generation, management, and validation; payment credential issuing for cards, wearables, and connected devices; securing keys and authentication data; and sensitive data protection for point-to-point encryption, security tokenization, and EMV payment tokenization. This also includes certifications such as PCI DSS, PCI 3DS, and PCI PIN. These are only supported by Azure Payment HSM.



The flowchart result is a starting point to identify the solution that best matches your needs.

Compare other customer requirements

Azure provides multiple key management solutions to allow customers to choose a product based on both high-level requirements and management responsibilities. There is a spectrum of management responsibilities ranging from Azure Key Vault and Azure Managed HSM having less customer responsibility, followed by Azure Cloud HSM and Azure Payment HSM having the most customer responsibility.

This trade-off of management responsibility between the customer and Microsoft and other requirements is detailed in the table below.

Provisioning and hosting are managed by Microsoft across all solutions. Key generation and management, roles and permissions granting, and monitoring and auditing are the responsibility of the customer across all solutions.

Use the table to compare all the solutions side by side. Begin from top to bottom, answering each question found on the left-most column to help you choose the solution that meets all your needs, including management overhead and costs.

 Expand table

	AKV Standard	AKV Premium	Azure Key Vault Managed HSM	Azure Dedicated HSM	Azure Payment HSM
What level of compliance do you need?	FIPS 140-2 level 1	FIPS 140-2 level 2	FIPS 140-2 level 3, PCI DSS, PCI 3DS	FIPS 140-2 level 3, HIPAA, PCI DSS, PCI 3DS, eIDAS	FIPS 140-2 level 3, PCI HSM v3, PCI PTS HSM v3, PCI DSS, PCI 3DS, PCI PIN
Do you need key sovereignty ?	No	No	Yes	Yes	Yes
What kind of tenancy are you looking for?	Multitenant	Multitenant	Single Tenant	Single Tenant	Single Tenant
What are your use cases ?	Encryption at Rest, CMK, custom	Encryption at Rest, CMK, custom	Encryption at Rest, TLS Offload, CMK, custom	PKCS11, TLS Offload, code/document signing, custom	Payment PIN processes, custom

	AKV Standard	AKV Premium	Azure Key Vault Managed HSM	Azure Dedicated HSM	Azure Payment HSM
Do you want HSM hardware protection ?	No	Yes	Yes	Yes	Yes
What is your budget ?	\$	\$\$	\$\$\$	\$\$\$	\$\$\$\$
Who takes responsibility for patching and maintenance ?	Microsoft	Microsoft	Microsoft	Microsoft	Customer
Who takes responsibility for service health and hardware failover ?	Microsoft	Microsoft	Shared	Shared	Customer
What kind of objects are you using?	Asym Keys, Secrets, Certs	Asym Keys, Secrets, Certs	Asym/Sym Keys	Asym/Sym Keys, Certs	Local Master Key
Root of trust control	Microsoft	Microsoft	Customer	Customer	Customer

Common key management solution uses by industry segments

Here is a list of the key management solutions we commonly see being utilized based on industry.

[Expand table](#)

Industry	Suggested Azure solution	Considerations for suggested solutions
I am an enterprise or an organization with strict security and compliance requirements (ex: banking, government, highly regulated industries).	Azure Key Vault Managed HSM, Azure	Azure Key Vault Managed HSM provides FIPS 140-2 Level 3 compliance, and it is a PCI compliant solution for ecommerce. It supports encryption for PCI DSS 4.0. It provides HSM backed keys and gives customers key

Industry	Suggested Azure solution	Considerations for suggested solutions
	Dedicated HSM	sovereignty and single tenancy. Azure Dedicated HSM provides FIPS 140-2 Level 3 compliance, customer ownership of HSM clusters, and support for PKCS#11 and other standard APIs for cryptographic operations.
I am a direct-to-consumer ecommerce merchant who needs to store, process, and transmit my customers' credit cards to my external payment processor/gateway and looking for a PCI compliant solution.	Azure Key Vault Managed HSM	Azure Key Vault Managed HSM provides FIPS 140-2 Level 3 compliance, and it is a PCI compliant solution for ecommerce. It supports encryption for PCI DSS 4.0. It provides HSM backed keys and gives customers key sovereignty and single tenancy.
I am a service provider for financial services, an issuer, a card acquirer, a card network, a payment gateway/PSP, or 3DS solution provider looking for a single tenant service that can meet PCI and multiple major compliance frameworks.	Azure Payment HSM	Azure Payment HSM provides FIPS 140-2 Level 3, PCI HSM v3, PCI DSS, PCI 3DS, and PCI PIN compliance. It provides key sovereignty and single tenancy, common internal compliance requirements around payment processing. Azure Payment HSM provides full payment transaction and PIN processing support.
I am an early-stage startup customer looking to prototype a cloud-native application.	Azure Key Vault Standard	Azure Key Vault Standard provides software-backed keys at an economy price.
I am a startup customer looking to produce a cloud-native application.	Azure Key Vault Premium, Azure Key Vault Managed HSM	Both Azure Key Vault Premium and Azure Key Vault Managed HSM provide HSM-backed keys* and are the best solutions for building cloud native applications.
I am an IaaS customer wanting to move my application to use Azure VM/HSMs.	Azure Dedicated HSM	Azure Dedicated HSM supports SQL IaaS customers. It is the only solution that supports PKCS11 and custom noncloud native applications.

For detailed information about each Azure key management solution, including technical specifications and use cases, see [Key management in Azure](#).

What's next

- [Key management in Azure](#)
 - [Azure Key Vault](#)
 - [Azure Key Vault Managed HSM](#)
 - [Azure Dedicated HSM](#)
 - [Azure Payment HSM](#)
 - [What is Zero Trust?](#)
-

Last updated on 11/05/2025

Services that support customer managed keys (CMKs) in Azure Key Vault and Azure Managed HSM

The following services support server-side encryption with customer managed keys in [Azure Key Vault](#) and [Azure Managed HSM](#). For implementation details, see the service-specific documentation or the service's [Microsoft Cloud Security Benchmark: security baseline](#) (section DP-5).

AI and machine learning

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure AI Search	Yes	Yes	Configure customer-managed keys for data encryption in Azure AI Search
Azure AI services	Yes	Yes	Customer-managed keys for encryption
Azure AI Foundry	Yes		Encryption of data at rest in Azure AI services
Azure Bot Service	Yes		Encryption of bot data in Azure Bot Service
Azure Health Bot	Yes		Configure customer-managed keys (CMK) for Azure Health Bot
Azure Machine Learning	Yes		Customer-managed keys for workspace encryption in Azure Machine Learning
Azure OpenAI	Yes	Yes	Azure OpenAI Service encryption of data at rest
Content Moderator	Yes	Yes	Content Moderator encryption of data at rest
Dataverse	Yes	Yes	Customer-managed keys in Dataverse
Dynamics 365	Yes	Yes	Customer-managed keys for encryption
Face	Yes	Yes	Face service encryption of data at rest
Language Understanding	Yes	Yes	Customer-managed keys with Azure Key Vault
Personalizer	Yes	Yes	Encryption of data at rest in Personalizer
Power Platform	Yes	Yes	Customer-managed keys in Power Platform

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
QnA Maker	Yes	Yes	QnA Maker encryption of data at rest
Speech Services	Yes	Yes	Speech service encryption of data at rest
Translator Text	Yes	Yes	Translator encryption of data at rest

Analytics

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Data Explorer	Yes		Configure customer-managed keys (CMK) in Azure Data Explorer
Azure Data Factory	Yes	Yes	Encryption with customer-managed keys for Azure Data Factory
Azure Data Lake Store	Yes (RSA 2048-bit)		
Azure Data Manager for Energy	Yes		Manage data security and encryption
Azure Databricks	Yes	Yes	Customer-managed keys for managed services
Azure HDInsight	Yes		Azure HDInsight double encryption for data at rest
Azure Monitor Application Insights	Yes		Customer-managed keys in Azure Monitor
Azure Monitor Log Analytics	Yes	Yes	Customer-managed keys in Azure Monitor
Azure Stream Analytics	Yes*	Yes	Data protection in Azure Stream Analytics
Azure Synapse Analytics	Yes (RSA 3072-bit)	Yes	Configure encryption at rest with customer-managed keys
Microsoft Fabric	Yes		Customer-managed key (CMK) encryption and Microsoft Fabric
Power BI Embedded	Yes		Using your own key for Power BI encryption

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
(Preview)			

Containers

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Kubernetes Service	Yes	Yes	Enable host encryption on your AKS cluster nodes
Azure Red Hat OpenShift	Yes		Bring your own keys (BYOK) with Azure Red Hat OpenShift
Container Instances	Yes		Encrypt data with a customer-managed key
Container Registry	Yes		Encrypt container images with a customer-managed key

Compute

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
App Service	Yes*	Yes	Configure customer-managed keys for App Service
Azure Functions	Yes*	Yes	Configure customer-managed keys for Azure Functions
Azure HPC Cache	Yes		Use customer-managed keys with HPC Cache
Azure Managed Applications	Yes*	Yes	Azure managed applications overview
Azure portal	Yes*	Yes	Security in the Azure portal
Azure VMware Solution	Yes	Yes	Configure customer-managed keys in Azure VMware Solution
Batch	Yes		Use customer-managed keys with Batch accounts

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
SAP HANA	Yes		
Site Recovery	Yes		Enable replication with customer-managed keys
Virtual Machine Scale Set	Yes	Yes	Overview of managed disk encryption options
Virtual Machines	Yes	Yes	Overview of managed disk encryption options

Databases

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Cosmos DB	Yes	Yes	Configure customer-managed keys using Azure Key Vault, Configure customer-managed keys using Azure Key Vault Managed HSM
Azure Database for MySQL - Flexible Server	Yes	Yes	Data encryption with customer-managed keys in Azure Database for MySQL - Flexible Server
Azure Database for MySQL - Single Server	Yes		Azure Database for MySQL data encryption with a customer-managed key
Azure Database for PostgreSQL - Flexible Server	Yes	Yes	Data encryption with customer-managed keys in Azure Database for PostgreSQL - Flexible Server
Azure Database for PostgreSQL - Single Server	Yes	Yes	Data encryption with customer-managed keys in Azure Database for PostgreSQL - Single Server
Azure Managed Instance for Apache Cassandra	Yes		Configure customer-managed keys for encryption
Azure SQL Database	Yes (RSA 3072-bit)	Yes	Bring your own key (BYOK) support for Transparent Data Encryption (TDE)
Azure SQL Managed Instance	Yes (RSA 3072-bit)	Yes	Bring your own key (BYOK) support for Transparent Data Encryption (TDE)

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
SQL Server on Azure VM	Yes		Configure Azure Key Vault integration for SQL Server on Azure VMs
SQL Server on Virtual Machines	Yes		Transparent data encryption for SQL Server on Azure VM
SQL Server Stretch Database	Yes (RSA 3072-bit)		
Table Storage	Yes		Customer-managed keys for Azure Storage encryption

Hybrid + multicloud

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Stack Edge	Yes		Protect data at rest on Azure Stack Edge Pro R

Integration

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Fluid Relay	Yes	Yes	Customer-managed keys for Azure Fluid Relay
Azure Health Data Services	Yes		Configure customer-managed keys for Azure Health Data Services DICOM, Configure customer-managed keys for Azure Health Data Services FHIR
Event Hubs	Yes	Yes	Configure customer-managed keys for encryption
Logic Apps	Yes		
Service Bus	Yes	Yes	Configure customer-managed keys for encryption

IoT services

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Device Update for IoT Hub	Yes	Yes	Data encryption for Device Update for IoT Hub
IoT Hub Device Provisioning	Yes		

Management and governance

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
App Configuration	Yes		Use customer-managed keys to encrypt data
Automation	Yes		Encryption of automation assets
Azure Migrate	Yes		Tutorial: Migrate VMware VMs to Azure
Azure Monitor	Yes	Yes	Customer-managed keys in Azure Monitor

Media

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Communication Services	Yes		Data encryption in Azure Communication Services
Media Services	Yes		Use your own encryption keys with Azure Media Services

Security

[Expand table](#)

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Azure Information Protection	Yes		How are the Azure Rights Management cryptographic keys managed and secured?
Microsoft Defender for Cloud	Yes		Customer-managed keys in Azure Monitor
Microsoft Defender for IoT	Yes		
Microsoft Sentinel	Yes	Yes	Encryption at rest in Microsoft Sentinel

Storage

[Expand table](#)

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Archive Storage	Yes		Customer-managed keys for Azure Storage encryption
Azure Backup	Yes	Yes	Encrypt backup data using customer-managed keys
Azure Cache for Redis	Yes**	Yes	Configure disk encryption for Azure Cache for Redis instances using customer managed keys
Azure Data Box	Yes		Use a customer-managed key to secure your Data Box
Azure Managed Lustre	Yes		Use customer-managed encryption keys with Azure Managed Lustre
Azure NetApp Files	Yes	Yes	Configure customer-managed keys for Azure NetApp Files volume encryption
Blob Storage	Yes	Yes	Customer-managed keys for Azure Storage encryption
Data Lake Storage Gen2	Yes	Yes	Customer-managed keys for Azure Storage encryption
Disk Storage	Yes	Yes	Azure Disk Encryption for Windows and Linux VMs
File Storage	Yes	Yes	Customer-managed keys for Azure Storage encryption
File Sync	Yes	Yes	Customer-managed keys for Azure Storage encryption

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Managed Disk Storage	Yes	Yes	Azure Disk Encryption for Windows and Linux VMs
Premium Blob Storage	Yes	Yes	Customer-managed keys for Azure Storage encryption
Queue Storage	Yes	Yes	Customer-managed keys for Azure Storage encryption
StorSimple	Yes		Azure StorSimple security features
Ultra Disk Storage	Yes	Yes	Azure Disk Encryption for Windows and Linux VMs

Other

 Expand table

Product, Feature, or Service	Key Vault	Managed HSM	Documentation
Universal Print	Yes		Data encryption in Universal Print

Caveats

* This service supports storing data in your own Key Vault, Storage Account, or other data persisting service that already supports Server-Side Encryption with Customer-Managed Key.

** Any transient data stored temporarily on disk such as pagefiles or swap files are encrypted with a Microsoft key (all tiers) or a customer-managed key (using the Enterprise and Enterprise Flash tiers). For more information, see [Configure disk encryption in Azure Cache for Redis](#).

Related content

- [Data encryption models in Microsoft Azure](#)
- [How encryption is used in Azure](#)
- [Double encryption](#)

Last updated on 12/03/2025

Double encryption

Article • 04/23/2025

Double encryption is where two or more independent layers of encryption are enabled to protect against compromises of any one layer of encryption. Using two layers of encryption mitigates threats that come with encrypting data. For example:

- Configuration errors in the data encryption
- Implementation errors in the encryption algorithm
- Compromise of a single encryption key

Azure provides double encryption for data at rest and data in transit.

Data at rest

Microsoft's approach to enabling two layers of encryption for data at rest is:

- **Encryption at rest using customer-managed keys.** You provide your own key for data encryption at rest. You can bring your own keys to your Key Vault (BYOK – Bring Your Own Key), or generate new keys in Azure Key Vault to encrypt the desired resources.
- **Infrastructure encryption using platform-managed keys.** By default, data is automatically encrypted at rest using platform-managed encryption keys.

Data in transit

Microsoft's approach to enabling two layers of encryption for data in transit is:

- **Transit encryption using Transport Layer Security (TLS) 1.2 to protect data when it's traveling between the cloud services and you.** All traffic leaving a datacenter is encrypted in transit, even if the traffic destination is another domain controller in the same region. TLS 1.2 is the default security protocol used. TLS provides strong authentication, message privacy, and integrity (enabling detection of message tampering, interception, and forgery), interoperability, algorithm flexibility, and ease of deployment and use.
- **Additional layer of encryption provided at the infrastructure layer.** Whenever Azure customer traffic moves between datacenters-- outside physical boundaries not controlled by Microsoft or on behalf of Microsoft-- a data-link layer encryption method using the [IEEE 802.1AE MAC Security Standards](#) (also known as MACsec) is applied from point-to-point across the underlying network hardware. The packets are encrypted and decrypted on the devices before being sent, preventing physical "man-in-the-middle" or snooping/wiretapping attacks. Because this technology is integrated on the network

hardware itself, it provides line rate encryption on the network hardware with no measurable link latency increase. This MACsec encryption is on by default for all Azure traffic traveling within a region or between regions, and no action is required on customers' part to enable.

Next steps

Learn how [encryption is used in Azure](#).

Azure Certificate Authority details

This article outlines the specific root and subordinate Certificate Authorities (CAs) that are employed by Azure's service endpoints. It is important to note that this list is distinct from the trust anchors provided on Azure VMs and hosted services, which leverage the trust anchors provided by the operating systems themselves. The scope includes government and national clouds. The minimum requirements for public key encryption and signature algorithms, links to certificate downloads and revocation lists, and information about key concepts are provided below the CA details tables. The host names for the URIs that should be added to your firewall allowlists are also provided.

Certificate Authority details

Any entity trying to access Microsoft Entra identity services via the TLS/SSL protocols will be presented with certificates from the CAs listed in this article. Different services may use different root or intermediate CAs. The following root and subordinate CAs are relevant to entities that use [certificate pinning](#).

How to read the certificate details:

- The Serial Number (top string in the table) contains the hexadecimal value of the certificate serial number.
- The Thumbprint (bottom string in the table) is the SHA1 thumbprint.
- CAs listed in italics are the most recently added CAs.

Root and Subordinate CAs list

Root Certificate Authorities

 Expand table

Certificate Authority	Serial Number / Thumbprint
DigiCert Global Root CA	0x083be056904246b1a1756ac95991c74a A8985D3A65E5E5C4B2D7D66D40C6DD2FB19C5436
DigiCert Global Root G2	0x033af1e6a711a9a0bb2864b11d09fae5 DF3C24F9BFD666761B268073FE06D1CC8D4F82A4
DigiCert Global Root G3	0x055556bcf25ea43535c3a40fd5ab4572 7E04DE896A3E666D00E687D33FFAD93BE83D349E

Certificate Authority	Serial Number / Thumbprint
DigiCert TLS ECC P384 Root G5 ↗	0x09e09365acf7d9c8b93e1c0b042a2ef3 17F3DE5E9F0F19E98EF61F32266E20C407AE30EE
DigiCert TLS RSA 4096 Root G5 ↗	0x08f9b478a8fa7eda6a333789de7ccf8a A78849DC5D7C758C8CDE399856B3AAD0B2A57135
Microsoft ECC Root Certificate Authority 2017 ↗	0x66f23daf87de8bb14aea0c573101c2ec 999A64C37FF47D9FAB95F14769891460EEC4C3C5
Microsoft RSA Root Certificate Authority 2017 ↗	0x1ed397095fd8b4b347701eaabe7f45b3 73A5E64A3BFF8316FF0EDCCC618A906E4EAE4D74

Subordinate Certificate Authorities

 Expand table

Certificate Authority	Serial Number / Thumbprint
DigiCert Basic OV G2 TLS CN RSA406 SHA256 2022 CA1 ↗	0x02361857392757c7ce46336cbc47cda2 60707270F2100EE2B771FEC9EFFAD8C9BFFE3358
DigiCert Basic OV G3 TLS CN ECC P-384 SHA384 2022 CA1 ↗	0x01484637b5aacf53c276a96852c2bf32 9355B7783FFF5DF69FE98FAF08F7B66CD4B94B99
DigiCert Basic RSA CN CA G2 ↗	0x02f7e1f982bad009aff47dc95741b2f6 4D1FA5D1FB1AC3917C08E43F65015E6AEA571179
DigiCert Cloud Services CA-1 ↗	0x019ec1c6bd3f597bb20c3338e551d877 81B68D6CD2F221F8F534E677523BB236BBA1DC56
DigiCert Global G2 TLS RSA SHA256 2020 CA1 ↗	0cf5bd062b5602f47ab8502c23ccf066 1B511ABEAD59C6CE207077C0BF0E0043B1382612
DigiCert Global G3 TLS ECC SHA384 2020 CA1 ↗	0b00e92d4d6d731fca3059c7cb1e1886 9577F91FE86C27D9912129730E8166373FC2EEB8
DigiCert SHA2 Secure Server CA ↗	0x02742eaa17ca8e21c717bb1ffcfcd0ca0 626D44E704D1CEABE3BF0D53397464AC8080142C
DigiCert TLS Hybrid ECC SHA384 2020 CA1 ↗	0x0a275fe704d6eeeb23d5cd5b4b1a4e04 51E39A8BDB08878C52D6186588A0FA266A69CF28
DigiCert TLS RSA SHA256 2020 CA1 ↗	0x06d8d904d5584346f68a2fa754227ec4 1C58A3A8518E8759BF075B76B750D4F2DF264FCD

Certificate Authority	Serial Number Thumbprint
GeoTrust Global TLS RSA4096 SHA256 2022 CA1 ↗	0x0f622f6f21c2ff5d521f723a1d47d62d 7E6DB7B7584D8CF2003E0931E6CFC41A3A62D3DF
Microsoft Azure ECC TLS Issuing CA 03 ↗	0x01529ee8368f0b5d72ba433e2d8ea62d 56D955C849887874AA1767810366D90ADF6C8536
Microsoft Azure ECC TLS Issuing CA 03 ↗	0x330000003322a2579b5e698bcc00000000033 91503BE7BF74E2A10AA078B48B71C3477175FEC3
Microsoft Azure ECC TLS Issuing CA 04 ↗	0x02393d48d702425a7cb41c000b0ed7ca FB73FDC24F06998E070A06B6AFC78FDF2A155B25
Microsoft Azure ECC TLS Issuing CA 04 ↗	0x33000000322164aedab61f509d000000000032 406E3B38EFF35A727F276FE993590B70F8224AED
Microsoft Azure ECC TLS Issuing CA 07 ↗	0x0f1f157582cdcd33734bdc5fcd941a33 3BE6CA5856E3B9709056DA51F32CBC8970A83E28
Microsoft Azure ECC TLS Issuing CA 07 ↗	0x3300000034c732435db22a0a2b000000000034 AB3490B7E37B3A8A1E715036522AB42652C3CFE
Microsoft Azure ECC TLS Issuing CA 08 ↗	0x0ef2e5d83681520255e92c608fbc2ff4 716DF84638AC8E6EEBE64416C8DD38C2A25F6630
Microsoft Azure ECC TLS Issuing CA 08 ↗	0x3300000031526979844798bbb8000000000031 CF33D5A1C2F0355B207FCE940026E6C1580067FD
Microsoft Azure RSA TLS Issuing CA 03 ↗	0x05196526449a5e3d1a38748f5dcfebcc F9388EA2C9B7D632B66A2B0B406DF1D37D3901F6
Microsoft Azure RSA TLS Issuing CA 03 ↗	0x330000003968ea517d8a7e30ce000000000039 37461AACFA5970F7F2D2BAC5A659B53B72541C68
Microsoft Azure RSA TLS Issuing CA 04 ↗	0x09f96ec295555f24749eaf1e5dced49d BE68D0ADAA2345B48E507320B695D386080E5B25
Microsoft Azure RSA TLS Issuing CA 04 ↗	0x330000003cd7cb44ee579961d000000000003c 7304022CA8A9FF7E3E0C1242E0110E643822C45E
Microsoft Azure RSA TLS Issuing CA 07 ↗	0x0a43a9509b01352f899579ec7208ba50 3382517058A0C20228D598EE7501B61256A76442
Microsoft Azure RSA TLS Issuing CA 07 ↗	0x330000003bf980b0c83783431700000000003b 0E5F41B697DAADD808BF55AD080350A2A5DFCA93
Microsoft Azure RSA TLS Issuing CA 08 ↗	0x0efb7e547edf0ff1069aee57696d7ba0 31600991ED5FEC63D355A5484A6DCC787EAD89BC

Certificate Authority	Serial Number Thumbprint
Microsoft Azure RSA TLS Issuing CA 08	0x330000003a5dc2ffc321c16d9b00000000003a512C8F3FB71EDACF7ADA490402E710B10C73026E
Microsoft TLS G2 ECC CA OCSP 02	0x3300000005516a21c89814ccf7000000000005336D3E5451315294F6E1E7CFDA84E51C8B1E4054
Microsoft TLS G2 ECC CA OCSP 04	0x3300000008d690fff489d9c4d70000000000085BFF3CC51FB9C1CCE0D52F754427347545E583C5
Microsoft TLS G2 ECC CA OCSP 06	0x33000000066fa24c31c3956fab00000000000069BC57546ACBDB29E388CD1791B43DE4F659B844B
Microsoft TLS G2 ECC CA OCSP 08	0x3300000007737deb49d007dc64000000000007F1C0A1828C10B43F9C651FDFC494832174BE8D07
Microsoft TLS G2 RSA CA OCSP 02	0x330000000c4964a16f44203b2200000000000cF8A408B8FBEEA077B75738658641BE77154720EB
Microsoft TLS G2 RSA CA OCSP 04	0x330000000b13e5667d4a9b558000000000000bDA6D0400641B45AECC595D24E5037AA6BC09C358
Microsoft TLS G2 RSA CA OCSP 06	0x330000000def69cd2d7223735600000000000d88283C78D08489718F78CD2B797333765164E911
Microsoft TLS G2 RSA CA OCSP 08	0x3300000011f100a7eb05eba100000000000011DDBD3DA5C09386CC36CFEC4680F17084BA7A5628
Microsoft TLS G2 RSA CA OCSP 10	0x330000000f33206537ee42ae4f000000000000fB663D8413CA35F3F22643CCBC2D3250700916B1A
Microsoft TLS G2 RSA CA OCSP 12	0x33000000107d59860fd64d7243000000000010D3089DF9F47490887045B91D3FDEF6ED87F02F52
Microsoft TLS G2 RSA CA OCSP 14	0x33000000127e739347e75be1a100000000001255D9B33BDC16DDB3B663F1771F96D4324C118073
Microsoft TLS G2 RSA CA OCSP 16	0x330000000e42938f25dc51a99b00000000000e27E6663299D340DFE54571CECCEC01BE74495DC6

Client compatibility for public PKIs

The CAs used by Azure are compatible with the following OS versions:

[Expand table](#)

Windows	Firefox	iOS	macOS	Android	Java
Windows XP SP3+	Firefox 32+	iOS 7+	OS X Mavericks (10.9)+	Android SDK 5.x+	Java JRE 1.8.0_101+

Review the following action steps when CAs expire or change:

- Update to a supported version of the required OS.
- If you can't change the OS version, you may need to manually update the trusted root store to include the new CAs. Refer to documentation provided by the manufacturer.
- If your scenario includes disabling the trusted root store or running the Windows client in disconnected environments, ensure that all root CAs are included in the Trusted Root CA store and all sub CAs listed in this article are included in the Intermediate CA store.
- Many distributions of **Linux** require you to add CAs to /etc/ssl/certs. Refer to the distribution's documentation.
- Ensure that the **Java** key store contains the CAs listed in this article. For more information, see the [Java applications](#) section of this article.
- If your application explicitly specifies a list of acceptable CAs, check to see if you need to update the pinned certificates when CAs change or expire. For more information, see [Certificate pinning](#).

Public key encryption and signature algorithms

Support for the following algorithms, elliptical curves, and key sizes are required:

Signature algorithms:

- ES256
- ES384
- ES512
- RS256
- RS384
- RS512

Elliptical curves:

- P256
- P384
- P521

Key sizes:

- ECDSA 256

- ECDSA 384
- ECDSA 521
- RSA 2048
- RSA 3072
- RSA 4096

Certificate downloads and revocation lists

The following domains (HTTP/Port 80) may need to be included in your firewall allowlists to optimize connectivity:

AIA:

- `cacerts.digicert.com`
- `cacerts.digicert.cn`
- `cacerts.geotrust.com`
- `caissuers.microsoft.com`
- `www.microsoft.com`

CRL:

- `crl3.digicert.com`
- `crl4.digicert.com`
- `crl.digicert.cn`
- `www.microsoft.com`

OCSP:

- `ocsp.digicert.com`
- `ocsp.digicert.cn`
- `oneocsp.microsoft.com`

Certificate Pinning

Certificate Pinning is a security technique where only authorized, or *pinned*, certificates are accepted when establishing a secure session. Any attempt to establish a secure session using a different certificate is rejected. Learn about the history and implications of [certificate pinning](#).

How to address certificate pinning

If your application explicitly specifies a list of acceptable CAs, you may periodically need to update pinned certificates when Certificate Authorities change or expire.

To detect certificate pinning, we recommend the taking the following steps:

- If you're an application developer, search your source code for references to certificate thumbprints, Subject Distinguished Names, Common Names, serial numbers, public keys, and other certificate properties of any of the Sub CAs involved in this change.
 - If there's a match, update the application to include the missing CAs.
- If you have an application that integrates with Azure APIs or other Azure services and you're unsure if it uses certificate pinning, check with the application vendor.

Java Applications

To determine if the **Microsoft ECC Root Certificate Authority 2017** and **Microsoft RSA Root Certificate Authority 2017** root certificates are trusted by your Java application, you can check the list of trusted root certificates used by the Java Virtual Machine (JVM).

1. Open a terminal window on your system.
2. Run the following command:

Bash

```
keytool -list -keystore $JAVA_HOME/jre/lib/security/cacerts
```

- `$JAVA_HOME` refers to the path to the Java home directory.
- If you're unsure of the path, you can find it by running the following command:

Bash

```
readlink -f $(which java) | xargs dirname | xargs dirname
```

3. Look for the **Microsoft RSA Root Certificate Authority 2017** in the output. It should look something like this:

- If the **Microsoft ECC Root Certificate Authority 2017** and **Microsoft RSA Root Certificate Authority 2017** root certificates are trusted, they should appear in the list of trusted root certificates used by the JVM.
- If it's not in the list, you'll need to add it.
- The output should look like the following sample:

Bash

```
...
Microsoft ECC Root Certificate Authority 2017, 20-Aug-2022, Root CA,
Microsoft RSA Root Certificate Authority 2017, 20-Aug-2022, Root CA,
...
```

4. To add a root certificate to the trusted root certificate store in Java, you can use the `keytool` utility. The following example adds the **Microsoft RSA Root Certificate Authority 2017** root certificate:

Bash

```
keytool -import -file microsoft-ecc-root-ca.crt -alias microsoft-ecc-root-ca -
keystore $JAVA_HOME/jre/lib/security/cacerts
keytool -import -file microsoft-rsa-root-ca.crt -alias microsoft-rsa-root-ca -
keystore $JAVA_HOME/jre/lib/security/cacerts
```

ⓘ Note

In this example, `microsoft-ecc-root-ca.crt` and `microsoft-rsa-root-ca.crt` are the names of the files that contain the **Microsoft ECC Root Certificate Authority 2017** and **Microsoft RSA Root Certificate Authority 2017** root certificates, respectively.

Past changes

The CA/Browser Forum updated the Baseline Requirements to require all publicly trusted Public Key Infrastructures (PKIs) to end usage of the SHA-1 hash algorithms for Online Certificate Standard Protocol (OCSP) on May 31, 2022. Microsoft updated all remaining OCSP Responders that used the SHA-1 hash algorithm to use the SHA-256 hash algorithm.

Microsoft updated Azure services to use TLS certificates from a different set of Root Certificate Authorities (CAs) on February 15, 2021, to comply with changes set forth by the CA/Browser Forum Baseline Requirements. Some services finalized these updates in 2022. For the latest information regarding managed TLS and Azure, see [Managed TLS changes](#).

Article change log

- September 09, 2025:
 - Added the following CAs:

[Expand table](#)

Certificate Authority	Serial Number Thumbprint
Microsoft TLS G2 ECC CA OCSP 04	0x3300000008d690fff489d9c4d7000000000008 5BFF3CC51FB9C1CCE0D52F754427347545E583C5
Microsoft TLS G2 ECC CA OCSP 08	0x3300000007737deb49d007dc64000000000007 F1C0A1828C10B43F9C651FDFC494832174BE8D07
Microsoft Azure ECC TLS Issuing CA 03	0x330000003322a2579b5e698bcc0000000000033 91503BE7BF74E2A10AA078B48B71C3477175FEC3
Microsoft Azure ECC TLS Issuing CA 04	0x33000000322164aedab61f509d0000000000032 406E3B38EFF35A727F276FE993590B70F8224AED
Microsoft Azure ECC TLS Issuing CA 07	0x3300000034c732435db22a0a2b0000000000034 AB3490B7E37B3A8A1E715036522AB42652C3CFFE
Microsoft Azure ECC TLS Issuing CA 08	0x3300000031526979844798bbb80000000000031 CF33D5A1C2F0355B207FCE940026E6C1580067FD

- Removed the following CAs:

 Expand table

Certificate Authority	Serial Number Thumbprint
Microsoft RSA TLS Issuing AOC CA 01	0x330000002ffaf06f6697e2469c00000000002f 4697FDBED95739B457B347056F8F16A975BAF8EE
Microsoft RSA TLS Issuing AOC CA 02	0x3300000030c756cc88f5c1e7eb0000000000030 90ED2E9CB40D0CB49A20651033086B1EA2F76E0E
Microsoft RSA TLS Issuing EOC CA 01	0x33000000310c4914b18c8f339a0000000000031 A04D3750DEBFCCF1259D553DBEC33162C6B42737
Microsoft RSA TLS Issuing EOC CA 02	0x3300000032444d7521341496a90000000000032 697C6404399CC4E7BB3C0D4A8328B71DD3205563
Microsoft ECC TLS Issuing AOC CA 01	0x33000000282bfd23e7d1add707000000000028 30AB5C33EB4B77D4CBFF00A11EE0A7507D9DD316
Microsoft ECC TLS Issuing AOC CA 02	0x33000000290f8a6222ef6a5695000000000029 3709CD92105D074349D00EA8327F7D5303D729C8
Microsoft ECC TLS Issuing EOC CA 01	0x330000002a2d006485fdacbfefb00000000002a 5FA13B879B2AD1B12E69D476E6CAD90D01013B46
Microsoft ECC TLS Issuing EOC CA 02	0x330000002be6902838672b667900000000002b 58A1D8B1056571D32BE6A7C77ED27F73081D6E7A

Certificate Authority	Serial Number Thumbprint
DigiCert TLS RSA SHA256 2020 CA1 ↗	0a3508d55c292b017df8ad65c00ff7e4 6938FD4D98BAB03FAADB97B34396831E3780AEA1

- September 5, 2025:
 - Added the following CAs:

 Expand table

Certificate Authority	Serial Number Thumbprint
DigiCert Global G2 TLS RSA SHA256 2020 CA1 ↗	0cf5bd062b5602f47ab8502c23ccf066 1B511ABEAD59C6CE207077C0BF0E0043B1382612
DigiCert Global G3 TLS ECC SHA384 2020 CA1 ↗	0b00e92d4d6d731fca3059c7cb1e1886 9577F91FE86C27D9912129730E8166373FC2EEB8
DigiCert Basic OV G2 TLS CN RSA406 SHA256 2022 CA1 ↗	02361857392757c7ce46336cbc47cda2 60707270F2100EE2B771FEC9EFFAD8C9BFFE3358
DigiCert Basic OV G3 TLS CN ECC P-384 SHA384 2022 CA1 ↗	01484637b5aacf53c276a96852c2bf32 9355B7783FFF5DF69FE98FAF08F7B66CD4B94B99
Microsoft TLS G2 ECC CA OCSP 02 ↗	3300000005516a21c89814ccf7000000000005 336D3E5451315294F6E1E7CFDA84E51C8B1E4054
Microsoft TLS G2 ECC CA OCSP 06 ↗	33000000066fa24c31c3956fab000000000006 9BC57546ACBDB29E388CD1791B43DE4F659B844B
Microsoft TLS G2 RSA CA OCSP 02 ↗	330000000c4964a16f44203b2200000000000c F8A408B8FBEEA077B75738658641BE77154720EB
Microsoft TLS G2 RSA CA OCSP 04 ↗	330000000b13e5667d4a9b558000000000000b DA6D0400641B45AECC595D24E5037AA6BC09C358
Microsoft TLS G2 RSA CA OCSP 06 ↗	330000000def69cd2d7223735600000000000d 88283C78D08489718F78CD2B797333765164E911
Microsoft TLS G2 RSA CA OCSP 08 ↗	3300000011f100a7eb05eba100000000000011 DDBD3DA5C09386CC36CFEC4680F17084BA7A5628
Microsoft TLS G2 RSA CA OCSP 10 ↗	330000000f33206537ee42ae4f00000000000f B663D8413CA35F3F22643CCBC2D3250700916B1A
Microsoft TLS G2 RSA CA OCSP 12 ↗	33000000107d59860fd64d7243000000000010 D3089DF9F47490887045B91D3FDEF6ED87F02F52

Certificate Authority	Serial Number Thumbprint
Microsoft TLS G2 RSA CA OCSP 14 ↗	33000000127e739347e75be1a1000000000012 55D9B33BDC16DDB3B663F1771F96D4324C118073
Microsoft TLS G2 RSA CA OCSP 16 ↗	330000000e42938f25dc51a99b00000000000e 27E6663299D340DFE54571CECCEC01BE74495DC6

- Removed the following CAs:

 Expand table

Certificate Authority	Serial Number Thumbprint
Entrust Root Certification Authority G2 ↗	4a538c28 8CF427FD790C3AD166068DE81E57EFBB932272D4
Entrust Root Certification Authority G2 ↗	4a538c28 8CF427FD790C3AD166068DE81E57EFBB932272D4
↳ Entrust Certification Authority - L1K ↗	0ee94cc30000000051d37785 F21C12F46CDB6B2E16F09F9419CDFF328437B2D7
↳ Entrust Certification Authority - L1M ↗	61a1e7d20000000051d366a6 CC136695639065FAB47074D28C55314C66077E90
Entrust Certification Authority - L1K ↗	0ee94cc30000000051d37785 F21C12F46CDB6B2E16F09F9419CDFF328437B2D7
Entrust Certification Authority - L1M ↗	61a1e7d20000000051d366a6 CC136695639065FAB47074D28C55314C66077E90

- Updated AIA, CRL, and OCSP URLs to reflect recent CA changes.
- October 8, 2024: Removed the following CAs and CDP endpoints: [crl.microsoft.com](#), [mscrl.microsoft.com](#), and [ocsp.msocsp.com](#).

 Expand table

Certificate Authority	Serial Number Thumbprint
Baltimore CyberTrust Root ↗	0x20000b9 D4DE20D05E66FC53FE1A50882C78DB2852CAE474
Microsoft RSA TLS CA 01 ↗	0x0f14965f202069994fd5c7ac788941e2 703D7A8F0EBF55AAA59F98EAF4A206004EB2516A

Certificate Authority	Serial Number Thumbprint
Microsoft RSA TLS CA 02 ↗	0x0fa74722c53d88c80f589efb1f9d4a3a B0C2D2D13CDD56CDAA6AB6E2C04440BE4A429C75
DigiCert Cloud Services CA-1 ↗	0f171a48c6f223809218cd2ed6ddc0e8 B3F6B64A07BB9611F47174407841F564FB991F29

- July 22, 2024: Added Entrust CAs from a parallel Microsoft 365 article to provide a comprehensive list.
- June 27, 2024: Removed the following CAs, which were superseded by both versions of Microsoft Azure ECC TLS Issuing CAs 03, 04, 07, 08.

 Expand table

Certificate Authority	Serial Number Thumbprint
Microsoft Azure ECC TLS Issuing CA 01 ↗	0x09dc42a5f574ff3a389ee06d5d4de440 92503D0D74A7D3708197B6EE13082D52117A6AB0
Microsoft Azure ECC TLS Issuing CA 01 ↗	0x330000001aa9564f44321c54b900000000001a CDA57423EC5E7192901CA1BF6169DBE48E8D1268
Microsoft Azure ECC TLS Issuing CA 02 ↗	0x0e8dbe5ea610e6cbb569c736f6d7004b 1E981CCDDC69102A45C6693EE84389C3CF2329F1
Microsoft Azure ECC TLS Issuing CA 02 ↗	0x330000001b498d6736ed5612c200000000001b 489FF5765030EB28342477693EB183A4DED4D2A6
Microsoft Azure ECC TLS Issuing CA 05 ↗	x0ce59c30fd7a83532e2d0146b332f965 C6363570AF8303CDF31C1D5AD81E19DBFE172531
Microsoft Azure ECC TLS Issuing CA 05 ↗	0x330000001cc0d2a3cd78cf2c1000000000001c 4C15BC8D7AA5089A84F2AC4750F040D064040CD4
Microsoft Azure ECC TLS Issuing CA 06 ↗	0x066e79cd7624c63130c77abeb6a8bb94 7365ADAEDFEA4909C1BAADBAB68719AD0C381163
Microsoft Azure ECC TLS Issuing CA 06 ↗	0x330000001d0913c309da3f05a6000000000001d DFEB65E575D03D0CC59FD60066C6D39421E65483
Microsoft Azure TLS Issuing CA 01 ↗	0x0aafa6c5ca63c45141ea3be1f7c75317 2F2877C5D778C31E0F29C7E371DF5471BD673173
Microsoft Azure TLS Issuing CA 01 ↗	0x1dbe9496f3db8b8de70000000000001d B9ED88EB05C15C79639493016200FDAB08137AF3

Certificate Authority	Serial Number Thumbprint
Microsoft Azure TLS Issuing CA 02 ↗	0x0c6ae97cced599838690a00a9ea53214 E7EEA674CA718E3BEFD90858E09F8372AD0AE2AA
Microsoft Azure TLS Issuing CA 02 ↗	0x330000001ec6749f058517b4d000000000001e C5FB956A0E7672E9857B402008E7CCAD031F9B08
Microsoft Azure TLS Issuing CA 05 ↗	0x0d7bede97d8209967a52631b8bdd18bd 6C3AF02E7F269AA73AFD0EFF2A88A4A1F04ED1E5
Microsoft Azure TLS Issuing CA 05 ↗	0x330000001f9f1fa2043bc28db900000000001f 56F1CA470BB94E274B516A330494C792C419CF87
Microsoft Azure TLS Issuing CA 06 ↗	0x02e79171fb8021e93fe2d983834c50c0 30E01761AB97E59A06B41EF20AF6F2DE7EF4F7B0
Microsoft Azure TLS Issuing CA 06 ↗	0x3300000020a2f1491a37fbd31f000000000020 8F1FD57F27C828D7BE29743B4D02CD7E6E5F43E6

- July 17, 2023: Added 16 new subordinate Certificate Authorities.
- February 7, 2023: Added eight new subordinate Certificate Authorities.

Next steps

To learn more about Certificate Authorities and PKI, see:

- [Microsoft PKI Repository](#) ↗
- [Microsoft PKI Repository, including CRL and policy information](#) ↗
- [Azure Firewall Premium certificates](#)
- [PKI certificates and Configuration Manager](#)
- [Securing PKI](#)

Last updated on 11/17/2025

What is Certificate pinning?

Article • 12/06/2023

Certificate Pinning is a security technique where only authorized, or *pinned*, certificates are accepted when establishing a secure session. Any attempt to establish a secure session using a different certificate is rejected.

Certificate pinning history

Certificate pinning was originally devised as a means of thwarting Man-in-the-Middle (MITM) attacks. Certificate pinning first became popular in 2011 as the result of the DigiNotar Certificate Authority (CA) compromise, where an attacker was able to create wildcard certificates for several high-profile websites including Google. Chrome was updated to "pin" the current certificates for Google's websites and would reject any connection if a different certificate was presented. Even if an attacker found a way to convince a CA into issuing a fraudulent certificate, it would still be recognized by Chrome as invalid, and the connection rejected.

Though web browsers such as Chrome and Firefox were among the first applications to implement this technique, the range of use cases rapidly expanded. Internet of Things (IoT) devices, iOS and Android mobile apps, and a disparate collection of software applications began using this technique to defend against Man-in-the-Middle attacks.

For several years, certificate pinning was considered good security practice. Oversight over the public Public Key Infrastructure (PKI) landscape has improved with transparency into issuance practices of publicly trusted CAs.

How to address certificate pinning in your application

Typically, an application contains a list of authorized certificates or properties of certificates including Subject Distinguished Names, thumbprints, serial numbers, and public keys. Applications might pin against individual leaf or end-entity certificates, subordinate CA certificates, or even Root CA certificates.

If your application explicitly specifies a list of acceptable CAs, you might periodically need to update pinned certificates when Certificate Authorities change or expire. To detect certificate pinning, we recommend the taking the following steps:

- If you're an application developer, search your source code for any of the following references for the CA that is changing or expiring. If there's a match, update the application to include the missing CAs.
 - Certificate thumbprints
 - Subject Distinguished Names
 - Common Names
 - Serial numbers
 - Public keys
 - Other certificate properties
- If your custom client application integrates with Azure APIs or other Azure services and you're unsure if it uses certificate pinning, check with the application vendor.

Certificate pinning limitations

The practice of certificate pinning has become widely disputed as it carries unacceptable certificate agility costs. One specific implementation, HTTP Public Key Pinning (HPKP), has been deprecated altogether

As there's no single web standard for how certificate pinning is performed, we can't offer direct guidance in detecting its usage. While we don't recommend against certificate pinning, customers should be aware of the limitations this practice creates if they choose to use it.

- Ensure that the pinned certificates can be updated on short notice.
- Industry requirements, such as the [CA/Browser Forum's Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates](#) [↗] require rotating and revoking certificates in as little as 24 hours in certain situations.

