
Microsoft Azure Fundamentals (AZ-900)

Exam Guide

Azure Fundamentals – Guide

Summary

Basics

Candidates for the Azure Fundamentals certification should have foundational knowledge of cloud services and how those services are provided with Microsoft Azure. This certification is intended for candidates who are just beginning to work with cloud-based solutions and services or are new to Azure. Azure Fundamentals can be used to prepare for other Azure role-based or specialty certifications, but it is not a prerequisite for any of them.

Job Role

Administrator, Business User, Developer, Student, Technology Manager

Required Exams

AZ-900 Exam - Microsoft Azure Fundamentals (£69 GBP)

This exam measures your ability to describe the following concepts: cloud concepts; core Azure services; core solutions and management tools on Azure; general security and network security features; identity, governance, privacy, and compliance features; and Azure cost management and Service Level Agreements.

Domain Area	
cloud concepts	
core Azure services	
core solutions and management tools on Azure	
general security and network security features	
identity, governance, privacy, and compliance features	
Azure cost management and Service Level Agreements	

(<https://docs.microsoft.com/en-us/learn/certifications/azure-fundamentals>)

Azure Basics

What is Cloud Computing?

Cloud computing is the delivery of computing services over the internet, these services include servers, storage, databases, networking, software, analytics, and intelligence. Cloud computing offers faster innovation, flexible resources, and economies of scale.

Cloud computing is the delivery of computing services over the internet by using a pay-as-you-go pricing model. Cloud computing is typically less expensive than alternatives. You typically pay only for the cloud services you use, which helps you:

- Lower your operating costs.
- Run your infrastructure more efficiently.
- Scale as your business needs change.

Instead of maintaining CPUs and storage in your datacenter, you rent them for the time that you need them. The cloud provider takes care of maintaining the underlying infrastructure for you. The cloud enables you to quickly solve your toughest business challenges, and bring cutting-edge solutions to your users.

What is Azure?

Azure is a continually expanding set of cloud services that help your organization meet your current and future business challenges. Azure gives you the freedom to build, manage, and deploy applications on a massive global network using your favorite tools and frameworks.

What is the Azure portal?

The Azure portal is a web-based, unified console that provides an alternative to command-line tools. With the Azure portal, you can manage your Azure subscription by using a graphical user interface. You can:

Build, manage, and monitor everything from simple web apps to complex cloud deployments.

Create custom dashboards for an organized view of resources.

Configure accessibility options for an optimal experience.

What is Azure Marketplace?

Azure Marketplace helps connect users with Microsoft partners, independent software vendors, and startups that are offering their solutions and services, which are optimized to run on Azure. Azure Marketplace customers can find, try, purchase, and provision applications and services from hundreds of leading service providers. All solutions and services are certified to run on Azure.

The solution catalog spans several industry categories such as open-source container platforms, virtual machine images, databases, application build and deployment software, developer tools, threat detection, and blockchain. Using Azure Marketplace, you can provision end-to-end solutions quickly and reliably, hosted in your own Azure environment. At the time of writing, there are more than 8,000 listings.

Cloud Computing Advantages

There are several advantages that a cloud environment has over a physical environment that Tailwind Traders can use following its migration to Azure.

High availability: Depending on the service-level agreement (SLA) that you choose, your cloud-based apps can provide a continuous user experience with no apparent downtime, even when things go wrong.

Scalability: Apps in the cloud can scale vertically and horizontally:

- Scale vertically to increase compute capacity by adding RAM or CPUs to a virtual machine.
- Scaling horizontally increases compute capacity by adding instances of resources, such as adding VMs to the configuration.

Elasticity: You can configure cloud-based apps to take advantage of autoscaling, so your apps always have the resources they need.

Agility: Deploy and configure cloud-based resources quickly as your app requirements change.

Geo-distribution: You can deploy apps and data to regional datacenters around the globe, thereby ensuring that your customers always have the best performance in their region.

Disaster recovery: By taking advantage of cloud-based backup services, data replication, and geo-distribution, you can deploy your apps with the confidence that comes from knowing that your data is safe in the event of disaster.