Next steps

- [Check the Azure Certificate Authority details for upcoming changes](#)
- [Review the Azure Security Fundamentals best practices and patterns](#)

Feedback

Was this page helpful?

 Yes

 No

[Provide product feedback](#) [↗] | [Get help at Microsoft Q&A](#)

Changes to the managed TLS feature

Microsoft Azure offers a comprehensive managed TLS solution integrated with several Microsoft services. This capability includes managed TLS server certificates for customer vanity domains, provided by DigiCert.

As a result of evolving industry compliance standards, security requirements, and PKI lifecycle changes, this offering will undergo several major updates in 2025 and 2026 that will impact customers utilizing this feature.

PKI updates

Starting in late 2025, Azure began updating its managed TLS solution to align with upcoming browser requirements. These changes affect all managed TLS certificates issued for the following Azure services:

- Azure Front Door (AFD) and CDN Classic
- Azure Front Door Standard/Premium SKU
- Azure API Management
- Azure App Service
- Azure Container Apps
- Azure Static Web Apps

Key Changes

This update includes two key changes:

- **New Root and Subordinate Certificate Authorities (CAs):**
 - All managed TLS certificates will migrate from Certificate Authorities (CAs) under *DigiCert Global Root CA* to CAs under *DigiCert Global Root G2* and *DigiCert Global Root G3*. This transition ensures compliance with browser trusted root program requirements.
- **Removal of Client Authentication EKU**
 - These new CAs will not support client authentication in accordance with browser trusted root program requirements. All managed TLS certificates under the new CAs will only include the Server Authentication Extended Key Usage (EKU).

Potential Customer impact

To prepare for the change, it's important to know how the changes could potentially affect customers.

- **Certificate Pinning**
 - If you pin certificates or public keys, you must update your pin sets to include the new roots and intermediates.
 - Static pinning is strongly discouraged due to operational risk.
- **Client Authentication**
 - If your application relies on the Client Authentication EKU in public certificates, you must update your configuration to use certificates from other CAs.
 - Managed TLS certificates will only support the Server Authentication EKU.

Domain validation

Starting in late 2025, DigiCert is transitioning to a new open-source software (OSS) domain control validation (DCV) platform designed to enhance transparency and accountability in domain validation processes. DigiCert will no longer support the legacy CNAME Delegation DCV workflow for domain control validation in the specified Azure services.

Consequently, these Azure services will be introducing an enhanced domain control validation process, aiming to significantly expedite domain validation and address key vulnerabilities in the user experience.

This change doesn't impact the standard CNAME DCV process for DigiCert customers, where validation uses a random value in the CNAME record. Only this one workflow for validation previously used by Microsoft is being retired.

Warning

Customers who haven't updated their configurations to comply with the managed TLS changes will have a service outage if they don't update the configuration.

- An outage is *guaranteed* to occur when the current certificate expires.
- An outage *could* occur if the certificate is revoked.

In the event of a revocation, certificates must be revoked within 24 hours, as mandated by the [CA/Browser Forum Baseline Requirements](#) , leaving very little time to respond.

Customers should update their configurations with urgency to avoid disruption.

Frequently asked questions

Q: Is support for custom domains being retired?

No. The feature is very much supported and in fact is receiving several key updates that improve the overall user experience.

ⓘ Note

AFD classic and CDN Classic SKUs, which are on the path to deprecation, are retiring support for adding new custom domains. For more information, see [Azure Front Door \(classic\) and Azure CDN from Microsoft Classic SKU ending CNAME based domain validation and new domain/profile creations by August 15, 2025](#) [↗]. Customers are recommended to use managed TLS certificates with AFD Standard and Premium SKUs for new custom domains.

Q: What is domain control validation?

Domain Control Validation (DCV) is a critical process used to verify that an entity requesting a TLS/SSL certificate has legitimate control over the domain(s) listed in the certificate.

Q: Is DigiCert retiring CNAME domain control validation?

No. Only this specific CNAME validation method unique to Azure services is being retired. The CNAME DCV method used by DigiCert customers, such as the one described for DigiCert [OV/EV certificates](#) [↗] and [DV certificates](#) [↗] isn't impacted.

Only Azure is impacted by this change.

Q: Why is Microsoft migrating to the DigiCert Global Root G2 and G3 roots?

This change aligns with industry standards and upcoming browser requirements. On April 15, 2026, Mozilla and Chrome will distrust the *DigiCert Global Root CA*. To maintain trust, all managed TLS certificates will move to *DigiCert Global Root G2* and *DigiCert Global Root G3* before this date. For more information, see [DigiCert root and intermediate CA certificate updates 2023](#) [↗].

Q: Why is the Client Authentication EKU being removed?

This is an industry-wide change driven by the Chrome Trusted Root Program. Chrome is restricting TLS certificates to server authentication to improve security and compliance. For

more information, see [Sunsetting the client authentication EKU from DigiCert public TLS certificates](#) .

Last updated on 12/02/2025

Azure data security and encryption best practices

This article describes best practices for data security and encryption.

The best practices are based on a consensus of opinion, and they work with current Azure platform capabilities and feature sets. Opinions and technologies change over time and this article is updated on a regular basis to reflect those changes.

Protect data

To help protect data in the cloud, you need to account for the possible states in which your data can occur, and what controls are available for that state. Best practices for Azure data security and encryption relate to the following data states:

- **At rest:** This includes all information storage objects, containers, and types that exist statically on physical media, whether magnetic or optical disk.
- **In transit:** When data is being transferred between components, locations, or programs, it's in transit. Examples are transfer over the network, across a service bus (from on-premises to cloud and vice-versa, including hybrid connections such as ExpressRoute), or during an input/output process.
- **In Use:** When data is being processed, the specialized AMD & Intel chipset based Confidential compute VMs keep the data encrypted in memory using hardware managed keys.

Choose a key management solution

Protecting your keys is essential to protecting your data in the cloud.

Azure offers several different services to protect your cryptographic keys using HSMs. These offerings provide cloud scalability and availability while giving you complete control over your keys. For more information and guidance on choosing among these key management offerings, see [How to choose the right Azure key management solution](#). [Azure Key Vault Premium](#) or [Azure Key Vault Managed HSM](#) are recommended for managing your encryption at rest keys.

Manage with secure workstations

📌 Note

The subscription administrator or owner should use a secure access workstation or a privileged access workstation.

Because the vast majority of attacks target the end user, the endpoint becomes one of the primary points of attack. An attacker who compromises the endpoint can use the user's credentials to gain access to the organization's data. Most endpoint attacks take advantage of the fact that users are administrators in their local workstations.

Best practice: Use a secure management workstation to protect sensitive accounts, tasks, and data. **Detail:** Use a [privileged access workstation](#) to reduce the attack surface in workstations. These secure management workstations can help you mitigate some of these attacks and ensure that your data is safer.

Best practice: Ensure endpoint protection. **Detail:** Enforce security policies across all devices that are used to consume data, regardless of the data location (cloud or on-premises).

Protect data at rest

[Data encryption at rest](#) is a mandatory step toward data privacy, compliance, and data sovereignty.

Best practice: Apply encryption at host to help safeguard your data. **Detail:** Use [Encryption at host - End-to-end encryption for your VM](#). Encryption at host is a Virtual Machine option that enhances Azure Disk Storage Server-Side Encryption to ensure that all temp disks and disk caches are encrypted at rest and flow encrypted to the Storage clusters..

Most Azure services, such as Azure Storage and Azure SQL Database, encrypt data at rest by default. You can use Azure Key Vault to maintain control of keys that access and encrypt your data. See [Azure resource providers encryption model support to learn more](#).

Best practices: Use encryption to help mitigate risks related to unauthorized data access. **Detail:** Encrypt encryption on your services before you write sensitive data to them.

Organizations that don't enforce data encryption are more exposed to data-confidentiality issues. Companies also must prove that they are diligent and using correct security controls to enhance their data security in order to comply with industry regulations.

Protect data in transit

Protecting data in transit should be an essential part of your data protection strategy. Because data is moving back and forth from many locations, we generally recommend that you always use SSL/TLS protocols to exchange data across different locations. In some circumstances, you

might want to isolate the entire communication channel between your on-premises and cloud infrastructures by using a VPN.

For data moving between your on-premises infrastructure and Azure, consider appropriate safeguards such as HTTPS or VPN. When sending encrypted traffic between an Azure virtual network and an on-premises location over the public internet, use [Azure VPN Gateway](#).

Following are best practices specific to using Azure VPN Gateway, SSL/TLS, and HTTPS.

Best practice: Secure access from multiple workstations located on-premises to an Azure virtual network. **Detail:** Use [site-to-site VPN](#).

Best practice: Secure access from an individual workstation located on-premises to an Azure virtual network. **Detail:** Use [point-to-site VPN](#).

Best practice: Move larger data sets over a dedicated high-speed WAN link. **Detail:** Use [ExpressRoute](#). If you choose to use ExpressRoute, you can also encrypt the data at the application level by using SSL/TLS or other protocols for added protection.

Best practice: Interact with Azure Storage through the Azure portal. **Detail:** All transactions occur via HTTPS. You can also use [Storage REST API](#) over HTTPS to interact with [Azure Storage](#).

Organizations that fail to protect data in transit are more susceptible to [man-in-the-middle attacks](#), [eavesdropping](#), and session hijacking. These attacks can be the first step in gaining access to confidential data.

Protect data in use

Lessen the need for trust Running workloads on the cloud requires trust. You give this trust to various providers enabling different components of your application.

- App software vendors: Trust software by deploying on-premises, using open-source, or by building in-house application software.
- Hardware vendors: Trust hardware by using on-premises hardware or in-house hardware.
- Infrastructure providers: Trust cloud providers or manage your own on-premises data centers.

Reducing the attack surface The Trusted Computing Base (TCB) refers to all of a system's hardware, firmware, and software components that provide a secure environment. The components inside the TCB are considered "critical." If one component inside the TCB is compromised, the entire system's security may be jeopardized. A lower TCB means higher security. There's less risk of exposure to various vulnerabilities, malware, attacks, and malicious people.

Azure confidential computing can help you:

- Prevent unauthorized access: Run sensitive data in the cloud. Trust that Azure provides the best data protection possible, with little to no change from what gets done today.
- Meet regulatory compliance: Migrate to the cloud and keep full control of data to satisfy government regulations for protecting personal information and secure organizational IP.
- Ensure secure and untrusted collaboration: Tackle industry-wide work-scale problems by combining data across organizations, even competitors, to unlock broad data analytics and deeper insights.
- Isolate processing: Offer a new wave of products that remove liability on private data with blind processing. User data can't even be retrieved by the service provider.

Learn more about [Confidential computing](#).

Secure email, documents, and sensitive data

You want to control and secure email, documents, and sensitive data that you share outside your company. [Azure Information Protection](#) is a cloud-based solution that helps an organization to classify, label, and protect its documents and emails. This can be done automatically by administrators who define rules and conditions, manually by users, or a combination where users get recommendations.

Classification is identifiable at all times, regardless of where the data is stored or with whom it's shared. The labels include visual markings such as a header, footer, or watermark. Metadata is added to files and email headers in clear text. The clear text ensures that other services, such as solutions to prevent data loss, can identify the classification and take appropriate action.

The protection technology uses Azure Rights Management (Azure RMS). This technology is integrated with other Microsoft cloud services and applications, such as Microsoft 365 and Microsoft Entra ID. This protection technology uses encryption, identity, and authorization policies. Protection that is applied through Azure RMS stays with the documents and emails, independently of the location—inside or outside your organization, networks, file servers, and applications.

This information protection solution keeps you in control of your data, even when it's shared with other people. You can also use Azure RMS with your own line-of-business applications and information protection solutions from software vendors, whether these applications and solutions are on-premises or in the cloud.

We recommend that you:

- [Deploy Azure Information Protection](#) for your organization.

- Apply labels that reflect your business requirements. For example: Apply a label named "highly confidential" to all documents and emails that contain top-secret data, to classify and protect this data. Then, only authorized users can access this data, with any restrictions that you specify.
- Configure [usage logging for Azure RMS](#) so that you can monitor how your organization is using the protection service.

Organizations that are weak on [data classification](#) [↗] and file protection might be more susceptible to data leakage or data misuse. With proper file protection, you can analyze data flows to gain insight into your business, detect risky behaviors and take corrective measures, track access to documents, and so on.

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

The following resources are available to provide more general information about Azure security and related Microsoft services:

- [Azure Security Team Blog](#) - for up to date information on the latest in Azure Security
- [Microsoft Security Response Center](#) [↗] - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com

Last updated on 10/25/2025

Azure Data Encryption at rest

Microsoft Azure includes tools to safeguard data according to your company's security and compliance needs. This paper focuses on:

- How data is protected at rest across Microsoft Azure
- Discusses the various components taking part in the data protection implementation,
- Reviews pros and cons of the different key management protection approaches.

Encryption at Rest is a common security requirement. In Azure, organizations can encrypt data at rest without the risk or cost of a custom key management solution. Organizations have the option of letting Azure completely manage Encryption at Rest. Additionally, organizations have various options to closely manage encryption or encryption keys.

What is encryption at rest?

Encryption is the secure encoding of data used to protect confidentiality of data. The Encryption at Rest designs in Azure use symmetric encryption to encrypt and decrypt large amounts of data quickly according to a simple conceptual model:

- A symmetric encryption key is used to encrypt data as it is written to storage.
- The same encryption key is used to decrypt that data as it is readied for use in memory.
- Data may be partitioned, and different keys may be used for each partition.
- Keys must be stored in a secure location with identity-based access control and audit policies. Data encryption keys which are stored outside of secure locations are encrypted with a key encryption key kept in a secure location.

In practice, key management and control scenarios, as well as scale and availability assurances, require additional constructs. Microsoft Azure Encryption at Rest concepts and components are described below.

The purpose of encryption at rest

Encryption at rest provides data protection for stored data (at rest). Attacks against data at-rest include attempts to obtain physical access to the hardware on which the data is stored, and then compromise the contained data. In such an attack, a server's hard drive may have been mishandled during maintenance allowing an attacker to remove the hard drive. Later the attacker would put the hard drive into a computer under their control to attempt to access the data.

Encryption at rest is designed to prevent the attacker from accessing the unencrypted data by ensuring the data is encrypted when on disk. If an attacker obtains a hard drive with encrypted

data but not the encryption keys, the attacker must defeat the encryption to read the data. This attack is much more complex and resource consuming than accessing unencrypted data on a hard drive. For this reason, encryption at rest is highly recommended and is a high priority requirement for many organizations.

Encryption at rest may also be required by an organization's need for data governance and compliance efforts. Industry and government regulations such as HIPAA, PCI and FedRAMP, lay out specific safeguards regarding data protection and encryption requirements. Encryption at rest is a mandatory measure required for compliance with some of those regulations. For more information on Microsoft's approach to FIPS 140-2 validation, see [Federal Information Processing Standard \(FIPS\) Publication 140-2](#).

In addition to satisfying compliance and regulatory requirements, encryption at rest provides defense-in-depth protection. Microsoft Azure provides a compliant platform for services, applications, and data. It also provides comprehensive facility and physical security, data access control, and auditing. However, it's important to provide additional "overlapping" security measures in case one of the other security measures fails and encryption at rest provides such a security measure.

Microsoft is committed to encryption at rest options across cloud services and giving customers control of encryption keys and logs of key use. Additionally, Microsoft is working towards encrypting all customer data at rest by default.

Azure Encryption at Rest Components

As described previously, the goal of encryption at rest is that data that is persisted on disk is encrypted with a secret encryption key. To achieve that goal secure key creation, storage, access control, and management of the encryption keys must be provided. Though details may vary, Azure services Encryption at Rest implementations can be described in terms illustrated in the following diagram.



Azure Key Vault

The storage location of the encryption keys and access control to those keys is central to an encryption at rest model. The keys need to be highly secured but manageable by specified users and available to specific services. For Azure services, Azure Key Vault is the recommended key storage solution and provides a common management experience across services. Keys are stored and managed in key vaults, and access to a key vault can be given to users or services. Azure Key Vault supports customer creation of keys or import of customer keys for use in customer-managed encryption key scenarios.

Microsoft Entra ID

Permissions to use the keys stored in Azure Key Vault, either to manage or to access them for Encryption at Rest encryption and decryption, can be given to Microsoft Entra accounts.

Envelope Encryption with a Key Hierarchy

More than one encryption key is used in an encryption at rest implementation. Storing an encryption key in Azure Key Vault ensures secure key access and central management of keys. However, service local access to encryption keys is more efficient for bulk encryption and decryption than interacting with Key Vault for every data operation, allowing for stronger encryption and better performance. Limiting the use of a single encryption key decreases the risk that the key will be compromised and the cost of re-encryption when a key must be replaced. Azure encryption at rest models use envelope encryption, where a key encryption key encrypts a data encryption key. This model forms a key hierarchy which is better able to address performance and security requirements:

- **Data Encryption Key (DEK)** – A symmetric AES256 key used to encrypt a partition or block of data, sometimes also referred to as simply a Data Key. A single resource may have many partitions and many Data Encryption Keys. Encrypting each block of data with a different key makes crypto analysis attacks more difficult. And keeping DEKs local to the service encrypting and decrypting data maximizes performance.
- **Key Encryption Key (KEK)** – An encryption key used to encrypt the Data Encryption Keys using envelope encryption, also referred to as wrapping. Use of a Key Encryption Key that never leaves Key Vault allows the data encryption keys themselves to be encrypted and controlled. The entity that has access to the KEK may be different than the entity that requires the DEK. An entity may broker access to the DEK to limit the access of each DEK to a specific partition. Since the KEK is required to decrypt the DEKs, customers can cryptographically erase DEKs and data by disabling of the KEK.

Resource providers and application instances store the encrypted Data Encryption Keys as metadata. Only an entity with access to the Key Encryption Key can decrypt these Data Encryption Keys. Different models of key storage are supported. For more information, see [data encryption models](#).

Encryption at rest in Microsoft cloud services

Microsoft Cloud services are used in all three cloud models: IaaS, PaaS, SaaS. Below you have examples of how they fit on each model:

- Software services, referred to as Software as a Service or SaaS, which have applications provided by the cloud such as Microsoft 365.
- Platform services in which customers use the cloud for things like storage, analytics, and service bus functionality in their applications.
- Infrastructure services, or Infrastructure as a Service (IaaS) in which customer deploys operating systems and applications that are hosted in the cloud and possibly leveraging other cloud services.

Encryption at rest for SaaS customers

Software as a Service (SaaS) customers typically have encryption at rest enabled or available in each service. Microsoft 365 has several options for customers to verify or enable encryption at rest. For information about Microsoft 365 services, see [Encryption in Microsoft 365](#).

Encryption at rest for PaaS customers

Platform as a Service (PaaS) customer's data typically resides in a storage service such as Blob Storage but may also be cached or stored in the application execution environment, such as a

virtual machine. To see the encryption at rest options available to you, examine the [Data encryption models](#) for the storage and application platforms that you use.

Encryption at rest for IaaS customers

Infrastructure as a Service (IaaS) customers can have a variety of services and applications in use. IaaS services can enable encryption at rest in their Azure hosted virtual machines and VHDs using encryption at host.

Encrypted storage

Like PaaS, IaaS solutions can leverage other Azure services that store data encrypted at rest. In these cases, you can enable the Encryption at Rest support as provided by each consumed Azure service. The [Data encryption models](#) enumerates the major storage, services, and application platforms and the model of Encryption at Rest supported.

Encrypted compute

All Managed Disks, Snapshots, and Images are encrypted using Storage Service Encryption using a service-managed key. A more complete Encryption at Rest solution ensures that the data is never persisted in unencrypted form. While processing the data on a virtual machine, data can be persisted to the Windows page file or Linux swap file, a crash dump, or to an application log. To ensure this data is encrypted at rest, IaaS applications can use encryption at host on an Azure IaaS virtual machine.

Custom encryption at rest

It is recommended that whenever possible, IaaS applications leverage encryption at host and Encryption at Rest options provided by any consumed Azure services. In some cases, such as irregular encryption requirements or non-Azure based storage, a developer of an IaaS application may need to implement encryption at rest themselves. Developers of IaaS solutions can better integrate with Azure management and customer expectations by leveraging certain Azure components. Specifically, developers should use the Azure Key Vault service to provide secure key storage as well as provide their customers with key management options consistent with that of Azure platform services. Additionally, custom solutions should use Azure managed service identities to enable service accounts to access encryption keys. For developer information on Azure Key Vault and Managed Service Identities, see their respective SDKs.

Azure resource providers encryption model support

Microsoft Azure Services each support one or more of the encryption at rest models. For some services, however, one or more of the encryption models may not be applicable. For services that support customer-managed key scenarios, they may support only a subset of the key types that Azure Key Vault supports for key encryption keys. Additionally, services may release support for these scenarios and key types at different schedules. This section describes the encryption at rest support at the time of this writing for each of the major Azure data storage services.

Azure VM disk encryption

Any customer using Azure Infrastructure as a Service (IaaS) features can achieve encryption at rest for their IaaS VM disks through encryption at host. For more information, see [Encryption at host - End-to-end encryption for your VM](#).

Azure storage

All Azure Storage services (Blob storage, Queue storage, Table storage, and Azure Files) support server-side encryption at rest with customer-managed keys; some services additionally support client-side encryption.

- Server-side: All Azure Storage Services enable server-side encryption by default using service-managed keys, which is transparent to the application. For more information, see [Azure Storage Service Encryption for Data at Rest](#). They also support customer-managed keys in Azure Key Vault. For more information, see [Storage Service Encryption using customer-managed keys in Azure Key Vault](#).
- Client-side: Azure Blobs, Tables, and Queues support client-side encryption. When using client-side encryption, customers encrypt the data and upload the data as an encrypted blob. Key management is done by the customer. For more information, see [Client-Side Encryption and Azure Key Vault for Microsoft Azure Storage](#).

Azure SQL Database

Azure SQL Database currently supports encryption at rest for Microsoft-managed service side and client-side encryption scenarios.

Support for server encryption is currently provided through the SQL feature called Transparent Data Encryption. Once an Azure SQL Database customer enables TDE, keys are automatically created and managed for them. Encryption at rest can be enabled at the database and server

levels. As of June 2017, [Transparent Data Encryption \(TDE\)](#) is enabled by default on newly created databases. Azure SQL Database supports RSA 2048-bit customer-managed keys in Azure Key Vault. For more information, see [Transparent Data Encryption with Bring Your Own Key support for Azure SQL Database and Data Warehouse](#).

Client-side encryption of Azure SQL Database data is supported through the [Always Encrypted](#) feature. Always Encrypted uses a key that is created and stored by the client. Customers can store the master key in a Windows certificate store, Azure Key Vault, or a local Hardware Security Module. Using SQL Server Management Studio, SQL users choose what key they'd like to use to encrypt which column.

Conclusion

Protection of customer data stored within Azure Services is of paramount importance to Microsoft. All Azure hosted services are committed to providing Encryption at Rest options. Azure services support either service-managed keys, customer-managed keys, or client-side encryption. Azure services are broadly enhancing Encryption at Rest availability and new options are planned for preview and general availability in the upcoming months.

Next steps

- See [data encryption models](#) to learn more about service-managed keys and customer-managed keys.
- Learn how Azure uses [double encryption](#) to mitigate threats that come with encrypting data.
- Learn what Microsoft does to ensure [platform integrity and security](#) of hosts traversing the hardware and firmware build-out, integration, operationalization, and repair pipelines.

Last updated on 10/25/2025

Data encryption models

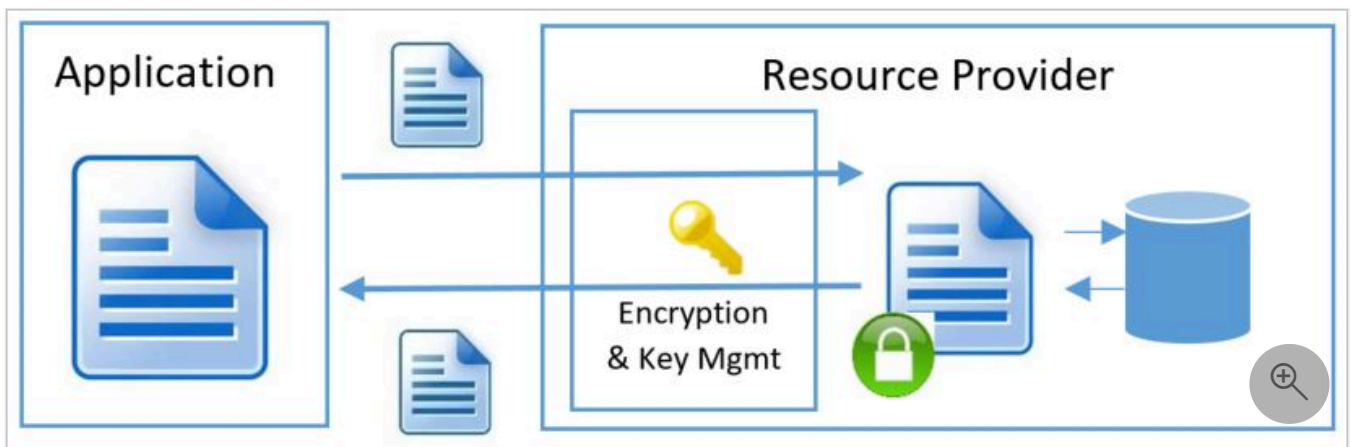
Article • 04/23/2025

An understanding of the various encryption models and their pros and cons is essential for understanding how the various resource providers in Azure implement encryption at Rest. These definitions are shared across all resource providers in Azure to ensure common language and taxonomy.

There are three scenarios for server-side encryption:

- Server-side encryption using Service-Managed keys
 - Azure Resource Providers perform the encryption and decryption operations
 - Microsoft manages the keys
 - Full cloud functionality
- Server-side encryption using customer-managed keys in Azure Key Vault
 - Azure Resource Providers perform the encryption and decryption operations
 - Customer controls keys via Azure Key Vault
 - Full cloud functionality
- Server-side encryption using customer-managed keys on customer-controlled hardware
 - Azure Resource Providers perform the encryption and decryption operations
 - Customer controls keys on customer-controlled hardware
 - Full cloud functionality

Server-side Encryption models refer to encryption that performed by the Azure service. In that model, the Resource Provider performs the encrypt and decrypt operations. For example, Azure Storage might receive data in plain text operations and perform the encryption and decryption internally. The Resource Provider might use encryption keys that managed by Microsoft or by the customer depending on the provided configuration.



Each of the server-side encryption at rest models implies distinctive characteristics of key management, including where and how encryption keys are created and stored, as well as the

access models and the key rotation procedures.

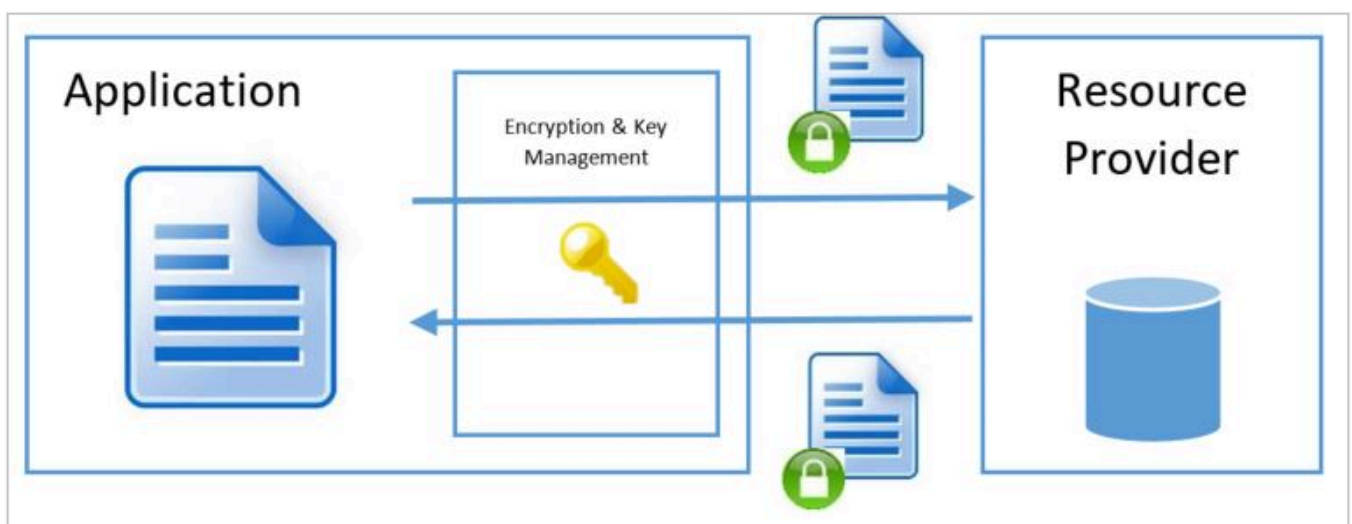
For client-side encryption, consider:

- Azure services cannot see decrypted data
- Customers manage and store keys on-premises (or in other secure stores). Keys are not available to Azure services
- Reduced cloud functionality

The supported encryption models in Azure split into two main groups: "Client Encryption" and "Server-side Encryption" as mentioned previously. Independent of the encryption at rest model used, Azure services always recommend the use of a secure transport such as TLS or HTTPS. Therefore, encryption in transport should be addressed by the transport protocol and should not be a major factor in determining which encryption at rest model to use.

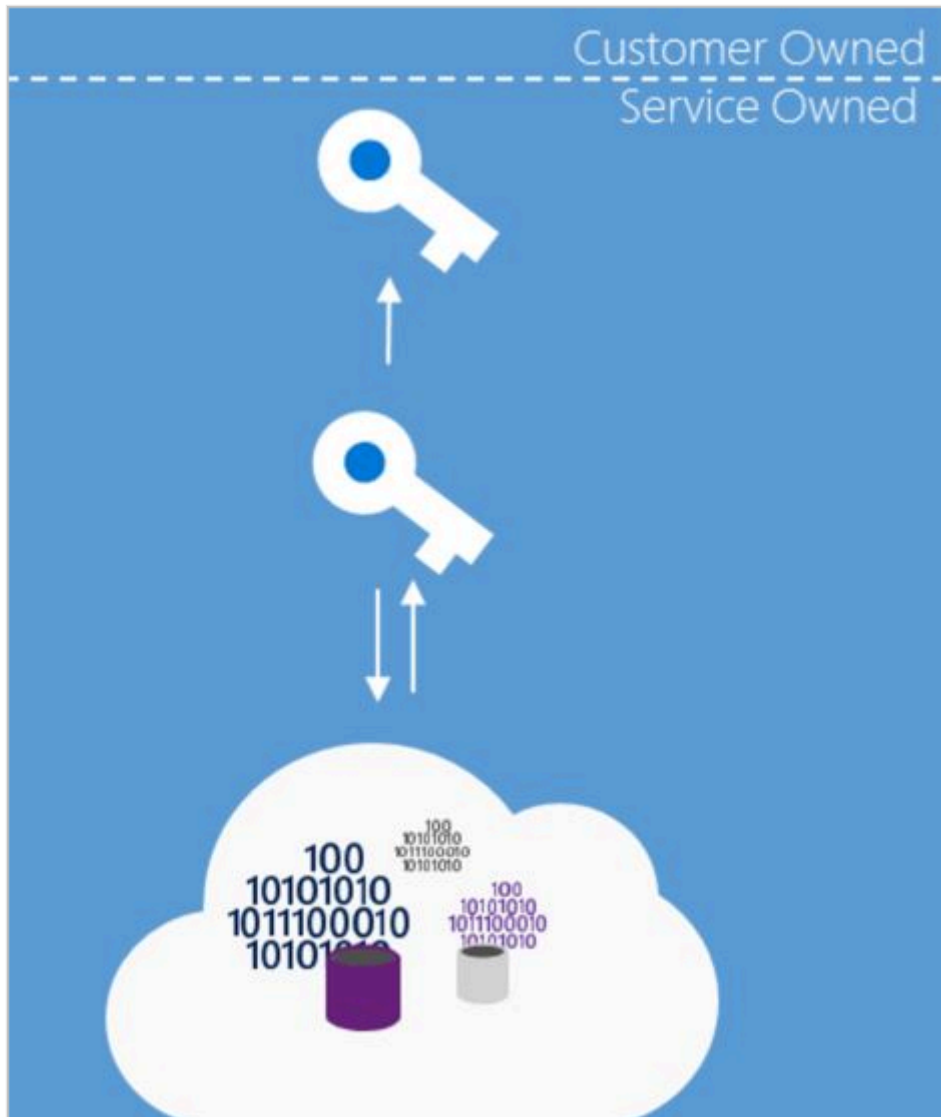
Client encryption model

Client Encryption model refers to encryption that is performed outside of the Resource Provider or Azure by the service or calling application. The encryption can be performed by the service application in Azure, or by an application running in the customer data center. In either case, when using this encryption model, the Azure Resource Provider receives an encrypted blob of data without the ability to decrypt the data in any way or have access to the encryption keys. In this model, the key management is done by the calling service/application and is opaque to the Azure service.



Server-side encryption using service-managed keys

For many customers, the essential requirement is to ensure that the data is encrypted whenever it is at rest. Server-side encryption using service-managed Keys enables this model by allowing customers to mark the specific resource (Storage Account, SQL DB, etc.) for encryption and leaving all key management aspects such as key issuance, rotation, and back up to Microsoft. Most Azure services that support encryption at rest typically support this model of offloading the management of the encryption keys to Azure. The Azure resource provider creates the keys, places them in secure storage, and retrieves them when needed. The service has full access to the keys and the service has full control over the credential lifecycle management.



Server-side encryption using service-managed keys therefore quickly addresses the need to have encryption at rest with low overhead to the customer. When available a customer typically opens the Azure portal for the target subscription and resource provider and checks a box indicating, they would like the data to be encrypted. In some Resource Managers, server-side encryption with service-managed keys is on by default.

Server-side encryption with Microsoft-managed keys does imply the service has full access to store and manage the keys. While some customers might want to manage the keys because they feel they gain greater security, the cost and risk associated with a custom key storage

solution should be considered when evaluating this model. In many cases, an organization might determine that resource constraints or risks of an on-premises solution might be greater than the risk of cloud management of the encryption at rest keys. However, this model might not be sufficient for organizations that have requirements to control the creation or lifecycle of the encryption keys or to have different personnel manage a service's encryption keys than those managing the service (that is, segregation of key management from the overall management model for the service).

Key access

When Server-side encryption with service-managed keys is used, the key creation, storage, and service access are all managed by the service. Typically, the foundational Azure resource providers store the Data Encryption Keys in a store that is close to the data and quickly available and accessible while the Key Encryption Keys are stored in a secure internal store.

Advantages

- Simple setup
- Microsoft manages key rotation, backup, and redundancy
- Customer does not have the cost associated with implementation or the risk of a custom key management scheme.

Disadvantages

- No customer control over the encryption keys (key specification, lifecycle, revocation, etc.)
- No ability to segregate key management from overall management model for the service

Server-side encryption using customer-managed keys in Azure Key Vault and Azure Managed HSM

For scenarios where the requirement is to encrypt the data at rest and control the encryption keys customers can use server-side encryption using customer-managed Keys in Key Vault. Some services might store only the root Key Encryption Key in Azure Key Vault and store the encrypted Data Encryption Key in an internal location closer to the data. In that scenario customers can bring their own keys to Key Vault (BYOK – Bring Your Own Key), or generate new ones, and use them to encrypt the desired resources. While the Resource Provider performs the encryption and decryption operations, it uses the configured key encryption key as the root key for all encryption operations.

Loss of key encryption keys means loss of data. For this reason, keys should not be deleted. Keys should be backed up whenever created or rotated. [Soft-Delete and purge protection](#) must be enabled on any vault storing key encryption keys to protect against accidental or malicious

cryptographic erasure. Instead of deleting a key, it is recommended to set enabled to false on the key encryption key. Use access controls to revoke access to individual users or services in [Azure Key Vault](#) or [Managed HSM](#).

ⓘ Note

For a list of services that support customer-managed keys in Azure Key Vault and Azure Managed HSM, see [Services that support CMKs in Azure Key Vault and Azure Managed HSM](#).

Key Access

The server-side encryption model with customer-managed keys in Azure Key Vault involves the service accessing the keys to encrypt and decrypt as needed. Encryption-at-rest keys are made accessible to a service through an access control policy. This policy grants the service identity access to receive the key. An Azure service running on behalf of an associated subscription can be configured with an identity in that subscription. The service can perform Microsoft Entra authentication and receive an authentication token identifying itself as that service acting on behalf of the subscription. That token can then be presented to Key Vault to obtain a key to which it has been given access.

For operations using encryption keys, a service identity can be granted access to any of the following operations: decrypt, encrypt, unwrapKey, wrapKey, verify, sign, get, list, update, create, import, delete, backup, and restore.

To obtain a key for use in encrypting or decrypting data at rest the service identity that the Resource Manager service instance will run as must have UnwrapKey (to get the key for decryption) and WrapKey (to insert a key into key vault when creating a new key).

ⓘ Note

For more detail on Key Vault authorization see the secure your key vault page in the [Azure Key Vault documentation](#).

Advantages

- Full control over the keys used – encryption keys are managed in the customer's Key Vault under the customer's control.
- Ability to encrypt multiple services to one master
- Can segregate key management from overall management model for the service
- Can define service and key location across regions

Disadvantages

- Customer has full responsibility for key access management
- Customer has full responsibility for key lifecycle management
- Additional Setup & configuration overhead

Server-side encryption using customer-managed keys in customer-controlled hardware

Some Azure services enable the Host Your Own Key (HYOK) key management model. This management mode is useful in scenarios where there is a need to encrypt the data at rest and manage the keys in a proprietary repository outside of Microsoft's control. In this model, the service must use the key from an external site to decrypt the Data Encryption Key (DEK). Performance and availability guarantees are affected, and configuration is more complex. Additionally, since the service does have access to the DEK during the encryption and decryption operations the overall security guarantees of this model are similar to when the keys are customer-managed in Azure Key Vault. As a result, this model is not appropriate for most organizations unless they have specific key management requirements. Due to these limitations, most Azure services do not support server-side encryption using customer-managed keys in customer-controlled hardware. One of two keys in [Double Key Encryption](#) follows this model.

Key Access

When server-side encryption using customer-managed keys in customer-controlled hardware is used, the key encryption keys are maintained on a system configured by the customer. Azure services that support this model provide a means of establishing a secure connection to a customer supplied key store.

Advantages

- Full control over the root key used – encryption keys are managed by a customer provided store
- Ability to encrypt multiple services to one master
- Can segregate key management from overall management model for the service
- Can define service and key location across regions

Disadvantages

- Full responsibility for key storage, security, performance, and availability
- Full responsibility for key access management
- Full responsibility for key lifecycle management

- Significant setup, configuration, and ongoing maintenance costs
- Increased dependency on network availability between the customer datacenter and Azure datacenters.

Related content

- [Services that support CMKs in Azure Key Vault and Azure Managed HSM](#)
- [How encryption is used in Azure](#)
- [Double encryption](#)

Overview of managed disk encryption options

09/24/2025

There are several types of encryption available for your managed disks, including Azure Disk Encryption (ADE), Server-Side Encryption (SSE), and encryption at host.

- **Azure Disk Storage Server-Side Encryption** (also referred to as encryption-at-rest or Azure Storage encryption) is always enabled and automatically encrypts data stored on Azure managed disks (OS and data disks) when persisting on the Storage Clusters. When configured with a Disk Encryption Set (DES), it supports customer-managed keys as well. It doesn't encrypt temp disks or disk caches. For full details, see [Server-side encryption of Azure Disk Storage](#).
- **Encryption at host** is a Virtual Machine option that enhances Azure Disk Storage Server-Side Encryption to ensure that all temp disks and disk caches are encrypted at rest and flow encrypted to the Storage clusters. For full details, see [Encryption at host - End-to-end encryption for your VM data](#).
- **Confidential disk encryption** binds disk encryption keys to the virtual machine's TPM and makes the protected disk content accessible only to the VM. The TPM and VM guest state is always encrypted in attested code using keys released by a secure protocol that bypasses the hypervisor and host operating system. Currently only available for the OS disk; [temp disk support is in preview](#) [↗](#). Encryption at host may be used for other disks on a Confidential VM in addition to Confidential Disk Encryption. For full details, see [DCasv5 and ECasv5 series confidential VMs](#).
- **Azure Disk Encryption** helps protect and safeguard your data to meet your organizational security and compliance commitments. ADE encrypts the OS and data disks of Azure virtual machines (VMs) inside your VMs by using the [DM-Crypt](#) [↗](#) feature of Linux or the [BitLocker](#) [↗](#) feature of Windows. ADE is integrated with Azure Key Vault to help you control and manage the disk encryption keys and secrets, with the option to encrypt with a key encryption key (KEK). For full details, see [Azure Disk Encryption for Linux VMs](#) or [Azure Disk Encryption for Windows VMs](#).

Important

Azure Disk Encryption for Virtual Machines and Virtual Machine Scale Sets will be retired on **September 15, 2028**. New customers should use [encryption at host](#) for all new VMs. Existing customers should plan to migrate current ADE-enabled VMs to

encryption at host before the retirement date to avoid service disruption -- see [Migrate from Azure Disk Encryption to encryption at host](#).

Encryption is part of a layered approach to security and should be used with other recommendations to secure Virtual Machines and their disks. For full details, see [Security recommendations for virtual machines in Azure](#) and [Restrict import/export access to managed disks](#).

Comparison

Here's a comparison of Disk Storage SSE, ADE, encryption at host, and Confidential disk encryption.

 Expand table

	Azure Disk Storage Server-Side Encryption	Encryption at Host	Azure Disk Encryption	Confidential disk encryption (For the OS disk only)
Encryption at rest (OS and data disks)	✓	✓	✓	✓
Temp disk encryption	✗	✓ Only supported with platform managed key	✓	✓ In Preview 
Encryption of caches	✗	✓	✓	✓
Data flows encrypted between Compute and Storage	✗	✓	✓	✓
Customer control of keys	✓ When configured with DES	✓ When configured with DES	✓ When configured with KEK	✓ When configured with DES
HSM Support	Azure Key Vault Premium and Managed HSM	Azure Key Vault Premium and Managed HSM	Azure Key Vault Premium	Azure Key Vault Premium and Managed HSM
Does not use your VM's CPU	✓	✓	✗	✗

	Azure Disk Storage Server-Side Encryption	Encryption at Host	Azure Disk Encryption	Confidential disk encryption (For the OS disk only)
Works for custom images	✓	✓	✗ Does not work for custom Linux images	✓
Enhanced Key Protection	✗	✗	✗	✓
Microsoft Defender for Cloud disk encryption status*	Unhealthy	Healthy	Healthy	Not applicable

Important

For Confidential disk encryption, Microsoft Defender for Cloud does not currently have a recommendation that is applicable.

* Microsoft Defender for Cloud has the following disk encryption recommendations:

- [Virtual machines and virtual machine scale sets should have encryption at host enabled](#)  (Only detects Encryption at Host)
- [Virtual machines should encrypt temp disks, caches, and data flows between Compute and Storage resources](#)  (Only detects Azure Disk Encryption)
- [Windows virtual machines should enable Azure Disk Encryption or EncryptionAtHost](#)  (Detects both Azure Disk Encryption and EncryptionAtHost)
- [Linux virtual machines should enable Azure Disk Encryption or EncryptionAtHost](#)  (Detects both Azure Disk Encryption and EncryptionAtHost)

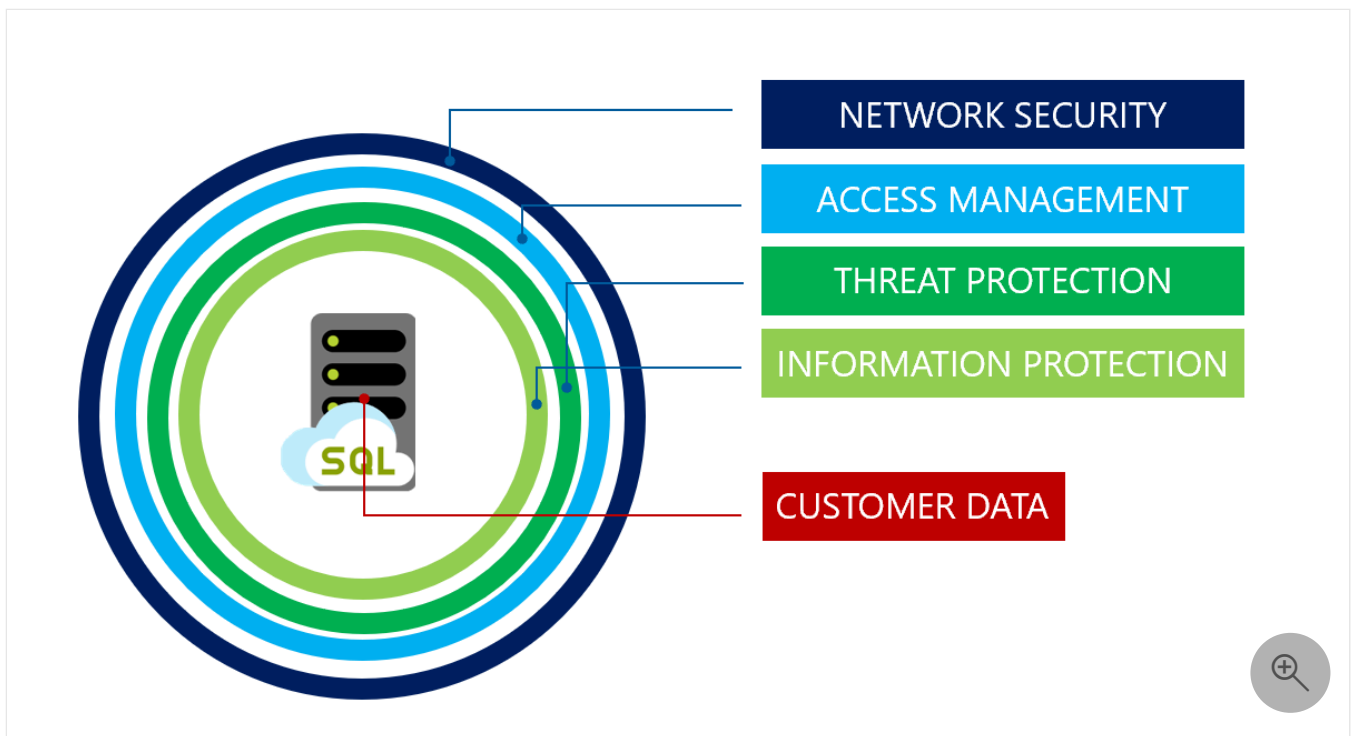
Next steps

- [Azure Disk Encryption for Linux VMs](#)
- [Azure Disk Encryption for Windows VMs](#)
- [Server-side encryption of Azure Disk Storage](#)
- [Encryption at host](#)
- [Migrate from Azure Disk Encryption to server-side encryption](#)
- [DCasv5 and ECasv5 series confidential VMs](#)
- [Azure Security Fundamentals - Azure encryption overview](#)

An overview of Azure SQL Database and SQL Managed Instance security capabilities

Applies to:  [Azure SQL Database](#)  [Azure SQL Managed Instance](#)  [Azure Synapse Analytics](#)

This article outlines the basics of securing the data tier of an application that uses [Azure SQL Database](#), [Azure SQL Managed Instance](#), and [Azure Synapse Analytics](#). The security strategy described in this article follows the layered defense-in-depth approach as shown in the following diagram, and moves from the outside in:



Network security

Azure SQL Database, Azure SQL Managed Instance, and Azure Synapse Analytics provide a relational database service for cloud and enterprise applications. To help protect customer data, firewalls prevent network access to the server until you explicitly grant access based on IP address or Azure Virtual network traffic origin.

IP firewall rules

IP firewall rules grant access to databases based on the originating IP address of each request. For more information, see [Overview of Azure SQL Database and Azure Synapse Analytics firewall rules](#).

Virtual network firewall rules

[Virtual network service endpoints](#) extend your virtual network connectivity over the Azure backbone and enable Azure SQL Database to identify the virtual network subnet that traffic originates from. To allow traffic to reach Azure SQL Database, use the SQL [service tags](#) to allow outbound traffic through Network Security Groups.

- [Virtual network rules](#) enable Azure SQL Database to only accept communications that are sent from selected subnets inside a virtual network.
- Controlling access with firewall rules doesn't apply to **SQL Managed Instance**. For more information about the networking configuration needed, see [Connecting to a managed instance](#)

ⓘ Note

Controlling access with firewall rules doesn't apply to **SQL Managed Instance**. For more information about the networking configuration needed, see [Connecting to a managed instance](#)

Network security perimeter

An [Azure network security perimeter](#) creates logical network boundaries around your platform-as-a-service (PaaS) resources that you deploy outside your virtual networks.

- An Azure network security perimeter helps you control public network access to Azure SQL Database.
- Controlling access with an Azure network security perimeter doesn't apply to Azure SQL Managed Instance.

ⓘ Important

Azure SQL Database with Network Security Perimeter is currently in preview. The previews are provided without a service level agreement, and it's not recommended for production workloads. Certain features might not be supported or might have constrained capabilities. For more information, see [Supplemental Terms of Use for Microsoft Azure Previews](#) [↗](#).

Authentication

Authentication is the process of proving the user is who they claim to be. Azure SQL Database and SQL Managed Instance support authentication with Microsoft Entra ID ([formerly Azure Active Directory](#)) and SQL authentication. SQL Managed instance additionally supports [Windows authentication](#) for Microsoft Entra principals.

- **Microsoft Entra authentication:**

Microsoft Entra authentication is a mechanism to connect to [Azure SQL Database](#), [Azure SQL Managed Instance](#), and [Azure Synapse Analytics](#) by using identities in Microsoft Entra ID. Microsoft Entra authentication allows administrators to centrally manage the identities and permissions of database users along with other Azure services in one central location. This feature can help eliminate the use of secrets and passwords.

To use Microsoft Entra authentication with SQL Database, create a server admin called the **Microsoft Entra administrator**. For more information, see [Connecting to SQL Database with Microsoft Entra authentication](#). Microsoft Entra authentication supports both managed and federated accounts. The federated accounts support Windows users and groups for a customer domain federated with Microsoft Entra ID.

Microsoft Entra supports several different authentication options, including [multifactor authentication](#), [Integrated Windows authentication](#), and [Conditional Access](#).

- **Windows authentication for Microsoft Entra principals:**

[Kerberos authentication for Microsoft Entra principals](#) enables Windows authentication for Azure SQL Managed Instance. Windows authentication for managed instances empowers customers to move existing services to the cloud while maintaining a seamless user experience and provides the basis for infrastructure modernization.

To enable Windows authentication for Microsoft Entra principals, turn your Microsoft Entra tenant into an independent Kerberos realm and create an incoming trust in the customer domain. Learn [how Windows authentication for Azure SQL Managed Instance is implemented with Microsoft Entra ID and Kerberos](#).

- **SQL authentication:**

SQL authentication refers to the authentication of a user when connecting to Azure SQL Database or Azure SQL Managed Instance by using a username and password. You must specify a **server admin** login with a username and password when creating the server. Using these credentials, a **server admin** can authenticate to any database on that server or instance as the database owner. After that, the server admin can create other SQL logins and users, which enable users to connect by using a username and password.

Authorization and access management

Authorization refers to controlling access to management of servers and databases, and to data, resources, and commands within a database. You assign permissions to a user within a database in Azure SQL Database or Azure SQL Managed Instance. Your portal user account's role assignments control managing databases and servers within Azure. For more information, see [Azure role-based access control in the Azure portal](#).

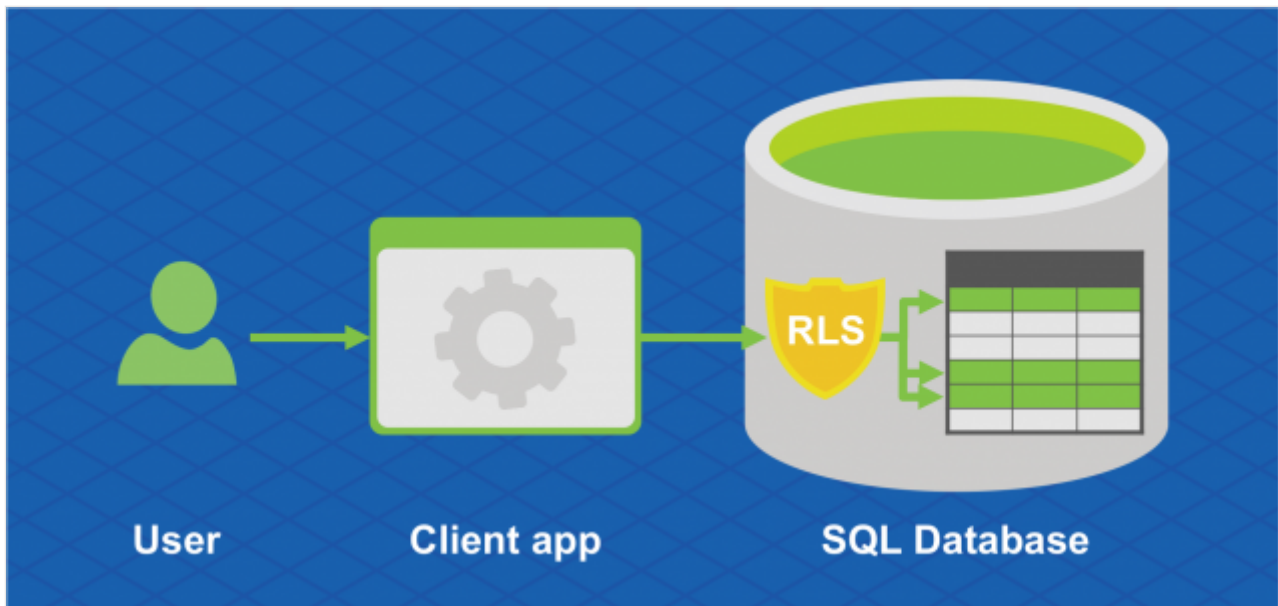
Manage permissions by adding user accounts to [database roles](#) and assigning database-level permissions to those roles. Alternatively, grant certain [object-level permissions](#) to an individual user. For more information, see [Logins and users](#).

In addition, Azure SQL Managed Instance provides [server-level roles](#) (fixed or custom) to manage permissions for an instance. Server-level roles have server-wide permissions scope. You can add server-level principals into server-level roles.

As a best practice, create custom roles when needed. Add users to the role with the least privileges required to do their job function. Don't assign permissions directly to users. The server admin account is a member of the built-in db_owner role, which has extensive permissions and should only be granted to a few users with administrative duties. To further limit the scope of what a user can do, use the [EXECUTE AS](#) to specify the execution context of the called module. Following these best practices is also a fundamental step toward Separation of Duties.

Row-level security

Row-Level Security enables you to control access to rows in a database table based on the characteristics of the user executing a query (for example, group membership or execution context). Use Row-Level Security to implement custom Label-based security concepts. For more information, see [Row-Level security](#).



Threat protection

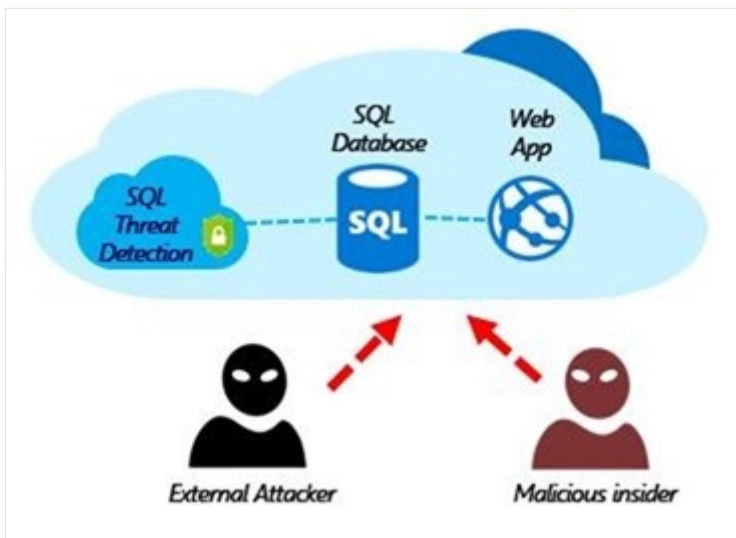
Azure SQL Database and SQL Managed Instance secure customer data by providing auditing and threat detection capabilities.

SQL auditing in Azure Monitor logs and Event Hubs

SQL Database and SQL Managed Instance auditing tracks database activities and helps maintain compliance with security standards by recording database events to an audit log in a customer-owned Azure storage account. Auditing allows you to monitor ongoing database activities, as well as analyze and investigate historical activity to identify potential threats or suspected abuse and security violations. For more information, see [Get started with SQL Database Auditing](#).

Advanced Threat Protection

Advanced Threat Protection analyzes your logs to detect unusual behavior and potentially harmful attempts to access or exploit databases. It creates alerts for suspicious activities such as SQL injection, potential data infiltration, and brute force attacks or for anomalies in access patterns to catch privilege escalations and breached credentials use. You can view alerts from [Microsoft Defender for Cloud](#), where the details of the suspicious activities are provided and recommendations for further investigation given along with actions to mitigate the threat. You can enable Advanced Threat Protection per server for an additional fee. For more information, see [Get started with SQL Database Advanced Threat Protection](#).



Information protection and encryption

Transport Layer Security (encryption-in-transit)

SQL Database, SQL Managed Instance, and Azure Synapse Analytics secure customer data by encrypting data in motion with [Transport Layer Security \(TLS\)](#). These services always enforce TLS encrypted connections to ensure all data is encrypted in transit between the client and server.

Specifically, SQL Database, SQL Managed Instance, and Azure Synapse Analytics set the configuration flag `ForceEncryption` to `Yes`. Clients and drivers must support encrypted connections to connect to these services. The lowest version of the TDS protocol that can connect is TDS 7.1.

As a best practice, if you have [TDS 8.0](#)-capable SQL drivers, use [Strict connection encryption](#).

If your drivers don't support TDS 8.0, use mandatory encryption and don't trust the server certificate. For example, when using the ADO.NET driver, use `Encrypt=True` and `TrustServerCertificate=False` in the connection string to accomplish this. The connection string you get from the Azure portal is already configured with these values.

Avoid setting the parameter `TrustServerCertificate` to `True` in production use. `TrustServerCertificate=True` is too permissive and doesn't protect against man-in-the-middle attacks. Instead, if your client expects a different domain name in the server certificate, use the `HostNameInCertificate` parameter to provide the correct domain name for validation.

For example, when using the ADO.NET driver to connect to your managed instance `contoso-instance.123456.database.windows.net` through a custom domain name `contoso-instance.contoso.com`, set the connection parameters `Encrypt=True` and set

`HostNameInCertificate=contoso-instance.123456.database.windows.net`. This configuration allows the driver to validate the server certificate against an expected VNet-local endpoint domain name.

Important

Some non-Microsoft drivers might not use TLS by default or might rely on an older version of TLS (earlier than 1.2) to function. In this case, the server still allows you to connect to your database. However, evaluate the security risks of allowing such drivers and applications to connect to SQL Database, especially if you store sensitive data.

For more information about TLS and connectivity, see [TLS considerations](#).

Transparent data encryption (encryption-at-rest) with service-managed keys

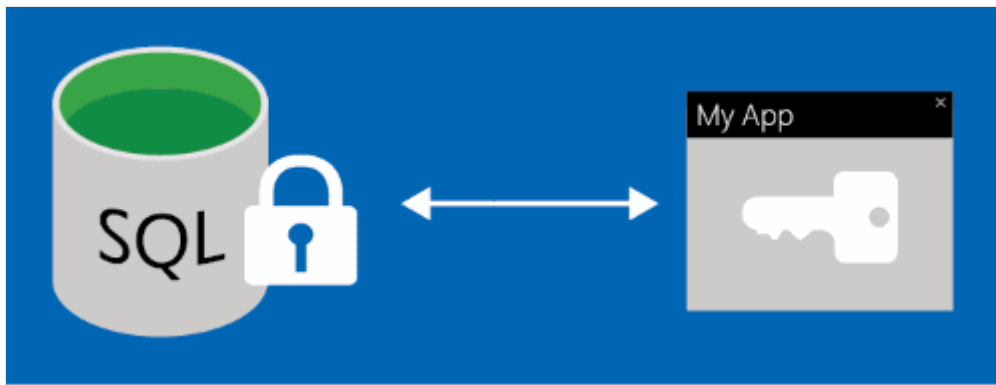
[Transparent data encryption \(TDE\) for SQL Database, SQL Managed Instance, and Azure Synapse Analytics](#) adds a layer of security to help protect data at rest from unauthorized or offline access to raw files or backups. Common scenarios include data center theft or unsecured disposal of hardware or media such as disk drives and backup tapes. TDE encrypts the entire database by using an AES encryption algorithm, which doesn't require application developers to make any changes to existing applications.

In Azure, all newly created databases are encrypted by default and the database encryption key is protected by a built-in server certificate. The service manages certificate maintenance and rotation and requires no input from the user. If you prefer to take control of the encryption keys, you can manage the keys in [Azure Key Vault](#).

Transparent data encryption (encryption-at-rest) with customer-managed keys

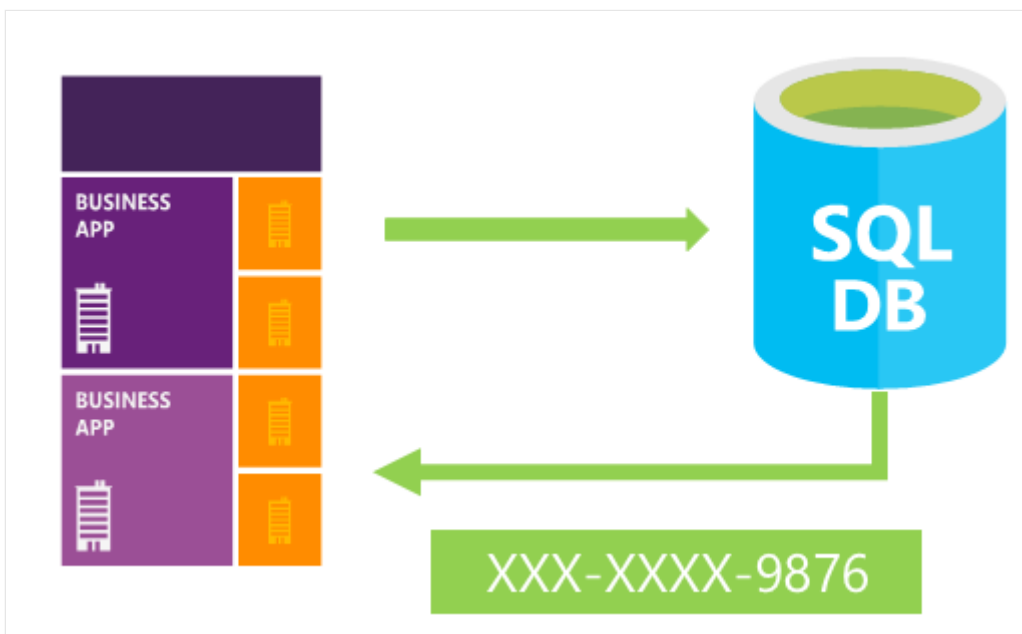
If you need greater control over encryption keys, [transparent data encryption](#) (TDE) supports [customer-managed keys \(CMK\)](#). This CMK is associated with the logical server and wraps the database encryption keys for all databases within that server. Alternatively, you can configure a CMK at the individual [database level](#). By managing the CMK, you can control key rotation, revocation, and auditing, which is often necessary for compliance or strict security policies.

Always Encrypted and Always Encrypted with secure enclaves (encryption-in-use)



[Always Encrypted](#) and [Always Encrypted with secure enclaves](#) are features designed to protect sensitive data stored in specific database columns from access (for example, credit card numbers, national/regional identification numbers, or data on a *need to know* basis). This protection includes database administrators or other privileged users who are authorized to access the database to perform management tasks but have no business need to access the particular data in the encrypted columns. The data is always encrypted, which means the encrypted data is decrypted only for processing by client applications with access to the encryption key. The encryption key is never exposed to SQL Database or SQL Managed Instance and can be stored either in the [Windows Certificate Store](#) or in [Azure Key Vault](#).

Dynamic data masking



Dynamic data masking limits sensitive data exposure by masking it to nonprivileged users. Dynamic data masking automatically discovers potentially sensitive data in Azure SQL Database and SQL Managed Instance and provides actionable recommendations to mask these fields, with minimal impact to the application layer. It works by obfuscating the sensitive data in the result set of a query over designated database fields, while the data in the database isn't changed. For more information, see [Get started with SQL Database and SQL Managed Instance dynamic data masking](#).

Ledger

[Ledger](#) in Azure SQL Database and SQL Managed Instance is a feature that provides cryptographic proof of data integrity. With ledger, you have tamper-evidence capabilities for your data. You can cryptographically attest to other parties, such as auditors or other business parties, that your data hasn't been tampered with.

Ledger uses tamper-evident technology to record database changes in an immutable ledger, ensuring that any unauthorized modifications can be detected. This feature is particularly useful for scenarios requiring regulatory compliance, auditability, and trust between multiple parties. By enabling ledger, you can verify the integrity of your data, reducing the risk of fraud or data manipulation.

Security management

Vulnerability assessment

[Vulnerability assessment](#) is an easy to configure service that can discover, track, and help remediate potential database vulnerabilities with the goal to proactively improve overall database security. Vulnerability assessment (VA) is part of the Microsoft Defender for SQL offering, which is a unified package for advanced SQL security capabilities. You can access and manage vulnerability assessment via the central Microsoft Defender for SQL portal.

Data discovery and classification

Data discovery and classification provides basic capabilities built into Azure SQL Database and SQL Managed Instance for discovering, classifying, and labeling the sensitive data in your databases. Discovering and classifying your most sensitive data (business, financial, healthcare, personal data, and more) plays a pivotal role in your organizational information protection stature. It serves as infrastructure for:

- Various security scenarios, such as monitoring (auditing) and alerting on anomalous access to sensitive data.
- Controlling access to, and hardening the security of, databases containing highly sensitive data.
- Helping meet data privacy standards and regulatory compliance requirements.

For more information, see [Get started with data discovery and classification](#).

Compliance

In addition to the features and functionality that help your application meet various security requirements, Azure SQL Database also participates in regular audits. It has been certified against a number of compliance standards. For more information, see the [Microsoft Azure Trust Center](#) where you can find the most current list of SQL Database compliance certifications.

Related content

- [SQL Server and client encryption summary](#)
- [Manage logins and user accounts](#)
- [auditing](#)
- [threat detection](#)

Last updated on 12/01/2025

Playbook for addressing common security requirements with Azure SQL Database and Azure SQL Managed Instance

Applies to:  [Azure SQL Database](#)  [Azure SQL Managed Instance](#)

This article provides best practices on how to solve common security requirements. Not all requirements are applicable to all environments, and you should consult your database and security team on which features to implement.

Note

[Microsoft Entra ID](#) was previously known as Azure Active Directory (Azure AD).

Solve common security requirements

This document provides guidance on how to solve common security requirements for new or existing applications using Azure SQL Database and Azure SQL Managed Instance. It's organized by high-level security areas. For addressing specific threats, refer to the [Common security threats and potential mitigations](#) section. Although some of the presented recommendations are applicable when migrating applications from on-premises to Azure, migration scenarios aren't the focus of this document.

Azure SQL Database deployment offers covered in this guide

- [Azure SQL Database: single databases](#) and [elastic pools](#) in [servers](#)
- [What is Azure SQL Managed Instance?](#)

Deployment offers not covered in this guide

- Azure Synapse Analytics
- Azure SQL VMs (IaaS)
- SQL Server

Audience

The intended audiences for this guide are customers facing questions on how to secure Azure SQL Database. The roles interested in this best practice article include, but not limited to:

- Security Architects
- Security Managers
- Compliance Officers
- Privacy Officers
- Security Engineers

How to use this guide

This document is intended as a companion to our existing [overview of Azure SQL Database and SQL Managed Instance security capabilities](#).

Unless otherwise stated, we recommend you follow all best practices listed in each section to achieve the respective goal or requirement. To meet specific security compliance standards or best practices, important regulatory compliance controls are listed under the Requirements or Goals section wherever applicable. These are the security standards and regulations that are referenced in this paper:

- [FedRAMP](#) [↗]: AC-04, AC-06
- [SOC](#) [↗]: CM-3, SDL-3
- [ISO/IEC 27001](#): Access Control, Cryptography
- [Microsoft Operational Security Assurance \(OSA\) practices](#) [↗]: Practice #1-6 and #9
- [NIST Special Publication 800-53 Security Controls](#) [↗]: AC-5, AC-6
- [PCI DSS](#) [↗]: 6.3.2, 6.4.2

We plan on continuing to update the recommendations and best practices listed here. Provide input or any corrections for this document using the **Feedback** link at the bottom of this article.

Authentication

Authentication is the process of proving the user is who they claim to be. Azure SQL Database and SQL Managed Instance support two types of authentication:

- SQL authentication
- Microsoft Entra authentication

ⓘ Note

Microsoft Entra authentication might not be supported for all tools and 3rd party applications.

Central management for identities

Central identity management offers the following benefits:

- Manage group accounts and control user permissions without duplicating logins across servers, databases and managed instances.
- Simplified and flexible permission management.
- Management of applications at scale.

How to implement

- Use Microsoft Entra authentication for centralized identity management.

Best practices

- Create a Microsoft Entra tenant and [create users](#) to represent human users and create [service principals](#) to represent apps, services, and automation tools. Service principals are equivalent to service accounts in Windows and Linux.
- Assign access rights to resources to Microsoft Entra principals via group assignment: Create Microsoft Entra groups, grant access to groups, and add individual members to the groups. In your database, create contained database users that map to your Microsoft Entra groups. To assign permissions inside the database, add the contained database users representing your groups to database roles, or grant permissions to them directly.
 - For more information, see [Configure and manage Microsoft Entra authentication with Azure SQL](#) and [Microsoft Entra authentication for Azure SQL](#).

ⓘ Note

In SQL Managed Instance, you can also create logins that map to Microsoft Entra principals in the `master` database. See [CREATE LOGIN \(Transact-SQL\)](#).

- Using Microsoft Entra groups simplifies permission management and both the group owner, and the resource owner can add/remove members to/from the group.
- Create a separate group for Microsoft Entra administrators for each server or managed instance.
 - See the article, [Provision a Microsoft Entra administrator for your server](#).
- Monitor Microsoft Entra group membership changes using Microsoft Entra ID audit activity reports.
- For a managed instance, a separate step is required to create a Microsoft Entra admin.
 - See the article, [Provision a Microsoft Entra administrator for your managed instance](#).

ⓘ Note

- Microsoft Entra authentication is recorded in Azure SQL audit logs, but not in Microsoft Entra sign-in logs.
- Azure RBAC permissions granted in Azure do not apply to Azure SQL Database or SQL Managed Instance permissions. Such permissions must be created/mapped manually using existing SQL permissions.
- On the client-side, Microsoft Entra authentication needs access to the internet or via User Defined Route (UDR) to a virtual network.
- The Microsoft Entra access token is cached on the client side and its lifetime depends on token configuration. See the article, [Configurable token lifetimes in Microsoft Entra ID](#)
- For guidance on troubleshooting Microsoft Entra authentication issues, see the following blog: [Troubleshooting Microsoft Entra ID](#) .

Microsoft Entra multifactor authentication

Mentioned in: OSA Practice #2, ISO Access Control (AC)

Microsoft Entra multifactor authentication helps provides additional security by requiring more than one form of authentication.

How to implement

- [Enable multifactor authentication](#) in Microsoft Entra ID using Conditional Access and use interactive authentication.
- The alternative is to enable multifactor authentication for the entire Microsoft Entra tenant or Active Directory domain.

Best practices

- Activate Conditional Access in Microsoft Entra ID (requires Premium subscription).
 - See the article, [Conditional Access in Microsoft Entra ID](#).
- Create Microsoft Entra group(s) and enable multifactor authentication policy for selected groups using Microsoft Entra Conditional Access.
 - See the article, [Plan Conditional Access Deployment](#).

- Multifactor authentication can be enabled for the entire Microsoft Entra tenant or for Active Directory federated with Microsoft Entra ID.
- Use Microsoft Entra interactive authentication mode for Azure SQL Database and Azure SQL Managed Instance where a password is requested interactively, followed by multifactor authentication:
 - Use universal authentication in SSMS. See the article, [Using Microsoft Entra multifactor authentication with Azure SQL Database, SQL Managed Instance, Azure Synapse \(SSMS support for multifactor authentication\)](#).
 - Use interactive authentication supported in SQL Server Data Tools (SSDT). See the article, [Microsoft Entra ID support in SQL Server Data Tools \(SSDT\)](#).
 - Use other SQL tools supporting multifactor authentication.
 - SSMS Wizard support for export/extract/deploy database
 - [SqlPackage](#): option '/ua'
 - [sqlcmd Utility](#): option -G (interactive)
 - [bcp Utility](#): option -G (interactive)
- Implement your applications to connect to Azure SQL Database or Azure SQL Managed Instance using interactive authentication with multifactor authentication support.
 - See the article, [Connect to Azure SQL Database with Microsoft Entra multifactor authentication](#).

ⓘ Note

This authentication mode requires user-based identities. In cases where a trusted identity model is used that is bypassing individual Microsoft Entra user authentication (for example, using managed identity for Azure resources), multifactor authentication does not apply.

Minimize the use of password-based authentication for users

Mentioned in: OSA Practice #4, ISO Access Control (AC)

Password-based authentication methods are a weaker form of authentication. Credentials can be compromised or mistakenly given away.

How to implement

- Use Microsoft Entra integrated authentication that eliminates the use of passwords.

Best practices

- Use single sign-on authentication using Windows credentials. Federate the on-premises Active Directory domain with Microsoft Entra ID and use integrated Windows authentication (for domain-joined machines with Microsoft Entra ID).
 - See the article, [SSMS support for Microsoft Entra integrated authentication](#).

Minimize the use of password-based authentication for applications

Mentioned in: OSA Practice #4, ISO Access Control (AC)

How to implement

- Enable Azure Managed Identity. You can also use integrated or certificate-based authentication.

Best practices

- Use [managed identities for Azure resources](#).
 - [System-assigned managed identity](#)
 - [User-assigned managed identity](#)
 - [Use Azure SQL Database from Azure App Service with managed identity \(without code changes\)](#)[↗]
- Use cert-based authentication for an application.
 - See this [code sample](#) [↗].
- Use Microsoft Entra authentication for integrated federated domain and domain-joined machine (see section above).
 - See the [sample application for integrated authentication](#) [↗].

Protect passwords and secrets

For cases when passwords aren't avoidable, make sure they're secured.

How to implement

- Use Azure Key Vault to store passwords and secrets. Whenever applicable, use multifactor authentication for Azure SQL Database with Microsoft Entra users.

Best practices

- If avoiding passwords or secrets aren't possible, store user passwords and application secrets in Azure Key Vault, and manage access through Key Vault access policies.

- Various app development frameworks might also offer framework-specific mechanisms for protecting secrets in the app. For example: [ASP.NET core app](#).

Use SQL authentication for legacy applications

SQL authentication refers to the authentication of a user when connecting to Azure SQL Database or SQL Managed Instance using username and password. A login will need to be created in each server or managed instance, and a user created in each database.

How to implement

- Use SQL authentication.

Best practices

- As a server or instance admin, create logins and users. Unless using contained database users with passwords, all passwords are stored in `master` database. In Azure SQL Database, the `master` database is part of the [logical server](#).
 - See the article, [Authorize database access to SQL Database, SQL Managed Instance, and Azure Synapse Analytics](#).

Access management

Access management (also called Authorization) is the process of controlling and managing authorized users' access and privileges to Azure SQL Database or SQL Managed Instance.

Implement principle of least privilege

Mentioned in: FedRamp controls AC-06, NIST: AC-6, OSA Practice #3

The principle of least privilege states that users shouldn't have more privileges than needed to complete their tasks. For more information, see [Just enough administration](#).

How to implement

Assign only the necessary [permissions](#) to complete the required tasks:

- In SQL Databases:
 - Use granular permissions and user-defined database roles (or server-roles in SQL Managed Instance):
 1. Create the required roles
 - [CREATE ROLE](#)

- [CREATE SERVER ROLE](#)
- 2. Create required users
 - [CREATE USER](#)
- 3. Add users as members to roles
 - [ALTER ROLE](#)
 - [ALTER SERVER ROLE](#)
- 4. Then assign permissions to roles.
 - [GRANT](#)
- Make sure to not assign users to unnecessary roles.
- In Azure Resource Manager:
 - Use built-in roles if available or Azure custom roles and assign the necessary permissions.
 - [Azure built-in roles](#)
 - [Azure custom roles](#)

Best practices

The following best practices are optional but will result in better manageability and supportability of your security strategy:

- If possible, start with the least possible set of permissions and start adding permissions one by one if there's a real necessity (and justification) – as opposed to the opposite approach: taking permissions away step by step.
- Refrain from assigning permissions to individual users. Use roles (database or server roles) consistently instead. Roles helps greatly with reporting and troubleshooting permissions. (Azure RBAC only supports permission assignment via roles.)
- Create and use custom roles with the exact permissions needed. Typical roles that are used in practice:
 - Security deployment
 - Administrator
 - Developer
 - Support personnel
 - Auditor
 - Automated processes
 - End user
- Use built-in roles only when the permissions of the roles match exactly the needed permissions for the user. You can assign users to multiple roles.
- Remember that permissions in the database engine can be applied within the following scopes (the smaller the scope, the smaller the impact of the granted permissions):

- Server (special roles in the `master` database) in Azure
- Database
- Schema
 - It is a best practice to use schemas to grant permissions inside a database.
- Object (table, view, procedure, and so on)

⚠ Note

It is not recommended to apply permissions on the object level because this level adds unnecessary complexity to the overall implementation. If you decide to use object-level permissions, those should be clearly documented. The same applies to column-level-permissions, which are even less recommendable for the same reasons. Also be aware that by default a table-level **DENY** does not override a column-level GRANT. This would require the [common criteria compliance Server Configuration](#) to be activated.

- Perform regular checks using [Vulnerability Assessment \(VA\)](#) to test for too many permissions.

Implement Separation of Duties

Mentioned in: FedRamp: AC-04, NIST: AC-5, ISO: A.6.1.2, PCI 6.4.2, SOC: CM-3, SDL-3

Separation of Duties, also called Segregation of Duties describes the requirement to split sensitive tasks into multiple tasks that are assigned to different users. Separation of Duties helps prevent data breaches.

How to implement

- Identify the required level of Separation of Duties. Examples:
 - Between Development/Test and Production environments
 - Security-wise sensitive tasks vs Database Administrator (DBA) management level tasks vs developer tasks.
 - Examples: Auditor, creation of security policy for Role-level Security (RLS), Implementing SQL Database objects with DDL-permissions.
- Identify a comprehensive hierarchy of users (and automated processes) that access the system.
- Create roles according to the needed user-groups and assign permissions to roles.

- For management-level tasks in Azure portal or via PowerShell-automation use Azure roles. Either find a built-in role matching the requirement, or create an Azure custom role using the available permissions
- Create Server roles for server-wide tasks (creating new logins, databases) in a managed instance.
- Create Database Roles for database-level tasks.
- For certain sensitive tasks, consider creating special stored procedures signed by a certificate to execute the tasks on behalf of the users. One important advantage of digitally signed stored procedures is that if the procedure is changed, the permissions that were granted to the previous version of the procedure are immediately removed.
 - Example: [Tutorial: Signing Stored Procedures with a Certificate](#)
- Implement Transparent Data Encryption (TDE) with customer-managed keys in Azure Key Vault to enable Separation of Duties between data owner and security owner.
 - See the article, [Configure customer-managed keys for Azure Storage encryption from the Azure portal](#).
- To ensure that a DBA can't see data that is considered highly sensitive and can still do DBA tasks, you can use Always Encrypted with role separation.
 - See the articles, [Overview of Key Management for Always Encrypted](#), [Key Provisioning with Role Separation](#), and [Column Master Key Rotation with Role Separation](#).
- In cases where the use of Always Encrypted isn't feasible, or at least not without major costs and efforts that might even render the system near unusable, compromises can be made and mitigated through the use of compensating controls such as:
 - Human intervention in processes.
 - Audit trails – for more information on Auditing, see, [Audit critical security events](#).

Best practices

- Make sure that different accounts are used for Development/Test and Production environments. Different accounts help to comply with separation of Test and Production systems.
- Refrain from assigning permissions to individual users. Use roles (database or server roles) consistently instead. Having roles helps greatly with reporting and troubleshooting permissions.
- Use built-in roles when the permissions match exactly the needed permissions – if the union of all permissions from multiple built-in roles leads to a 100% match, you can assign multiple roles concurrently as well.

- Create and use user-defined roles when built-in roles grant too many permissions or insufficient permissions.
- Role assignments can also be done temporarily, also known as Dynamic Separation of Duties (DSD), either within SQL Agent Job steps in T-SQL or using Azure PIM for Azure roles.
- Make sure that DBAs don't have access to the encryption keys or key stores, and that Security Administrators with access to the keys have no access to the database in turn. The use of [Extensible Key Management \(EKM\)](#) can make this separation easier to achieve. [Azure Key Vault](#) [↗](#) can be used to implement EKM.
- Always make sure to have an Audit trail for security-related actions.
- You can retrieve the definition of the Azure built-in roles to see the permissions used and create a custom role based on excerpts and cumulations of these via PowerShell.
- Because any member of the db_owner database role can change security settings like Transparent Data Encryption (TDE), or change the SLO, this membership should be granted with care. However, there are many tasks that require db_owner privileges. Task like changing any database setting such as changing DB options. Auditing plays a key role in any solution.
- It is not possible to restrict permissions of a db_owner, and therefore prevent an administrative account from viewing user data. If there's highly sensitive data in a database, Always Encrypted can be used to safely prevent db_owners or any other DBA from viewing it.

ⓘ Note

Achieving Separation of Duties (SoD) is challenging for security-related or troubleshooting tasks. Other areas like development and end-user roles are easier to segregate. Most compliance related controls allow the use of alternate control functions such as Auditing when other solutions aren't practical.

For the readers that want to dive deeper into SoD, we recommend the following resources:

- For Azure SQL Database and SQL Managed Instance:
 - [Authorize database access](#)
 - [Engine Separation of Duties for the Application Developer](#)
 - [Separation of Duties](#) [↗](#)
 - [Signing Stored Procedures](#)

- For Azure Resource Management:
 - [Azure built-in roles](#)
 - [Azure custom roles](#)

Perform regular code reviews

Mentioned in: PCI: 6.3.2, SOC: SDL-3

Separation of Duties is not limited to the data in a database, but includes application code. Malicious code can potentially circumvent security controls. Before deploying custom code to production, it is essential to review what's being deployed.

How to implement

- Use a database tool like SQL Server Management Studio that supports source control.
- Implement a segregated code deployment process.
- Before committing to main branch, a person (other than the author of the code itself) has to inspect the code for potential elevation of privileges risks as well as malicious data modifications to protect against fraud and rogue access. This can be done using source control mechanisms.

Best practices

- Standardization: It helps to implement a standard procedure that is to be followed for any code updates.
- Vulnerability Assessment contains rules that check for excessive permissions, the use of old encryption algorithms, and other security problems within a database schema.
- Further checks can be done in a QA or test environment using Advanced Threat Protection that scans for code that is vulnerable to SQL-injection.
- Examples of what to look out for:
 - Creation of a user or changing security settings from within an automated SQL-code-update deployment.
 - A stored procedure, which, depending on the parameters provided, updates a monetary value in a cell in a non-conforming way.
- Make sure the person conducting the review is an individual other than the originating code author and knowledgeable in code-reviews and secure coding.

- Be sure to know all sources of code-changes. Code can be in T-SQL Scripts. It can be ad hoc commands to be executed or be deployed in forms of Views, Functions, Triggers, and Stored Procedures. It can be part of SQL Agent Job definitions (Steps). It can also be executed from within SSIS packages, Azure Data Factory, and other services.

Data protection

Data protection is a set of capabilities for safeguarding important information from compromise by encryption or obfuscation.

ⓘ Note

Microsoft attests to Azure SQL Database and Azure SQL Managed Instance as being FIPS 140-2 Level 1 compliant. This is done after verifying the strict use of FIPS 140-2 Level 1 acceptable algorithms and FIPS 140-2 Level 1 validated instances of those algorithms including consistency with required key lengths, key management, key generation, and key storage. This attestation is meant to allow our customers to respond to the need or requirement for the use of FIPS 140-2 Level 1 validated instances in the processing of data or delivery of systems or applications. We define the terms "FIPS 140-2 Level 1 compliant" and "FIPS 140-2 Level 1 compliance" used in the above statement to demonstrate their intended applicability to U.S. and Canadian government use of the different term "FIPS 140-2 Level 1 validated."

Encrypt data in transit

Mentioned in: OSA Practice #6, ISO Control Family: Cryptography

Protects your data while data moves between your client and server. Refer to [Network Security](#).

Encrypt data at rest

Mentioned in: OSA Practice #6, ISO Control Family: Cryptography

Encryption at rest is the cryptographic protection of data when it is persisted in database, log, and backup files.

How to implement

- [Transparent data encryption \(TDE\)](#) with service managed keys are enabled by default for any databases created after 2017 in Azure SQL Database and SQL Managed Instance.

- In a managed instance, if the database is created from a restore operation using an on-premises server, the TDE setting of the original database will be honored. If the original database doesn't have TDE enabled, we recommend that TDE be manually turned on for the managed instance.

Best practices

- Don't store data that requires encryption-at-rest in the `master` database. The `master` database can't be encrypted with TDE.
- Use customer-managed keys in Azure Key Vault if you need increased transparency and granular control over the TDE protection. Azure Key Vault allows the ability to revoke permissions at any time to render the database inaccessible. You can centrally manage TDE protectors along with other keys, or rotate the TDE protector at your own schedule using Azure Key Vault.
- If you're using customer-managed keys in Azure Key Vault, follow the articles, [Guidelines for configuring TDE with Azure Key Vault](#) and [How to configure Geo-DR with Azure Key Vault](#).

ⓘ Note

Some items considered customer content, such as table names, object names, and index names, might be transmitted in log files for support and troubleshooting by Microsoft.

Protect sensitive data in use from high-privileged, unauthorized users

Data in use is the data stored in memory of the database system during the execution of SQL queries. If your database stores sensitive data, your organization could be required to ensure that high-privileged users are prevented from viewing sensitive data in your database. High-privilege users, such as Microsoft operators or DBAs in your organization should be able to manage the database, but prevented from viewing and potentially exfiltrating sensitive data from the memory of the SQL process or by querying the database.

The policies that determine which data is sensitive and whether the sensitive data must be encrypted in memory and not accessible to administrators in plaintext, are specific to your organization and compliance regulations you need to adhere to. Please see the related requirement: [Identify and tag sensitive data](#).

How to implement

- Use [Always Encrypted](#) to ensure sensitive data isn't exposed in plaintext in Azure SQL Database or SQL Managed Instance, even in memory/in use. Always Encrypted protects the data from Database Administrators (DBAs) and cloud admins (or bad actors who can impersonate high-privileged but unauthorized users) and gives you more control over who can access your data.

Best practices

- Always Encrypted isn't a substitute to encrypt data at rest (TDE) or in transit (SSL/TLS). Always Encrypted shouldn't be used for non-sensitive data to minimize performance and functionality impact. Using Always Encrypted in conjunction with TDE and Transport Layer Security (TLS) is recommended for comprehensive protection of data at-rest, in-transit, and in-use.
- Assess the impact of encrypting the identified sensitive data columns before you deploy Always Encrypted in a production database. In general, Always Encrypted reduces the functionality of queries on encrypted columns and has other limitations, listed in [Always Encrypted - Feature Details](#). Therefore, you might need to rearchitect your application to reimplement the functionality, a query does not support, on the client side or/and refactor your database schema, including the definitions of stored procedures, functions, views and triggers. Existing applications might not work with encrypted columns if they do not adhere to the restrictions and limitations of Always Encrypted. While the ecosystem of Microsoft tools, products and services supporting Always Encrypted is growing, a number of them do not work with encrypted columns. Encrypting a column can also affect query performance, depending on the characteristics of your workload.
- Manage Always Encrypted keys with role separation if you're using Always Encrypted to protect data from malicious DBAs. With role separation, a security admin creates the physical keys. The DBA creates the column master key and column encryption key metadata objects describing the physical keys in the database. During this process, the security admin doesn't need access to the database, and the DBA doesn't need access to the physical keys in plaintext.
 - See the article, [Managing Keys with Role Separation](#) for details.
- Store your column master keys in Azure Key Vault for ease of management. Avoid using Windows Certificate Store (and in general, distributed key store solutions, as opposed central key management solutions) that make key management hard.
- Think carefully through the tradeoffs of using multiple keys (column master key or column encryption keys). Keep the number of keys small to reduce key management cost. One column master key and one column encryption key per database is typically sufficient in steady-state environments (not in the middle of a key rotation). You might

need additional keys if you have different user groups, each using different keys and accessing different data.

- Rotate column master keys per your compliance requirements. If you also need to rotate column encryption keys, consider using online encryption to minimize application downtime.
 - For more information, see [Performance and Availability Considerations](#).
- Use deterministic encryption if computations (equality) on data need to be supported. Otherwise, use randomized encryption. Avoid using deterministic encryption for low-entropy data sets, or data sets with publicly known distribution.
- If you're concerned about third parties accessing your data legally without your consent, ensure that all application and tools that have access to the keys and data in plaintext run outside of Microsoft Azure Cloud. Without access to the keys, the third party will have no way of decrypting the data unless they bypass the encryption.
- Always Encrypted doesn't easily support granting temporary access to the keys (and the protected data). For example, if you need to share the keys with a DBA to allow the DBA to do some cleansing operations on sensitive and encrypted data. The only way to reliably revoke the access to the data from the DBA will be to rotate both the column encryption keys and the column master keys protecting the data, which is an expensive operation.
- To access the plaintext values in encrypted columns, a user needs to have access to the Column Master Key (CMK) that protects columns, which is configured in the key store holding the CMK. The user also needs to have the **VIEW ANY COLUMN MASTER KEY DEFINITION** and **VIEW ANY COLUMN ENCRYPTION KEY DEFINITION** database permissions.