Consumption-based Model

Cloud service providers operate on a consumption-based model, which means that end users only pay for the resources that they use. Whatever they use is what they pay for.

A consumption-based model has many benefits, including:

- No upfront costs.
- No need to purchase and manage costly infrastructure that users might not use to its fullest.
- The ability to pay for additional resources when they are needed.

The ability to stop paying for resources that are no longer needed.

There are two different types of expenses that you should consider for IT resources:

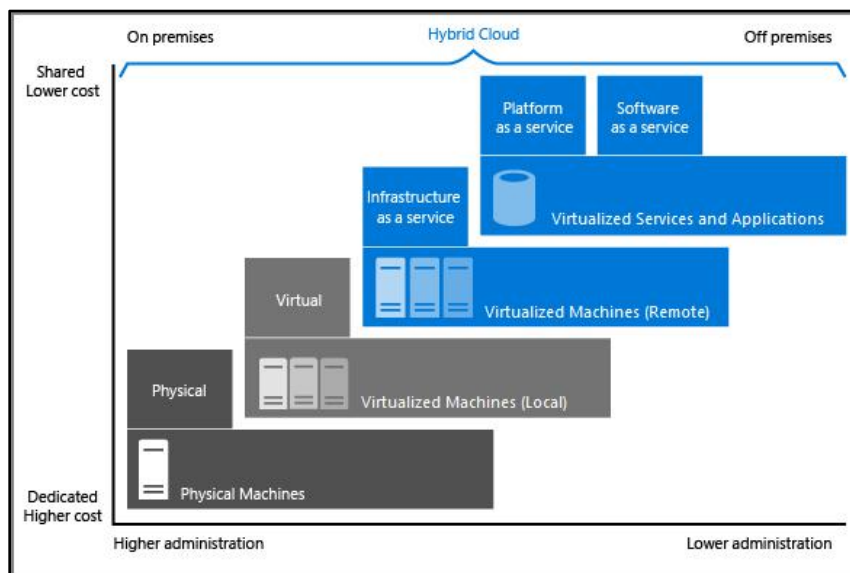
Capital Expenditure (CapEx) is the up-front spending of money on physical infrastructure, and then deducting that up-front expense over time. The up-front cost from CapEx has a value that reduces over time.

Operational Expenditure (OpEx) is spending money on services or products now, and being billed for them now. You can deduct this expense in the same year you spend it. There is no up-front cost, as you pay for a service or product as you use it.

In traditional premises-based IT infrastructure, an IT company for example, buys IT equipment that goes onto its balance sheets as assets. Because a capital investment was made, accountants categorize this transaction as a CapEx. Over time, to account for the assets' limited useful lifespan, assets are depreciated or amortized.

Cloud services, on the other hand, are categorized as an OpEx, because of their consumption model. There's no asset for the IT company to amortize, and its cloud service provider (Azure) manages the costs that are associated with the purchase and lifespan of the physical equipment. As a result, OpEx has a direct impact on net profit, taxable income, and the associated expenses on the balance sheet.

To summarize, CapEx requires significant up-front financial costs, as well as ongoing maintenance and support expenditures. By contrast, OpEx is a consumption-based model, so the IT company is only responsible for the cost of the computing resources that it uses.



Cloud Service Models

Cloud computing falls into one of the following computing models. If you've been around cloud computing for a while, you've probably seen the terms infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS) for the different cloud service models. These models define the different level of shared responsibility that a cloud provider and cloud tenant are responsible for.

IaaS

This cloud service model is the closest to managing physical servers. A cloud provider keeps the hardware up to date, but operating system maintenance and network configuration is left to the cloud tenant. For example, Azure virtual machines are fully operational virtual compute devices running in Microsoft's datacenters. An advantage of this cloud service model is rapid deployment of new compute devices. Setting up a new virtual machine is considerably faster than procuring, installing, and configuring a physical server.