Control access of application users to sensitive data through encryption

Encryption can be used as a way to ensure that only specific application users who have access to cryptographic keys can view or update the data.

How to implement

- Use Cell-level Encryption (CLE). See the article, [Encrypt a Column of Data](#) for details.
- Use Always Encrypted, but be aware of its limitation. The limitations are listed below.

Best practices:

When using CLE:

- Control access to keys through SQL permissions and roles.
- Use AES (AES 256 recommended) for data encryption. Algorithms, such as RC4, DES and TripleDES, are deprecated and shouldn't be used because of known vulnerabilities.
- Protect symmetric keys with asymmetric keys/certificates (not passwords) to avoid using 3DES.
- Be careful when migrating a database using Cell-Level Encryption via export/import (bacpac files).
 - See the article, [Recommendations for using Cell Level Encryption in Azure SQL Database](#) on how to prevent losing keys when migrating data, and for other best practice guidance.

Keep in mind that Always Encrypted is primarily designed to protect sensitive data in use from high-privilege users of Azure SQL Database (cloud operators, DBAs). For more information, see [Protect sensitive data in use from high-privileged, unauthorized users](#). Be aware of the following challenges when using Always Encrypted to protect data from application users:

- By default, all Microsoft client drivers supporting Always Encrypted maintain a global (one per application) cache of column encryption keys. Once a client driver acquires a plaintext column encryption key by contacting a key store holding a column master key, the plaintext column encryption key is cached. This makes isolating data from users of a multi-user application challenging. If your application impersonates end users when interacting with a key store (such as Azure Key Vault), after a user's query populates the cache with a column encryption key, a subsequent query that requires the same key but is triggered by another user will use the cached key. The driver won't call the key store and it won't check if the second user has a permission to access the column encryption key. As a result, the user can see the encrypted data even if the user doesn't have access to the keys. To achieve the isolation of users within a multi-user application, you can disable column encryption key caching. Disabling caching will cause additional performance overheads, as the driver will need to contact the key store for each data encryption or decryption operation.

Protect data against unauthorized viewing by application users while preserving data format

Another technique for preventing unauthorized users from viewing data is to obfuscate or mask the data while preserving data types and formats to ensure that user applications can continue handle and display the data.

How to implement

- Use [Dynamic Data Masking](#) to obfuscate table columns.

ⓘ Note

Always Encrypted does not work with Dynamic Data Masking. It is not possible to encrypt and mask the same column, which implies that you need to prioritize protecting data in use vs. masking the data for your app users via Dynamic Data Masking.

Best practices

ⓘ Note

Dynamic Data Masking cannot be used to protect data from high-privilege users. Masking policies do not apply to users with administrative access like `db_owner`.

- Don't permit app users to run ad hoc queries (as they might be able to work around Dynamic Data Masking). For more information, see [Bypassing masking using inference or brute-force techniques](#).
- Use a proper access control policy (via SQL permissions, roles, RLS) to limit user permissions to make updates in the masked columns. Creating a mask on a column doesn't prevent updates to that column. Users that receive masked data when querying the masked column, can update the data if they have write-permissions.
- Dynamic Data Masking doesn't preserve the statistical properties of the masked values. This might affect query results (for example, queries containing filtering predicates or joins on the masked data).

Network security

Network security refers to access controls and best practices to secure your data in transit to Azure SQL Database.

Configure my client to connect securely to SQL Database/SQL Managed Instance

Best practices on how to prevent client machines and applications with well-known vulnerabilities (for example, using older TLS protocols and cipher suites) from connecting to Azure SQL Database and SQL Managed Instance.

How to implement

- Ensure that client machines connecting to Azure SQL Database and SQL Managed Instance are using the latest [Transport Layer Security \(TLS\)](#) version.

Best practices

- Enforce a minimal TLS version in the [logical server in Azure SQL Database](#) or [Azure SQL Managed Instance](#) by using the minimal TLS version setting. We recommend setting the minimal TLS version to 1.2, after testing to confirm your applications supports it. TLS 1.2 includes fixes for vulnerabilities found in previous versions.
- Configure all your apps and tools to connect to SQL Database with encryption enabled
 - Encrypt = On, TrustServerCertificate = Off (or equivalent with non-Microsoft drivers).
- If your app uses a driver that doesn't support TLS or supports an older version of TLS, replace the driver, if possible. If not possible, carefully evaluate the security risks.
 - Reduce attack vectors via vulnerabilities in SSL 2.0, SSL 3.0, TLS 1.0, and TLS 1.1 by disabling them on client machines connecting to Azure SQL Database per [Transport Layer Security \(TLS\) registry settings](#).
 - Check cipher suites available on the client: [Cipher Suites in TLS/SSL \(Schannel SSP\)](#). Specifically, disable 3DES per [Configuring TLS Cipher Suite Order](#).

Minimize attack surface

Minimize the number of features that can be attacked by a malicious user. Implement network access controls for Azure SQL Database.

Mentioned in: OSA Practice #5

How to implement

In Azure SQL Database:

- Set Allow Access to Azure services to OFF at the server-level
- Use VNet Service endpoints and VNet Firewall Rules.
- Use Private Link.

In SQL Managed Instance:

- Follow the guidelines in [Network requirements](#).

Best practices

- Restricting access to Azure SQL Database and SQL Managed Instance by connecting on a private endpoint (for example, using a private data path):

- A managed instance can be isolated inside a virtual network to prevent external access. Applications and tools that are in the same or peered virtual network in the same region could access it directly. Applications and tools that are in different region could use virtual-network-to-virtual-network connection or ExpressRoute circuit peering to establish connection. Customer should use Network Security Groups (NSG) to restrict access over port 1433 only to resources that require access to a managed instance.
- For Azure SQL Database, use [Private Link](#), which provides a dedicated private IP for the server inside your virtual network. You can also use [virtual network service endpoints](#) to restrict access to your logical servers.
- Mobile users should use point-to-site VPN connections to connect over the data path.
- Users connected to their on-premises network should use site-to-site VPN connection or ExpressRoute to connect over the data path.
- You can access Azure SQL Database and SQL Managed Instance by connecting to a public endpoint (for example, using a public data path). The following best practices should be considered:
 - For a server in SQL Database, use [Azure SQL Database and Azure Synapse IP firewall rules](#) to restrict access to only authorized IP addresses.
 - For SQL Managed Instance, use Network Security Groups (NSG) to restrict access over port 3342 only to required resources. For more information, see [Use Azure SQL Managed Instance securely with public endpoints](#).

ⓘ **Note**

The SQL Managed Instance public endpoint is not enabled by default and it must be explicitly enabled. If company policy disallows the use of public endpoints, use [Azure Policy](#) to prevent enabling public endpoints in the first place.

- Set up Azure Networking components:
 - Follow [Azure best practices for network security](#).
 - Plan Virtual Network configuration per best practices outlined in [Azure Virtual Network frequently asked questions \(FAQ\)](#) and plan.
 - Segment a virtual network into multiple subnets and assign resources for similar role to the same subnet (for example, front-end vs back-end resources).
 - Use [Network Security Groups \(NSGs\)](#) to control traffic between subnets inside the Azure virtual network boundary.
 - Enable [Azure Network Watcher](#) for your subscription to monitor inbound and outbound network traffic.

Configure Power BI for secure connections to SQL Database/SQL Managed Instance

Best practices

- For Power BI Desktop, use private data path whenever possible.
- Ensure that Power BI Desktop is connecting using TLS1.2 by setting the registry key on the client machine as per [Transport Layer Security \(TLS\)](#) registry settings.
- Restrict data access for specific users via [Row-level security \(RLS\) with Power BI](#).
- For Power BI Service, use the [on-premises data gateway](#), keeping in mind [Limitations and Considerations](#).

Configure App Service for secure connections to SQL Database/SQL Managed Instance

Best practices

- For a simple Web App, connecting over public endpoint requires setting **Allow Azure Services** to ON.
- [Integrate your app with an Azure Virtual Network](#) for private data path connectivity to a managed instance. Optionally, you can also deploy a Web App with [App Service Environments \(ASE\)](#).
- For Web App with ASE or virtual network Integrated Web App connecting to a database in SQL Database, you can use [virtual network service endpoints and virtual network firewall rules](#) to limit access from a specific virtual network and subnet. Then set **Allow Azure Services** to OFF. You can also connect ASE to a managed instance in SQL Managed Instance over a private data path.
- Ensure that your Web App is configured per the article, [Best practices for securing platform as a service \(PaaS\) web and mobile applications using Azure App Service](#).
- Install [Web Application Firewall \(WAF\)](#) to protect your web app from common exploits and vulnerabilities.

Configure Azure Virtual Machine hosting for secure connections to SQL Database/SQL Managed Instance

Best practices

- Use a combination of Allow and Deny rules on the NSGs of Azure virtual machines to control which regions can be accessed from the VM.
- Ensure that your VM is configured per the article, [Security best practices for IaaS workloads in Azure](#).
- Ensure that all VMs are associated with a specific virtual network and subnet.
- Evaluate if you need the default route 0.0.0.0/Internet per the guidance at [about forced tunneling](#).
 - If yes – for example, front-end subnet - then keep the default route.
 - If no – for example, middle tier or back-end subnet – then enable force tunneling so no traffic goes over Internet to reach on-premises (a.k.a cross-premises).
- Implement [optional default routes](#) if you're using peering or connecting to on-premises.
- Implement [User Defined Routes](#) if you need to send all traffic in the virtual network to a Network Virtual Appliance for packet inspection.
- Use [virtual network service endpoints](#) for secure access to PaaS services like Azure Storage via the Azure backbone network.

Protect against Distributed Denial of Service (DDoS) attacks

Distributed Denial of Service (DDoS) attacks are attempts by a malicious user to send a flood of network traffic to Azure SQL Database with the aim of overwhelming the Azure infrastructure and causing it to reject valid logins and workload.

Mentioned in: OSA Practice #9

How to implement

DDoS protection is automatically enabled as part of the Azure Platform. It includes always-on traffic monitoring and real-time mitigation of network-level attacks on public endpoints.

- Use [Azure DDoS Protection](#) to monitor public IP addresses associated to resources deployed in virtual networks.
- Use [SQL Advanced Threat Protection](#) to detect Denial of Service (DoS) attacks against databases.

Best practices

- Follow the practices described in [Minimize Attack Surface](#) helps minimize DDoS attack threats.

- The Advanced Threat Protection **Brute force SQL credentials** alert helps to detect brute force attacks. In some cases, the alert can even distinguish penetration testing workloads.
- For Azure VM hosting applications connecting to SQL Database:
 - Follow recommendation to Restrict access through Internet-facing endpoints in Microsoft Defender for Cloud.
 - Use virtual machine scale sets to run multiple instances of your application on Azure VMs.
 - Disable RDP and SSH from Internet to prevent brute force attack.

Monitoring, logging, and auditing

This section refers to capabilities to help you detect anomalous activities indicating unusual and potentially harmful attempts to access or exploit databases. It also describes best practices to configure database auditing to track and capture database events.

Protect databases against attacks

Advanced threat protection enables you to detect and respond to potential threats as they occur by providing security alerts on anomalous activities.

How to implement

- Use [Advanced Threat Protection for SQL](#) to detect unusual and potentially harmful attempts to access or exploit databases, including:
 - SQL injection attack.
 - Credentials theft/leak.
 - Privilege abuse.
 - Data exfiltration.

Best practices

- Configure [Microsoft Defender for SQL](#) for a specific server or a managed instance. You can also configure Microsoft Defender for SQL for all servers and managed instances in a subscription by enabling [Microsoft Defender for Cloud](#).
- For a full investigation experience, it's recommended to enable [Auditing for Azure SQL Database](#). With auditing, you can track database events and write them to an audit log in an Azure Storage account or Azure Log Analytics workspace.

Audit critical security events

Tracking of database events helps you understand database activity. You can gain insight into discrepancies and anomalies that could indicate business concerns or suspected security violations. It also enables and facilitates adherence to compliance standards.

How to implement

- Enable [Azure SQL Database Auditing](#) or [SQL Managed Instance auditing](#) to track database events and write them to an audit log in your Azure Storage account, Log Analytics workspace (preview), or Event Hubs (preview).
- Audit logs can be written to an Azure Storage account, to a Log Analytics workspace for consumption by Azure Monitor logs, or to event hub for consumption using event hub. You can configure any combination of these options, and audit logs will be written to each.

Best practices

- By configuring [Auditing for Azure SQL Database](#) or [Azure SQL Managed Instance](#) to audit events, all existing and newly created databases on that server will be audited.
- By default auditing policy includes all actions (queries, stored procedures and successful and failed logins) against the databases, which might result in high volume of audit logs. It's recommended for customers to [configure auditing for different types of actions and action groups using PowerShell](#). Configuring this will help control the number of audited actions, and minimize the risk of event loss. Custom audit configurations allow customers to capture only the audit data that is needed.
- Audit logs can be consumed directly in the [Azure portal](#) , or from the storage location that was configured.

Note

Enabling auditing to Log Analytics will incur cost based on ingestion rates. Please be aware of the associated cost with using this [option](#) , or consider storing the audit logs in an Azure storage account.

Further resources

- [Auditing for Azure SQL Database](#)
- [Auditing for Azure SQL Managed Instance](#)
- [SQL Server Auditing](#)

Secure audit logs

Restrict access to the storage account to support Separation of Duties and to separate DBA from Auditors.

How to implement

- When saving Audit logs to Azure Storage, make sure that access to the Storage Account is restricted to the minimal security principles. Control who has access to the storage account.
- For more information, see [Authorizing access to Azure Storage](#).

Best practices

- Controlling Access to the Audit Target is a key concept in separating DBA from Auditors.
- When auditing access to sensitive data, consider securing the data with data encryption to avoid information leakage to the Auditor. For more information, see the section [Protect sensitive data in use from high-privileged, unauthorized users](#).

Security Management

This section describes the different aspects and best practices for managing your databases security posture. It includes best practices for ensuring your databases are configured to meet security standards, for discovering and for classifying and tracking access to potentially sensitive data in your databases.

Ensure that the databases are configured to meet security best practices

Proactively improve your database security by discovering and remediating potential database vulnerabilities.

How to implement

- Enable [SQL Vulnerability Assessment](#) (VA) to scan your database for security issues, and to automatically run periodically on your databases.

Best practices

- Initially, run VA on your databases and iterate by remediating failing checks that oppose security best practices. Set up baselines for acceptable configurations until the scan comes out *clean*, or all checks has passed.
- Configure periodic recurring scans to run once a week and configure the relevant person to receive summary emails.

- Review the VA summary following each weekly scan. For any vulnerabilities found, evaluate the drift from the previous scan result and determine if the check should be resolved. Review if there's a legitimate reason for the change in configuration.
- Resolve checks and update baselines where relevant. Create ticket items for resolving actions and track these until they're resolved.

Further resources

- [SQL Vulnerability Assessment](#)
- [SQL Vulnerability Assessment service helps you identify database vulnerabilities](#)

Identify and tag sensitive data

Discover columns that potentially contain sensitive data. What is considered sensitive data heavily depends on the customer, compliance regulation, etc., and needs to be evaluated by the users in charge of that data. Classify the columns to use advanced sensitivity-based auditing and protection scenarios.

How to implement

- Use [Data Discovery and Classification](#) to discover, classify, label, and protect the sensitive data in your databases.
 - View the classification recommendations that are created by the automated discovery in the SQL Data Discovery and Classification dashboard. Accept the relevant classifications, such that your sensitive data is persistently tagged with classification labels.
 - Manually add classifications for any additional sensitive data fields that were not discovered by the automated mechanism.
- For more information, see [SQL Data Discovery and Classification](#).

Best practices

- Monitor the classification dashboard on a regular basis for an accurate assessment of the database's classification state. A report on the database classification state can be exported or printed to share for compliance and auditing purposes.
- Continuously monitor the status of recommended sensitive data in SQL Vulnerability Assessment. Track the sensitive data discovery rule and identify any drift in the recommended columns for classification.
- Use classification in a way that is tailored to the specific needs of your organization. Customize your Information Protection policy (sensitivity labels, information types, discovery logic) in the [SQL Information Protection](#) policy in Microsoft Defender for Cloud.

Track access to sensitive data

Monitor who accesses sensitive data and capture queries on sensitive data in audit logs.

How to implement

- Use SQL Audit and Data Classification in combination.
 - In your [SQL Database Audit](#) log, you can track access specifically to sensitive data. You can also view information such as the data that was accessed, as well as its sensitivity label. For more information, see [Data Discovery and Classification](#) and [Auditing access to sensitive data](#).

Best practices

- See best practices for the Auditing and Data Classification sections:
 - [Audit critical security events](#)
 - [Identify and tag sensitive data](#)

Visualize security and compliance status

Use a unified infrastructure security management system that strengthens the security posture of your data centers (including databases in SQL Database). View a list of recommendations concerning the security of your databases and compliance status.

How to implement

- Monitor SQL-related security recommendations and active threats in [Microsoft Defender for Cloud](#).

Common security threats and potential mitigations

This section helps you find security measures to protect against certain attack vectors. It's expected that most mitigations can be achieved by following one or more of the security guidelines above.

Security threat: Data exfiltration

Data exfiltration is the unauthorized copying, transfer, or retrieval of data from a computer or server. See a definition for [data exfiltration](#) [↗](#) on Wikipedia.

Connecting to server over a public endpoint presents a data exfiltration risk as it requires customers open their firewalls to public IPs.

Scenario 1: An application on an Azure VM connects to a database in Azure SQL Database. A rogue actor gets access to the VM and compromises it. In this scenario, data exfiltration means that an external entity using the rogue VM connects to the database, copies personal data, and stores it in a blob storage or a different SQL Database in a different subscription.

Scenario 2: A rogue DBA. This scenario is often raised by security sensitive customers from regulated industries. In this scenario, a high privilege user might copy data from Azure SQL Database to another subscription not controlled by the data owner.

Potential mitigations

Today, Azure SQL Database and SQL Managed Instance offer the following techniques for mitigating data exfiltration threats:

- Use a combination of Allow and Deny rules on the NSGs of Azure VMs to control which regions can be accessed from the VM.
- If using a server in SQL Database, set the following options:
 - Allow Azure Services to OFF.
 - Only allow traffic from the subnet containing your Azure VM by setting up a VNet Firewall rule.
 - Use [Private Link](#)
- For SQL Managed Instance, using private IP access by default addresses the first data exfiltration concern of a rogue VM. Turn on the subnet delegation feature on a subnet to automatically set the most restrictive policy on a SQL Managed Instance subnet.
- The Rogue DBA concern is more exposed with SQL Managed Instance as it has a larger surface area and networking requirements are visible to customers. The best mitigation for this is applying all of the practices in this security guide to prevent the Rogue DBA scenario in the first place (not only for data exfiltration). Always Encrypted is one method to protect sensitive data by encrypting it and keeping the key inaccessible for the DBA.

Security aspects of business continuity and availability

Most security standards address data availability in terms of operational continuity, achieved by implementing redundancy and fail-over capabilities to avoid single points of failure. For disaster scenarios, it's a common practice to keep backups of Data and Log files. The following section provides a high-level overview of the capabilities that are built-into Azure. It also provides additional options that can be configured to meet specific needs:

- Azure offers built-in high-availability: [Availability through redundancy - Azure SQL Database](#)

- The Business Critical tier includes failover groups, full and differential log backups, and point-in-time-restore backups enabled by default:
 - [Automated backups in Azure SQL Database](#)
 - [Recover a database using automated database backups - Point-in-time restore](#)
- Additional business continuity features such as the zone redundant configuration and failover groups across different Azure geos can be configured:
 - [High-availability - Zone redundant configuration for Premium & Business Critical service tiers](#)
 - [High-availability - Zone redundant configuration for General Purpose service tier](#)
 - [Business continuity in Azure SQL Database](#)

Next step

An overview of Azure SQL Database and SQL Managed Instance security capabilities

Last updated on 06/13/2025

Azure database security checklist

To help improve security, Azure SQL Database and Azure SQL Managed Instance include built-in security controls that you can use to limit and control access, protect data, and monitor threats.

Security controls include:

- Firewall rules limiting connectivity by IP address and virtual network
- Microsoft Entra authentication for centralized identity management
- Secure connectivity using TLS encryption
- Access management and authorization
- Data encryption at rest and in transit
- Database auditing and threat detection
- Advanced data security features

Introduction

Cloud computing requires new security paradigms that may be unfamiliar to many application users, database administrators, and programmers. Organizations can leverage Azure SQL's comprehensive security features to protect sensitive data and meet regulatory compliance requirements.

Checklist

We recommend that you read the [Azure SQL Database security best practices](#) article before reviewing this checklist. Understanding the best practices will help you get the most value from this checklist. Use this checklist to verify that you've addressed the important security controls in Azure database security.

 Expand table

Checklist Category	Description
Protect Data	
Encryption in transit	• Transport Layer Security (TLS) encrypts data in motion between clients and databases. Azure SQL requires TLS 1.2 or higher for secure connections. • Database requires secure communication from clients based on the TDS (Tabular Data Stream) protocol over TLS.

Checklist Category	Description
Encryption at rest	• Transparent Data Encryption (TDE) encrypts data and log files at rest. TDE is enabled by default on all new Azure SQL databases. • Bring Your Own Key (BYOK) allows you to manage TDE encryption keys in Azure Key Vault.
Encryption in use	• Always Encrypted protects sensitive data by encrypting it within client applications. Encryption keys never reach the database engine, ensuring separation between data owners and data managers. • Column-Level Encryption (CLE) encrypts specific columns using symmetric encryption for additional protection of sensitive data.
Control Access	
Database access	• Microsoft Entra authentication provides centralized identity management with single sign-on (SSO) capabilities. • SQL authentication with strong passwords provides an alternative authentication method. • Authorization grants users the minimum privileges necessary using role-based access control.
Network access control	• Server-level IP firewall rules restrict access based on originating IP addresses. • Database-level IP firewall rules provide granular access control per database. • Virtual Network service endpoints allow connectivity from specific Azure virtual networks. • Private Link provides private connectivity to Azure SQL Database using a private IP address within your virtual network.
Application access control	• Row-Level Security (RLS) restricts row-level access based on a user's identity, role, or execution context. • Dynamic Data Masking limits sensitive data exposure by masking it to non-privileged users without changing the underlying data. • Data Discovery and Classification identifies, classifies, and labels sensitive data for improved protection and compliance.
Proactive Monitoring	
Auditing and detection	• Auditing tracks database events and writes them to an audit log in your Azure Storage account, Log Analytics workspace, or Event Hubs. • Track Azure SQL Database health using Azure Monitor and diagnostic settings. • Microsoft Defender for SQL detects anomalous database activities indicating potential security threats including SQL injection, brute-force attacks, and vulnerability exploits.

Checklist Category	Description
Vulnerability assessment	• Vulnerability Assessment discovers, tracks, and helps remediate potential database vulnerabilities. • Provides actionable security recommendations and risk reports for compliance.
Centralized security management	• Microsoft Defender for Cloud provides centralized security monitoring and management for Azure SQL Database and other Azure services. • Security recommendations based on the Microsoft cloud security benchmark.
Data Integrity	
Ledger capability	• Ledger provides tamper-evident capabilities by creating an immutable record of database transactions. • Helps meet compliance requirements for data integrity verification.

Conclusion

Azure SQL Database and Azure SQL Managed Instance provide robust database platforms with comprehensive security features meeting organizational and regulatory compliance requirements. You can protect data throughout its lifecycle—at rest, in transit, and in use—using Transparent Data Encryption, Always Encrypted, and TLS. Fine-grained access controls including Row-Level Security, Dynamic Data Masking, and Microsoft Entra authentication ensure only authorized users access sensitive data. Continuous monitoring through auditing, Microsoft Defender for SQL, and Vulnerability Assessment helps identify and remediate security threats proactively.

Next steps

You can improve the protection of your database against malicious users or unauthorized access with a few simple steps:

- Configure [firewall rules](#) for your server and databases
- Protect your data with [encryption](#)
- Enable [SQL Database auditing](#)
- Enable [Microsoft Defender for SQL](#) for threat detection
- Review [security best practices](#)

Security recommendations for Blob storage

Article • 03/04/2025

This article contains security recommendations for Blob storage. Implementing these recommendations will help you fulfill your security obligations as described in our shared responsibility model. For more information on how Microsoft fulfills service provider responsibilities, see [Shared responsibility in the cloud](#).

Some of the recommendations included in this article can be automatically monitored by Microsoft Defender for Cloud, which is the first line of defense in protecting your resources in Azure. For information on Microsoft Defender for Cloud, see [What is Microsoft Defender for Cloud?](#)

Microsoft Defender for Cloud periodically analyzes the security state of your Azure resources to identify potential security vulnerabilities. It then provides you with recommendations on how to address them. For more information on Microsoft Defender for Cloud recommendations, see [Review your security recommendations](#).

Data protection

 Expand table

Recommendation	Comments	Defender for Cloud
Use the Azure Resource Manager deployment model	Create new storage accounts using the Azure Resource Manager deployment model for important security enhancements, including superior Azure role-based access control (Azure RBAC) and auditing, Resource Manager-based deployment and governance, access to managed identities, access to Azure Key Vault for secrets, and Microsoft Entra authentication and authorization for access to Azure Storage data and resources. Migrate all existing storage accounts that use the classic deployment model to use Azure Resource Manager. For more information about Azure Resource Manager, see Azure Resource Manager overview .	-
Enable Microsoft Defender for all of your storage accounts	Microsoft Defender for Storage provides an additional layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts. Security alerts are triggered in Microsoft	Yes

Recommendation	Comments	Defender for Cloud
	Defender for Cloud when anomalies in activity occur and are also sent via email to subscription administrators, with details of suspicious activity and recommendations on how to investigate and remediate threats. For more information, see Configure Microsoft Defender for Storage .	
Turn on soft delete for blobs	Soft delete for blobs enables you to recover blob data after it has been deleted. For more information on soft delete for blobs, see Soft delete for Azure Storage blobs .	-
Turn on soft delete for containers	Soft delete for containers enables you to recover a container after it has been deleted. For more information on soft delete for containers, see Soft delete for containers .	-
Lock storage account to prevent accidental or malicious deletion or configuration changes	Apply an Azure Resource Manager lock to your storage account to protect the account from accidental or malicious deletion or configuration change. Locking a storage account does not prevent data within that account from being deleted. It only prevents the account itself from being deleted. For more information, see Apply an Azure Resource Manager lock to a storage account .	
Store business-critical data in immutable blobs	Configure legal holds and time-based retention policies to store blob data in a WORM (Write Once, Read Many) state. Blobs stored immutably can be read, but cannot be modified or deleted for the duration of the retention interval. For more information, see Store business-critical blob data with immutable storage .	-
Use Encryption to Protect Data	Azure Storage encrypts all data at rest by default using Microsoft-managed keys. For enhanced control, configure customer-managed keys with Azure Key Vault to manage encryption keys directly. To further strengthen security, implement client-side encryption before uploading data.	-
Require secure transfer (HTTPS) to the storage account	When you require secure transfer for a storage account, all requests to the storage account must be made over HTTPS. Any requests made over HTTP are rejected. Microsoft recommends that you always require secure transfer for all of your storage accounts. For more information, see Require secure transfer to ensure secure connections .	-
Limit shared access signature (SAS) tokens	Requiring HTTPS when a client uses a SAS token to access blob data helps to minimize the risk of eavesdropping. For	-

Recommendation	Comments	Defender for Cloud
to HTTPS connections only	more information, see Grant limited access to Azure Storage resources using shared access signatures (SAS) .	
Disallow cross-tenant object replication	By default, an authorized user is permitted to configure an object replication policy where the source account is in one Microsoft Entra tenant and the destination account is in a different tenant. Disallow cross-tenant object replication to require that the source and destination accounts participating in an object replication policy are in the same tenant. For more information, see Prevent object replication across Microsoft Entra tenants .	-

Identity and access management

 Expand table

Recommendation	Comments	Defender for Cloud
Use Microsoft Entra ID to authorize access to blob data	Microsoft Entra ID provides superior security and ease of use over Shared Key for authorizing requests to Blob storage. For more information, see Authorize access to data in Azure Storage .	-
Keep in mind the principle of least privilege when assigning permissions to a Microsoft Entra security principal via Azure RBAC	When assigning a role to a user, group, or application, grant that security principal only those permissions that are necessary for them to perform their tasks. Limiting access to resources helps prevent both unintentional and malicious misuse of your data.	-
Use a user delegation SAS to grant limited access to blob data to clients	A user delegation SAS is secured with Microsoft Entra credentials and also by the permissions specified for the SAS. A user delegation SAS is analogous to a service SAS in terms of its scope and function, but offers security benefits over the service SAS. For more information, see Grant limited access to Azure Storage resources using shared access signatures (SAS) .	-
Secure your account access keys with Azure Key Vault	Microsoft recommends using Microsoft Entra ID to authorize requests to Azure Storage. However, if you must use Shared Key authorization, then secure your account keys with Azure Key Vault. You can retrieve the keys from the key vault at runtime, instead of saving them with your application. For more	-

Recommendation	Comments	Defender for Cloud
	information about Azure Key Vault, see Azure Key Vault overview .	
Regenerate your account keys periodically	Rotating the account keys periodically reduces the risk of exposing your data to malicious actors.	-
Disallow Shared Key authorization	When you disallow Shared Key authorization for a storage account, Azure Storage rejects all subsequent requests to that account that are authorized with the account access keys. Only secured requests that are authorized with Microsoft Entra ID will succeed. For more information, see Prevent Shared Key authorization for an Azure Storage account .	-
Keep in mind the principle of least privilege when assigning permissions to a SAS	When creating a SAS, specify only those permissions that are required by the client to perform its function. Limiting access to resources helps prevent both unintentional and malicious misuse of your data.	-
Have a revocation plan in place for any SAS that you issue to clients	If a SAS is compromised, you will want to revoke that SAS as soon as possible. To revoke a user delegation SAS, revoke the user delegation key to quickly invalidate all signatures associated with that key. To revoke a service SAS that is associated with a stored access policy, you can delete the stored access policy, rename the policy, or change its expiry time to a time that is in the past. For more information, see Grant limited access to Azure Storage resources using shared access signatures (SAS) .	-
If a service SAS is not associated with a stored access policy, then set the expiry time to one hour or less	A service SAS that is not associated with a stored access policy cannot be revoked. For this reason, limiting the expiry time so that the SAS is valid for one hour or less is recommended.	-
Disable anonymous read access to containers and blobs	anonymous read access to a container and its blobs grants read-only access to those resources to any client. Avoid enabling anonymous read access unless your scenario requires it. To learn how to disable anonymous access for a storage account, see Overview: Remediating anonymous read access for blob data .	-

Networking

Recommendation	Comments	Defender for Cloud
Configure the minimum required version of Transport Layer Security (TLS) for a storage account.	Require that clients use a more secure version of TLS to make requests against an Azure Storage account by configuring the minimum version of TLS for that account. For more information, see Configure minimum required version of Transport Layer Security (TLS) for a storage account	-
Enable the Secure transfer required option on all of your storage accounts	When you enable the Secure transfer required option, all requests made against the storage account must take place over secure connections. Any requests made over HTTP will fail. For more information, see Require secure transfer in Azure Storage .	Yes
Enable firewall rules	Configure firewall rules to limit access to your storage account to requests that originate from specified IP addresses or ranges, or from a list of subnets in an Azure Virtual Network (VNet). For more information about configuring firewall rules, see Configure Azure Storage firewalls and virtual networks .	-
Allow trusted Microsoft services to access the storage account	Turning on firewall rules for your storage account blocks incoming requests for data by default, unless the requests originate from a service operating within an Azure Virtual Network (VNet) or from allowed public IP addresses. Requests that are blocked include those from other Azure services, from the Azure portal, from logging and metrics services, and so on. You can permit requests from other Azure services by adding an exception to allow trusted Microsoft services to access the storage account. For more information about adding an exception for trusted Microsoft services, see Configure Azure Storage firewalls and virtual networks .	-
Use private endpoints	A private endpoint assigns a private IP address from your Azure Virtual Network (VNet) to the storage account. It secures all traffic between your VNet and the storage account over a private link. For more information about private endpoints, see Connect privately to a storage account using Azure Private Endpoint .	-
Use VNet service tags	A service tag represents a group of IP address prefixes from a given Azure service. Microsoft manages the address prefixes encompassed by the service tag and automatically updates the service tag as addresses change. For more information about service tags	-

Recommendation	Comments	Defender for Cloud
	supported by Azure Storage, see Azure service tags overview . For a tutorial that shows how to use service tags to create outbound network rules, see Restrict access to PaaS resources .	
Limit network access to specific networks	Limiting network access to networks hosting clients requiring access reduces the exposure of your resources to network attacks.	Yes
Configure network routing preference	You can configure network routing preference for your Azure storage account to specify how network traffic is routed to your account from clients over the Internet using the Microsoft global network or Internet routing. For more information, see Configure network routing preference for Azure Storage .	-

Logging/Monitoring

 Expand table

Recommendation	Comments	Defender for Cloud
Track how requests are authorized	Enable logging for Azure Storage to track how requests to the service are authorized. The logs indicate whether a request was made anonymously, by using an OAuth 2.0 token, by using Shared Key, or by using a shared access signature (SAS). For more information, see Monitoring Azure Blob Storage with Azure Monitor or Azure Storage analytics logging with Classic Monitoring .	-
Set up alerts in Azure Monitor	Configure log alerts to evaluate resources logs at a set frequency and fire an alert based on the results. For more information, see Log alerts in Azure Monitor .	-

Next steps

- [Azure security documentation](#)
- [Secure development documentation](#).

Feedback

Was this page helpful?



Yes



No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

Customer Lockbox for Microsoft Azure

Article • 04/16/2025

ⓘ Note

To use this feature, your organization must have an [Azure support plan](#) [↗] with a minimal level of **Developer**.

Most operations and support performed by Microsoft personnel and subprocessors do not require access to customer data. In those rare circumstances where such access is required, Customer Lockbox for Microsoft Azure provides an interface for customers to review and approve or reject customer data access requests. It is used in cases where a Microsoft engineer needs to access customer data, whether in response to a customer-initiated support ticket or a problem identified by Microsoft.

This article covers how to enable Customer Lockbox for Microsoft Azure and how requests are initiated, tracked, and stored for later reviews and audits.

Supported services

The following services are currently supported for Customer Lockbox for Microsoft Azure:

- Azure API Management
- Azure App Service
- Azure AI Search
- Azure AI Services
- Azure Chaos Studio
- Azure Communications Gateway
- Azure Container Registry
- Azure Data Box
- Azure Data Explorer
- Azure Data Factory
- Azure Data Manager for Energy
- Azure Database for MySQL
- Azure Database for MySQL Flexible Server
- Azure Database for PostgreSQL
- Azure Edge Zone Platform Storage
- Azure Energy
- Azure Functions
- Azure HDInsight

- Azure Health Bot
- Azure Intelligent Recommendations
- Azure Information Protection
- Azure Kubernetes Service
- Azure Load Testing (CloudNative Testing)
- Azure Logic Apps
- Azure Monitor (Log Analytics)
- Azure Red Hat OpenShift
- Azure Spring Apps
- Azure SQL Database
- Azure SQL Managed Instance
- Azure Storage
- Azure Subscription Transfers
- Azure Synapse Analytics
- Commerce AI (Intelligent Recommendations)
- DevCenter / DevBox
- ElasticSan
- Kusto (Dashboards)
- Microsoft Azure Attestation
- Microsoft Entra Diagnostics Data
- OpenAI
- Spring Cloud
- Unified Vision Service
- Virtual Machines in Azure

Enable Customer Lockbox for Microsoft Azure

You can now enable Customer Lockbox for Microsoft Azure from the [Administration module](#).

ⓘ Note

To enable Customer Lockbox for Microsoft Azure, the user account needs to have the [Global Administrator role assigned](#).

Workflow

The following steps outline a typical workflow for a Customer Lockbox for Microsoft Azure request.

1. Someone at an organization has an issue with their Azure workload.

2. After this person troubleshoots the issue, but can't fix it, they open a support ticket from the [Azure portal](#). The ticket is assigned to an Azure Customer Support Engineer.
3. An Azure Support Engineer reviews the service request and determines the next steps to resolve the issue.
4. If the support engineer can't troubleshoot the issue by using standard tools and service generated data, the next step is to request elevated permissions by using a Just-In-Time (JIT) access service. This request can be from the original support engineer or from a different engineer because the problem is escalated to the Azure DevOps team.
5. After the Azure Engineer submits an access request, Just-In-Time service evaluates the request taking into account factors such as:
 - The scope of the resource.
 - Whether the requester is an isolated identity or using multifactor authentication.
 - Permissions levels. Based on the JIT rule, this request might also include an approval from Internal Microsoft Approvers. For example, the approver might be the Customer support lead or the DevOps Manager.
6. When the request requires direct access to customer data, a Customer Lockbox request is initiated.

The request is now in a **Customer Notified** state, waiting for the customer's approval before granting access.

7. One or more approvers at the customer organization for a given Customer Lockbox request are determined as follows:
 - For Subscription scoped requests (requests to access specific resources contained within a subscription), users with the Owner role or the Azure Customer Lockbox Approver for Subscription role on the associated subscription.
 - For Tenant scope requests (requests to access the Microsoft Entra tenant), users with the Global Administrator role on the Tenant.

Note

Role assignments must be in place before Customer Lockbox for Microsoft Azure starts to process a request. Any role assignments made after Customer Lockbox for Microsoft Azure starts to process a given request will not be recognized. Because of this, to use PIM eligible assignments for the Subscription Owner role, users are required to activate the role before the Customer Lockbox request is initiated. Refer to [Activate Microsoft Entra roles in PIM](#) / [Activate Azure resource roles in PIM](#) for more information on activating PIM eligible roles.

Role assignments scoped to management groups are not supported in Customer Lockbox for Microsoft Azure at this time.

8. At the customer organization, designated lockbox approvers ([Azure Subscription Owner/Microsoft Entra Global admin](#)/Azure Customer Lockbox Approver for Subscription receive an email from Microsoft to notify them about the pending access request. You can also use the [Azure Lockbox alternate email notifications](#) feature to configure an alternate email address to receive lockbox notifications in scenarios where Azure account is not email enabled or if a service principal is defined as the lockbox approver.

Example email:



Customer Lockbox for Microsoft Azure

A customer Lockbox request is pending your approval.

Please [sign in](#) to the Azure portal and go to the "Customer Lockbox for Microsoft Azure" service blade to approve this request.

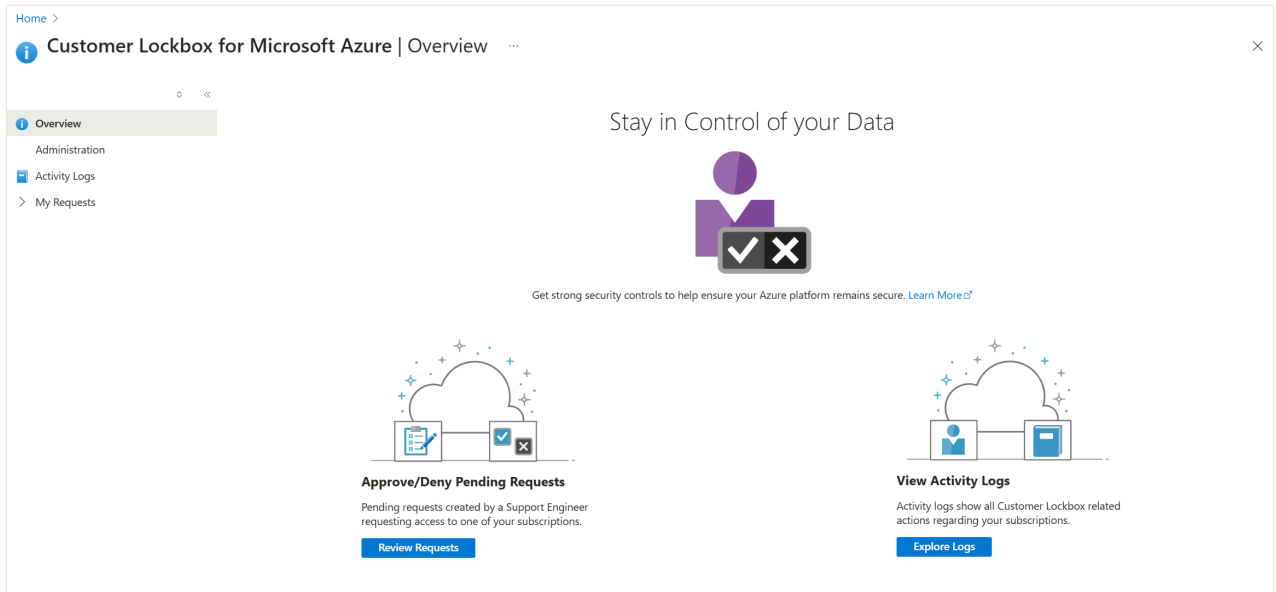
Request Information

Support request #	119041124002216
Requestor	Microsoft Support Engineer
Resource ID	demo.demo
Resource type	Virtual Machine
Justification	Microsoft Support Team is requesting access to your resource temporarily for troubleshooting.
Request start time	April 23, 2019
Duration of access	08:00:00
Request end time	April 27, 2019
Client Request ID	da365854-3d24-425d-9701-b90110279f3f

By approving this request, the Microsoft support organization will be given direct access to your virtual machine for the purpose of troubleshooting and/or resolving the technical issue described in the Microsoft support case.



9. The email notification provides a link to the **Customer Lockbox** blade in the Administration module. The designated approver signs in to the Azure portal to view any pending requests that their organization has for Customer Lockbox for Microsoft Azure:



The request remains in the customer queue for four days. After this time, the access request automatically expires and no access is granted to Microsoft engineers.

10. To get the details of the pending request, the designated approver can select the Customer Lockbox request from **Pending Requests**:

The screenshot shows the 'Customer Lockbox for Microsoft Azure - Pending Requests' page. The left sidebar has a menu with 'Overview', 'Activity Logs', 'My Requests', and 'Pending Requests' (selected). The main content area has a 'Columns' button and a table with the following data:

CREATED TIME	REQUESTOR	SERVICE REQUEST ID	REASON	RESOURCE TYPE	EXPIRES ON	REQUESTED DURATION
4/22/2019, 2:54 PM	Microsoft Support Engineer	119041124002216	Microsoft Support Team is ...	Virtual Machine	4/26/2019, 2:54 PM	08:00:00
4/22/2019, 2:54 PM	Microsoft Support Engineer	119041124002216	Microsoft Support Team is ...	Virtual Machine	4/26/2019, 2:54 PM	08:00:00

A magnifying glass icon is visible on the right side of the image.

11. The designated approver can also select the **SERVICE REQUEST ID** to view the support ticket request that was created by the original user. This information provides context for

why Microsoft Support is engaged, and the history of the reported problem. For example:

 **Support Request: Support Request Created for Testing purposes**

Status
Closed

Severity
C

Title

Support Request Created for Testing purposes

Support request ID

119041124002216

Support plan

Azure Support Plan – Developer

Created on

Thu, Apr 11, 2019, 5:50:27 PM UTC



12. The designated approver reviews the request and selects **Approve** or **Deny**:

Home > Customer Lockbox for Microsoft Azure - Pending Requests

Customer Lockbox for Microsoft Azure - Pending Requests

Overview

Activity Logs

My Requests

Pending Requests

Columns

CREATED TIME	REQUESTOR	SERVICE REQUEST ID	REASON	RESOURCE TYPE
4/22/2019, 2:54 PM	Microsoft Support En...	119041124002216	Microsoft Support Tea...	Virtual Machine
4/22/2019, 2:54 PM	Microsoft Support En...	119041124002216	Microsoft Support Tea...	Virtual Machine

Microsoft Support Engineer
lockboxsupport@microsoft.com

Created Time

4/22/2019, 2:54 PM

Requestor

Microsoft Support Engineer

Service Request Id

119041124002216

Reason

Microsoft Support Team is requesting access to your resource temporarily for troubleshooting.

Resource Type

Virtual Machine

Resource Name

demo.demo

Expires on

4/26/2019, 2:54 PM

Requested Duration

08:00:00

Client Request Id

7663b899-c756-4b75-9515-3a49e208e29c

* Justification

Approve

Deny

As a result of the selection:

- **Approve:** Access is granted to the Microsoft engineer for the duration specified in the request details, which is shown in the email notification and in the Azure portal.
- **Deny:** The elevated access request by the Microsoft engineer is rejected and no further action is taken.

For auditing purposes, the actions taken in this workflow are logged in [Customer Lockbox request logs](#).