IaaS is the most flexible category of cloud services. It aims to give you complete control over the hardware that runs your application. Instead of buying hardware, with IaaS, you rent it.

Advantages:

- No CapEx. Users have no up-front costs.
- Agility. Applications can be made accessible quickly, and deprovisioned whenever needed.
- Management. The shared responsibility model applies; the user manages and maintains the services they have provisioned, and the cloud provider manages and maintains the cloud infrastructure.
- Consumption-based model. Organizations pay only for what they use and operate under an Operational Expenditure (OpEx) model.
- Skills. No deep technical skills are required to deploy, use, and gain the benefits of a public cloud. Organizations can use the skills and expertise of the cloud provider to ensure workloads are secure, safe, and highly available.
- Cloud benefits. Organizations can use the skills and expertise of the cloud provider to ensure workloads are made secure and highly available.
- Flexibility. IaaS is the most flexible cloud service because you have control to configure and manage the hardware running your application.

PaaS

This cloud service model is a managed hosting environment. The cloud provider manages the virtual machines and networking resources, and the cloud tenant deploys their applications into the managed hosting environment. For example, Azure App Services provides a managed hosting environment where developers can upload their web applications without having to deal with the physical hardware and software requirements.

PaaS provides the same benefits and considerations as IaaS, but there are some additional benefits to be aware of.

Advantages

- **No CapEx.** Users have no up-front costs.
- **Agility.** PaaS is more agile than IaaS, and users don't need to configure servers for running applications.
- **Consumption-based model.** Users pay only for what they use, and operate under an OpEx model.
- **Skills.** No deep technical skills are required to deploy, use, and gain the benefits of PaaS.
- **Cloud benefits.** Users can take advantage of the skills and expertise of the cloud provider to ensure that their workloads are made secure and highly available. In addition, users can gain access to more cutting-edge development tools. They can then apply these tools across an application's lifecycle.
- **Productivity.** Users can focus on application development only, because the cloud provider handles all platform management. Working with distributed teams as services is easier because the platform is accessed over the internet. You can make the platform available globally more easily.

Disadvantages

- **Platform limitations.** There can be some limitations to a cloud platform that might affect how an application runs. When you're evaluating which PaaS platform is best suited for a workload, be sure to consider any limitations in this area.

SaaS

In this cloud service model, the cloud provider manages all aspects of the application environment, such as virtual machines, networking resources, data storage, and applications. The cloud tenant only needs to provide their data to the application managed by the cloud provider. For example, Office 365 provides a fully working version of Office that runs in the cloud. All that you need to do is create your content, and Office 365 takes care of everything else.

SaaS is software that's centrally hosted and managed for you and your users or customers. Usually one version of the application is used for all customers, and it's licensed through a monthly or annual subscription.

SaaS provides the same benefits as IaaS, but again there are some additional benefits to be aware of too.