Auditing logs

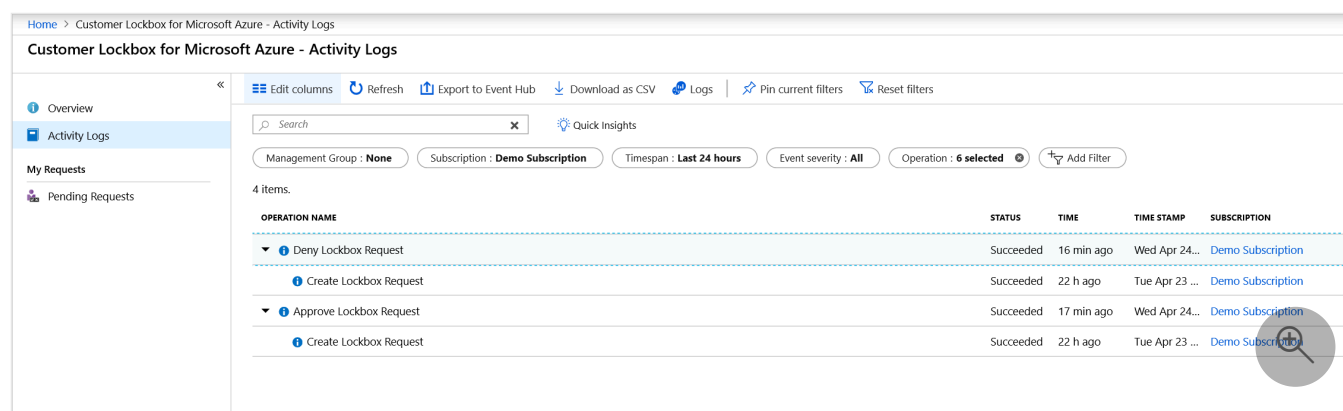
The auditing logs for Customer Lockbox for Azure are written to the activity logs for subscription-scoped requests and to the [Entra Audit Log](#) for tenant-scoped requests.

Subscription-scoped requests - Activity Logs

In the Azure portal, Customer Lockbox for Microsoft Azure blade, select **Activity Logs** to view auditing information related to Customer Lockbox requests. You can also view the **Activity Logs** in the subscription details blade for the subscription in question. In both cases, you can filter for specific operations, such as:

- Deny Lockbox Request
- Create Lockbox Request
- Approve Lockbox Request
- Lockbox Request Expiry

As an example:



OPERATION NAME	STATUS	TIME	TIME STAMP	SUBSCRIPTION
Deny Lockbox Request	Succeeded	16 min ago	Wed Apr 24...	Demo Subscription
Create Lockbox Request	Succeeded	22 h ago	Tue Apr 23 ...	Demo Subscription
Approve Lockbox Request	Succeeded	17 min ago	Wed Apr 24...	Demo Subscription
Create Lockbox Request	Succeeded	22 h ago	Tue Apr 23 ...	Demo Subscription

Tenant-Scoped requests - Audit Log

For tenant-scoped Customer Lockbox requests, log entries are written to the [Entra Audit Log](#). These Log entries are created by the Access Reviews service with activities such as:

- Create request
- Request approved
- Request denied

You can filter for `Service = Access Reviews` and `Activity = one of the above activities`.

As an example:

LockboxRunner | Audit Logs

Download

Export Data Settings

Refresh

Columns

Got feedback?

Manage

Users

Groups

External Identities

Roles and administrators

Administrative units

Delegated admin partners

Enterprise applications

Devices

App registrations

Identity Governance

Application proxy

Custom security attributes

Licenses

Cross-tenant synchronization

Microsoft Entra Connect

Custom domain names

Mobility (MDM and WIP)

Password reset

User settings

Properties

Security

Monitoring

Sign-in logs

Audit logs

Date : Last 24 hours

Show dates as : Local

Service : Access Reviews

Category : All

Activity : All

Date	Service	Category	Activity	Status
7/17/2024, 3:57:21 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:56:55 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:54:43 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:54:42 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:54:14 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:54:14 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:47:17 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:47:17 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:46:50 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:46:50 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:46:47 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:46:47 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:46:20 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:46:18 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:46:18 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:45:52 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:45:49 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:45:23 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:44:00 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:43:37 PM	Access Reviews	UserManagement	Request approved	Success
7/17/2024, 3:41:24 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:41:23 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:40:54 PM	Access Reviews	Policy	Create request	Success
7/17/2024, 3:40:54 PM	Access Reviews	UserManagement	Create business flow	Success
7/17/2024, 3:33:58 PM	Access Reviews	Policy	Create request	Success

Note

The History tab in the Azure Lockbox portal has been removed due to existing technical limitations. To see Customer Lockbox request history, please use the Activity Log for subscription-scoped requests and the [Entra Audit Log](#) for tenant-scoped requests.

Customer Lockbox for Microsoft Azure integration with the Microsoft cloud security benchmark

We introduced a new baseline control ([PA-8: Determine access process for cloud provider support](#)) in the Microsoft cloud security benchmark that covers Customer Lockbox applicability. Customers can now use the benchmark to review Customer Lockbox applicability for a service.

Exclusions

Customer Lockbox requests are not triggered in the following scenarios:

- Emergency scenarios that fall outside of standard operating procedures and require urgent action from Microsoft to restore access to online services or to prevent corruption or loss of customer data, or to investigate a security or abuse incident. For instance, a major service outage or a security incident demands immediate attention to recover or restore services under unexpected or unpredictable circumstances. These "break glass" events are rare and, in most cases, do not necessitate access to customer data for resolution. The controls and processes governing Microsoft's access to customer data in core online services align with NIST 800-53 and are validated through SOC 2 audits. For further information, refer to the [Azure security baseline for Customer Lockbox for Microsoft Azure](#).
- A Microsoft engineer accesses the Azure platform as part of troubleshooting and is inadvertently exposed to customer data. For example, the Azure Network Team performs troubleshooting that results in a packet capture on a network device. It is rare that such scenarios would result in access to meaningful quantities of customer data. Customers can further protect their data through the use of Customer-managed keys (CMK), which is available for some Azure service. For more information see [Overview of Key Management in Azure](#).

External legal demands for data also do not trigger Customer Lockbox requests. For details, see the discussion of [government requests for data](#) on the Microsoft Trust Center.

Next steps

Enable Customer Lockbox from the [Administration module](#) in the Customer Lockbox blade. Customer Lockbox for Microsoft Azure is available for all customers who have an [Azure support plan](#) with a minimal level of Developer.

- [Customer Lockbox for Microsoft Azure alternate email notifications](#)
- [Customer Lockbox for Microsoft Azure FAQ](#)

Customer Lockbox for Microsoft Azure alternate email notifications

05/19/2025

ⓘ Note

To use this feature, your organization must have an [Azure support plan](#) with a minimal level of **Developer**.

Alternate email notification feature enables customers to use alternate email IDs for getting Customer Lockbox notifications. This feature lets Customer Lockbox for Microsoft Azure customers receive notifications when their Azure account isn't email enabled or when a service principal is defined as the tenant admin or subscription owner.

ⓘ Important

This feature only enables Customer Lockbox notifications to be sent to alternate email IDs. It does not enable alternate users to act as approvers for Customer Lockbox requests.

For example, Alice has the subscription owner role for subscription X, and she adds Bob's email address as an alternate email in her user profile. Bob has a reader role. When a Customer Lockbox request is created for a resource scoped to subscription 'X', Bob receives the email notification, but he can't approve or reject the Customer Lockbox request because he doesn't have the required privileges (subscription owner role).

Prerequisites

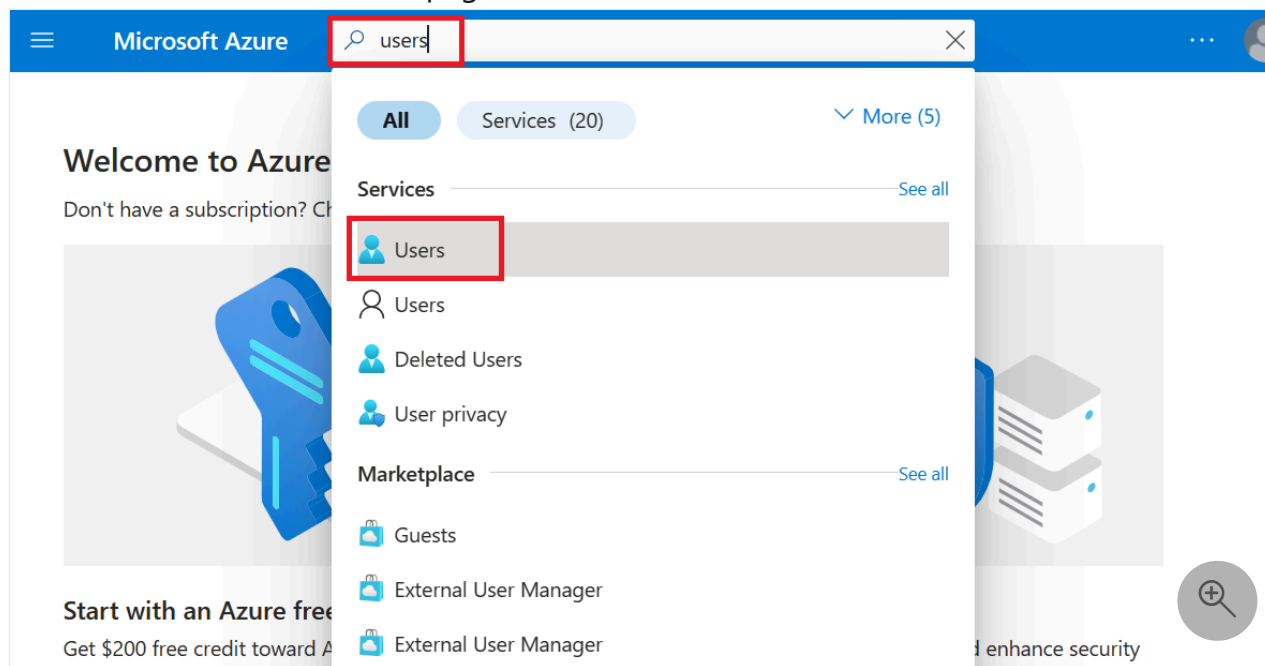
To take advantage of the Customer Lockbox for Microsoft Azure alternate email feature, you must have:

- A Microsoft Entra ID tenant that has Customer Lockbox for Microsoft Azure enabled on it.
- A Developer or above Azure support plan.
- Role assignments:
 - A user account with the Tenant admin, privileged authentication administrator, or User administrator role to update user settings.
 - [Optional] Subscription owner or the new Azure Customer Lockbox Approver for Subscription role if you want to approve or reject Customer Lockbox requests.

Set up customer Lockbox for Microsoft Azure alternate email feature

Here are the steps to set up the Customer Lockbox for Microsoft Azure alternate email feature.

1. Access the [Azure portal](#).
2. Sign in with the user account with tenant/privileged authentication administrator/User administrator role privileges.
3. Search for Users at the home page:



4. Search for the user to add an alternate email address.

ⓘ Note

The user must have tenant admin/subscription owner/Azure Customer Lockbox Approver for Subscription role privileges to act on Lockbox requests.

Home >

Users

MSB-test-org - Microsoft Entra ID

Search << + New user ↓ ↓ Download users Bulk operations ↓ Refresh Manage v

All users Azure Active Directory is now Microsoft Entra ID. [\[i\]](#)

Audit logs

Sign-in logs

Diagnose and solve problems

Manage

Deleted users

testuser X Add filter

1 user found

☐	Display name ↑↓	User principal name ↑↓	User type
☐	 Test User	testuser@contoso.com 	Member 

5. Select the user, then select **Edit properties**.

Home > Users >

Test User

User

Search << **Edit properties** Delete Refresh Reset password Revoke sessions ...

Overview

Audit logs

Sign-in logs

Diagnose and solve problems

Manage

Custom security attributes

Assigned roles

Administrative units

Groups

Applications

Licenses

Devices

Azure role assignments

Authentication methods

Overview Monitoring Properties

Basic info

 Test User
testuser@contoso.com
Member

User principal name	testuser@contoso.com 
Object ID	aaaaaaaa-0000-1111-2222-bbbbbbbbbbbb 
Created date time	Mar 19, 2024, 12:33 PM
User type	Member
Identities	contoso.com
Group memberships	0

6. Navigate to Contact Information tab.

Home > Users > Test User >

Test User

...

Properties

Refresh

Got feedback?

All

Identity

Job Information

Contact Information

Parental controls

Settings

On-premises

This view only contains properties that can be updated.

Learn more

Search "Contact Informati..."

Showing 11 results under "Contact Information"

Street address

City

State or province

ZIP or postal code

Country or region

Business phone

Mobile phone

Email

testuser@contoso.com

Other emails

+ Add email

Fax number

Mail nickname

testuser

Save

Cancel

7. Select **Add email** under the 'Other emails' category, then select **Add**.

Home > Users > Test User >

Test User

Properties

Refresh

Got feedback?

All

Identity

Job Information

Contact Information

This view only contains properties that can be updated. [Learn more](#)

Search "Contact Information"

Showing 11 results under "Contact Information"

Street address

City

State or province

ZIP or postal code

Country or region

Business phone

Mobile phone

Email

Other emails

Fax number

testuser@contoso.

+ Add email

Edit Other emails

+ Add

8. Enter the alternate email address in the text field, then select **Save**.

Micr...

Search resources, services, and docs (G+/)

2

Home > Users > Test User >

Test User

...

Properties

Refresh

Got feedback?

All

Identity

Job Information

Contact Information

This view only contains properties that can be updated. [Learn more](#)

Search "Contact Information"

Showing 11 results under "Contact Information"

Street address

City

State or province

ZIP or postal code

Country or region

Business phone

Mobile phone

Email

testuser@contoso.com

Other emails

+ Add email

Fax number

Mail nickname

testuser

Save

Cancel

Save

Cancel

Edit Other emails

testuser2@contoso.com

+ Add

9. Select the "Save" button in the Contact Information tab to save the updates.

Home > Users > Test User >

Test User

Properties

Refresh

Got feedback?

All

Identity

Job Information

Contact Information

Parental controls

Settings

On-p

This view only contains properties that can be updated. [Learn more](#)

Search "Contact Informati...

Showing 11 results under "Contact Information"

Street address

City

State or province

ZIP or postal code

Country or region

Business phone

Mobile phone

Email

testuser@contoso.com

Other emails

testuser2@contoso.com

+ Add email

Fax number

Mail nickname

testuser

Save

Cancel

10. The **Contact information** tab now shows the updated information with the alternate email:

Home > Users > Test User >

Test User

Properties

 Refresh |  Got feedback?

All Identity Job Information **Contact Information** Parental controls Settings On-pr

This view only contains properties that can be updated. [Learn more](#) 

 Search "Contact Informati..."

Showing 11 results under "Contact Information"

Street address	
City	
State or province	
ZIP or postal code	
Country or region	
Business phone	
Mobile phone	
Email	testuser@contoso.com
Other emails	testuser2@contoso.com  Add email
Fax number	
Mail nickname	testuser



11. If the primary 'Email' field has a value, emails are sent only to that address. To send Lockbox email notifications to 'Other emails', clear the primary 'Email' field.

Home > Users > Test User >

Test User

Properties

 Refresh |  Got feedback?

All Identity Job Information **Contact Information** Parental controls Settings On-p...

This view only contains properties that can be updated. [Learn more](#)

 Search "Contact Informati..."

Showing 11 results under "Contact Information"

Street address	
City	
State or province	
ZIP or postal code	
Country or region	
Business phone	
Mobile phone	
Email	
Other emails	testuser2@contoso.com  Add email
Fax number	
Mail nickname	testuser



12. When a Lockbox request is triggered and the user is identified as a Lockbox approver, the email notification is sent to the primary email if it has a value. If the primary email is empty, the notification is sent to other email addresses. These emails notify the approver that Microsoft Support is trying to access a resource in their tenant, and they need to sign in to the Azure portal to approve or reject the request. Here is an example screenshot:

Customer Lockbox for Microsoft Azure: Please take action

A Customer Lockbox request is pending action from you.

If you have an alternate email address on file, this email has been sent to both your primary email address and your alternate email address. To see the Lockbox approval request, please be sure you sign in to the correct production account.

Required action

To view and act on this Customer Lockbox request, sign in to the Azure portal and go to the **Customer Lockbox for Microsoft Azure** service blade.

[Sign in >](#)

Request Information

Support request number	1
Requester	Microsoft Support Engineer
Resource ID	subscriptions/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx/resourcegroups/demo/providers/Microsoft.Compute/VIRTUALMACHINES/DEMO
Resource type	Virtual Machine
Justification	Microsoft Support Team is requesting access to your resource temporarily for troubleshooting.
Request start time	March 11, 2024
Duration of access	04:00:00
Request end time	March 15, 2024
Client Request ID	ba61065c-badf-426f-99bd-375b768a10da



Known Issues

These are the known issues with this feature:

- Duplicate emails are sent if the value for the primary email and other email is the same.

- Notifications are sent to only the first email address in 'other emails' despite multiple email IDs configured in other email field.
- If the primary email is not set, and the other email is set, two emails are sent to the alternate email address.

Next steps

- [Customer Lockbox for Microsoft Azure](#)
- [Customer Lockbox for Microsoft Azure frequently asked questions](#)

Customer Lockbox for Microsoft Azure frequently asked questions

This article answers frequently asked questions about Customer Lockbox for Microsoft Azure.

General

Can I enable Customer Lockbox for Microsoft Azure at management group or subscription level?

No, Customer Lockbox for Microsoft Azure can only be enabled at tenant-level, and is applicable to all the subscriptions and resources under that tenant.

What does Microsoft do when a customer rejects a Customer Lockbox request?

If a customer rejects a Customer Lockbox request, no access to customer content occurs. If a user in your organization continues to experience a service issue requiring Microsoft to access customer content to resolve the issue, then the service issue might persist and Microsoft will inform the user.

Can I assign the Customer Lockbox approver role at the management group level?

No, role assignments scoped to management groups are not supported in Customer Lockbox for Microsoft Azure at this time.

Can I use Privileged Identity Management (PIM) to activate the Customer Lockbox approver role after a Customer Lockbox request is initiated?

Role assignments must be in place before Customer Lockbox for Microsoft Azure starts to process a request. Any role assignments made after Customer Lockbox for Microsoft Azure starts to process a given request will not be recognized. Using PIM eligible assignments for the Customer Lockbox approver role requires users to activate the role before the Customer Lockbox request is initiated.

Customer Lockbox Approver Role for Subscriptions

Can I use the new Customer Lockbox approver role for tenant-scoped requests as well?

No, Azure Customer Lockbox Approver for Subscription role works only for subscription-scoped requests. The Customer Lockbox for Microsoft Azure team will be creating a lesser privilege role for tenant-scoped requests in subsequent releases.

Can I use the new Customer Lockbox approver role with Microsoft Purview Customer Lockbox or Customer Lockbox for Power Platform and Dynamics 365?

No, the Azure Customer Lockbox Approver for Subscription role works only for subscription-scoped requests created by Customer Lockbox for Microsoft Azure.

Can I use PIM to activate the new Customer Lockbox approver role after a Customer Lockbox request is initiated?

Role assignments must be in place before Customer Lockbox starts to process a request. Any role assignments made after Customer Lockbox for Microsoft Azure starts to process a given request will not be recognized. Because of this, to use PIM eligible assignments for the Customer Lockbox approver role, users are required to activate the role before the Customer Lockbox request is initiated.

Alternative email feature

Can I add a different user email address as an alternate email to another user's account?

Yes, you can add any email address in the other emails field to be used as alternate email for receiving Customer Lockbox notifications.

If I add a second user's email address as an alternate email to an existing Customer Lockbox approver user's account, will the second user be able to see and approve/reject Customer Lockbox requests?

No, this feature only allows customers to receive Customer Lockbox request notifications on alternate email addresses, but it does not provide the ability to configure other users as Customer Lockbox approvers. For example, Alice has the subscription owner role for subscription X and she adds Bob's email address as alternate email/other email in her user profile who has a reader role. When a Customer Lockbox request is created for a resource scoped to subscription "X", Bob receives the email notification, but he'll not be able to approve/reject the Customer Lockbox request as he does not have the required privileges for it (subscription owner role).

Can I add more than one alternate email address to a user account?

You can add multiple email addresses in the other field but currently Customer Lockbox for Microsoft Azure supports sending notifications only to the first email address in "other emails" despite multiple email IDs configured.

Can I use alternate email notification functionality with Microsoft Purview Customer Lockbox or Customer Lockbox for Power Platform and Dynamics 365?

No, this feature is limited to Customer Lockbox for Microsoft Azure.

Will the alternate email notification work for both tenant-scoped and subscription-scoped Customer Lockbox requests?

Yes, alternate email notifications work for all Customer Lockbox requests.

Next steps

- [Customer Lockbox for Microsoft Azure overview](#)
- [Customer Lockbox for Microsoft Azure alternate email notifications](#)

Azure security baseline for Customer Lockbox for Microsoft Azure

Article • 02/25/2025

This security baseline applies guidance from the [Microsoft cloud security benchmark version 1.0](#) to Customer Lockbox for Microsoft Azure. The Microsoft cloud security benchmark provides recommendations on how you can secure your cloud solutions on Azure. The content is grouped by the security controls defined by the Microsoft cloud security benchmark and the related guidance applicable to Customer Lockbox for Microsoft Azure.

You can monitor this security baseline and its recommendations using Microsoft Defender for Cloud. Azure Policy definitions will be listed in the Regulatory Compliance section of the Microsoft Defender for Cloud portal page.

When a feature has relevant Azure Policy Definitions, they are listed in this baseline to help you measure compliance with the Microsoft cloud security benchmark controls and recommendations. Some recommendations may require a paid Microsoft Defender plan to enable certain security scenarios.

 Note

Features not applicable to Customer Lockbox for Microsoft Azure have been excluded. To see how Customer Lockbox for Microsoft Azure completely maps to the Microsoft cloud security benchmark, see the [full Customer Lockbox for Microsoft Azure security baseline mapping file](#).

Security profile

The security profile summarizes high-impact behaviors of Customer Lockbox for Microsoft Azure, which may result in increased security considerations.

 Expand table

Service Behavior Attribute	Value
Product Category	Security
Customer can access HOST / OS	No Access
Service can be deployed into customer's virtual network	False
Stores customer content at rest	False

Network security

For more information, see the [Microsoft cloud security benchmark: Network security](#).

NS-1: Establish network segmentation boundaries

Features

Virtual Network Integration

Description: Service supports deployment into customer's private Virtual Network (VNet).
[Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Network Security Group Support

Description: Service network traffic respects Network Security Groups rule assignment on its subnets. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

NS-2: Secure cloud services with network controls

Features

Azure Private Link

Description: Service native IP filtering capability for filtering network traffic (not to be confused with NSG or Azure Firewall). [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Disable Public Network Access

Description: Service supports disabling public network access either through using service-level IP ACL filtering rule (not NSG or Azure Firewall) or using a 'Disable Public Network Access' toggle switch. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Identity management

For more information, see the [Microsoft cloud security benchmark: Identity management](#).

IM-1: Use centralized identity and authentication system

Features

Azure AD Authentication Required for Data Plane Access

Description: Service supports using Azure AD authentication for data plane access. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
True	True	Microsoft

Configuration Guidance: No additional configurations are required as this is enabled on a default deployment.

Local Authentication Methods for Data Plane Access

Description: Local authentications methods supported for data plane access, such as a local username and password. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

IM-3: Manage application identities securely and automatically

Features

Managed Identities

Description: Data plane actions support authentication using managed identities. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Service Principals

Description: Data plane supports authentication using service principals. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

IM-7: Restrict resource access based on conditions

Features

Conditional Access for Data Plane

Description: Data plane access can be controlled using Azure AD Conditional Access Policies.
[Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

IM-8: Restrict the exposure of credential and secrets

Features

Service Credential and Secrets Support Integration and Storage in Azure Key Vault

Description: Data plane supports native use of Azure Key Vault for credential and secrets store.
[Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Privileged access

For more information, see the [Microsoft cloud security benchmark: Privileged access](#).

PA-1: Separate and limit highly privileged/administrative users

Features

Local Admin Accounts

Description: Service has the concept of a local administrative account. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

PA-7: Follow just enough administration (least privilege) principle

Features

Azure RBAC for Data Plane

Description: Azure Role-Based Access Control (Azure RBAC) can be used to managed access to service's data plane actions. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

PA-8: Determine access process for cloud provider support

Features

Customer Lockbox

Description: Customer Lockbox can be used for Microsoft support access. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
True	False	Customer

Feature notes: This is the Customer Lockbox service.

Configuration Guidance: In support scenarios where Microsoft needs to access your data, use Customer Lockbox to review, then approve or reject each of Microsoft's data access requests.

Reference: [Customer Lockbox for Microsoft Azure](#)

Data protection

For more information, see the [Microsoft cloud security benchmark: Data protection](#).

DP-1: Discover, classify, and label sensitive data

Features

Sensitive Data Discovery and Classification

Description: Tools (such as Azure Purview or Azure Information Protection) can be used for data discovery and classification in the service. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

DP-2: Monitor anomalies and threats targeting sensitive data

Features

Data Leakage/Loss Prevention

Description: Service supports DLP solution to monitor sensitive data movement (in customer's content). [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

DP-3: Encrypt sensitive data in transit

Features

Data in Transit Encryption

Description: Service supports data in-transit encryption for data plane. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

DP-6: Use a secure key management process

Features

Key Management in Azure Key Vault

Description: The service supports Azure Key Vault integration for any customer keys, secrets, or certificates. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

DP-7: Use a secure certificate management process

Features

Certificate Management in Azure Key Vault

Description: The service supports Azure Key Vault integration for any customer certificates. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

Asset management

For more information, see the [Microsoft cloud security benchmark: Asset management](#).

AM-2: Use only approved services

Features

Azure Policy Support

Description: Service configurations can be monitored and enforced via Azure Policy. [Learn more.](#)

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Feature notes: Private preview that requires manual onbaording

Configuration Guidance: This feature is not supported to secure this service.

Logging and threat detection

For more information, see the [Microsoft cloud security benchmark: Logging and threat detection](#).

LT-1: Enable threat detection capabilities

Features

Microsoft Defender for Service / Product Offering

Description: Service has an offering-specific Microsoft Defender solution to monitor and alert on security issues. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Configuration Guidance: This feature is not supported to secure this service.

LT-4: Enable logging for security investigation

Features

Azure Resource Logs

Description: Service produces resource logs that can provide enhanced service-specific metrics and logging. The customer can configure these resource logs and send them to their own data sink like a storage account or log analytics workspace. [Learn more](#).

 Expand table

Supported	Enabled By Default	Configuration Responsibility
False	Not Applicable	Not Applicable

Feature notes: Though Customer Lockbox does not support this feature, the customer does have access to the activity logs for the service.

For more information, please visit: [Auditing Logs](#)

Configuration Guidance: This feature is not supported to secure this service.

Next steps

- See the [Microsoft cloud security benchmark overview](#)
- Learn more about [Azure security baselines](#)

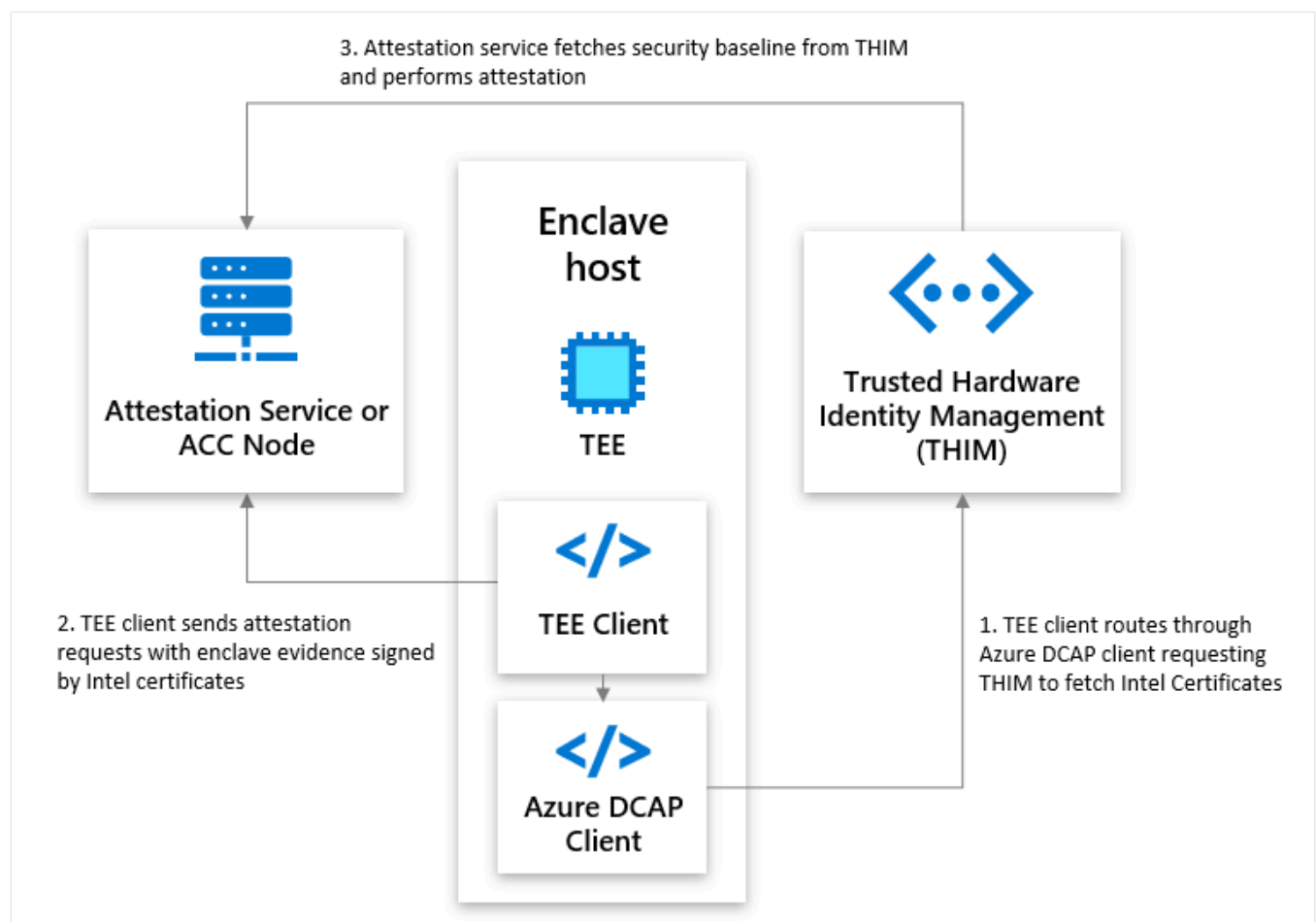
Trusted Hardware Identity Management

Article • 04/16/2025

The Trusted Hardware Identity Management service handles cache management of certificates for all trusted execution environments (TEEs) that reside in Azure. It also provides trusted computing base (TCB) information to enforce a minimum baseline for attestation solutions.

Trusted Hardware Identity Management and attestation interactions

Trusted Hardware Identity Management defines the Azure security baseline for Azure confidential computing (ACC) nodes and caches collateral from TEE providers. Attestation services and ACC nodes can use the cached information to validate TEEs. The following diagram shows the interactions between an attestation service or node, Trusted Hardware Identity Management, and an enclave host.



Frequently asked questions

How do I use Trusted Hardware Identity Management with Intel processors?

To generate Intel SGX and Intel TDX quotes, the Intel Quote Generation Library (QGL) needs access to quote generation/validation collateral. All or parts of this collateral must be fetched from Trusted Hardware Identity Management. You can fetch it by using the [Intel Quote Provider Library \(QPL\)](#) or the [Azure Data Center Attestation Primitives \(DCAP\) client library](#).

The "next update" date of the Azure-internal caching service API that Azure Attestation uses seems to be out of date. Is it still in operation and can I use it?

The `tcbinfo` field contains the TCB information. The Trusted Hardware Identity Management service provides older `tcbinfo` information by default. Updating to the latest `tcbinfo` information from Intel would cause attestation failures for customers who haven't migrated to the latest Intel SDK, and it could result in outages.

The Open Enclave SDK and Azure Attestation don't look at the `nextUpdate` date, however, and will pass attestation.

What is the Azure DCAP library?

The Azure Data Center Attestation Primitives (DCAP) library, a replacement for Intel Quote Provider Library (QPL), fetches quote generation collateral and quote validation collateral directly from the Trusted Hardware Identity Management service. Fetching collateral directly from the Trusted Hardware Identity Management service ensures that all Azure hosts have collateral readily available within the Azure cloud to reduce external dependencies. It is recommended to use the latest version available for your target OS.

Where can I download the latest Azure DCAP library?

Use the following links to download the packages:

- [Ubuntu 22.04](#) ↗
- [Ubuntu 20.04](#) ↗
- [Ubuntu 18.04](#) ↗
- [Windows](#) ↗

For newer versions of Ubuntu (for example, Ubuntu 24.04), you have to use the [Intel QPL](#).

Why do Trusted Hardware Identity Management and Intel have different baselines?

Trusted Hardware Identity Management and Intel provide different baseline levels of the trusted computing base. When customers assume that Intel has the latest baselines, they must ensure that all the requirements are satisfied. This approach can lead to a breakage if customers haven't updated to the specified requirements.

Trusted Hardware Identity Management takes a slower approach to updating the TCB baseline, so customers can make the necessary changes at their own pace. Although this approach provides an older TCB baseline, customers won't experience a breakage if they haven't met the requirements of the new TCB baseline. This is why the TCB baseline from Trusted Hardware Identity Management is a different version from Intel's baseline. We want to empower customers to meet the requirements of the new TCB baseline at their pace, instead of forcing them to update and causing a disruption that would require reprioritization of workstreams.

With Intel Xeon E Processors, I could get my certificates directly from the Intel PCS. Why, with Intel Xeon Scalable processors starting from the 4th generation, do I need to get the certificates from Trusted Hardware Identity Management? And how can I fetch those certificates?

Starting with the 4th Generation of Intel® Xeon® Scalable Processors, Azure performs indirect registration at Intel's Registration Service using the Platform Manifest and stores the resulting PCK certificate in the Trusted Hardware Identity Management (THIM) service Azure uses indirect registration, because Intel's registration service will not store root keys for a platform in this case and this is reflected by `false` in the `CachedKeys` flag in PCK Certificates. As indirect registration is used, all following communication to Intel PCS would require the Platform Manifest, which Azure does not provide to virtual machines (VMs). Instead, VMs have to reach out to THIM to receive PCK certificates. To retrieve a PCK certificate, you can either use the [Intel QPL](#) or the [Azure DCAP library](#).

How do I use Intel QPL with Trusted Hardware Identity Management?

Customers might want the flexibility to use Intel QPL to interact with Trusted Hardware Identity Management without having to download another dependency from Microsoft (that is, the Azure DCAP client library). Customers who want to use Intel QPL with the Trusted Hardware Identity Management service must adjust the Intel QPL configuration file, `sgx_default_qcnl.conf`.

The quote generation/verification collateral that's used to generate the Intel SGX or Intel TDX quotes can be split into:

- The PCK certificate. To retrieve it, customers must use a Trusted Hardware Identity Management endpoint.
- All other quote generation/verification collateral. To retrieve it, customers can either use a Trusted Hardware Identity Management endpoint or an Intel Provisioning Certification Service (PCS) endpoint.

The Intel QPL configuration file (*sgx_default_qcnl.conf*) contains three keys for defining the collateral endpoints. The `pccs_url` key defines the endpoint that's used to retrieve the PCK certificates. The `collateral_service` key can define the endpoint that's used to retrieve all other quote generation/verification collateral. If the `collateral_service` key is not defined, all quote verification collateral is retrieved from the endpoint defined with the `pccs_url` key.

The following table shows how these keys can be set.

 Expand table

Name	Possible endpoints
<code>pccs_url</code>	Trusted Hardware Identity Management endpoint: <code>https://global.accache.azure.net/sgx/certification/v3</code> .
<code>collateral_service</code>	Trusted Hardware Identity Management endpoint (<code>https://global.accache.azure.net/sgx/certification/v3</code>) or Intel PCS endpoint. The sgx_default_qcnl.conf file always lists the most up-to-date endpoint in the <code>collateral_service</code> key.

The following code snippet is from an example of an Intel QPL configuration file:

Bash

```
{
  "pccs_url": "https://global.accache.azure.net/sgx/certification/v3/",
  "use_secure_cert": true,
  "collateral_service":
"https://global.accache.azure.net/sgx/certification/v3/",
  "pccs_api_version": "3.1",
  "retry_times": 6,
  "retry_delay": 5,
  "local_pck_url":
"http://169.254.169.254/metadata/THIM/sgx/certification/v3/",
  "pck_cache_expire_hours": 24,
  "verify_collateral_cache_expire_hours": 24,
  "custom_request_options": {
    "get_cert": {
      "headers": {
```

```
        "metadata": "true"
      },
      "params": {
        "api-version": "2021-07-22-preview"
      }
    }
  }
}
```

The following procedures explain how to change the Intel QPL configuration file and activate the changes.

On Windows

1. Make changes to the configuration file.
2. Ensure that there are read permissions to the file from the following registry location and key/value:

Bash

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Intel\SGX\QCNL]
"CONFIG_FILE"="<Full File Path>"
```

3. Restart the AESMD service. For instance, open PowerShell as an administrator and use the following commands:

Bash

```
Restart-Service -Name "AESMService" -ErrorAction Stop
Get-Service -Name "AESMService"
```

On Linux

1. Make changes to the configuration file. For example, you can use Vim for the changes via the following command:

Bash

```
sudo vim /etc/sgx_default_qcnl.conf
```

2. Restart the AESMD service. Open any terminal and run the following commands:

Bash


```
sudo systemctl restart aesmd
systemctl status aesmd
```

How do I request collateral in a confidential virtual machine?

Use the following sample in a confidential virtual machine (CVM) guest for requesting AMD collateral that includes the VCEK certificate and certificate chain. For details on this collateral and where it comes from, see [Versioned Chip Endorsement Key \(VCEK\) Certificate and KDS Interface Specification](#).

URI parameters

```
Bash

GET "http://169.254.169.254/metadata/THIM/amd/certification"
```

Request body

 Expand table

Name	Type	Description
Metadata	Boolean	Setting to <code>True</code> allows for collateral to be returned.

Sample request

```
Bash

curl GET "http://169.254.169.254/metadata/THIM/amd/certification" -H "Metadata: true"
```

Responses

 Expand table

Name	Description
200 OK	Lists available collateral in the HTTP body within JSON format
Other Status Codes	Describes why the operation failed

Key	Description
VcekCert	X.509v3 certificate as defined in RFC 5280
tcbm	Trusted computing base
certificateChain	AMD SEV Key (ASK) and AMD Root Key (ARK) certificates

How do I request AMD collateral in an Azure Kubernetes Service Container on a CVM node?

Follow these steps to request AMD collateral in a confidential container:

1. Start by creating an Azure Kubernetes Service (AKS) cluster on a CVM node or by adding a CVM node pool to an existing cluster:
 - Create an AKS cluster on a CVM node:
 - a. Create a resource group in one of the CVM supported regions:

Bash

```
az group create --resource-group <RG_NAME> --location <LOCATION>
```

- b. Create an AKS cluster with one CVM node in the resource group:

Bash

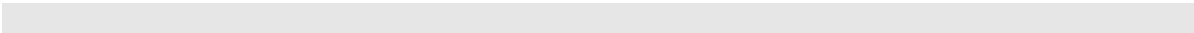
```
az aks create --name <CLUSTER_NAME> --resource-group <RG_NAME> -l <LOCATION> --node-vm-size Standard_DC4as_v5 --nodepool-name <POOL_NAME> --node-count 1
```

- c. Configure kubectl to connect to the cluster:

Bash

```
az aks get-credentials --resource-group <RG_NAME> --name <CLUSTER_NAME>
```

- Add a CVM node pool to an existing AKS cluster:



Bash

```
az aks nodepool add --cluster-name <CLUSTER_NAME> --resource-group  
<RG_NAME> --name <POOL_NAME> --node-vm-size Standard_DC4as_v5 --node-  
count 1
```

2. Verify the connection to your cluster by using the `kubectl get` command. This command returns a list of the cluster nodes.

Bash

```
kubectl get nodes
```

The following output example shows the single node that you created in the previous steps. Make sure that the node status is `Ready`.

[Expand table](#)

NAME	STATUS	ROLES	AGE	VERSION
aks-nodepool1-31718369-0	Ready	agent	6m44s	v1.12.8

3. Create a `curl.yaml` file with the following content. It defines a job that runs a curl container to fetch AMD collateral from the Trusted Hardware Identity Management endpoint. For more information about Kubernetes Jobs, see the [Kubernetes documentation](#).

Bash

```
apiVersion: batch/v1  
kind: Job  
metadata:  
  name: curl  
spec:  
  template:  
    metadata:  
      labels:  
        app: curl  
    spec:  
      nodeSelector:  
        kubernetes.azure.com/security-type: ConfidentialVM  
      containers:  
        - name: curlcontainer  
          image: alpine/curl:3.14  
          imagePullPolicy: IfNotPresent  
          args: ["-H", "Metadata:true",
```

```
"http://169.254.169.254/metadata/THIM/amd/certification"]  
restartPolicy: "Never"
```

The *curl.yaml* file contains the following arguments.

 Expand table

Name	Type	Description
Metadata	Boolean	Setting to <code>True</code> allows for collateral to be returned.

4. Run the job by applying the *curl.yaml* file:

```
Bash
```

```
kubectl apply -f curl.yaml
```

5. Check and wait for the pod to complete its job:

```
Bash
```

```
kubectl get pods
```

Here's an example response:

 Expand table

Name	Ready	Status	Restarts	Age
Curl-w7nt8	0/1	Completed	0	72 s

6. Run the following command to get the job logs and validate if it's working. A successful output should include `vcekCert`, `tcbm`, and `certificateChain`.

```
Bash
```

```
kubectl logs job/curl
```

Next steps

- Learn more about [Azure Attestation documentation](#).
- Learn more about [Azure confidential computing](#) .

Securing PaaS deployments

This article provides information that helps you:

- Understand the security advantages of hosting applications in the cloud
- Evaluate the security advantages of platform as a service (PaaS) versus other cloud service models
- Change your security focus from a network-centric to an identity-centric perimeter security approach
- Implement general PaaS security best practices recommendations

[Develop secure applications on Azure](#) is a general guide to the security questions and controls you should consider at each phase of the software development lifecycle when developing applications for the cloud.

Cloud security advantages

It's important to understand the [division of responsibility](#) between you and Microsoft. On-premises, you own the whole stack, but as you move to the cloud some responsibilities transfer to Microsoft.

There are security advantages to being in the cloud. In an on-premises environment, organizations likely have unmet responsibilities and limited resources available to invest in security, which creates an environment where attackers can exploit vulnerabilities at all layers.

Organizations can improve their threat detection and response times by using a provider's cloud-based security capabilities and cloud intelligence. By shifting responsibilities to the cloud provider, organizations can get more security coverage, which enables them to reallocate security resources and budget to other business priorities.

Security advantages of a PaaS cloud service model

Let's look at the security advantages of an Azure PaaS deployment versus on-premises.

 Security advantages of PaaS

Starting at the bottom of the stack, the physical infrastructure, Microsoft mitigates common risks and responsibilities. Because the Microsoft cloud is continually monitored by Microsoft, it is hard to attack. It doesn't make sense for an attacker to pursue the Microsoft cloud as a target. Unless the attacker has lots of money and resources, the attacker is likely to move on to another target.

In the middle of the stack, there is no difference between a PaaS deployment and on-premises. At the application layer and the account and access management layer, you have similar risks. In the next steps section of this article, we guide you to best practices for eliminating or minimizing these risks.

At the top of the stack, data governance and rights management, you take on one risk that can be mitigated by key management. While key management is an additional responsibility, you have areas in a PaaS deployment that you no longer have to manage so you can shift resources to key management.

The Azure platform also provides strong DDoS protection by using various network-based technologies. However, all types of network-based DDoS protection methods have their limits on a per-link and per-datacenter basis. To help avoid the impact of large DDoS attacks, you can take advantage of Azure's core cloud capability of enabling you to quickly and automatically scale out to defend against DDoS attacks.

Identity as the primary security perimeter

With PaaS deployments comes a shift in your overall approach to security. You shift from needing to control everything yourself to sharing responsibility with Microsoft.

Another significant difference between PaaS and traditional on-premises deployments is a new view of what defines the primary security perimeter. Historically, the primary on-premises security perimeter was your network and most on-premises security designs use the network as its primary security pivot. For PaaS deployments, you are better served by considering identity to be the primary security perimeter.

One of the five essential characteristics of cloud computing is broad network access, which makes network-centric thinking less relevant. The goal of much of cloud computing is to allow users to access resources regardless of location. For most users, their location is going to be somewhere on the Internet.

The following figure shows how the security perimeter has evolved from a network perimeter to an identity perimeter. Security becomes less about defending your network and more about defending your data, as well as managing the security of your apps and users. The key difference is that you want to push security closer to what's important to your company.

 Identity as new security perimeter

Initially, Azure PaaS services (for example, Azure App Service and Azure SQL) provided little or no traditional network perimeter defenses. It was understood that the element's purpose was to be exposed to the Internet (web role) and that authentication provides the new perimeter (for example, Azure SQL).

Modern security practices assume that the adversary has breached the network perimeter. Therefore, modern defense practices have moved to identity. Organizations must establish an identity-based security perimeter with strong authentication and authorization hygiene.

Best practices for identity management

The following are best practices for managing the identity perimeter.

Best practice: You should first consider using managed identities for Azure resources to securely access other services without storing credentials. **Detail:** [Managed identities](#) automatically provide an identity for applications running in Azure services, enabling them to authenticate to services that support Microsoft Entra ID without requiring credentials in code or configuration files. This reduces the risk of credential exposure and simplifies identity management for your applications.

Best practice: Secure your keys and credentials to secure your PaaS deployment. **Detail:** Losing keys and credentials is a common problem. You can use a centralized solution where keys and secrets can be stored in hardware security modules (HSMs). [Azure Key Vault](#) safeguards your keys and secrets by encrypting authentication keys, storage account keys, data encryption keys, .pfx files, and passwords using keys that are protected by HSMs.

Best practice: Don't put credentials and other secrets in source code or GitHub. **Detail:** The only thing worse than losing your keys and credentials is having an unauthorized party gain access to them. Attackers can take advantage of bot technologies to find keys and secrets stored in code repositories such as GitHub. Do not put keys and secrets in these public code repositories.

Best practice: Use strong authentication and authorization platforms. **Detail:** Use [Microsoft Entra ID](#) for authentication instead of custom user stores. When you use Microsoft Entra ID, you take advantage of a platform-based approach and delegate the management of authorized identities. A Microsoft Entra ID approach is especially important when employees are terminated and that information needs to be reflected through multiple identity and authorization systems.

Use platform-supplied authentication and authorization mechanisms instead of custom code. The reason is that developing custom authentication code can be error prone. Most of your developers are not security experts and are unlikely to be aware of the subtleties and the latest developments in authentication and authorization. Commercial code (for example, from Microsoft) is often extensively security reviewed.

Use [multifactor authentication \(MFA\)](#), and ensure that phishing-resistant MFA methods — such as [Passkeys](#), [FIDO2](#), or [Certificate-Based Authentication \(CBA\)](#) — are enforced through

[Conditional Access policies](#). At a minimum, require these for all administrators, and for optimal security, implement them tenant-wide. Access to both Azure management (portal/remote PowerShell) interfaces and customer-facing services should be designed and configured to use Microsoft Entra multifactor authentication.

For app sign-in, use [OpenID Connect \(OIDC\)](#) with [OAuth 2.0](#) via Microsoft Entra ID. These protocols have been extensively peer reviewed and are likely implemented as part of your platform libraries for authentication and authorization.

Use threat modeling during application design

The Microsoft [Security Development Lifecycle](#) specifies that teams should engage in a process called threat modeling during the design phase. To help facilitate this process, Microsoft has created the [SDL Threat Modeling Tool](#). Modeling the application design and enumerating [STRIDE](#) threats across all trust boundaries can catch design errors early on.

The following table lists the STRIDE threats and gives some example mitigations that use Azure features. These mitigations won't work in every situation.

 Expand table

Threat	Security property	Potential Azure platform mitigations
Spoofing	Authentication	Require HTTPS connections.
Tampering	Integrity	Validate TLS/SSL certificates.
Repudiation	Non-repudiation	Enable Azure monitoring and diagnostics .
Information disclosure	Confidentiality	Encrypt sensitive data at rest by using service certificates .
Denial of service	Availability	Monitor performance metrics for potential denial-of-service conditions. Implement connection filters.
Elevation of privilege	Authorization	Use Privileged Identity Management .

Azure App Service

[Azure App Service](#) is a PaaS offering that lets you create web and mobile apps for any platform or device and connect to data anywhere, in the cloud or on-premises. App Service includes the web and mobile capabilities that were previously delivered separately as Azure Websites and

Azure Mobile Services. It also includes new capabilities for automating business processes and hosting cloud APIs.

Following are best practices for using App Service.

Best practice: [Authenticate through Microsoft Entra ID](#). **Detail:** App Service provides an OAuth 2.0 service for your identity provider. OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications, and mobile phones. Microsoft Entra ID uses OAuth 2.0 to enable you to authorize access to mobile and web applications.

Best practice: Restrict access based on the need to know and least privilege security principles. **Detail:** Restricting access is imperative for organizations that want to enforce security policies for data access. You can use [Azure RBAC](#) to assign permissions to users, groups, and applications at a certain scope.

Best practice: Protect your keys. **Detail:** Azure Key Vault helps safeguard cryptographic keys and secrets that cloud applications and services use. With Key Vault, you can encrypt keys and secrets (such as authentication keys, storage account keys, data encryption keys, .PFX files, and passwords) by using keys that are protected by hardware security modules (HSMs). For added assurance, you can import or generate keys in HSMs. See [Azure Key Vault](#) to learn more. You can also use Key Vault to manage your TLS certificates with auto-renewal.

Best practice: Restrict incoming source IP addresses. **Detail:** [App Service Environment](#) has a virtual network integration feature that helps you restrict incoming source IP addresses through network security groups. Virtual networks enable you to place Azure resources in a non-internet, routable network that you control access to. To learn more, see [Integrate your app with an Azure virtual network](#). In addition, you can also use [private link \(private endpoint\)](#) and disable the public network to force the private network connection between App Service and other services.

Best practice: Enforce HTTPS-only traffic and require TLS 1.2 or higher for all connections. Disable FTP access where possible; if file transfer is necessary, use FTPS to ensure secure, encrypted transfers. **Detail:** Configuring your App Service to accept only HTTPS traffic ensures data is encrypted in transit, protecting sensitive information from interception. Requiring TLS 1.2 or higher provides stronger security against vulnerabilities found in earlier protocol versions. Disabling FTP reduces the risk of credentials or data being transmitted unencrypted. If file transfer is required, enable only FTPS, which encrypts both credentials and data during transit.

Best practice: Monitor the security state of your App Service environments. **Detail:** Use Microsoft Defender for Cloud to monitor your App Service environments. When Defender for Cloud identifies potential security vulnerabilities, it creates recommendations that guide you

through the process of configuring the needed controls. Microsoft Defender for App Service provides threat protection for your App Service resources.

For more information, see [Microsoft Defender for App Service](#).

Web Application Firewall

Web applications are increasingly targets of malicious attacks that exploit common known vulnerabilities. Common among these exploits are SQL injection attacks and cross-site scripting attacks. Preventing such attacks in application code can be challenging and may require rigorous maintenance, patching, and monitoring at many layers of the application topology. A centralized web application firewall helps make security management much simpler and gives better assurance to application administrators against threats or intrusions. A WAF solution can also react to a security threat faster by patching a known vulnerability at a central location versus securing each individual web application.

[Azure Web Application Firewall \(WAF\)](#) provides centralized protection of your web applications from common exploits and vulnerabilities. WAF is available through [Azure Application Gateway](#) and [Azure Front Door](#).

DDoS protection

Azure offers two main DDoS protection tiers: [DDoS IP Protection](#) and [DDoS Network Protection](#). These options cover different scenarios and have distinct features and pricing.

- **DDoS IP Protection:** Best for protecting specific public IP addresses, ideal for smaller or targeted deployments needing essential DDoS mitigation at the IP level.
- **DDoS Network Protection:** Covers entire virtual networks with advanced mitigation, analytics, and integration; suited for larger or enterprise environments needing broader security.

Choose DDoS IP Protection for focused, cost-sensitive cases; select DDoS Network Protection for comprehensive coverage and advanced features.

DDoS Protection defends at the network layer (3/4). For application-layer (7) defense, add a WAF. See [Application DDoS protection](#).

Monitor application performance

[Azure Monitor](#) collects, analyzes, and acts on telemetry from your cloud and on-premises environments. An effective monitoring strategy helps you understand the detailed operation of

the components of your application. It helps you increase your uptime by notifying you of critical issues so that you can resolve them before they become problems. It also helps you detect anomalies that might be security related.

Use [Application Insights](#) to monitor availability, performance, and usage of your application, whether it's hosted in the cloud or on-premises. By using Application Insights, you can quickly identify and diagnose errors in your application without waiting for a user to report them. With the information that you collect, you can make informed choices on your application's maintenance and improvements.

Application Insights has extensive tools for interacting with the data that it collects. Application Insights stores its data in a common repository. It can take advantage of shared functionality such as alerts, dashboards, and deep analysis with the Kusto query language.

Perform security penetration testing

Validating security defenses is as important as testing any other functionality. Make [penetration testing](#) a standard part of your build and deployment process. Schedule regular security tests and vulnerability scanning on deployed applications, and monitor for open ports, endpoints, and attacks.

Next steps

In this article, we focused on security advantages of an Azure PaaS deployment and security best practices for cloud applications. Next, learn recommended practices for securing your PaaS web and mobile solutions using specific Azure services. We'll start with Azure App Service, Azure SQL Database and Azure Synapse Analytics, and Azure Storage. As articles on recommended practices for other Azure services become available, links will be provided in the following list:

- [Azure App Service](#)
- [Azure SQL Database and Azure Synapse Analytics](#)
- [Azure Storage](#)

See [Develop secure applications on Azure](#) for security questions and controls you should consider at each phase of the software development lifecycle when developing applications for the cloud.

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

The following resources are available to provide more general information about Azure security and related Microsoft services:

- [Azure Security Documentation](#) - for comprehensive security guidance
- [Microsoft Security Response Center](#)  - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com

Last updated on 11/18/2025

Best practices for securing PaaS web and mobile applications using Azure App Service

In this article, we discuss a collection of [Azure App Service](#) security best practices for securing your PaaS web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

Azure App Service is a platform-as-a-service (PaaS) offering that lets you create web and mobile apps for any platform or device and connect to data anywhere, in the cloud or on-premises. App Service includes the web and mobile capabilities that were previously delivered separately as Azure Websites and Azure Mobile Services. It also includes new capabilities for automating business processes and hosting cloud APIs. As a single integrated service, App Service brings a rich set of capabilities to web, mobile, and integration scenarios.

Authenticate through Microsoft Entra ID

App Service provides an OAuth 2.0 service for your identity provider. OAuth 2.0 focuses on client developer simplicity while providing specific authorization flows for web applications, desktop applications, and mobile phones. Microsoft Entra ID uses OAuth 2.0 to enable you to authorize access to mobile and web applications. To learn more, see [Authentication and authorization in Azure App Service](#).

Restrict access based on role

Restricting access is imperative for organizations that want to enforce security policies for data access. You can use Azure role-based access control (Azure RBAC) to assign permissions to users, groups, and applications at a certain scope, such as the need to know and least privilege security principles. To learn more about granting users access to applications, see [What is Azure role-based access control \(Azure RBAC\)](#).

Protect your keys

It doesn't matter how good your security is if you lose your subscription keys. Azure Key Vault helps safeguard cryptographic keys and secrets used by cloud applications and services. With Key Vault, you can encrypt keys and secrets (such as authentication keys, storage account keys, data encryption keys, .PFX files, and passwords) by using keys that are protected by hardware security modules (HSMs). For added assurance, you can import or generate keys in HSMs. You

can also use Key Vault to manage your TLS certificates with auto-renewal. See [What is Azure Key Vault](#) to learn more.

Restrict incoming source IP addresses

[App Service Environments](#) has a virtual network integration feature that helps you restrict incoming source IP addresses through network security groups (NSGs). If you are unfamiliar with Azure Virtual Networks (VNETs), this is a capability that allows you to place many of your Azure resources in a non-internet, routable network that you control access to. To learn more, see [Integrate your app with an Azure Virtual Network](#).

For App Service on Windows, you can also restrict IP addresses dynamically by configuring the web.config. For more information, see [Dynamic IP Security](#).

Next steps

This article introduced you to a collection of App Service security best practices for securing your PaaS web and mobile applications. To learn more about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS databases in Azure](#)

Last updated on 12/03/2025

Best practices for securing PaaS web and mobile applications using Azure Storage

In this article, we discuss a collection of Azure Storage security best practices for securing your platform-as-a-service (PaaS) web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

Azure makes it possible to deploy and use storage in ways not easily achievable on-premises. With Azure storage, you can reach high levels of scalability and availability with relatively little effort. Not only is Azure Storage the foundation for Windows and Linux Azure Virtual Machines, it can also support large distributed applications.

Azure Storage provides the following four services: Blob storage, Table storage, Queue storage, and File storage. To learn more, see [Introduction to Microsoft Azure Storage](#).

This article addresses the following best practices:

- Shared access signatures (SAS)
- Azure role-based access control (Azure RBAC)
- Client side encryption for high value data
- Storage Service Encryption

Use a shared access signature instead of a storage account key

Access control is critical. To help you control access to Azure Storage, Azure generates two 512-bit storage account keys (SAKs) when you create a storage account. The level of key redundancy makes it possible for you to avoid service interruptions during routine key rotation.

Storage access keys are high priority secrets and should only be accessible to people responsible for storage access control. If the wrong people get access to these keys, they'll have complete control of storage and could replace, delete, or add files to storage. This includes malware and other types of content that can potentially compromise your organization or your customers.

You still need a way to provide access to objects in storage. To provide more granular access, you can take advantage of shared access signature (SAS). The SAS makes it possible for you to share specific objects in storage for a pre-defined time-interval and with specific permissions. A shared access signature allows you to define:

- The interval over which the SAS is valid, including the start time and the expiry time.

- The permissions granted by the SAS. For example, a SAS on a blob might grant a user read and write permissions to that blob, but not delete permissions.
- An optional IP address or range of IP addresses from which Azure Storage accepts the SAS. For example, you might specify a range of IP addresses belonging to your organization. This provides another measure of security for your SAS.
- The protocol over which Azure Storage accepts the SAS. You can use this optional parameter to restrict access to clients using HTTPS.

SAS allows you to share content the way you want to share it without giving away your storage account keys. Always using SAS in your application is a secure way to share your storage resources without compromising your storage account keys.

To learn more about shared access signature, see [Using shared access signatures](#).

Use Azure role-based access control

Another way to manage access is to use [Azure role-based access control \(Azure RBAC\)](#). With Azure RBAC, you focus on giving employees the exact permissions they need, based on the need to know and least privilege security principles. Too many permissions can expose an account to attackers. Too few permissions means that employees can't get their work done efficiently. Azure RBAC helps address this problem by offering fine-grained access management for Azure. Access control is imperative for organizations that want to enforce security policies for data access.

You can use Azure built-in roles in Azure to assign privileges to users. For example, use Storage Account Contributor for cloud operators that need to manage storage accounts and Classic Storage Account Contributor role to manage classic storage accounts. For cloud operators that need to manage VMs but not the virtual network or storage account to which they're connected, you can add them to the Virtual Machine Contributor role.

Organizations that don't enforce data access control by using capabilities such as Azure RBAC may be giving more privileges than necessary for their users. More privileges than necessary can lead to data compromise by allowing some users access to data they shouldn't have in the first place.

To learn more about Azure RBAC see:

- [Assign Azure roles using the Azure portal](#)
- [Azure built-in roles](#)
- [Security recommendations for Blob storage](#)

Use client-side encryption for high value data

Client-side encryption enables you to programmatically encrypt data in transit before uploading to Azure Storage, and programmatically decrypt data when retrieving it. Client-side encryption provides encryption of data in transit but it also provides encryption of data at rest. Client-side encryption is the most secure method of encrypting your data but it does require you to make programmatic changes to your application and put key management processes in place.

Client-side encryption also enables you to have sole control over your encryption keys. You can generate and manage your own encryption keys. It uses an envelope technique where the Azure storage client library generates a content encryption key (CEK) that is then wrapped (encrypted) using the key encryption key (KEK). The KEK is identified by a key identifier and can be an asymmetric key pair or a symmetric key and can be managed locally or stored in [Azure Key Vault](#).

Client-side encryption is built into the Java and the .NET storage client libraries. See [Client-side encryption and Azure Key Vault for Microsoft Azure Storage](#) for information on encrypting data within client applications and generating and managing your own encryption keys.

Enable Storage Service Encryption for data at rest

When [Storage Service Encryption](#) for File storage is enabled, the data is encrypted automatically using AES-256 encryption. Microsoft handles all the encryption, decryption, and key management. This feature is available for LRS and GRS redundancy types.

Next steps

This article introduced you to a collection of Azure Storage security best practices for securing your PaaS web and mobile applications. To learn more about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS web and mobile applications using Azure App Services](#)
- [Securing PaaS databases in Azure](#)

Best practices for securing PaaS databases in Azure

Article • 04/16/2025

In this article, we discuss a collection of [Azure SQL Database](#) and [Azure Synapse Analytics](#) security best practices for securing your platform-as-a-service (PaaS) web and mobile applications. These best practices are derived from our experience with Azure and the experiences of customers like yourself.

Azure SQL Database and Azure Synapse Analytics provide a relational database service for your internet-based applications. Let's look at services that help protect your applications and data when using Azure SQL Database and Azure Synapse Analytics in a PaaS deployment:

- Microsoft Entra authentication (instead of SQL Server authentication)
- Azure SQL firewall
- Transparent Data Encryption (TDE)

Use a centralized identity repository

Azure SQL Database can be configured to use one of two types of authentication:

- **SQL authentication** uses a username and password. When you created the server for your database, you specified a "server admin" login with a username and password. Using these credentials, you can authenticate to any database on that server as the database owner.
- **Microsoft Entra authentication** uses identities managed by Microsoft Entra ID and is supported for managed and integrated domains. To use Microsoft Entra authentication, you must create another server admin called the "Microsoft Entra admin," which is allowed to administer Microsoft Entra users and groups. This admin can also perform all operations that a regular server admin can.

[Microsoft Entra authentication](#) is a mechanism of connecting to Azure SQL Database and Azure Synapse Analytics by using identities in Microsoft Entra ID. Microsoft Entra ID provides an alternative to SQL Server authentication so you can stop the proliferation of user identities across database servers. Microsoft Entra authentication enables you to centrally manage the identities of database users and other Microsoft services in one central location. Central ID management provides a single place to manage database users and simplifies permission management.

Benefits of using Microsoft Entra ID instead of SQL authentication

- Allows password rotation in a single place.
- Manages database permissions using external Microsoft Entra groups.
- Eliminates storing passwords by enabling integrated Windows authentication and other forms of authentication supported by Microsoft Entra ID.
- Uses contained database users to authenticate identities at the database level.
- Supports token-based authentication for applications connecting to SQL Database.
- Supports domain federation with Active Directory Federation Services (ADFS) or native user/password authentication for a local Microsoft Entra ID without domain synchronization.
- Supports connections from SQL Server Management Studio that use Active Directory Universal Authentication, which includes [Multi-Factor Authentication \(MFA\)](#). MFA includes strong authentication with a range of easy verification options. Verification options are phone call, text message, smart cards with pin, or mobile app notification. For more information, see [Universal Authentication with SQL Database and Azure Synapse Analytics](#).

To learn more about Microsoft Entra authentication, see:

- [Use Microsoft Entra authentication for authentication with SQL Database, Managed Instance, or Azure Synapse Analytics](#)
- [Authentication to Azure Synapse Analytics](#)
- [Token-based authentication support for Azure SQL Database using Microsoft Entra authentication](#)

ⓘ Note

To ensure that Microsoft Entra ID is a good fit for your environment, see [Microsoft Entra features and limitations](#).

Restrict access based on IP address

You can create firewall rules that specify ranges of acceptable IP addresses. These rules can be targeted at both the server and database levels. We recommend using database-level firewall rules whenever possible to enhance security and to make your database more portable. Server-level firewall rules are best used for administrators and when you have many databases that have the same access requirements but you don't want to spend time configuring each database individually.

SQL Database default source IP address restrictions allow access from any Azure address, including other subscriptions and tenants. You can restrict this to only allow your IP addresses to access the instance. Even with your SQL firewall and IP address restrictions, strong authentication is still needed. See the recommendations made earlier in this article.

To learn more about Azure SQL Firewall and IP restrictions, see:

- [Azure SQL Database and Azure Synapse Analytics access control](#)
- [Azure SQL Database and Azure Synapse Analytics firewall rules](#)

Encrypt data at rest

[Transparent Data Encryption \(TDE\)](#) is enabled by default. TDE transparently encrypts SQL Server, Azure SQL Database, and Azure Synapse Analytics data and log files. TDE protects against a compromise of direct access to the files or their backup. This enables you to encrypt data at rest without changing existing applications. TDE should always stay enabled; however, this will not stop an attacker using the normal access path. TDE provides the ability to comply with many laws, regulations, and guidelines established in various industries.

Azure SQL manages key related issues for TDE. As with TDE, on-premises special care must be taken to ensure recoverability and when moving databases. In more sophisticated scenarios, the keys can be explicitly managed in Azure Key Vault through extensible key management. See [Enable TDE on SQL Server Using EKM](#). This also allows for Bring Your Own Key (BYOK) through Azure Key Vaults BYOK capability.

Azure SQL provides encryption for columns through [Always Encrypted](#). This allows only authorized applications access to sensitive columns. Using this kind of encryption limits SQL queries for encrypted columns to equality-based values.

Application level encryption should also be used for selective data. Data sovereignty concerns can sometimes be mitigated by encrypting data with a key that is kept in the correct country/region. This prevents even accidental data transfer from causing an issue since it is impossible to decrypt the data without the key, assuming a strong algorithm is used (such as AES 256).

You can use additional precautions to help secure the database, such as designing a secure system, encrypting confidential assets, and building a firewall around the database servers.

Next steps

This article introduced you to a collection of SQL Database and Azure Synapse Analytics security best practices for securing your PaaS web and mobile applications. To learn more

about securing your PaaS deployments, see:

- [Securing PaaS deployments](#)
- [Securing PaaS web and mobile applications using Azure App Services](#)

Azure Service Fabric security best practices

In addition to this article, please also review [Service Fabric security checklist](#) for more information.

Deploying an application on Azure is fast, easy, and cost-effective. Before you deploy your cloud application into production, review our list of essential and recommended best practices for implementing secure clusters in your application.

Azure Service Fabric is a distributed systems platform that makes it easy to package, deploy, and manage scalable and reliable microservices. Service Fabric also addresses the significant challenges in developing and managing cloud applications. Developers and administrators can avoid complex infrastructure problems and focus on implementing mission-critical, demanding workloads that are scalable, reliable, and manageable.

For each best practice, we explain:

- What the best practice is.
- Why you should implement the best practice.
- What might happen if you don't implement the best practice.
- How you can learn to implement the best practice.

We recommend the following Azure Service Fabric security best practices:

- Use Azure Resource Manager templates and the Service Fabric PowerShell module to create secure clusters.
- Use X.509 certificates.
- Configure security policies.
- Implement the Reliable Actors security configuration.
- Configure TLS for Azure Service Fabric.
- Use network isolation and security with Azure Service Fabric.
- Configure Azure Key Vault for security.
- Assign users to roles.
- Things to consider if hosting untrusted applications in a Service Fabric cluster.

Best practices for securing your clusters

Always use a secure cluster:

- Implement cluster security by using certificates.
- Provide client access (admin and read-only) by using Microsoft Entra ID.

Use automated deployments:

- Use scripts to generate, deploy, and roll over the secrets.
- Store the secrets in Azure Key Vault and use Microsoft Entra ID for all other client access.
- Require authentication for human access to the secrets.

Additionally, consider the following configuration options:

- Create perimeter networks (also known as demilitarized zones, DMZs, and screened subnets) by using Azure Network Security Groups (NSGs).
- Access cluster virtual machines (VMs) or manage your cluster by using jump servers with Remote Desktop Connection.

Your clusters must be secured to prevent unauthorized users from connecting, especially when a cluster is running in production. Although it's possible to create an unsecured cluster, anonymous users can connect to your cluster if the cluster exposes management endpoints to the public internet.

There are three [scenarios](#) for implementing cluster security by using various technologies:

- Node-to-node security: This scenario secures communication between the VMs and the computers in the cluster. This form of security ensures that only those computers that are authorized to join the cluster can host applications and services in the cluster. In this scenario, the clusters that run on Azure, or standalone clusters that run on Windows, can use either [certificate security](#) or [Windows security](#) for Windows Server machines.
- Client-to-node security: This scenario secures communication between a Service Fabric client and the individual nodes in the cluster.
- Service Fabric role-based access control (Service Fabric RBAC): This scenario uses separate identities (certificates, Microsoft Entra ID, and so on) for each administrator and user client role that accesses the cluster. You specify the role identities when you create the cluster.

 Note

Security recommendation for Azure clusters: Use Microsoft Entra security to authenticate clients and certificates for node-to-node security.

To configure a standalone Windows cluster, see [Configure settings for a standalone Windows cluster](#).

Use Azure Resource Manager templates and the Service Fabric PowerShell module to create a secure cluster. For step-by-step instructions to create a secure Service Fabric cluster by using Azure Resource Manager templates, see [Creating a Service Fabric cluster](#).

Use the Azure Resource Manager template:

- Customize your cluster by using the template to configure managed storage for VM virtual hard disks (VHDs).
- Drive changes to your resource group by using the template for easy configuration management and auditing.

Treat your cluster configuration as code:

- Be thorough when checking your deployment configurations.
- Avoid using implicit commands to directly modify your resources.

Many aspects of the [Service Fabric application lifecycle](#) can be automated. The [Service Fabric PowerShell module](#) automates common tasks for deploying, upgrading, removing, and testing Azure Service Fabric applications. Managed APIs and HTTP APIs for application management are also available.

Use X.509 certificates

Always secure your clusters by using X.509 certificates or Windows security. Security is only configured at cluster creation time. It's not possible to turn on security after the cluster is created.

To specify a [cluster certificate](#), set the value of the **ClusterCredentialType** property to X509. To specify a server certificate for outside connections, set the **ServerCredentialType** property to X509.

In addition, follow these practices:

- Create the certificates for production clusters by using a correctly configured Windows Server certificate service. You can also obtain the certificates from an approved certificate authority (CA).
- Never use a temporary or test certificate for production clusters if the certificate was created by using MakeCert.exe or a similar tool.
- Use a self-signed certificate for test clusters, but not for production clusters.

If the cluster is unsecure, anyone can connect to the cluster anonymously and perform management operations. For this reason, always secure production clusters by using X.509 certificates or Windows security.

To learn more about using X.509 certificates, see [Add or remove certificates for a Service Fabric cluster](#).

Configure security policies

Service Fabric also secures the resources that are used by applications. Resources like files, directories, and certificates are stored under the user accounts when the application is deployed. This feature makes running applications more secure from one another, even in a shared hosted environment.

- Use an Active Directory domain group or user: Run the service under the credentials for an Active Directory user or group account. Be sure to use Active Directory on-premises within your domain and not Microsoft Entra ID. Access other resources in the domain that have been granted permissions by using a domain user or group. For example, resources such as file shares.
- Assign a security access policy for HTTP and HTTPS endpoints: Specify the **SecurityAccessPolicy** property to apply a **RunAs** policy to a service when the service manifest declares endpoint resources with HTTP. Ports allocated to the HTTP endpoints are correctly access-controlled lists for the RunAs user account that the service runs under. When the policy isn't set, http.sys doesn't have access to the service and you can get failures with calls from the client.

To learn how to use security policies in a Service Fabric cluster, see [Configure security policies for your application](#).

Implement the Reliable Actors security configuration

Service Fabric Reliable Actors is an implementation of the actor design pattern. As with any software design pattern, the decision to use a specific pattern is based on whether a software problem fits the pattern.

In general, use the actor design pattern to help model solutions for the following software problems or security scenarios:

- Your problem space involves a large number (thousands or more) of small, independent, and isolated units of state and logic.
- You're working with single-threaded objects that don't require significant interaction from external components, including querying state across a set of actors.
- Your actor instances don't block callers with unpredictable delays by issuing I/O operations.

In Service Fabric, actors are implemented in the Reliable Actors application framework. This framework is based on the actor pattern and built on top of [Service Fabric Reliable Services](#). Each reliable actor service that you write is a partitioned stateful reliable service.

Every actor is defined as an instance of an actor type, identical to the way a .NET object is an instance of a .NET type. For example, an **actor type** that implements the functionality of a calculator can have many actors of that type that are distributed on various nodes across a cluster. Each of the distributed actors is uniquely characterized by an actor identifier.

[Replicator security configurations](#) are used to secure the communication channel that is used during replication. This configuration prevents services from seeing each other's replication traffic and ensures that highly available data is secure. By default, an empty security configuration section prevents replication security. Replicator configurations configure the replicator that is responsible for making the Actor State Provider state highly reliable.

Configure TLS for Azure Service Fabric

The server authentication process [authenticates](#) the cluster management endpoints to a management client. The management client then recognizes that it's talking to the real cluster. This certificate also provides a [TLS](#) for the HTTPS management API and for Service Fabric Explorer over HTTPS. You must obtain a custom domain name for your cluster. When you request a certificate from a certificate authority, the certificate's subject name must match the custom domain name that you use for your cluster.

To configure TLS for an application, you first need to obtain an SSL/TLS certificate that has been signed by a CA. The CA is a trusted third party that issues certificates for TLS security purposes. If you don't already have an SSL/TLS certificate, you need to obtain one from a company that sells SSL/TLS certificates.

The certificate must meet the following requirements for SSL/TLS certificates in Azure:

- The certificate must contain a private key.
- The certificate must be created for key exchange and be exportable to a personal information exchange (.pfx) file.
- The certificate's subject name must match the domain name that is used to access your cloud service.
 - Acquire a custom domain name to use for accessing your cloud service.
 - Request a certificate from a CA with a subject name that matches your service's custom domain name. For example, if your custom domain name is **contoso.com**, the certificate from your CA should have the subject name **.contoso.com** or **www.contoso.com**.

ⓘ **Note**

You cannot obtain an SSL/TLS certificate from a CA for the **cloudapp.net** domain.

- The certificate must use a minimum of 2,048-bit encryption.

The HTTP protocol is insecure and subject to eavesdropping attacks. Data that is transmitted over HTTP is sent as plain text from the web browser to the web server or between other endpoints. Attackers can intercept and view sensitive data that is sent via HTTP, such as credit card details and account logins. When data is sent or posted through a browser via HTTPS, SSL ensures that sensitive information is encrypted and secure from interception.

To learn more about using SSL/TLS certificates, see [Configuring TLS for an application in Azure](#).

Use network isolation and security with Azure Service Fabric

Set up a 3 nodetype secure cluster by using the [Azure Resource Manager template](#) as a sample. Control the inbound and outbound network traffic by using the template and Network Security Groups.

The template has an NSG for each of the virtual machine scale sets and is used to control the traffic in and out of the set. The rules are configured by default to allow all traffic necessary for the system services and the application ports specified in the template. Review these rules and make any changes to fit your needs, including adding new rules for your applications.

For more information, see [Common networking scenarios for Azure Service Fabric](#).

Set up Azure Key Vault for security

Service Fabric uses certificates to provide authentication and encryption for securing a cluster and its applications.

Service Fabric uses X.509 certificates to secure a cluster and to provide application security features. You use Azure Key Vault to [manage certificates](#) for Service Fabric clusters in Azure. The Azure resource provider that creates the clusters pulls the certificates from a key vault. The provider then installs the certificates on the VMs when the cluster is deployed on Azure.

A certificate relationship exists between [Azure Key Vault](#), the Service Fabric cluster, and the resource provider that uses the certificates. When the cluster is created, information about the certificate relationship is stored in a key vault.

There are two basic steps to set up a key vault:

1. Create a resource group specifically for your key vault.

We recommend that you put the key vault in its own resource group. This action helps to prevent the loss of your keys and secrets if other resource groups are removed, such as storage, compute, or the group that contains your cluster. The resource group that contains your key vault must be in the same region as the cluster that is using it.

2. Create a key vault in the new resource group.

The key vault must be enabled for deployment. The compute resource provider can then get the certificates from the vault and install them on the VM instances.

To learn more about how to set up a key vault, see [What is Azure Key Vault?](#).

Assign users to roles

After you've created the applications to represent your cluster, assign your users to the roles that are supported by Service Fabric: read-only and admin. You can assign these roles by using the Azure portal.

ⓘ Note

For more information about using roles in Service Fabric, see [Service Fabric role-based access control for Service Fabric clients](#).

Azure Service Fabric supports two access control types for clients that are connected to a [Service Fabric cluster](#): administrator and user. The cluster administrator can use access control to limit access to certain cluster operations for different groups of users. Access control makes the cluster more secure.

Things to consider if hosting untrusted applications in a Service Fabric cluster

Please see [Hosting untrusted applications in a Service Fabric cluster](#).

Next steps

- [Service Fabric security checklist](#)
- Set up your Service Fabric [development environment](#).
- Learn about [Service Fabric support options](#).

Last updated on 12/03/2025

Azure security logging and auditing

Azure provides a wide array of configurable security auditing and logging options to help you identify gaps in your security policies and mechanisms. This article discusses generating, collecting, and analyzing security logs from services hosted on Azure.

ⓘ Note

Certain recommendations in this article might result in increased data, network, or compute resource usage, and increase your license or subscription costs.

Types of logs in Azure

Cloud applications are complex with many moving parts. Logging data can provide insights about your applications and help you:

- Troubleshoot past problems or prevent potential ones
- Improve application performance or maintainability
- Automate actions that would otherwise require manual intervention

Azure logs are categorized into the following types:

- **Control/management logs** provide information about Azure Resource Manager CREATE, UPDATE, and DELETE operations. For more information, see [Azure activity logs](#).
- **Data plane logs** provide information about events raised as part of Azure resource usage. Examples of this type of log are the Windows event system, security, and application logs in a virtual machine (VM) and the [diagnostics logs](#) that are configured through Azure Monitor.
- **Processed events** provide information about analyzed events/alerts that have been processed on your behalf. Examples of this type are [Microsoft Defender for Cloud alerts](#) where [Microsoft Defender for Cloud](#) has processed and analyzed your subscription and provides concise security alerts.

The following table lists the most important types of logs available in Azure:

[⌵](#) Expand table

Log category	Log type	Usage	Integration
Activity logs	Control-plane events on Azure Resource Manager	Provides insight into the operations that were performed	REST API, Azure Monitor

Log category	Log type	Usage	Integration
	resources	on resources in your subscription.	
Azure Resource logs	Frequent data about the operation of Azure Resource Manager resources in subscription	Provides insight into operations that your resource itself performed.	Azure Monitor
Microsoft Entra ID reporting	Logs and reports	Reports user sign-in activities and system activity information about users and group management.	Microsoft Graph
Virtual machines and cloud services	Windows Event Log service and Linux Syslog	Captures system data and logging data on the virtual machines and transfers that data into a storage account of your choice.	Windows (using Azure Diagnostics storage) and Linux in Azure Monitor
Azure Storage Analytics	Storage logging, provides metrics data for a storage account	Provides insight into trace requests, analyzes usage trends, and diagnoses issues with your storage account.	REST API or the client library
Network security group (NSG) flow logs	JSON format, shows outbound and inbound flows on a per-rule basis	Displays information about ingress and egress IP traffic through a Network Security Group.	Azure Network Watcher
Application insight	Logs, exceptions, and custom diagnostics	Provides an application performance monitoring (APM) service for web developers on multiple platforms.	REST API, Power BI 
Process data / security alerts	Microsoft Defender for Cloud alerts, Azure Monitor logs alerts	Provides security information and alerts.	REST APIs, JSON

Log integration with on-premises SIEM systems

[Integrating Defender for Cloud alerts](#) discusses how to sync Defender for Cloud alerts, virtual machine security events collected by Azure diagnostics logs, and Azure audit logs with your Azure Monitor logs or SIEM solution.

Next steps

- [Auditing and logging](#): Protect data by maintaining visibility and responding quickly to timely security alerts.
- [Configure audit settings for a site collection](#) : If you're a site collection administrator, retrieve the history of individual users' actions and the history of actions taken during a particular date range.
- [Search the audit log in the Microsoft Defender Portal](#): Use the Microsoft Defender Portal to search the unified audit log and view user and administrator activity in your organization.

Last updated on 12/03/2025

Azure security management and monitoring overview

Azure provides comprehensive security management and monitoring capabilities to help you govern, secure, and maintain visibility across your Azure resources. This article covers key management and monitoring services that support secure operations.

Azure Monitor

[Azure Monitor](#) collects, analyzes, and acts on telemetry data from your Azure and on-premises environments. Monitor helps you maximize the availability and performance of your applications and proactively identify issues.

Azure Monitor provides:

- **Metrics and logs:** Collect and analyze data from Azure resources, operating systems, and applications
- **Log Analytics workspaces:** Centralized storage and analysis of log data with powerful query capabilities
- **Application Insights:** Application performance management (APM) for monitoring live web applications
- **Azure Monitor Alerts:** Proactive notifications based on metrics, logs, and activity data
- **Azure Workbooks:** Interactive visual reports combining text, queries, metrics, and parameters

For security monitoring, Azure Monitor integrates with Microsoft Sentinel and Microsoft Defender for Cloud to provide comprehensive threat detection and response capabilities.

Learn more:

- [Azure Monitor overview](#)
- [Azure Monitor Logs overview](#)
- [Threat detection overview](#)

Azure role-based access control

Azure role-based access control (Azure RBAC) provides fine-grained access management for Azure resources. With Azure RBAC, you grant users only the access they need to perform their jobs, following the principle of least privilege.

Azure RBAC enables you to:

- Assign built-in roles or create custom roles
- Control access at multiple scope levels (management group, subscription, resource group, resource)
- Separate duties within teams and grant only necessary access
- Integrate with Microsoft Entra ID for identity-based access control
- Audit role assignments through Azure Activity Log

Learn more:

- [What is Azure role-based access control \(Azure RBAC\)?](#)
- [Azure built-in roles](#)
- [Identity management security overview](#)

Azure Policy

[Azure Policy](#) helps you enforce organizational standards and assess compliance at scale. Azure Policy evaluates resources in Azure by comparing their properties to defined rules.

Azure Policy capabilities include:

- **Policy definitions:** Rules that describe compliance conditions and effects
- **Initiatives:** Collections of policy definitions grouped to achieve specific compliance goals
- **Compliance reporting:** Dashboard views showing compliant and non-compliant resources
- **Automatic remediation:** Deploy corrective configurations for non-compliant resources
- **Regulatory compliance:** Built-in policy sets aligned with standards like Microsoft cloud security benchmark, ISO 27001, and NIST

Common security use cases:

- Enforce encryption requirements for storage accounts and databases
- Require diagnostic settings for audit logging
- Restrict resource deployments to approved Azure regions
- Enforce naming conventions and tagging standards
- Require specific security configurations (TLS versions, firewall rules)

Learn more:

- [What is Azure Policy?](#)
- [Azure Policy regulatory compliance](#)

Azure Update Manager