Advantages

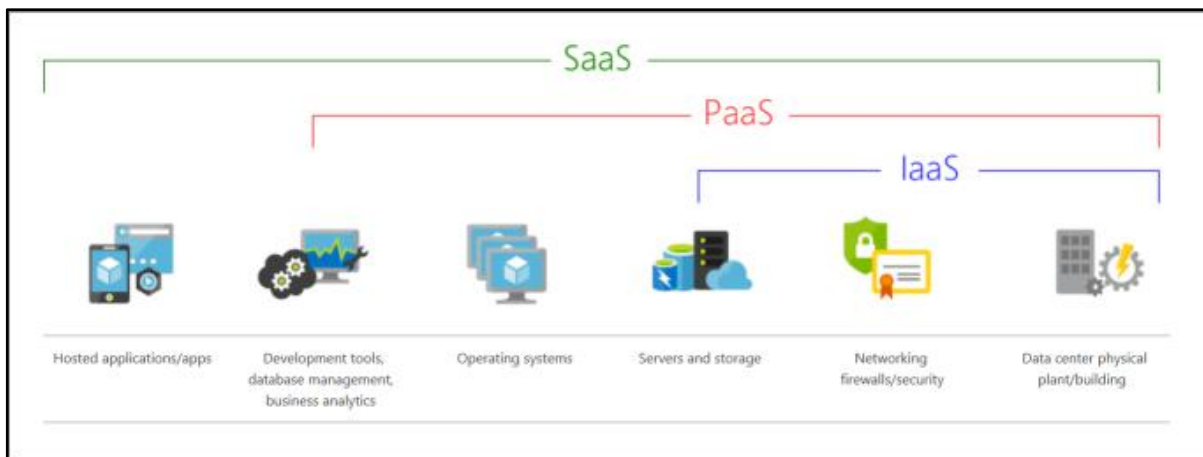
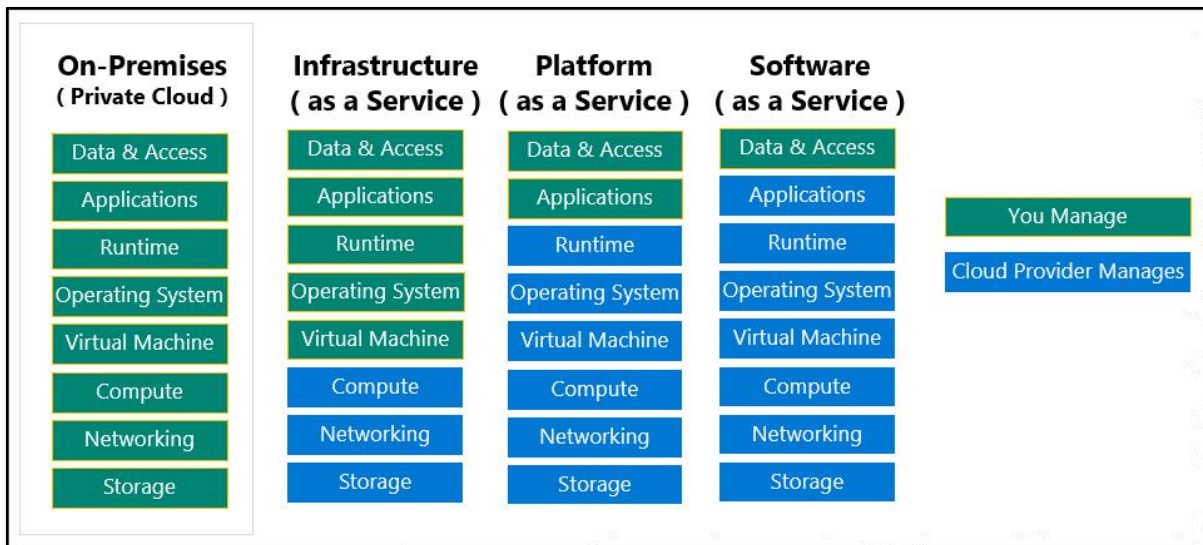
- **No CapEx.** Users have no up-front costs.
- **Agility.** Users can provide staff with access to the latest software quickly and easily.
- **Pay-as-you-go pricing model.** Users pay for the software they use on a subscription model, typically monthly or yearly, regardless of how much they use the software.
- **Skills.** No deep technical skills are required to deploy, use, and gain the benefits of SaaS.
- **Flexibility.** Users can access the same application data from anywhere.

Disadvantages

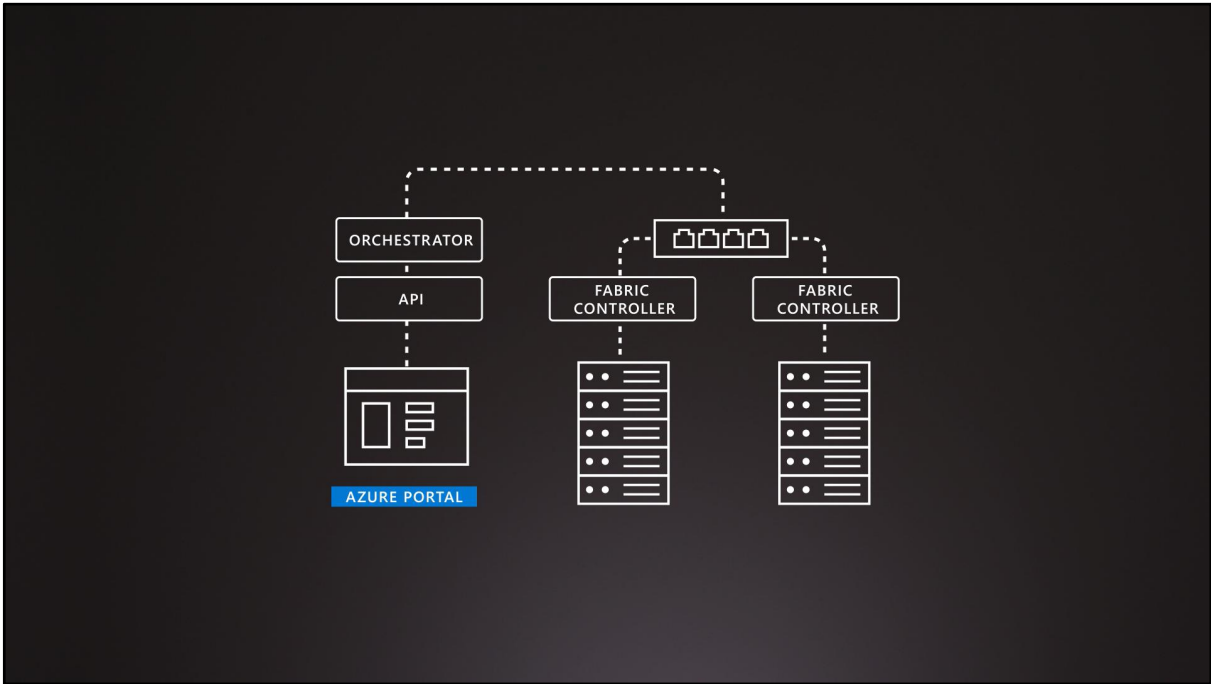
- **Software limitations.** There can be some limitations to a software application that might affect how users work. Because you're using as-is software, you don't have direct control of features. When you're evaluating which SaaS platform is best suited for a workload, be sure to consider any business needs and software limitations.

IaaS, PaaS & SaaS Compared

The following diagrams show how the cloud service models compare to each other:



Azure Infrastructure Overview



What is Serverless Computing?

Overlapping with PaaS, serverless computing enables developers to build applications faster by eliminating the need for them to manage infrastructure. Serverless computing is the abstraction of servers, infrastructure, and operating systems. With serverless applications, the cloud service provider automatically provisions, scales, and manages the infrastructure required to run the code. Serverless architectures are highly scalable and event-driven. They use resources only when a specific function or trigger occurs. You're billed only for the exact resources you use. There's no need to even reserve capacity.

In understanding the definition of serverless computing, it's important to note that servers are still running the code. The serverless name comes from the fact that the tasks associated with infrastructure provisioning and management are invisible to the developer. This approach enables developers to increase their focus on the business logic and deliver more value to the core of the business. Serverless computing helps teams increase their productivity and bring products to market faster. It allows organizations to better optimize resources and stay focused on innovation.

Serverless computing includes the abstraction of servers, an event-driven scale, and micro-billing:

- **Abstraction of servers:** Serverless computing abstracts the servers you run on. You never explicitly reserve server instances. The platform manages that for you. Each function execution can run on a different compute instance. This execution context is transparent to the code. With serverless architecture, you deploy your code, which then runs with high availability.
- **Event-driven scale:** Serverless computing is an excellent fit for workloads that respond to incoming events. Instead of writing an entire application, the developer authors a function, which contains both code and metadata about its triggers and bindings. The platform automatically schedules the function to run and scales the number of compute instances based on the rate of incoming events. Triggers define how a function is invoked. Bindings provide a declarative way to connect to services from within the code. Events include triggers by:
 - Timers, for example, if a function needs to run every day at 10:00 AM UTC.
 - HTTP, for example, API and webhook scenarios.
 - Queues, for example, with order processing.
 - And much more.
- **Micro-billing:** Traditional computing bills for a block of time like paying a monthly or annual rate for website hosting. This method of billing is convenient but isn't always cost effective. Even if a customer's website gets only one hit a day, they still pay for a full day's worth of availability. With serverless computing, they pay only for the time their code runs. If no active function executions occur, they're not charged. For example, if the code runs once a day for two minutes, they're charged for one execution and two minutes of computing time.