[Azure Update Manager](#) is a unified service that helps you manage and govern operating system updates for Windows and Linux virtual machines across Azure, on-premises, and multicloud environments.

Azure Update Manager provides:

- **Update assessment:** Automatic or on-demand assessment of available updates
- **Scheduled patching:** Configure recurring maintenance windows for update installation
- **One-time updates:** Install updates immediately for urgent security patches
- **Hotpatching:** Install security updates on Windows Server without requiring reboots (supported SKUs)
- **Update compliance reporting:** Dashboard views and Azure Workbooks showing update status
- **Integration with Azure Policy:** Enforce update policies at scale

Update Manager features include:

- Native Azure experience with zero onboarding required
- Granular access control at the resource level using Azure RBAC
- Support for Azure VMs and Azure Arc-enabled servers
- Pre and post-event scripts for custom automation
- Integration with Azure Monitor for alerts and notifications

Learn more:

- [About Azure Update Manager](#)
- [Scheduled patching in Update Manager](#)

Activity logging and auditing

Azure Activity Log records subscription-level events including administrative operations, service health events, and resource health changes. Activity Log provides visibility into who performed what operations and when.

Activity Log capabilities:

- **Administrative operations:** Create, update, delete operations on Azure resources
- **Service health:** Azure service incidents and maintenance notifications
- **Resource health:** Availability status changes for Azure resources
- **Retention and export:** Retain logs for up to 90 days; export to Log Analytics, Storage, or Event Hubs for longer retention
- **Integration with alerts:** Create alert rules based on Activity Log events

For comprehensive security auditing, you can configure diagnostic settings to send logs to Log Analytics workspaces for analysis with Microsoft Sentinel or Defender for Cloud.

Learn more:

- [Azure Activity Log](#)
- [Diagnostic settings](#)

Microsoft Cost Management

[Microsoft Cost Management](#) helps you monitor, allocate, and optimize your Azure spending. Understanding costs is essential for security management as unauthorized resource deployments can indicate security incidents.

Cost Management provides:

- **Cost analysis:** Visualize and analyze costs across subscriptions, resource groups, and tags
- **Budgets:** Set spending limits with proactive alerts
- **Recommendations:** Identify opportunities to reduce costs without compromising security
- **Cost allocation:** Distribute costs across business units using tags and subscriptions
- **Anomaly detection:** Identify unusual spending patterns that may indicate security issues

Learn more:

- [What is Microsoft Cost Management?](#)
- [Create and manage budgets](#)

Azure Resource Graph

[Azure Resource Graph](#) provides efficient resource exploration with the ability to query at scale across subscriptions. Resource Graph enables security teams to quickly identify resources with specific configurations or security postures.

Resource Graph capabilities:

- **Fast querying:** Query thousands of resources across multiple subscriptions in seconds
- **Complex queries:** Use Kusto Query Language (KQL) to analyze resource properties and relationships
- **Resource inventory:** Discover all resources of specific types or with particular configurations
- **Compliance verification:** Identify resources that don't meet security or compliance requirements
- **Change tracking:** Track resource property changes over time

Learn more:

- [What is Azure Resource Graph?](#)
- [Starter Resource Graph queries](#)

Azure Automation

[Azure Automation](#) delivers cloud-based automation and configuration management supporting consistent governance across Azure and non-Azure environments.

Azure Automation provides:

- **Process automation:** Automate frequent, time-consuming, and error-prone tasks using PowerShell and Python runbooks
- **Configuration management:** Apply and maintain desired state configurations using State Configuration (DSC)
- **Shared resources:** Centralized storage for credentials, certificates, connections, and variables used in automation
- **Change tracking:** Monitor configuration changes across files, registry, services, and software
- **Inventory collection:** Discover and track software and configurations across machines

Common security automation scenarios:

- Automated incident response workflows
- Scheduled security scans and remediation
- Configuration drift detection and correction
- Automated backup and disaster recovery operations

Learn more:

- [Azure Automation overview](#)
- [Change Tracking and Inventory overview](#)

Azure Advisor

[Azure Advisor](#) is a personalized cloud consultant that provides best practice recommendations to optimize your Azure deployments. Advisor includes security recommendations from Microsoft Defender for Cloud.

Advisor recommendation categories:

- **Reliability:** Improve availability and disaster recovery capabilities

- **Security:** Detect threats and vulnerabilities through Defender for Cloud integration
- **Performance:** Improve application speed and responsiveness
- **Cost:** Optimize and reduce overall Azure spending
- **Operational Excellence:** Achieve process and workflow efficiency

Learn more:

- [Azure Advisor overview](#)
- [Get started with Azure Advisor](#)

Azure Service Health

[Azure Service Health](#) provides personalized information about the health of your Azure services and regions. Service Health helps you plan for maintenance and respond to incidents that may affect availability.

Service Health components:

- **Azure status:** Global view of Azure service health across all regions
- **Service Health:** Personalized view of the health of Azure services you use in the regions you use them
- **Resource Health:** Health information about individual Azure resources
- **Health alerts:** Proactive notifications about service issues, planned maintenance, and health advisories

Learn more:

- [What is Azure Service Health?](#)
- [Create activity log alerts on service notifications](#)

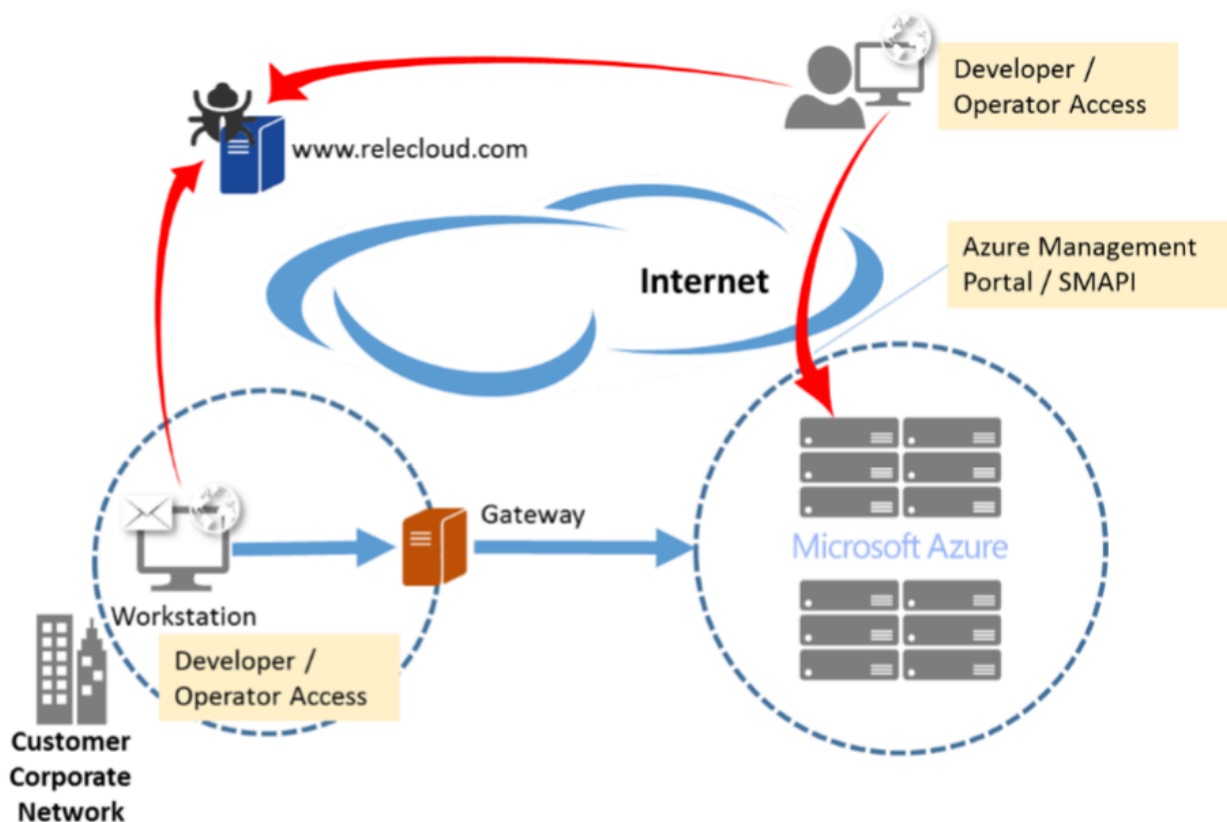
Next steps

- [Threat detection and protection](#)
- [Identity management security overview](#)
- [Network security overview](#)
- [Azure security best practices and patterns](#)

Security management in Azure

Azure subscribers may manage their cloud environments from multiple devices, including management workstations, developer PCs, and even privileged end-user devices that have task-specific permissions. In some cases, administrative functions are performed through web-based consoles such as the [Azure portal](#). In other cases, there may be direct connections to Azure from on-premises systems over Virtual Private Networks (VPNs), Terminal Services, client application protocols, or (programmatically) the Azure classic deployment model. Additionally, client endpoints can be either domain joined or isolated and unmanaged, such as tablets or smartphones.

Although multiple access and management capabilities provide a rich set of options, this variability can add significant risk to a cloud deployment. It can be difficult to manage, track, and audit administrative actions. This variability may also introduce security threats through unregulated access to client endpoints that are used for managing cloud services. Using general or personal workstations for developing and managing infrastructure opens unpredictable threat vectors such as web browsing (for example, watering hole attacks) or email (for example, social engineering and phishing).



The potential for attacks increases in this type of environment because it's challenging to construct security policies and mechanisms to appropriately manage access to Azure interfaces (such as SMAPI) from widely varied endpoints.

Remote management threats

Attackers often attempt to gain privileged access by compromising account credentials (for example, through password brute forcing, phishing, and credential harvesting), or by tricking users into running harmful code (for example, from harmful websites with drive-by downloads or from harmful email attachments). In a remotely managed cloud environment, account breaches can lead to an increased risk due to anywhere, anytime access.

Even with tight controls on primary administrator accounts, lower-level user accounts can be used to exploit weaknesses in one's security strategy. Lack of appropriate security training can also lead to breaches through accidental disclosure or exposure of account information.

When a user workstation is also used for administrative tasks, it can be compromised at many different points. Whether a user is browsing the web, using 3rd-party and open-source tools, or opening a harmful document file that contains a trojan.

In general, most targeted attacks that result in data breaches can be traced to browser exploits, plug-ins (such as Flash, PDF, Java), and spear phishing (email) on desktop machines. These machines may have administrative-level or service-level permissions to access live servers or network devices for operations when used for development or management of other assets.

Operational security fundamentals

For more secure management and operations, you can minimize a client's attack surface by reducing the number of possible entry points. This can be done through security principles: "separation of duties" and "segregation of environments."

Isolate sensitive functions from one another to decrease the likelihood that a mistake at one level leads to a breach in another. Examples:

- Administrative tasks shouldn't be combined with activities that might lead to a compromise (for example, malware in an administrator's email that then infects an infrastructure server).
- A workstation used for high-sensitivity operations shouldn't be the same system used for high-risk purposes such as browsing the Internet.

Reduce the system's attack surface by removing unnecessary software. Example:

- Standard administrative, support, or development workstation shouldn't require installation of an email client or other productivity applications if the device's main purpose is to manage cloud services.

Client systems that have administrator access to infrastructure components should be subjected to the strictest possible policy to reduce security risks. Examples:

- Security policies can include Group Policy settings that deny open Internet access from the device and use of a restrictive firewall configuration.
- Use Internet Protocol security (IPsec) VPNs if direct access is needed.
- Configure separate management and development Active Directory domains.
- Isolate and filter management workstation network traffic.
- Use antimalware software.
- Implement multi-factor authentication to reduce the risk of stolen credentials.

Consolidating access resources and eliminating unmanaged endpoints also simplifies management tasks.

Providing security for Azure remote management

Azure provides security mechanisms to aid administrators who manage Azure cloud services and virtual machines. These mechanisms include:

- Authentication and [Azure role-based access control \(Azure RBAC\)](#).
- Monitoring, logging, and auditing.
- Certificates and encrypted communications.
- A web management portal.
- Network packet filtering.

With client-side security configuration and datacenter deployment of a management gateway, it's possible to restrict and monitor administrator access to cloud applications and data.

ⓘ Note

Certain recommendations in this article may result in increased data, network, or compute resource usage, and may increase your license or subscription costs.

Hardened workstation for management

The goal of hardening a workstation is to eliminate all but the most critical functions required for it to operate, making the potential attack surface as small as possible. System hardening includes minimizing the number of installed services and applications, limiting application execution, restricting network access to only what is needed, and always keeping the system up to date. Furthermore, using a hardened workstation for management segregates administrative tools and activities from other end-user tasks.

Within an on-premises enterprise environment, you can limit the attack surface of your physical infrastructure through dedicated management networks, server rooms that have card access,

and workstations that run on protected areas of the network. In a cloud or hybrid IT model, being diligent about secure management services can be more complex because of the lack of physical access to IT resources. Implementing protection solutions requires careful software configuration, security-focused processes, and comprehensive policies.

Using a least-privilege minimized software footprint in a locked-down workstation for cloud management and for application development can reduce the risk of security incidents by standardizing the remote management and development environments. A hardened workstation configuration can help prevent the compromise of accounts that are used to manage critical cloud resources by closing many common avenues used by malware and exploits. Specifically, you can use [Windows AppLocker](#) and Hyper-V technology to control and isolate client system behavior and mitigate threats, including email or Internet browsing.

On a hardened workstation, the administrator runs a standard user account (which blocks administrative-level execution) and associated applications are controlled by an allowlist. The basic elements of a hardened workstation are as follows:

- Active scanning and patching. Deploy antimalware software, perform regular vulnerability scans, and update all workstations by using the latest security update in a timely fashion.
- Limited functionality. Uninstall any applications that aren't needed and disable unnecessary (startup) services.
- Network hardening. Use Windows Firewall rules to allow only valid IP addresses, ports, and URLs related to Azure management. Ensure that inbound remote connections to the workstation are also blocked.
- Execution restriction. Allow only a set of predefined executable files that are needed for management to run (referred to as "default-deny"). By default, users should be denied permission to run any program unless it's explicitly defined in the allowlist.
- Least privilege. Management workstation users shouldn't have any administrative privileges on the local machine itself. This way, they can't change the system configuration or the system files, either intentionally or unintentionally.

You can enforce all this by using [Group Policy Objects \(GPOs\)](#) in Active Directory Domain Services (AD DS) and applying them through your (local) management domain to all management accounts.

Managing services, applications, and data

Azure cloud services configuration is performed through either the Azure portal or SMAPI, via the Windows PowerShell command-line interface or a custom-built application that takes advantage of these RESTful interfaces. Services using these mechanisms include Microsoft Entra ID, Azure Storage, Azure Websites, and Azure Virtual Network, and others.

Virtual Machine deployed applications provide their own client tools and interfaces as needed, such as the Microsoft Management Console (MMC), an enterprise management console (such as Microsoft System Center or Windows Intune), or another management application Microsoft SQL Server Management Studio, for example. These tools typically reside in an enterprise environment or client network. They may depend on specific network protocols, such as Remote Desktop Protocol (RDP), that require direct, stateful connections. Some may have web-enabled interfaces that shouldn't be openly published or accessible via the Internet.

You can restrict access to infrastructure and platform services management in Azure by using [multi-factor authentication](#), X.509 management certificates, and firewall rules. The Azure portal and SMAPI require Transport Layer Security (TLS). However, services and applications that you deploy into Azure require you to take protection measures that are appropriate based on your application. These mechanisms can frequently be enabled more easily through a standardized hardened workstation configuration.

Security guidelines

In general, helping to secure administrator workstations for use with the cloud is similar to the practices used for any workstation on-premises. For example, minimized build and restrictive permissions. Some unique aspects of cloud management are more akin to remote or out-of-band enterprise management. These include the use and auditing of credentials, security-enhanced remote access, and threat detection and response.

Authentication

You can use Azure logon restrictions to constrain source IP addresses for accessing administrative tools and audit access requests. To help Azure identify management clients (workstations and/or applications), you can configure both SMAPI (via customer-developed tools such as Windows PowerShell cmdlets) and the Azure portal to require client-side management certificates to be installed, in addition to TLS/SSL certificates. We also recommend that administrator access require multi-factor authentication.

Some applications or services that you deploy into Azure may have their own authentication mechanisms for both end-user and administrator access, whereas others take full advantage of Microsoft Entra ID. Depending on whether you're federating credentials via Active Directory Federation Services (AD FS), using directory synchronization or maintaining user accounts solely in the cloud, using [Microsoft Identity Manager](#) (part of Microsoft Entra ID P1 or P2) helps you manage identity lifecycles between the resources.

Connectivity

Several mechanisms are available to help secure client connections to your Azure virtual networks. Two of these mechanisms, site-to-site VPN (S2S) and [point-to-site VPN](#) (P2S), enable the use of industry standard IPsec (S2S) for encryption and tunneling. When Azure is connecting to public-facing Azure services management such as the Azure portal, Azure requires Hypertext Transfer Protocol Secure (HTTPS).

A stand-alone hardened workstation that doesn't connect to Azure through an RD Gateway should use the SSTP-based point-to-site VPN to create the initial connection to the Azure Virtual Network, and then establish RDP connection to individual virtual machines from with the VPN tunnel.

Management auditing vs. policy enforcement

Typically, there are two approaches for helping to secure management processes: auditing and policy enforcement. Doing both provides comprehensive controls, but may not be possible in all situations. In addition, each approach has different levels of risk, cost, and effort associated with managing security, particularly as it relates to the level of trust placed in both individuals and system architectures.

Monitoring, logging, and auditing provide a basis for tracking and understanding administrative activities, but it may not always be feasible to audit all actions in complete detail due to the amount of data generated. Auditing the effectiveness of the management policies is a best practice, however.

Policy enforcement that includes strict access controls puts programmatic mechanisms in place that can govern administrator actions, and it helps ensure that all possible protection measures are being used. Logging provides proof of enforcement, in addition to a record of who did what, from where, and when. Logging also enables you to audit and crosscheck information about how administrators follow policies, and it provides evidence of activities

Client configuration

We recommend three primary configurations for a hardened workstation. The biggest differentiators between them are cost, usability, and accessibility, while maintaining a similar security profile across all options. The following table provides a short analysis of the benefits and risks to each. (Note that "corporate PC" refers to a standard desktop PC configuration that would be deployed for all domain users, regardless of roles.)

[Expand table](#)

Configuration	Benefits	Cons
Stand-alone hardened workstation	Tightly controlled workstation	higher cost for dedicated desktops
-	Reduced risk of application exploits	Increased management effort
-	Clear separation of duties	-
Corporate PC as virtual machine	Reduced hardware costs	-
-	Segregation of role and applications	-

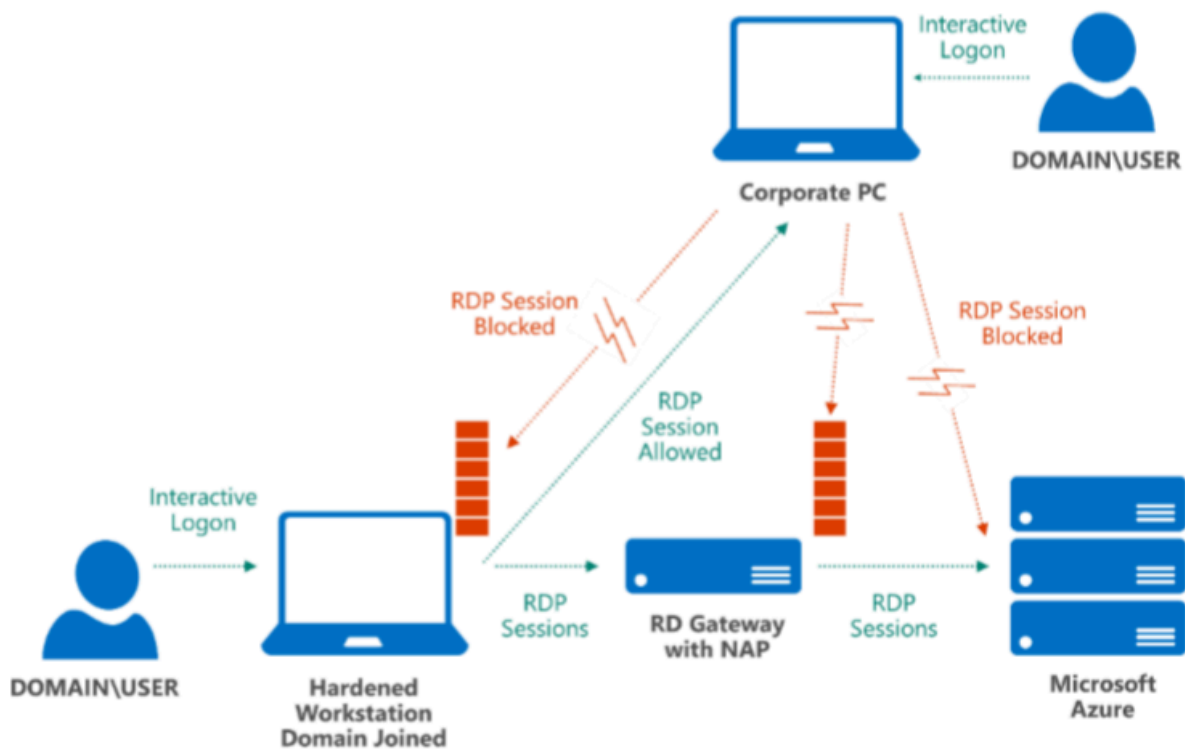
It's important that the hardened workstation is the host and not the guest, with nothing between the host operating system and the hardware. Following the "clean source principle" (also known as "secure origin") means that the host should be the most hardened. Otherwise, the hardened workstation (guest) is subject to attacks on the system on which it's hosted.

You can further segregate administrative functions through dedicated system images for each hardened workstation that have only the tools and permissions needed for managing select Azure and cloud applications, with specific local AD DS GPOs for the necessary tasks.

For IT environments that have no on-premises infrastructure (for example, no access to a local AD DS instance for GPOs because all servers are in the cloud), a service such as [Microsoft Intune](#) can simplify deploying and maintaining workstation configurations.

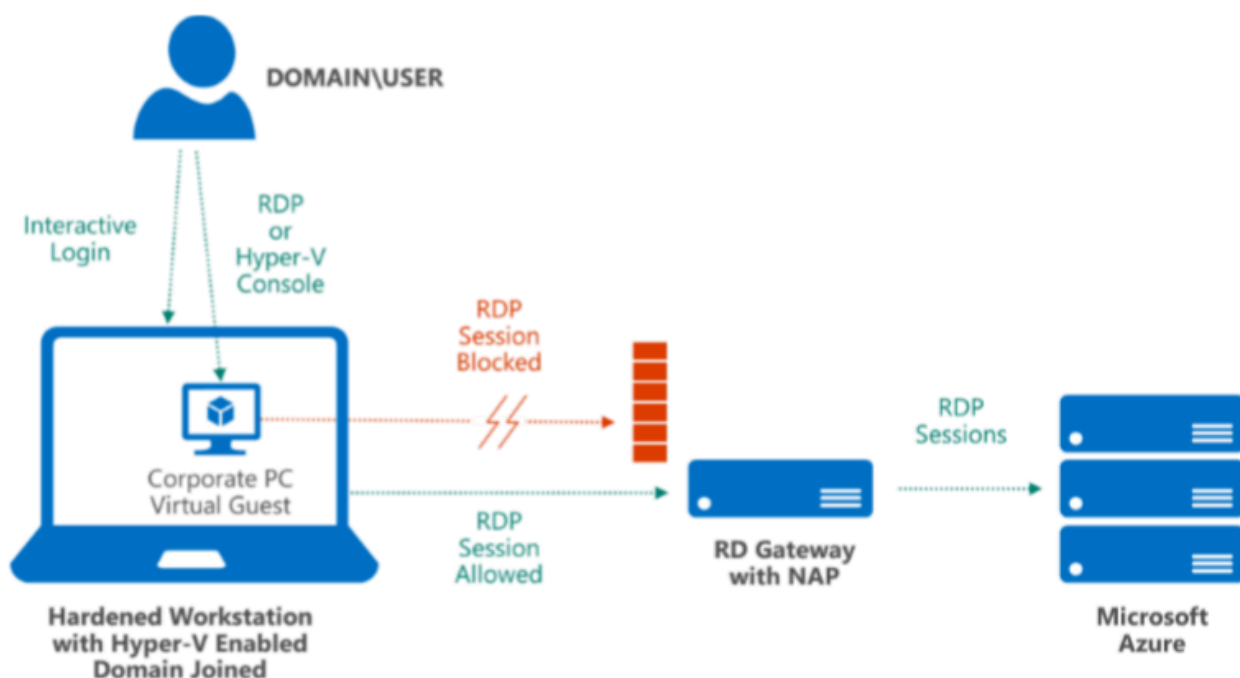
Stand-alone hardened workstation for management

With a stand-alone hardened workstation, administrators have a PC or laptop that they use for administrative tasks and another, separate PC or laptop for non-administrative tasks. In the stand-alone hardened workstation scenario (shown below), the local instance of Windows Firewall (or a non-Microsoft client firewall) is configured to block inbound connections, such as RDP. The administrator can log on to the hardened workstation and start an RDP session that connects to Azure after establishing a VPN connect with an Azure Virtual Network, but can't log on to a corporate PC and use RDP to connect to the hardened workstation itself.



Corporate PC as virtual machine

In cases where a separate stand-alone hardened workstation is cost prohibitive or inconvenient, the hardened workstation can host a virtual machine to perform non-administrative tasks.



To avoid several security risks that can arise from using one workstation for systems management and other daily work tasks, you can deploy a Windows Hyper-V virtual machine

to the hardened workstation. This virtual machine can be used as the corporate PC. The corporate PC environment can remain isolated from the Host, which reduces its attack surface and removes the user's daily activities (such as email) from coexisting with sensitive administrative tasks.

The corporate PC virtual machine runs in a protected space and provides user applications. The host remains a "clean source" and enforces strict network policies in the root operating system (for example, blocking RDP access from the virtual machine).

Best practices

Consider the following additional guidelines when you're managing applications and data in Azure.

Dos and don'ts

Don't assume that because a workstation has been locked down that other common security requirements don't need to be met. The potential risk is higher because of elevated access levels that administrator accounts generally possess. Examples of risks and their alternate safe practices are shown in the table below.

 Expand table

Don't	Do
Don't email credentials for administrator access or other secrets (for example, TLS/SSL or management certificates)	Maintain confidentiality by delivering account names and passwords by voice (but not storing them in voice mail), perform a remote installation of client/server certificates (via an encrypted session), download from a protected network share, or distribute by hand via removable media.
-	Proactively manage your management certificate life cycles.
Don't store account passwords unencrypted or un-hashed in application storage (such as in spreadsheets, SharePoint sites, or file shares).	Establish security management principles and system hardening policies, and apply them to your development environment.
Don't share accounts and passwords between administrators, or reuse passwords across multiple user accounts or services, particularly those for social media or other nonadministrative activities.	Create a dedicated Microsoft account to manage your Azure subscription, an account that is not used for personal email.

Don't	Do
Don't email configuration files.	Configuration files and profiles should be installed from a trusted source (for example, an encrypted USB flash drive), not from a mechanism that can be easily compromised, such as email.
Don't use weak or simple logon passwords.	Enforce strong password policies, expiration cycles (change-on-first-use), console timeouts, and automatic account lockouts. Use a client password management system with multi-factor authentication for password vault access.
Don't expose management ports to the Internet.	Lock down Azure ports and IP addresses to restrict management access.
-	Use firewalls, VPNs, and NAP for all management connections.

Azure operations

Within Microsoft's operation of Azure, operations engineers and support personnel who access Azure's production systems use [hardened workstation PCs with VMs](#) provisioned on them for internal corporate network access and applications (such as e-mail, intranet, etc.). All management workstation computers have TPMs, the host boot drive is encrypted with BitLocker, and they're joined to a special organizational unit (OU) in Microsoft's primary corporate domain.

System hardening is enforced through Group Policy, with centralized software updating. For auditing and analysis, event logs (such as security and AppLocker) are collected from management workstations and saved to a central location.

In addition, dedicated jump-boxes on Microsoft's network that require two-factor authentication are used to connect to Azure's production network.

Azure security checklist

Minimizing the number of tasks that administrators can perform on a hardened workstation helps minimize the attack surface in your development and management environment. Use the following technologies to help protect your hardened workstation:

- A web browser is a key entry point for harmful code due to its extensive interactions with external servers. Review your client policies and enforce running in protected mode, disabling add-ons, and disabling file downloads. Ensure that security warnings are

displayed. Take advantage of Internet zones and create a list of trusted sites for which you have configured reasonable hardening. Block all other sites and in-browser code, such as ActiveX and Java.

- Standard user. Running as a standard user brings a number of benefits, the biggest of which is that stealing administrator credentials via malware becomes more difficult. In addition, a standard user account doesn't have elevated privileges on the root operating system, and many configuration options and APIs are locked out by default.
- Code signing. Code signing all tools and scripts used by administrators provides a manageable mechanism for deploying application lockdown policies. Hashes don't scale with rapid changes to the code, and file paths don't provide a high level of security. [Set the PowerShell execution policies for Windows computers.](#)
- Group Policy. Create a global administrative policy that is applied to any domain workstation that is used for management (and block access from all others), and to user accounts authenticated on those workstations.
- Security-enhanced provisioning. Safeguard your baseline hardened workstation image to help protect against tampering. Use security measures like encryption and isolation to store images, virtual machines, and scripts, and restrict access (perhaps use an auditable check-in/check-out process).
- Patching. Maintain a consistent build (or have separate images for development, operations, and other administrative tasks), scan for changes and malware routinely, keep the build up to date, and only activate machines when they're needed.
- Governance. Use AD DS GPOs to control all the administrators' Windows interfaces, such as file sharing. Include management workstations in auditing, monitoring, and logging processes. Track all administrator and developer access and usage.

Summary

Using a hardened workstation configuration for administering your Azure cloud services, Virtual Machines, and applications can help you avoid numerous risks and threats that can come from remotely managing critical IT infrastructure. Both Azure and Windows provide mechanisms that you can employ to help protect and control communications, authentication, and client behavior.

Next steps

The following resources are available to provide more general information about Azure and related Microsoft services:

- Securing Privileged Access - get the technical details for designing and building a secure administrative workstation for Azure management

- [Microsoft Trust Center](#) - learn about Azure platform capabilities that protect the Azure fabric and the workloads that run on Azure
 - [Microsoft Security Response Center](#) - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com
-

Last updated on 12/03/2025

Azure operational security overview

[Azure operational security](#) refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure. It's a framework that incorporates the knowledge gained through a variety of capabilities that are unique to Microsoft. These capabilities include the Microsoft Security Development Lifecycle (SDL), the Microsoft Security Response Center program, and deep awareness of the cybersecurity threat landscape.

Azure management services

An IT operations team is responsible for managing datacenter infrastructure, applications, and data, including the stability and security of these systems. However, gaining security insights across increasing complex IT environments often requires organizations to cobble together data from multiple security and management systems.

[Microsoft Azure Monitor logs](#) is a cloud-based IT management solution that helps you manage and protect your on-premises and cloud infrastructure. Its core functionality is provided by the following services that run in Azure. Azure includes multiple services that help you manage and protect your on-premises and cloud infrastructure. Each service provides a specific management function. You can combine services to achieve different management scenarios.

Azure Monitor

[Azure Monitor](#) collects data from managed sources into central data stores. This data can include events, performance data, or custom data provided through the API. After the data is collected, it's available for alerting, analysis, and export.

You can consolidate data from a variety of sources and combine data from your Azure services with your existing on-premises environment. Azure Monitor logs also clearly separates the collection of the data from the action taken on that data, so that all actions are available to all kinds of data.

Automation

[Azure Automation](#) provides a way for you to automate the manual, long-running, error-prone, and frequently repeated tasks that are commonly performed in a cloud and enterprise environment. It saves time and increases the reliability of administrative tasks. It even schedules these tasks to be automatically performed at regular intervals. You can automate processes by using runbooks or automate configuration management by using Desired State Configuration.

Backup

[Azure Backup](#) is the Azure-based service that you can use to back up (or protect) and restore your data in the Microsoft Cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a cloud-based solution that's reliable, secure, and cost-competitive.

Azure Backup offers components that you download and deploy on the appropriate computer or server, or in the cloud. The component, or agent, that you deploy depends on what you want to protect. All Azure Backup components (whether you're protecting data on-premises or in the cloud) can be used to back up data to an Azure Recovery Services vault in Azure.

For more information, see the [Azure Backup components table](#).

Site Recovery

[Azure Site Recovery](#) provides business continuity by orchestrating the replication of on-premises virtual and physical machines to Azure, or to a secondary site. If your primary site is unavailable, you fail over to the secondary location so that users can keep working. You fail back when systems return to working order. Use Microsoft Defender for Cloud to perform more intelligent and effective threat detection.

Microsoft Entra ID

[Microsoft Entra ID](#) is a comprehensive identity service that:

- Enables identity and access management (IAM) as a cloud service.
- Provides central access management, single sign-on (SSO), and reporting.
- Supports integrated access management for [thousands of applications](#) [↗](#) in the Azure Marketplace, including Salesforce, Google Apps, Box, and Concur.

Microsoft Entra ID also includes a full suite of [identity management capabilities](#), including these:

- [Multi-factor authentication](#)
- [Self-service password management](#)
- [Self-service group management](#) [↗](#)
- [Privileged account management](#)
- [Azure role-based access control \(Azure RBAC\)](#)
- [Application usage monitoring](#)
- [Rich auditing](#)
- [Security monitoring and alerting](#)

With Microsoft Entra ID, all applications that you publish for your partners and customers (business or consumer) have the same identity and access management capabilities. This enables you to significantly reduce your operational costs.

Microsoft Defender for Cloud

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats with increased visibility into (and control over) the security of your Azure resources. It provides integrated security monitoring and policy management across your subscriptions. It helps detect threats that might otherwise go unnoticed, and it works with a broad ecosystem of security solutions.

[Safeguard virtual machine \(VM\) data](#) in Azure by providing visibility into your virtual machine's security settings and monitoring for threats. Defender for Cloud can monitor your virtual machines for:

- Operating system security settings with the recommended configuration rules.
- System security and critical updates that are missing.
- Endpoint protection recommendations.
- Disk encryption validation.
- Network-based attacks.

Defender for Cloud uses [Azure role-based access control \(Azure RBAC\)](#). Azure RBAC provides [built-in roles](#) that can be assigned to users, groups, and services in Azure.

Defender for Cloud assesses the configuration of your resources to identify security issues and vulnerabilities. In Defender for Cloud, you see information related to a resource only when you're assigned the role of owner, contributor, or reader for the subscription or resource group that a resource belongs to.

Note

To learn more about roles and allowed actions in Defender for Cloud, see [Permissions in Microsoft Defender for Cloud](#).

Defender for Cloud uses the Microsoft Monitoring Agent. This is the same agent that the Azure Monitor service uses. Data collected from this agent is stored in either an existing Log Analytics [workspace](#) associated with your Azure subscription or a new workspace, taking into account the geolocation of the VM.

Azure Monitor

Performance issues in your cloud app can affect your business. With multiple interconnected components and frequent releases, degradations can happen at any time. And if you're developing an app, your users usually discover issues that you didn't find in testing. You should know about these issues immediately, and you should have tools for diagnosing and fixing the problems.

[Azure Monitor](#) is basic tool for monitoring services running on Azure. It gives you infrastructure-level data about the throughput of a service and the surrounding environment. If you're managing your apps all in Azure and deciding whether to scale up or down resources, Azure Monitor is the place to start.

You can also use monitoring data to gain deep insights about your application. That knowledge can help you to improve application performance or maintainability, or automate actions that would otherwise require manual intervention.

Azure Monitor includes the following components.

Azure Activity Log

The [Azure Activity Log](#) provides insight into the operations that were performed on resources in your subscription. It was previously known as "Audit Log" or "Operational Log," because it reports control-plane events for your subscriptions.

Azure diagnostic logs

[Azure diagnostic logs](#) are emitted by a resource and provide rich, frequent data about the operation of that resource. The content of these logs varies by resource type.

Windows event system logs are one category of diagnostic logs for VMs. Blob, table, and queue logs are categories of diagnostic logs for storage accounts.

Diagnostic logs differ from the [Activity Log](#). The Activity log provides insight into the operations that were performed on resources in your subscription. Diagnostic logs provide insight into operations that your resource performed itself.

Metrics

Azure Monitor provides telemetry that gives you visibility into the performance and health of your workloads on Azure. The most important type of Azure telemetry data is the [metrics](#) (also called performance counters) emitted by most Azure resources. Azure Monitor provides several ways to configure and consume these metrics for monitoring and troubleshooting.

Azure Diagnostics

Azure Diagnostics enables the collection of diagnostic data on a deployed application. You can use the Diagnostics extension from various sources. Currently supported are [Azure cloud service roles](#), [Azure virtual machines](#) running Microsoft Windows, and [Azure Service Fabric](#).

Azure Network Watcher

Customers build an end-to-end network in Azure by orchestrating and composing individual network resources such as virtual networks, Azure ExpressRoute, Azure Application Gateway, and load balancers. Monitoring is available on each of the network resources.

The end-to-end network can have complex configurations and interactions between resources. The result is complex scenarios that need scenario-based monitoring through [Azure Network Watcher](#).

Network Watcher simplifies monitoring and diagnosing of your Azure network. You can use the diagnostic and visualization tools in Network Watcher to:

- Take remote packet captures on an Azure virtual machine.
- Gain insights into your network traffic by using flow logs.
- Diagnose Azure VPN Gateway and connections.

Network Watcher currently has the following capabilities:

- [Topology](#): Provides a view of the various interconnections and associations between network resources in a resource group.
- [Variable packet capture](#): Captures packet data in and out of a virtual machine. Advanced filtering options and fine-tuned controls, such as the ability to set time and size limitations, provide versatility. The packet data can be stored in a blob store or on the local disk in .cap format.
- [IP flow verify](#): Checks if a packet is allowed or denied based on 5-tuple packet parameters for flow information (destination IP, source IP, destination port, source port, and protocol). If a security group denies the packet, the rule and group that denied the packet are returned.
- [Next hop](#): Determines the next hop for packets being routed in the Azure network fabric, so you can diagnose any misconfigured user-defined routes.
- [Security group view](#): Gets the effective and applied security rules that are applied on a VM.
- [NSG flow logs for network security groups](#): Enable you to capture logs related to traffic that is allowed or denied by the security rules in the group. The flow is defined by 5-tuple information: source IP, destination IP, source port, destination port, and protocol.

- [Virtual network gateway and connection troubleshooting](#): Provides the ability to troubleshoot virtual network gateways and connections.
- [Network subscription limits](#): Enables you to view network resource usage against limits.
- [Diagnostic logs](#): Provides a single pane to enable or disable diagnostic logs for network resources in a resource group.

For more information, see [Configure Network Watcher](#).

Cloud Service Provider Access Transparency

[Customer Lockbox for Microsoft Azure](#) is a service integrated into Azure portal that gives you explicit control in the rare instance when a Microsoft Support Engineer may need access to your data to resolve an issue. There are very few instances, such as a debugging remote access issue, where a Microsoft Support Engineer requires elevated permissions to resolve this issue. In such cases, Microsoft engineers use just-in-time access service that provides limited, time-bound authorization with access limited to the service.

While Microsoft has always obtained customer consent for access, Customer Lockbox now gives you the ability to review and approve or deny such requests from the Azure portal. Microsoft support engineers will not be granted access until you approve the request.

Standardized and Compliant Deployments

[Azure Blueprints](#) enable cloud architects and central information technology groups to define a repeatable set of Azure resources that implement and adhere to an organization's standards, patterns, and requirements.

This makes it possible for DevOps teams to rapidly build and stand up new environments and trust that they're building them with infrastructure that maintains organizational compliance. Blueprints provide a declarative way to orchestrate the deployment of various resource templates and other artifacts such as:

- Role Assignments
- Policy Assignments
- Azure Resource Manager templates
- Resource Groups

DevOps

Before [Developer Operations \(DevOps\)](#) [↗](#) application development, teams were in charge of gathering business requirements for a software program and writing code. Then a separate QA team tested the program in an isolated development environment. If requirements were met,

the QA team released the code for operations to deploy. The deployment teams were further fragmented into groups like networking and database. Each time a software program was “thrown over the wall” to an independent team, it added bottlenecks.

DevOps enables teams to deliver more secure, higher-quality solutions faster and more cheaply. Customers expect a dynamic and reliable experience when consuming software and services. Teams must rapidly iterate on software updates and measure the impact of the updates. They must respond quickly with new development iterations to address issues or provide more value.

Cloud platforms such as Microsoft Azure have removed traditional bottlenecks and helped commoditize infrastructure. Software reigns in every business as the key differentiator and factor in business outcomes. No organization, developer, or IT worker can or should avoid the DevOps movement.

Mature DevOps practitioners adopt several of the following practices. These practices [involve people](#) to form strategies based on the business scenarios. Tooling can help automate the various practices.

- [Agile planning and project management](#) [↗](#) techniques are used to plan and isolate work into sprints, manage team capacity, and help teams quickly adapt to changing business needs.
- [Version control, usually with Git](#), enables teams located anywhere in the world to share source and integrate with software development tools to automate the release pipeline.
- [Continuous integration](#) drives the ongoing merging and testing of code, which leads to finding defects early. Other benefits include less time wasted on fighting merge issues and rapid feedback for development teams.
- [Continuous delivery](#) of software solutions to production and testing environments helps organizations quickly fix bugs and respond to ever-changing business requirements.
- [Monitoring](#) of running applications--including production environments for application health, as well as customer usage--helps organizations form a hypothesis and quickly validate or disprove strategies. Rich data is captured and stored in various logging formats.
- [Infrastructure as Code \(IaC\)](#) is a practice that enables the automation and validation of creation and teardown of networks and virtual machines to help with delivering secure, stable application hosting platforms.
- [Microservices](#) architecture is used to isolate business use cases into small reusable services. This architecture enables scalability and efficiency.

Next steps

To learn about the Security and Audit solution, see the following articles:

- [Security and compliance](#)[↗]
 - [Microsoft Defender for Cloud](#)
 - [Azure Monitor](#)
-

Last updated on 12/03/2025

Azure Operational Security best practices

Article • 04/23/2025

This article provides a set of operational best practices for protecting your data, applications, and other assets in Azure.

The best practices are based on a consensus of opinion, and they work with current Azure platform capabilities and feature sets. Opinions and technologies change over time and this article is updated on a regular basis to reflect those changes.

Define and deploy strong operational security practices

Azure operational security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Azure. Azure operational security is built on a framework that incorporates the knowledge gained through capabilities that are unique to Microsoft, including the [Security Development Lifecycle \(SDL\)](#), the [Microsoft Security Response Center](#) program, and deep awareness of the cybersecurity threat landscape.

Enforce multifactor verification for users

We recommend that you require two-step verification for all of your users. This includes administrators and others in your organization who can have a significant impact if their account is compromised (for example, financial officers).

There are multiple options for requiring two-step verification. The best option for you depends on your goals, the Microsoft Entra edition you're running, and your licensing program. See [How to require two-step verification for a user](#) to determine the best option for you. See the [Microsoft Entra ID](#) and [Microsoft Entra multifactor Authentication](#) pricing pages for more information about licenses and pricing.

Following are options and benefits for enabling two-step verification:

Option 1: Enable MFA for all users and login methods with Microsoft Entra Security Defaults

Benefit: This option enables you to easily and quickly enforce MFA for all users in your environment with a stringent policy to:

- Challenge administrative accounts and administrative logon mechanisms
- Require MFA challenge via Microsoft Authenticator for all users
- Restrict legacy authentication protocols.

This method is available to all licensing tiers but is not able to be mixed with existing Conditional Access policies. You can find more information in [Microsoft Entra Security Defaults](#)

Option 2: [Enable multifactor authentication by changing user state.](#)

Benefit: This is the traditional method for requiring two-step verification. It works with both [Microsoft Entra multifactor authentication in the cloud](#) and [Azure Multi-Factor Authentication Server](#). Using this method requires users to perform two-step verification every time they sign in and overrides Conditional Access policies.

To determine where multifactor authentication needs to be enabled, see [Which version of Microsoft Entra multifactor authentication is right for my organization?](#).

Option 3: [Enable multifactor authentication with Conditional Access policy.](#) **Benefit:** This option allows you to prompt for two-step verification under specific conditions by using [Conditional Access](#). Specific conditions can be user sign-in from different locations, untrusted devices, or applications that you consider risky. Defining specific conditions where you require two-step verification enables you to avoid constant prompting for your users, which can be an unpleasant user experience.

This is the most flexible way to enable two-step verification for your users. Enabling a Conditional Access policy works only for Microsoft Entra multifactor authentication in the cloud and is a premium feature of Microsoft Entra ID. You can find more information on this method in [Deploy cloud-based Microsoft Entra multifactor authentication](#).

Option 4: Enable multifactor authentication with Conditional Access policies by evaluating [Risk-based Conditional Access policies](#).

Benefit: This option enables you to:

- Detect potential vulnerabilities that affect your organization's identities.
- Configure automated responses to detected suspicious actions that are related to your organization's identities.
- Investigate suspicious incidents and take appropriate action to resolve them.

This method uses the Microsoft Entra ID Protection risk evaluation to determine if two-step verification is required based on user and sign-in risk for all cloud applications. This method requires Microsoft Entra ID P2 licensing. You can find more information on this method in [Microsoft Entra ID Protection](#).

 Note

Option 2, enabling multifactor authentication by changing the user state, overrides Conditional Access policies. Because options 3 and 4 use Conditional Access policies, you cannot use option 2 with them.

Organizations that don't add extra layers of identity protection, such as two-step verification, are more susceptible for credential theft attack. A credential theft attack can lead to data compromise.

Manage and monitor user passwords

The following table lists some best practices related to managing user passwords:

Best practice: Ensure you have the proper level of password protection in the cloud.

Detail: Follow the guidance in [Microsoft Password Guidance](#), which is scoped to users of the Microsoft identity platforms (Microsoft Entra ID, Active Directory, and Microsoft account).

Best practice: Monitor for suspicious actions related to your user accounts.

Detail: Monitor for [users at risk](#) and [risky sign-ins](#) by using Microsoft Entra security reports.

Best practice: Automatically detect and remediate high-risk passwords.

Detail: [Microsoft Entra ID Protection](#) is a feature of the Microsoft Entra ID P2 edition that enables you to:

- Detect potential vulnerabilities that affect your organization's identities
- Configure automated responses to detected suspicious actions that are related to your organization's identities
- Investigate suspicious incidents and take appropriate actions to resolve them

Receive incident notifications from Microsoft

Be sure your security operations team receives Azure incident notifications from Microsoft. An incident notification lets your security team know you have compromised Azure resources so they can quickly respond to and remediate potential security risks.

In the Azure enrollment portal, you can ensure admin contact information includes details that notify security operations. Contact information is an email address and phone number.

Organize Azure subscriptions into management groups

If your organization has many subscriptions, you might need a way to efficiently manage access, policies, and compliance for those subscriptions. [Azure management groups](#) provide a level of scope that's above subscriptions. You organize subscriptions into containers called management groups and apply your governance conditions to the management groups. All

subscriptions within a management group automatically inherit the conditions applied to the management group.

You can build a flexible structure of management groups and subscriptions into a directory. Each directory is given a single top-level management group called the root management group. This root management group is built into the hierarchy to have all management groups and subscriptions fold up to it. The root management group allows global policies and Azure role assignments to be applied at the directory level.

Here are some best practices for using management groups:

Best practice: Ensure that new subscriptions apply governance elements like policies and permissions as they are added.

Detail: Use the root management group to assign enterprise-wide security elements that apply to all Azure assets. Policies and permissions are examples of elements.

Best practice: Align the top levels of management groups with segmentation strategy to provide a point for control and policy consistency within each segment.

Detail: Create a single management group for each segment under the root management group. Don't create any other management groups under the root.

Best practice: Limit management group depth to avoid confusion that hampers both operations and security.

Detail: Limit your hierarchy to three levels, including the root.

Best practice: Carefully select which items to apply to the entire enterprise with the root management group.

Detail: Ensure root management group elements have a clear need to be applied across every resource and that they're low impact.

Good candidates include:

- Regulatory requirements that have a clear business impact (for example, restrictions related to data sovereignty)
- Requirements with near-zero potential negative effect on operations, like policy with audit effect or Azure RBAC permission assignments that have been carefully reviewed

Best practice: Carefully plan and test all enterprise-wide changes on the root management group before applying them (policy, Azure RBAC model, and so on).

Detail: Changes in the root management group can affect every resource on Azure. While they provide a powerful way to ensure consistency across the enterprise, errors or incorrect usage can negatively affect production operations. Test all changes to the root management group in a test lab or production pilot.

Streamline environment creation with blueprints

The [Azure Blueprints](#) service enables cloud architects and central information technology groups to define a repeatable set of Azure resources that implements and adheres to an organization's standards, patterns, and requirements. Azure Blueprints makes it possible for development teams to rapidly build and stand up new environments with a set of built-in components and the confidence that they're creating those environments within organizational compliance.

Monitor storage services for unexpected changes in behavior

Diagnosing and troubleshooting issues in a distributed application hosted in a cloud environment can be more complex than it is in traditional environments. Applications can be deployed in a PaaS or IaaS infrastructure, on-premises, on a mobile device, or in some combination of these environments. Your application's network traffic might traverse public and private networks, and your application might use multiple storage technologies.

You should continuously monitor the storage services that your application uses for any unexpected changes in behavior (such as slower response times). Use logging to collect more detailed data and to analyze a problem in depth. The diagnostics information that you obtain from both monitoring and logging helps you to determine the root cause of the issue that your application encountered. Then you can troubleshoot the issue and determine the appropriate steps to remediate it.