Serverless computing is a term used to describe an execution environment that's set up and managed for you. You merely specify what you want to happen by writing code or connecting and configuring components in a visual editor, and then specify the actions that trigger your functionality, such as a timer or an HTTP request. Best of all, you never have to worry about an outage, your code can scale instantly to meet demand, and you pay based only on the actual usage of your code. Scaling and performance are handled automatically, and you're billed only for the resources you use. You don't even need to reserve resources.

Serverless computing is ordinarily used to handle back-end scenarios. In other words, serverless computing is responsible for sending messages from one system to another, or processing messages that were sent from other systems. It's not used for user-facing systems but, rather, it works in the background.

What are public, private, and hybrid clouds?

There are three deployment models for cloud computing: public cloud, private cloud, and hybrid cloud. Each deployment model has different aspects that you should consider as you migrate to the cloud.

- **Public cloud** - Services are offered over the public internet and available to anyone who wants to purchase them. Cloud resources like servers and storage are owned and operated by a third-party cloud service provider and delivered over the internet.
 - No capital expenditures to scale up.
 - Applications can be quickly provisioned and deprovisioned.
 - Organizations pay only for what they use.
- **Private cloud** - Computing resources are used exclusively by users from one business or organization. A private cloud can be physically located at your organization's on-site datacenter. It also can be hosted by a third-party service provider.
 - Hardware must be purchased for start-up and maintenance.
 - Organizations have complete control over resources and security.
 - Organizations are responsible for hardware maintenance and updates.
- **Hybrid cloud** - This computing environment combines a public cloud and a private cloud by allowing data and applications to be shared between them.
 - Provides the most flexibility.
 - Organizations determine where to run their applications.
 - Organizations control security, compliance, or legal requirements.

Web

Having a great web experience is critical in today's business world. Azure includes first-class support to build and host web apps and HTTP-based web services. The following Azure services are focused on web hosting.

Azure App Service	Quickly create powerful cloud web-based apps
Azure Notification Hubs	Send push notifications to any platform from any back end
Azure API Management	Publish APIs to developers, partners, and employees securely and at scale
Azure Cognitive Search	Deploy this fully managed search as a service
Web Apps feature of Azure App Service	Create and deploy mission-critical web apps at scale
Azure SignalR Service	Add real-time web functionalities easily

Mobile

With Azure, developers can create mobile back-end services for iOS, Android, and Windows apps quickly and easily. Features that used to take time and increase project risks, such as adding corporate sign-in and then connecting to on-premises resources such as SAP, Oracle, SQL Server, and SharePoint, are now simple to include.

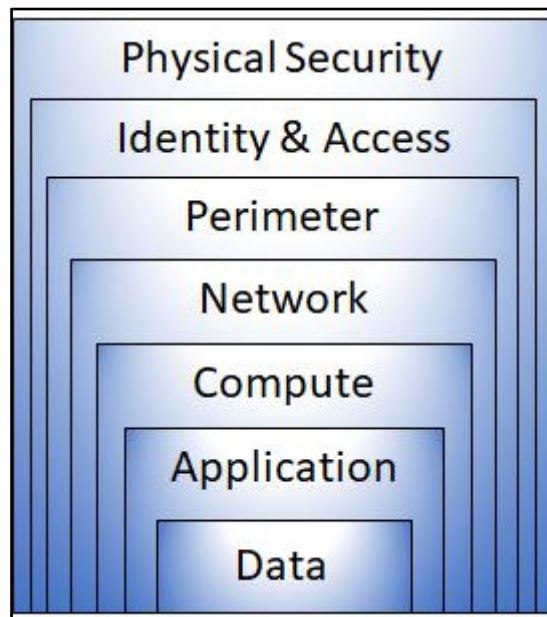
Other features of this service include:

- Offline data synchronization.
- Connectivity to on-premises data.
- Broadcasting push notifications.
- Autoscaling to match business needs.

What is defense in depth?

Tailwind Traders currently runs its workloads on-premises, in its datacenter. Running on-premises means that the company is responsible for all aspects of security, from physical access to buildings all the way down to how data travels in and out of the network. The company wants to know how its current defense-in-depth strategy compares to running in the cloud. The objective of defense in depth is to protect information and prevent it from being stolen by those who aren't authorized to access it.

A defense-in-depth strategy uses a series of mechanisms to slow the advance of an attack that aims at acquiring unauthorized access to data. You can visualize defense in depth as a set of layers, with the data to be secured at the center.



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

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

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

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

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



Physical security - Physically securing access to buildings and controlling access to computing hardware within the datacenter are the first line of defense. With physical security, the intent is to provide physical safeguards against access to assets. These safeguards ensure that other layers can't be bypassed, and loss or theft is handled appropriately. Microsoft uses various physical security mechanisms in its cloud datacenters.



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

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

nts and changes.



Perimeter - At the network perimeter, it's about protecting from network-based attacks against your resources. Identifying these attacks, eliminating their impact, and alerting you when they happen are important ways to keep your network secure. At this layer, it's important to:

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



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

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



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

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

current.



Application - Integrating security into the application development lifecycle helps reduce the number of vulnerabilities introduced in code. Every development team should ensure that its applications are secure by default. At this layer, it's important to:

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



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

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

Security posture

Your security posture is your organization's ability to protect from and respond to security threats. The common principles used to define a security posture are confidentiality, integrity, and availability, known collectively as CIA:

- Confidentiality - The principle of least privilege means restricting access to information only to individuals explicitly granted access, at only the level that they need to perform their work. This information includes protection of user passwords, email content, and access levels to applications and underlying infrastructure.
- Integrity - A common approach used in data transmission is for the sender to create a unique fingerprint of the data by using a one-way hashing algorithm. The hash is sent to the receiver along with the data. The receiver recalculates the data's hash and compares it to the original to ensure that the data wasn't lost or modified in transit. This also involves preventing unauthorized changes to information:
 - At rest -when it's stored.
 - In transit - when it's being transferred from one place to another, including from a local computer to the cloud.
- Availability - Ensure that services are functioning and can be accessed only by authorized users. Denial-of-service attacks are designed to degrade the availability of a system, affecting its users.

It's important to learn how costs are generated in Azure so that you can understand how your purchasing and solution design decisions can impact your final cost.

What types of Azure subscriptions can I use?

You probably know that an Azure *subscription* provides you with access to Azure resources, such as virtual machines (VMs), storage, and databases. The types of resources you use impact your monthly bill.

Azure offers both free and paid subscription options to fit your needs and requirements. They are:

- **Free trial** - A free trial subscription provides you with 12 months of popular free services, a credit to explore any Azure service for 30 days, and more than 25 services that are always free. Your Azure services are disabled when the trial ends or when your credit expires for paid products, unless you upgrade to a paid subscription.
- **Pay-as-you-go** - A pay-as-you-go subscription enables you to pay for what you use by attaching a credit or debit card to your account. Organizations can apply for volume discounts and prepaid invoicing.
- **Member offers** - Your existing membership to certain Microsoft products and services might provide you with credits for your Azure account and reduced rates on Azure services. For example, member offers are available to Visual Studio subscribers, Microsoft Partner Network members, Microsoft for Startups members, and Microsoft Imagine members.