[Azure Storage Analytics](#) performs logging and provides metrics data for an Azure storage account. We recommend that you use this data to trace requests, analyze usage trends, and diagnose issues with your storage account.

Prevent, detect, and respond to threats

[Microsoft Defender for Cloud](#) helps you prevent, detect, and respond to threats by providing increased visibility into (and control over) the security of your Azure resources. It provides integrated security monitoring and policy management across your Azure subscriptions, helps detect threats that might otherwise go unnoticed, and works with various security solutions.

The Free tier of Defender for Cloud offers limited security for your resources in Azure as well as Arc-enabled resources outside of Azure. The Enhanced Security Features extend these capabilities to include threat and vulnerability management, as well as regulatory compliance reporting. Defender for Cloud Plans help you find and fix security vulnerabilities, apply access

and application controls to block malicious activity, detect threats by using analytics and intelligence, and respond quickly when under attack. You can try Defender for Cloud Standard at no cost for the first 30 days. We recommend that you [enable enhanced security features on your Azure subscriptions in Defender for Cloud](#).

Use Defender for Cloud to get a central view of the security state of all your resources in your own data centers, Azure and other clouds. At a glance, verify that the appropriate security controls are in place and configured correctly, and quickly identify any resources that need attention.

Defender for Cloud also integrates with [Microsoft Defender for Endpoint](#), which provides comprehensive Endpoint Detection and Response (EDR) capabilities. With Microsoft Defender for Endpoint integration, you can spot abnormalities and detect vulnerabilities. You can also detect and respond to advanced attacks on server endpoints monitored by Defender for Cloud.

Almost all enterprise organizations have a security information and event management (SIEM) system to help identify emerging threats by consolidating log information from diverse signal gathering devices. The logs are then analyzed by a data analytics system to help identify what's "interesting" from the noise that is inevitable in all log gathering and analytics solutions.

[Microsoft Sentinel](#) is a scalable, cloud-native, security information and event management (SIEM) and security orchestration automated response (SOAR) solution. Microsoft Sentinel provides intelligent security analytics and threat intelligence via alert detection, threat visibility, proactive hunting, and automated threat response.

Here are some best practices for preventing, detecting, and responding to threats:

Best practice: Increase the speed and scalability of your SIEM solution by using a cloud-based SIEM.

Detail: Investigate the features and capabilities of [Microsoft Sentinel](#) and compare them with the capabilities of what you're currently using on-premises. Consider adopting Microsoft Sentinel if it meets your organization's SIEM requirements.

Best practice: Find the most serious security vulnerabilities so you can prioritize investigation.

Detail: Review your [Azure secure score](#) to see the recommendations resulting from the Azure policies and initiatives built into Microsoft Defender for Cloud. These recommendations help address top risks like security updates, endpoint protection, encryption, security configurations, missing WAF, internet-connected VMs, and many more.

The secure score, which is based on Center for Internet Security (CIS) controls, lets you benchmark your organization's Azure security against external sources. External validation helps validate and enrich your team's security strategy.

Best practice: Monitor the security posture of machines, networks, storage and data services, and applications to discover and prioritize potential security issues.

Detail: Follow the [security recommendations](#) in Defender for Cloud starting, with the highest priority items.

Best practice: Integrate Defender for Cloud alerts into your security information and event management (SIEM) solution.

Detail: Most organizations with a SIEM use it as a central clearinghouse for security alerts that require an analyst response. Processed events produced by Defender for Cloud are published to the Azure Activity Log, one of the logs available through Azure Monitor. Azure Monitor offers a consolidated pipeline for routing any of your monitoring data into a SIEM tool. See [Stream alerts to a SIEM, SOAR, or IT Service Management solution](#) for instructions. If you're using Microsoft Sentinel, see [Connect Microsoft Defender for Cloud](#).

Best practice: Integrate Azure logs with your SIEM.

Detail: Use [Azure Monitor to gather and export data](#). This practice is critical for enabling security incident investigation, and online log retention is limited. If you're using Microsoft Sentinel, see [Connect data sources](#).

Best practice: Speed up your investigation and hunting processes and reduce false positives by integrating Endpoint Detection and Response (EDR) capabilities into your attack investigation.

Detail: [Enable the Microsoft Defender for Endpoint integration](#) via your Defender for Cloud security policy. Consider using Microsoft Sentinel for threat hunting and incident response.

Monitor end-to-end scenario-based network monitoring

Customers build an end-to-end network in Azure by combining network resources like a virtual network, ExpressRoute, Application Gateway, and load balancers. Monitoring is available on each of the network resources.

[Azure Network Watcher](#) is a regional service. Use its diagnostic and visualization tools to monitor and diagnose conditions at a network scenario level in, to, and from Azure.

The following are best practices for network monitoring and available tools.

Best practice: Automate remote network monitoring with packet capture.

Detail: Monitor and diagnose networking issues without logging in to your VMs by using Network Watcher. Trigger [packet capture](#) by setting alerts and gain access to real-time performance information at the packet level. When you see an issue, you can investigate in detail for better diagnoses.

Best practice: Gain insight into your network traffic by using flow logs.

Detail: Build a deeper understanding of your network traffic patterns by using [network security group flow logs](#). Information in flow logs helps you gather data for compliance, auditing, and monitoring your network security profile.

Best practice: Diagnose VPN connectivity issues.

Detail: Use Network Watcher to [diagnose your most common VPN Gateway and connection issues](#). You can not only identify the issue but also use detailed logs to further investigate.

Secure deployment by using proven DevOps tools

Use the following DevOps best practices to ensure that your enterprise and teams are productive and efficient.

Best practice: Automate the build and deployment of services.

Detail: [Infrastructure as code](#) is a set of techniques and practices that help IT pros remove the burden of day-to-day build and management of modular infrastructure. It enables IT pros to build and maintain their modern server environment in a way that's like how software developers build and maintain application code.

You can use [Azure Resource Manager](#) to provision your applications by using a declarative template. In a single template, you can deploy multiple services along with their dependencies. You use the same template to repeatedly deploy your application in every stage of the application lifecycle.

Best practice: Automatically build and deploy to Azure web apps or cloud services.

Detail: You can configure your Azure DevOps Projects to [automatically build and deploy](#) to Azure web apps or cloud services. Azure DevOps automatically deploys the binaries after doing a build to Azure after every code check-in. The package build process is equivalent to the Package command in Visual Studio, and the publishing steps are equivalent to the Publish command in Visual Studio.

Best practice: Automate release management.

Detail: [Azure Pipelines](#) is a solution for automating multiple-stage deployment and managing the release process. Create managed continuous deployment pipelines to release quickly, easily, and often. With Azure Pipelines, you can automate your release process, and you can have predefined approval workflows. Deploy on-premises and to the cloud, extend, and customize as required.

Best practice: Check your app's performance before you launch it or deploy updates to production.

Detail: Run cloud-based [load tests](#) to:

- Find performance problems in your app.
- Improve deployment quality.
- Make sure that your app is always available.
- Make sure that your app can handle traffic for your next launch or marketing campaign.

[Apache JMeter](#) is a free, popular open source tool with a strong community backing.

Best practice: Monitor application performance.

Detail: [Azure Application Insights](#) is an extensible application performance management (APM) service for web developers on multiple platforms. Use Application Insights to monitor your live web application. It automatically detects performance anomalies. It includes analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability.

Mitigate and protect against DDoS

Distributed denial of service (DDoS) is a type of attack that tries to exhaust application resources. The goal is to affect the application's availability and its ability to handle legitimate requests. These attacks are becoming more sophisticated and larger in size and impact. They can be targeted at any endpoint that is publicly reachable through the internet.

Designing and building for DDoS resiliency requires planning and designing for a variety of failure modes. Following are best practices for building DDoS-resilient services on Azure.

Best practice: Ensure that security is a priority throughout the entire lifecycle of an application, from design and implementation to deployment and operations. Applications can have bugs that allow a relatively low volume of requests to use a lot of resources, resulting in a service outage.

Detail: To help protect a service running on Microsoft Azure, you should have a good understanding of your application architecture and focus on the [five pillars of software quality](#). You should know typical traffic volumes, the connectivity model between the application and other applications, and the service endpoints that are exposed to the public internet.

Ensuring that an application is resilient enough to handle a denial of service that's targeted at the application itself is most important. Security and privacy are built into the Azure platform, beginning with the [Security Development Lifecycle \(SDL\)](#). The SDL addresses security at every development phase and ensures that Azure is continually updated to make it even more secure.

Best practice: Design your applications to [scale horizontally](#) to meet the demand of an amplified load, specifically in the event of a DDoS attack. If your application depends on a single instance of a service, it creates a single point of failure. Provisioning multiple instances

makes your system more resilient and more scalable.

Detail: For [Azure App Service](#), select an [App Service plan](#) that offers multiple instances.

For Azure Cloud Services, configure each of your roles to use [multiple instances](#).

For [Azure Virtual Machines](#), ensure that your VM architecture includes more than one VM and that each VM is included in an [availability set](#). We recommend using Virtual Machine Scale Sets for autoscaling capabilities.

Best practice: Layering security defenses in an application reduces the chance of a successful attack. Implement secure designs for your applications by using the built-in capabilities of the Azure platform.

Detail: The risk of attack increases with the size (surface area) of the application. You can reduce the surface area by using an approval list to close down the exposed IP address space and listening ports that are not needed on the load balancers ([Azure Load Balancer](#) and [Azure Application Gateway](#)).

[Network security groups](#) are another way to reduce the attack surface. You can use [service tags](#) and [application security groups](#) to minimize complexity for creating security rules and configuring network security, as a natural extension of an application's structure.

You should deploy Azure services in a [virtual network](#) whenever possible. This practice allows service resources to communicate through private IP addresses. Azure service traffic from a virtual network uses public IP addresses as source IP addresses by default.

Using [service endpoints](#) switches service traffic to use virtual network private addresses as the source IP addresses when they're accessing the Azure service from a virtual network.

We often see customers' on-premises resources getting attacked along with their resources in Azure. If you're connecting an on-premises environment to Azure, minimize exposure of on-premises resources to the public internet.

Azure has two DDoS [service offerings](#) that provide protection from network attacks:

- Basic protection is integrated into Azure by default at no additional cost. The scale and capacity of the globally deployed Azure network provides defense against common network-layer attacks through always-on traffic monitoring and real-time mitigation. Basic requires no user configuration or application changes and helps protect all Azure services, including PaaS services like Azure DNS.
- Standard protection provides advanced DDoS mitigation capabilities against network attacks. It's automatically tuned to protect your specific Azure resources. Protection is simple to enable during the creation of virtual networks. It can also be done after creation and requires no application or resource changes.

Enable Azure Policy

[Azure Policy](#) is a service in Azure that you use to create, assign, and manage policies. These policies enforce rules and effects over your resources, so those resources stay compliant with your corporate standards and service-level agreements. Azure Policy meets this need by evaluating your resources for non-compliance with assigned policies.

Enable Azure Policy to monitor and enforce your organization's written policy. This will ensure compliance with your company or regulatory security requirements by centrally managing security policies across your hybrid cloud workloads. Learn how to [create and manage policies to enforce compliance](#). See [Azure Policy definition structure](#) for an overview of the elements of a policy.

Here are some security best practices to follow after you adopt Azure Policy:

Best practice: Policy supports several types of effects. You can read about them in [Azure Policy definition structure](#). Business operations can be negatively affected by the **deny** effect and the **remediate** effect, so start with the **audit** effect to limit the risk of negative impact from policy.

Detail: [Start policy deployments in audit mode](#) and then later progress to **deny** or **remediate**. Test and review the results of the audit effect before you move to **deny** or **remediate**.

For more information, see [Create and manage policies to enforce compliance](#).

Best practice: Identify the roles responsible for monitoring for policy violations and ensuring the right remediation action is taken quickly.

Detail: Have the assigned role monitor compliance through the [Azure portal](#) or via the [command line](#).

Best practice: Azure Policy is a technical representation of an organization's written policies. Map all Azure Policy definitions to organizational policies to reduce confusion and increase consistency.

Detail: Document mapping in your organization's documentation or in the Azure Policy definition itself by adding a reference to the organizational policy in the [policy definition](#) or the [initiative definition](#) description.

Monitor Microsoft Entra risk reports

The vast majority of security breaches take place when attackers gain access to an environment by stealing a user's identity. Discovering compromised identities is no easy task. Microsoft Entra ID uses adaptive machine learning algorithms and heuristics to detect suspicious actions that are related to your user accounts. Each detected suspicious action is stored in a record called a [risk detection](#). Risk detections are recorded in Microsoft Entra security reports. For

more information, read about the [users at risk security report](#) and the [risky sign-ins security report](#).

Next steps

See [Azure security best practices and patterns](#) for more security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure.

The following resources are available to provide more general information about Azure security and related Microsoft services:

- [Azure Security Team Blog](#) - for up to date information on the latest in Azure Security
- [Microsoft Security Response Center](#)  - where Microsoft security vulnerabilities, including issues with Azure, can be reported or via email to secure@microsoft.com

Azure operational security checklist

Article • 04/23/2025

Deploying a cloud application on Azure is fast, easy, and cost-effective. Before deploying an application, it's useful to have a checklist. A checklist can assist you in evaluating your application against a list of essential and recommended security actions.

Introduction

Azure provides a suite of infrastructure services that you can use to deploy your applications. Azure Operational Security refers to the services, controls, and features available to users for protecting their data, applications, and other assets in Microsoft Azure.

To get the maximum benefit out of the cloud platform, we recommend that you use Azure services and follow the checklist. Organizations that invest time and resources assessing the operational readiness of their applications before launch have a higher rate of satisfaction than those that don't. When performing this work, checklists can be an invaluable mechanism to ensure that applications are evaluated consistently and holistically.

Checklist

This checklist is intended to help enterprises think through various operational security considerations as they deploy sophisticated enterprise applications on Azure. It can also be used to help you build a secure cloud migration and operation strategy for your organization.

 Expand table

Checklist Category	Description
Security Roles & Access Controls	Use Azure role-based access control (Azure RBAC) to provide user-specific that used to assign permissions to users, groups, and applications at a certain scope.
Data Protection & Storage	- Use Management Plane Security to secure your Storage Account using Azure role-based access control (Azure RBAC). - Data Plane Security to Securing Access to your Data using Shared Access Signatures (SAS) and Stored Access Policies. - Use Transport-Level Encryption – Using HTTPS and the encryption used by SMB (Server message block protocols) 3.0 for Azure File Shares. - Use Client-side encryption to secure data that you send to storage accounts when you require sole control of encryption keys.

Checklist Category	Description
	• Use Storage Service Encryption (SSE) to automatically encrypt data in Azure Storage, and Azure Disk Encryption for Linux VMs and Azure Disk Encryption for Windows VMs to encrypt virtual machine disk files for the OS and data disks. • Use Azure Storage Analytics to monitor authorization type; like with Blob Storage, you can see if users have used a Shared Access Signature or the storage account keys. • Use Cross-Origin Resource Sharing (CORS) to access storage resources from different domains.
Security Policies & Recommendations	• Use Microsoft Defender for Cloud to deploy endpoint solutions. • Add a web application firewall (WAF) to secure web applications. • Use Azure Firewall to increase your security protections. • Apply security contact details for your Azure subscription. The Microsoft Security Response Center (MSRC) ↗ contacts you if it discovers that your customer data has been accessed by an unlawful or unauthorized party.
Identity & Access Management	• Synchronize your on-premises directory with your cloud directory using Microsoft Entra ID. • Use single sign-on to enable users to access their SaaS applications based on their organizational account in Azure AD. • Use the Password Reset Registration Activity report to monitor the users that are registering. • Enable multi-factor authentication (MFA) for users. • Developers to use secure identity capabilities for apps like Microsoft Security Development Lifecycle (SDL) ↗. • Actively monitor for suspicious activities by using Microsoft Entra ID P1 or P2 anomaly reports and Microsoft Entra ID Protection capability.
Ongoing Security Monitoring	• Use Malware Assessment Solution Azure Monitor logs to report on the status of antimalware protection in your infrastructure. • Use Update Management to determine the overall exposure to potential security problems, and whether or how critical these updates are for your environment. • The Microsoft Entra admin center ↗ provides visibility into the integrity and security of your organization's directory.
Microsoft Defender for Cloud detection capabilities	• Use Cloud Security Posture Management (CSPM) for hardening guidance that helps you efficiently and effectively improve your security. • Use alerts to be notified when threats are identified in your cloud, hybrid, or on-premises environment.

Checklist Category	Description
	• Use security policies, initiatives, and recommendations to improve your security posture.

Conclusion

Many organizations have successfully deployed and operated their cloud applications on Azure. The checklists provided highlight several checklists that are essential and help you to increase the likelihood of successful deployments and frustration-free operations. We highly recommend these operational and strategic considerations for your existing and new application deployments on Azure.

Next steps

To learn more about security in Azure, see the following articles:

- [Shared responsibility in the cloud.](#)
- [End-to-end security in Azure.](#)
- [Ransomware protection in Azure](#)

Security services and technologies available on Azure

Azure provides comprehensive security services and technologies across all layers of your cloud deployments. This article introduces the main security capabilities organized by domain, with links to detailed overview articles for more information.

For specific security best practices and detailed implementation guidance, refer to the domain-specific overview articles linked throughout this document.

Threat detection and response

 Expand table

Service	Description
Microsoft Defender for Cloud	Cloud workload protection with continuous security assessment, recommendations, and advanced threat detection across Azure, hybrid, and multicloud resources.
Microsoft Sentinel	Cloud-native SIEM and SOAR solution delivering intelligent security analytics, threat intelligence, attack detection, proactive hunting, and automated response.

For comprehensive information about threat detection capabilities and best practices, see [Azure threat protection](#).

Identity and access management

 Expand table

Service	Description
Microsoft Entra ID	Cloud-based identity and access management service supporting single sign-on, multifactor authentication, Conditional Access, and identity protection.
Azure role-based access control	Fine-grained access management enabling you to grant users only the permissions needed to perform their jobs.
Microsoft Entra Privileged Identity Management	Just-in-time privileged access to Azure and Microsoft Entra roles with approval workflows and access reviews.

For detailed identity security capabilities and best practices, see [Azure identity management security overview](#).

Key and secrets management

 Expand table

Service	Description
Azure Key Vault	Secure storage for keys, secrets, and certificates with FIPS 140-2 Level 1 (Standard tier) or FIPS 140-3 Level 3 (Premium tier with HSM) validation.
Azure Key Vault Managed HSM	Single-tenant, FIPS 140-2 Level 3 validated HSM service offering full control with confidential key support.

For comprehensive key management options including Azure Dedicated HSM and Azure Payment HSM, see [Key management in Azure](#).

Data encryption

 Expand table

Service	Description
Azure Storage Service Encryption	Automatic encryption for data at rest in Azure storage using AES 256 encryption.
Azure SQL Database Transparent Data Encryption	Real-time encryption of databases, backups, and transaction logs without application changes.
Azure Disk Encryption	Encryption for OS and data disks of Azure virtual machines using platform-managed or customer-managed keys.

For detailed encryption options and best practices, see [Azure encryption overview](#).

Network security

 Expand table

Service	Description
Azure Firewall	Cloud-native network firewall with threat intelligence, IDPS capabilities (Premium SKU), and TLS inspection.

Service	Description
Azure DDoS Protection	Always-on traffic monitoring and real-time mitigation of network-level DDoS attacks.
Azure Virtual Network	Network isolation with Network Security Groups, service endpoints, and Private Link for secure connectivity.
Azure VPN Gateway	Secure cross-premises connectivity to Azure virtual networks over IPsec/IKE VPN tunnels.
Azure Application Gateway with WAF	Layer 7 load balancing with integrated web application firewall protecting against OWASP Top 10 vulnerabilities.
Azure Front Door	Global HTTP load balancer with integrated WAF, DDoS protection, and SSL/TLS offloading.

For comprehensive network security guidance and best practices, see [Azure network security overview](#).

Monitoring and governance

 [Expand table](#)

Service	Description
Azure Monitor	Comprehensive monitoring solution collecting and analyzing telemetry with Log Analytics workspaces, metrics, alerts, and workbooks.
Azure Policy	Governance service enforcing organizational standards, assessing compliance at scale, and providing automatic remediation.
Microsoft Defender for Cloud regulatory compliance	Built-in and custom compliance assessments aligned with standards like Microsoft cloud security benchmark, ISO 27001, and NIST.

For detailed security management capabilities and best practices, see [Azure security management and monitoring overview](#).

Database security

 [Expand table](#)

Service	Description
Azure SQL Database security	Network access control, authentication, authorization, encryption at rest and in transit, auditing, and threat detection.

Service	Description
Microsoft Defender for SQL	Advanced threat protection detecting vulnerabilities, anomalous activities, and SQL injection attempts.
Azure Cosmos DB security	Encryption at rest and in transit, network isolation, RBAC, and audit logging for NoSQL workloads.

For a comprehensive database security checklist, see [Azure database security checklist](#).

Virtual machine security

 Expand table

Service	Description
Trusted launch	Default for Gen2 VMs providing Secure Boot, vTPM, and Boot Integrity Monitoring to protect against boot kits and rootkits.
Azure confidential computing	Hardware-based trusted execution environments using AMD SEV-SNP for data protection while in use.
Microsoft Defender for Servers	Threat detection and vulnerability management for Windows and Linux virtual machines.

For comprehensive VM security features and guidance, see [Azure Virtual Machines security overview](#).

Platform integrity

 Expand table

Service	Description
Azure platform security	Hardware and firmware security including Project Cerberus, measured boot, and host attestation.
Secure Boot and code integrity	UEFI Secure Boot and code integrity policies protecting Azure infrastructure from malicious code.

For detailed platform security architecture, see [Azure platform integrity and security overview](#).

Backup and disaster recovery

Service	Description
Azure Backup	Independent and isolated backups protecting application data with zero capital investment and built-in management.
Azure Site Recovery	Disaster recovery orchestration for replication, failover, and recovery of workloads to secondary locations or Azure.

PaaS deployment security

For guidance on securing platform-as-a-service deployments, including App Service, Azure Functions, and container services, see [Securing PaaS deployments](#).

Next steps

- [End-to-end security in Azure](#) - Comprehensive overview of Azure's security architecture and capabilities
- [Azure security best practices and patterns](#) - Collection of security best practices for various scenarios
- [Microsoft cloud security benchmark](#) - Comprehensive security guidance for Azure services
- [Shared responsibility in the cloud](#) - Understanding the security responsibilities shared between you and Microsoft

Last updated on 11/07/2025

Cloud feature availability for commercial and US Government customers

Article • 04/30/2025

This article describes feature availability in the Microsoft Azure and Azure Government clouds. Features are listed as **GA** (Generally Available), **Public Preview**, or **Not Available** for the following security services:

- [Azure Information Protection](#)
- [Microsoft Defender for Cloud](#)
- [Microsoft Sentinel](#)
- [Microsoft Defender for IoT](#)
- [Azure Attestation](#)

ⓘ Note

Additional security services will be added to this article soon.

Azure Government

Azure Government uses the same underlying technologies as Azure (sometimes referred to as Azure Commercial or Azure Public), which includes the core components of Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). Both Azure and Azure Government have comprehensive security controls in place, and the Microsoft commitment on the safeguarding of customer data.

Azure Government is a physically isolated cloud environment dedicated to US federal, state, local, and tribal governments, and their partners. Whereas both cloud environments are assessed and authorized at the FedRAMP High impact level, Azure Government provides an extra layer of protection to customers through contractual commitments regarding storage of customer data in the United States and limiting potential access to systems processing customer data to screened US persons. These commitments may be of interest to customers using the cloud to store or process data subject to US export control regulations such as the EAR, ITAR, and DoE 10 CFR Part 810.

For more information about Azure Government, see [What is Azure Government?](#)

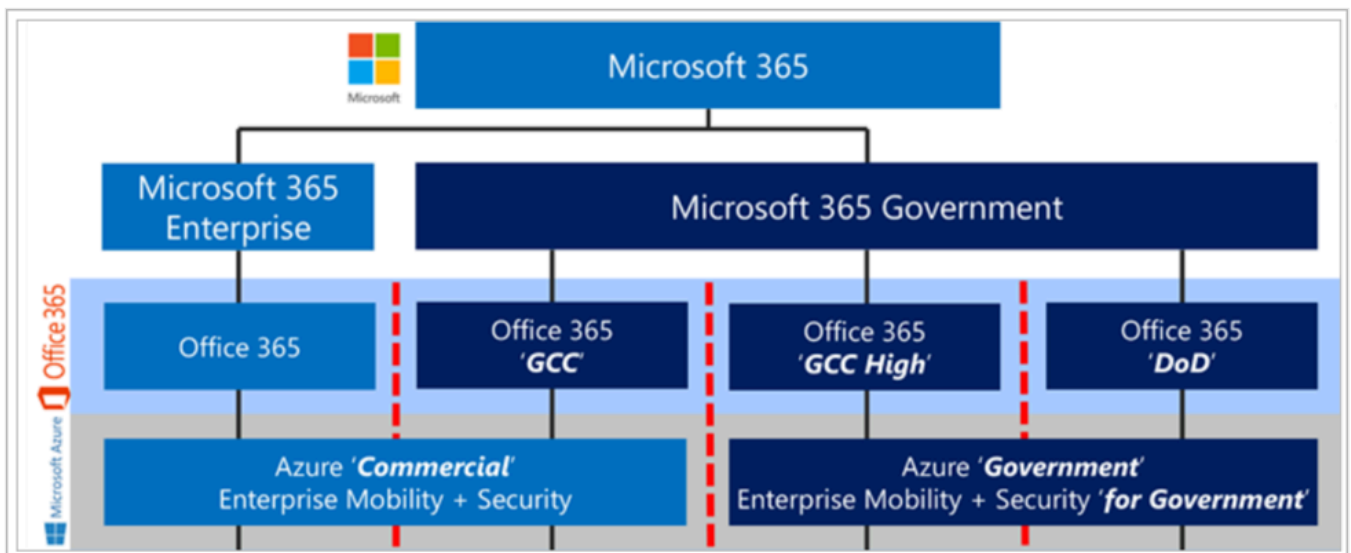
ⓘ Note

These lists and tables do not include feature or bundle availability in the Azure Government Secret or Azure Government Top Secret clouds. For more information about specific availability for air-gapped clouds, please contact your account team.

Microsoft 365 integration

Integrations between products rely on interoperability between Azure and Office platforms. Offerings hosted in the Azure environment are accessible from the Microsoft 365 Enterprise and Microsoft 365 Government platforms. Office 365 and Office 365 GCC are paired with Microsoft Entra ID in Azure. Office 365 GCC High and Office 365 DoD are paired with Microsoft Entra ID in Azure Government.

The following diagram displays the hierarchy of Microsoft clouds and how they relate to each other.



The Office 365 GCC environment helps customers comply with US government requirements, including FedRAMP High, CJIS, and IRS 1075. The Office 365 GCC High and DoD environments support customers who need compliance with DoD IL4/5, DFARS 7012, NIST 800-171, and ITAR.

For more information about Office 365 US Government environments, see:

- [Office 365 GCC](#)
- [Office 365 GCC High and DoD](#)

The following sections identify when a service has an integration with Microsoft 365 and the feature availability for Office 365 GCC, Office 365 High, and Office 365 DoD.

Azure Information Protection

Azure Information Protection (AIP) is a cloud-based solution that enables organizations to discover, classify, and protect documents and emails by applying labels to content.

AIP is part of the Microsoft Purview Information Protection (MIP) solution, and extends the [labeling](#) and [classification](#) functionality provided by Microsoft 365.

For more information, see the [Azure Information Protection product documentation](#).

- Office 365 GCC is paired with Microsoft Entra ID in Azure. Office 365 GCC High and Office 365 DoD are paired with Microsoft Entra ID in Azure Government. Make sure to pay attention to the Azure environment to understand where [interoperability is possible](#). In the following table, interoperability that is *not* possible is marked with a dash (-) to indicate that support is not relevant.
- Extra configurations are required for GCC-High and DoD customers. For more information, see [Azure Information Protection Premium Government Service Description](#).


Note

More details about support for government customers are listed in footnotes below the table.

Extra steps are required for configuring Azure Information Protection for GCC High and DoD customers. For more information, see the [Azure Information Protection Premium Government Service Description](#).


Expand table

Feature/Service	Azure	Azure Government
Azure Information Protection scanner ¹		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA
Administration		
Azure Information Protection portal for scanner administration		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA

Feature/Service	Azure	Azure Government
- Office 365 DoD	-	GA
Classification and labeling ²		
AIP scanner to apply a <i>default label</i> to all files in an on-premises file server / repository		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA
AIP scanner for automated classification, labeling, and protection of supported on-premises files		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA

¹ The scanner can function without Office 365 to scan files only. The scanner cannot apply labels to files without Office 365.

² The classification and labeling add-in is only supported for government customers with Microsoft 365 Apps (version 9126.1001 or higher), including Professional Plus (ProPlus) and Click-to-Run (C2R) versions. Office 2010, Office 2013, and other Office 2016 versions are not supported.

Office 365 features

 Expand table

Feature/Service	Office 365 GCC	Office 365 GCC High	Office 365 DoD
Administration			
- PowerShell for RMS service administration	GA	GA	GA
- PowerShell for AIP UL client bulk operations			
SDK			

Feature/Service	Office 365 GCC	Office 365 GCC High	Office 365 DoD
- MIP and AIP Software Development Kit (SDK)	GA	GA	GA
Customizations			
- Document tracking and revocation	GA	Not available	Not available
Key management			
- Bring Your Own Key (BYOK)	GA	GA	GA
- Double Key Encryption (DKE)	GA	GA	GA
Office files ³			
- Protection for Microsoft Exchange Online, Microsoft SharePoint Online, and Microsoft OneDrive for Business	GA	GA ⁴	GA ⁴
- Protection for on-premises Exchange and SharePoint content via the Rights Management connector	GA ⁵	GA ⁶	GA ⁶
- Office 365 Message Encryption	GA	GA	GA
- Set labels to automatically apply pre-configured M/MIME protection in Outlook	GA	GA	GA
- Control oversharing of information when using Outlook	GA	GA ⁷	GA ⁷
Classification and labeling ^{2/8}			
- Custom templates, including departmental templates	GA	GA	GA
- Manual, default, and mandatory document classification	GA	GA	GA
- Configure conditions for automatic and recommended classification GA	GA	GA	
- Protection for non-Microsoft Office file formats, including PTXT, PJPG, and PFILE (generic protection)	GA	GA	GA

³ The Mobile Device Extension for AD RMS is currently not available for government customers.

⁴ Information Rights Management with SharePoint Online (IRM-protected sites and libraries) is currently not available.

⁵ Information Rights Management (IRM) is supported only for Microsoft 365 Apps (version 9126.1001 or higher), including Professional Plus (ProPlus) and Click-to-Run (C2R) versions.

Office 2010, Office 2013, and other Office 2016 versions are not supported.

⁶ Only on-premises Exchange is supported. Outlook Protection Rules are not supported. File Classification Infrastructure is not supported. On-premises SharePoint is not supported.

⁷ Sharing of protected documents and emails from government clouds to users in the commercial cloud is not currently available. Includes Microsoft 365 Apps users in the commercial cloud, non-Microsoft 365 Apps users in the commercial cloud, and users with an RMS for Individuals license.

⁸ The number of [Sensitive Information Types](#) in your Microsoft Purview compliance portal may vary based on region.

Microsoft Defender for Cloud

Microsoft Defender for Cloud is a unified infrastructure security management system that strengthens the security posture of your data centers, and provides advanced threat protection across your hybrid workloads in the cloud - whether they're in Azure or not - as well as on premises.

For more information, see the [Microsoft Defender for Cloud product documentation](#).

The following table displays the current Defender for Cloud feature availability in Azure and Azure Government.

 Expand table

Feature/Service	Azure	Azure Government
Microsoft Defender for Cloud free features		
• Continuous export	GA	GA
• Workflow automation	GA	GA
• Recommendation exemption rules	Public Preview	Not Available
• Alert suppression rules	GA	GA
• Email notifications for security alerts	GA	GA
• Auto provisioning for agents and extensions	GA	GA
• Asset inventory	GA	GA
• Azure Monitor Workbooks reports in Microsoft Defender for Cloud's workbooks gallery	GA	GA

Feature/Service	Azure	Azure Government
Microsoft Defender plans and extensions		
• Microsoft Defender for servers	GA	GA
• Microsoft Defender AI Services	GA	Not Available
• Microsoft Defender for App Service	GA	Not Available
• Microsoft Defender for DNS	Not available for new subscriptions	Not available for new subscriptions
• Microsoft Defender for Containers ⁹	GA	GA
• Microsoft Defender for container registries ¹ (deprecated)	GA	GA ²
• Microsoft Defender for container registries scanning of images in CI/CD workflows ³	Public Preview	Not Available
• Microsoft Defender for Kubernetes ⁴ (deprecated)	GA	GA
• Defender extension for Arc-enabled Kubernetes, Servers, or Data services ⁵	Public Preview	Not Available
• Microsoft Defender for Azure SQL database servers	GA	GA
• Microsoft Defender for SQL servers on machines	GA	GA
• Microsoft Defender for open-source relational databases	GA	Not Available
• Microsoft Defender for Key Vault	GA	Not Available
• Microsoft Defender for Resource Manager	GA	GA
• Microsoft Defender for Storage ⁶	GA	GA (activity monitoring)
• Microsoft Defender for Azure Cosmos DB	GA	Not Available
• Kubernetes workload protection	GA	GA
• Bi-directional alert synchronization with Microsoft Sentinel	GA	GA
Microsoft Defender for servers features ⁷		
• Just-in-time VM access	GA	GA
• File integrity monitoring	GA	GA

Feature/Service	Azure	Azure Government
• Adaptive application controls	GA	GA
• Adaptive network hardening	GA	Not Available
• Docker host hardening	GA	GA
• Integrated vulnerability assessment for machines	GA	Not Available
• Regulatory compliance dashboard & reports ⁸	GA	GA
• Microsoft Defender for Endpoint deployment and integrated license	GA	GA
• Connect AWS account	GA	Not Available
• Connect GCP account	GA	Not Available

¹ Partially GA: The ability to disable specific findings from vulnerability scans is in public preview.

² Vulnerability scans of container registries on Azure Gov can only be performed with the scan on push feature.

³ Requires Microsoft Defender for container registries.

⁴ Partially GA: Support for Azure Arc-enabled clusters is in public preview and not available on Azure Government.

⁵ Requires Microsoft Defender for Kubernetes.

⁶ Partially GA: Some of the threat protection alerts from Microsoft Defender for Storage are in public preview.

⁷ These features all require [Microsoft Defender for servers](#).

⁸ There may be differences in the standards offered per cloud type.

⁹ Partially GA: Support for Arc-enabled Kubernetes clusters (and therefore AWS EKS too) is in public preview and not available on Azure Government. Run-time visibility of vulnerabilities in container images is also a preview feature.

Microsoft Sentinel

Microsoft Sentinel is a scalable, cloud-native, security information event management (SIEM), and security orchestration automated response (SOAR) solution. Microsoft Sentinel delivers

intelligent security analytics and threat intelligence across the enterprise, providing a single solution for alert detection, threat visibility, proactive hunting, and threat response.

For more information, see the [Microsoft Sentinel product documentation](#).

For Microsoft Sentinel feature availability in Azure, Azure Government, and Azure China 21 Vianet, see [Microsoft Sentinel feature support for Azure clouds](#).

Microsoft Purview Data Connectors

Office 365 GCC is paired with Microsoft Entra ID in Azure. Office 365 GCC High and Office 365 DoD are paired with Microsoft Entra ID in Azure Government.

 **Tip**

Make sure to pay attention to the Azure environment to understand where [interoperability is possible](#). In the following table, interoperability that is *not* possible is marked with a dash (-) to indicate that support is not relevant.

 Expand table

Connector	Azure	Azure Government
Office IRM		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
Dynamics 365		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
Microsoft Defender XDR		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Public Preview
- Office 365 DoD	-	Public Preview

Connector	Azure	Azure Government
Microsoft Defender for Cloud Apps		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA
Microsoft Defender for Cloud Apps Shadow IT logs		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Public Preview
- Office 365 DoD	-	Public Preview
Microsoft Defender for Cloud Apps Alerts		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Public Preview
- Office 365 DoD	-	Public Preview
Microsoft Defender for Endpoint		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA
Microsoft Defender for Identity		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
Microsoft Defender for Office 365		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
- Microsoft Power BI		

Connector	Azure	Azure Government
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
- Microsoft Project		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available
Office 365		
- Office 365 GCC	GA	-
- Office 365 GCC High	-	GA
- Office 365 DoD	-	GA
Teams 		
- Office 365 GCC	Public Preview	-
- Office 365 GCC High	-	Not Available
- Office 365 DoD	-	Not Available

Microsoft Defender for IoT

Microsoft Defender for IoT lets you accelerate IoT/OT innovation with comprehensive security across all your IoT/OT devices. For end-user organizations, Microsoft Defender for IoT offers agentless, network-layer security that is rapidly deployed, works with diverse industrial equipment, and interoperates with Microsoft Sentinel and other SOC tools. Deploy on-premises or in Azure-connected environments. For IoT device builders, the Microsoft Defender for IoT security agents allow you to build security directly into your new IoT devices and Azure IoT projects. The micro agent has flexible deployment options, including the ability to deploy as a binary package or modify source code. And the micro agent is available for standard IoT operating systems like Linux and Azure RTOS. For more information, see the [Microsoft Defender for IoT product documentation](#).

The following table displays the current Microsoft Defender for IoT feature availability in Azure, and Azure Government.

For organizations

 Expand table

Feature	Azure	Azure Government
On-premises device discovery and inventory	GA	GA
Vulnerability management	GA	GA
Threat detection with IoT, and OT behavioral analytics	GA	GA
Manual and automatic threat intelligence updates	GA	GA
Unify IT, and OT security with SIEM, SOAR and XDR		
Active Directory	GA	GA
ArcSight	GA	GA
ClearPass (Alerts & Inventory)	GA	GA
CyberArk PSM	GA	GA
Email	GA	GA
FortiGate	GA	GA
FortiSIEM	GA	GA
Microsoft Sentinel	GA	GA
NetWitness	GA	GA
Palo Alto NGFW	GA	GA
Palo Alto Panorama	GA	GA
ServiceNow (Alerts & Inventory)	GA	GA
SNMP MIB Monitoring	GA	GA
Splunk	GA	GA
SYSLOG Server (CEF format)	GA	GA
SYSLOG Server (LEEF format)	GA	GA
SYSLOG Server (Object)	GA	GA
SYSLOG Server (Text Message)	GA	GA

For device builders

 Expand table

Feature	Azure	Azure Government
Micro agent for Azure RTOS	GA	GA
Configure Sentinel with Microsoft Defender for IoT	GA	GA
Standalone micro agent for Linux		
Standalone agent binary installation	Public Preview	Public Preview

Azure Attestation

Microsoft Azure Attestation is a unified solution for remotely verifying the trustworthiness of a platform and integrity of the binaries running inside it. The service receives evidence from the platform, validates it with security standards, evaluates it against configurable policies, and produces an attestation token for claims-based applications (e.g., relying parties, auditing authorities).

Azure Attestation is currently available in multiple regions across Azure public and Government clouds. In Azure Government, the service is available in preview status across US Gov Virginia and US Gov Arizona.

For more information, see Azure Attestation [public documentation](#).

 Expand table

Feature	Azure	Azure Government
Portal experience to perform control-plane and data-plane operations	GA	-
PowerShell experience to perform control-plane and data-plane operations	GA	GA
TLS 1.2 enforcement	GA	GA
BCDR support	GA	-
Service tag integration	GA	GA
Immutable log storage	GA	GA

Feature	Azure	Azure Government
Network isolation using private link	Public Preview	-
FedRAMP High certification	GA	-
Customer lockbox	GA	-

Next steps

- Understand the [shared responsibility](#) model and which security tasks are handled by the cloud provider and which tasks are handled by you.
- Understand the [Azure Government Cloud](#) capabilities and the trustworthy design and security used to support compliance applicable to federal, state, and local government organizations and their partners.
- Understand the [Office 365 Government plan](#).
- Understand [compliance in Azure](#) for legal and regulatory standards.

Azure security best practices and patterns

This article contains security best practices to use when you're designing, deploying, and managing your cloud solutions by using Azure. These best practices come from our experience with Azure security and the experiences of customers like you.

Best practices

These best practices are intended to be a resource for IT pros. IT pros include designers, architects, developers, and testers who build and deploy secure Azure solutions.

- [Best practices for protecting secrets](#)
- [Azure database security best practices](#)
- [Azure data security and encryption best practices](#)
- [Azure identity management and access control security best practices](#)
- [Azure network security best practices](#)
- [Azure operational security best practices](#)
- [Azure PaaS Best Practices](#)
- [Azure Service Fabric security best practices](#)
- [Best practices for IaaS workloads in Azure](#)
- [Implementing a secure hybrid network architecture in Azure](#)
- [Internet of Things security best practices](#)
- [Securing PaaS databases in Azure](#)
- [Securing PaaS web and mobile applications using Azure App Service](#)
- [Securing PaaS web and mobile applications using Azure Storage](#)

Next steps

Microsoft finds that using security benchmarks can help you quickly secure cloud deployments. Benchmark recommendations from your cloud service provider give you a starting point for selecting specific security configuration settings in your environment and allow you to quickly reduce risk to your organization.

Microsoft Cloud Security Benchmark (MCSB)

The [Microsoft cloud security benchmark \(MCSB\)](#) provides comprehensive security best practices aligned with industry frameworks spanning identity, networking, compute, data protection, and management layers.

Microsoft Cloud Security Benchmark v2 (Preview): MCSB v2 is now available in preview with significant enhancements including:

- **Artificial Intelligence Security:** New control domain with 7 recommendations covering AI platform security, AI application security, and AI security monitoring to address threats and risks in AI deployments
- **Expanded Azure Policy Coverage:** Increased from 220+ to 420+ policy-based control measurements for comprehensive security posture monitoring
- **Enhanced Implementation Guidance:** More granular technical implementation examples with risk and threat-based control guides

MCSB v2 includes new guidance for confidential computing workloads and can be enforced and monitored through Azure Policy. For more information, see [Overview of Microsoft cloud security benchmark v2 \(preview\)](#).

Implementation recommendations:

- **Monitor compliance:** Use the [Microsoft Defender for Cloud regulatory compliance dashboard](#) to track Microsoft Cloud Security Benchmark compliance and identify security gaps
- **Enforce baselines:** Implement [Azure Policy](#) to audit and enforce secure configuration baselines based on Microsoft Cloud Security Benchmark v2 (preview) recommendations
- **Assess AI workloads:** Review the new Artificial Intelligence Security controls in Microsoft Cloud Security Benchmark v2 (preview) if deploying AI/ML workloads to ensure proper platform, application, and monitoring security

For a complete collection of high-impact security recommendations, see the [Microsoft cloud security benchmark](#).

Microsoft Services in Cybersecurity

Article • 04/23/2025

Microsoft Services provides a comprehensive approach to security, identity, and cybersecurity. They include an array of Security and Identity services across strategy, planning, implementation, and ongoing support. These services can help Enterprise customers implement security solutions that align with their strategic goals.

Microsoft services can create solutions that integrate, and enhance the latest security and identity capabilities of our products to help protect your business and drive innovation.

Our team of technical professionals consists of highly trained experts who offer a wealth of security and identity experience.

[Learn more](#)  about Microsoft Services Security consulting services.

Log a security issue

Visit the [Microsoft Security Response Center \(MSRC\)](#) to report a security specific issue.

You can also create a tailored, Azure support request in the Azure portal. Visit the Azure portal [here](#). Follow the prompts to receive recommended solutions or to log a support request.

Next steps

[MSRC](#) is part of the security community. Learn how MSRC helps to protect customers and the broader ecosystem.

Last updated on 12/03/2025

Penetration testing

One of the benefits of using Azure for application testing and deployment is that you can quickly get environments created. You don't have to worry about requisitioning, acquiring, and "racking and stacking" your own on-premises hardware.

Quickly creating environments is great but you still need to make sure you perform your normal security due diligence. One of the things you likely want to do is penetration test the applications you deploy in Azure. We don't perform penetration testing of your application for you, but we do understand that you want and need to perform testing on your own applications. That's a good thing, because when you enhance the security of your applications you help make the entire Azure ecosystem more secure.

As of June 15, 2017, Microsoft no longer requires pre-approval to conduct a penetration test against Azure resources. This process is only related to Microsoft Azure, and not applicable to any other Microsoft Cloud Service.

Important

While notifying Microsoft of pen testing activities is no longer required customers must still comply with the [Microsoft Cloud Unified Penetration Testing Rules of Engagement](#) .

Permitted testing

You can perform penetration testing on your own Azure-hosted applications and services without prior approval. This includes testing:

- Your endpoints hosted on Azure Virtual Machines
- Azure App Service applications (Web Apps, API Apps, Mobile Apps)
- Azure Functions and API endpoints
- Azure Websites
- Any other Azure services where you own or have explicit authorization to test the deployed resources

Standard tests you can perform include:

- Tests on your endpoints to uncover the [Open Web Application Security Project \(OWASP\) top 10 vulnerabilities](#) 
- Dynamic Application Security Testing (DAST) of your web applications and APIs
- [Fuzz testing](#)  of your endpoints
- [Port scanning](#)  of your endpoints

Prohibited testing

One type of pen test that you can't perform is any kind of [Denial of Service \(DoS\)](#) attack. This test includes initiating a DoS attack itself, or performing related tests that might determine, demonstrate, or simulate any type of DoS attack.

DDoS simulation testing

If you need to test your DDoS resilience, you can use Microsoft-approved simulation partners. These partners provide controlled DDoS simulation services that don't violate the penetration testing rules:

- [BreakingPoint Cloud](#): A self-service traffic generator where your customers can generate traffic against DDoS Protection-enabled public endpoints for simulations.
- [Red Button](#): Work with a dedicated team of experts to simulate real-world DDoS attack scenarios in a controlled environment.
- [RedWolf](#): A self-service or guided DDoS testing provider with real-time control.

To learn more about these simulation partners, see [testing with simulation partners](#).

Next steps

- Learn more about the [Penetration Testing Rules of Engagement](#).

Last updated on 01/07/2026

Reference list of Azure domains (not comprehensive)

This page is a partial list of the Azure domains in use. Some of them are REST API endpoints.

 Expand table

Service	Subdomain
Azure Access Control Service  (retired)	*.accesscontrol.windows.net
Microsoft Entra ID	*.graph.windows.net / *.onmicrosoft.com
Azure API Management 	*.azure-api.net
Azure BizTalk Services  (retired)	*.biztalk.windows.net
Azure Blob storage	*.blob.core.windows.net
Azure Cloud Services and Azure Virtual Machines	*.cloudapp.net
Azure Cloud Services and Azure Virtual Machines	*.cloudapp.azure.com
Azure Container Registry 	*.azurecr.io
Azure Container Service (deprecated)	*.azurecontainer.io
Azure Content Delivery Network (CDN) 	*.vo.msecnd.net
Azure Cosmos DB	*.cosmos.azure.com
Azure Cosmos DB	*.documents.azure.com
Azure Files	*.file.core.windows.net
Azure Front Door (classic)	*.azurefd.net
Azure Front Door Standard/Premium	*.z01.azurefd.net
Azure Key Vault	*.vault.azure.net
Azure Kubernetes Service	*.azmk8s.io
Azure Management Services	*.management.core.windows.net
Azure Media Services 	*.origin.mediaservices.windows.net
Azure Mobile Apps 	*.azure-mobile.net
Azure Queue Storage 	*.queue.core.windows.net

Service	Subdomain
Azure Service Bus	*.servicebus.windows.net
Azure SQL Database ↗	*.database.windows.net
Azure CDN (migrated to Azure Front Door)	*.azureedge.net
Azure Table Storage	*.table.core.windows.net
Azure Traffic Manager	*.trafficmanager.net
Azure Websites	*.azurewebsites.net
GitHub Codespaces ↗	*.visualstudio.com

Last updated on 01/07/2026