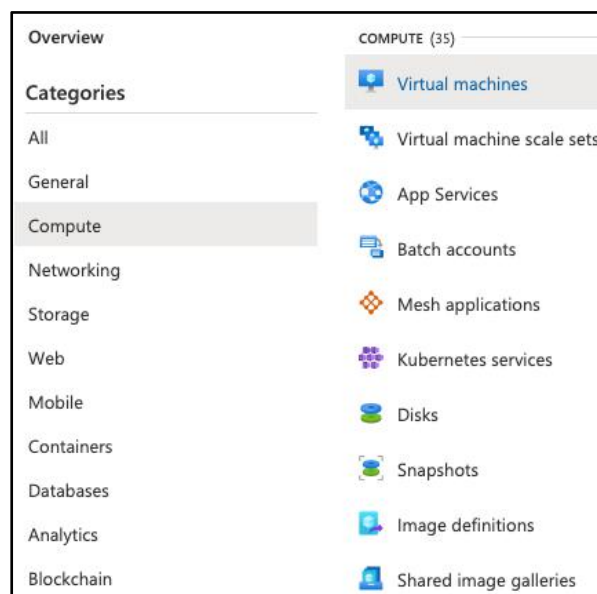
How do I purchase Azure services?

There are three main ways to purchase services on Azure. They are:

- **Through an Enterprise Agreement** - Larger customers, known as enterprise customers, can sign an Enterprise Agreement with Microsoft. This agreement commits them to spending a predetermined amount on Azure services over a period of three years. The service fee is typically paid annually. As an Enterprise Agreement customer, you'll receive the best customized pricing based on the kinds and amounts of services you plan on using.
- **Directly from the web** - Here, you purchase Azure services directly from the Azure portal website and pay standard prices. You're billed monthly, as a credit card payment or through an invoice. This purchasing method is known as Web Direct.
- **Through a Cloud Solution Provider** - A Cloud Solution Provider (CSP) is a Microsoft Partner who helps you build solutions on top of Azure. Your CSP bills you for your Azure usage at a price they determine. They also answer your support questions and escalate them to Microsoft, as needed.

You can bring up, or *provision*, Azure resources from the Azure portal or from the command line. The Azure portal arranges products and services by category. You select the services that fit your needs. Your account is billed according to Azure's "pay for what you use" model.

Here's an example that shows the Azure portal.



At the end of each month, you're billed for what you've used. At any time, you can check the cost management and billing page in the Azure portal to get a summary of your current usage and review invoices from prior months.

What factors affect cost?

The way you use resources, your subscription type, and pricing from third-party vendors are common factors. Let's take a quick look at each:

- **Resource type** - A number of factors influence the cost of Azure resources. They depend on the type of resource or how you customize it. For example, with a storage account you specify a type (such as block blob storage or table storage), a performance tier (standard or premium), and an access tier (hot, cool, or archive). These selections present different costs.
- **Usage meters** - When you provision a resource, Azure creates *meters* to track usage of that resource. Azure uses these meters to generate a usage record that's later used to help calculate your bill. Think of usage meters similar to how you use electricity or water in your home. You might pay a base price each month for electricity or water service, but your final bill is based on the total amount that you consumed. Each meter tracks a specific type of usage. For example, a meter might track bandwidth usage (ingress or egress network traffic in bits per second), number of operations, or its size (storage capacity in bytes). The usage that a meter tracks correlates to a quantity of billable units. Those units are charged to your account for each billing period. The rate per billable unit depends on the resource type you're using. Let's look at a single VM as an example. The following kinds of meters are relevant to tracking its usage:
 - Overall CPU time.
 - Time spent with a public IP address.
 - Incoming (ingress) and outgoing (egress) network traffic in and out of the VM.
 - Disk size and amount of disk read and disk write operations.
- **Resource usage** - In Azure, you're always charged based on what you use. As an example, let's look at how this billing applies to deallocating a VM. In Azure, you can delete or deallocate a VM. Deleting a VM means that you no longer need it. The VM is removed from your subscription, and then it's prepared for another customer. Deallocating a VM means that the VM is no longer running. But the associated hard disks and data are still kept in Azure. The VM isn't assigned to a CPU or network in Azure's datacenter, so it doesn't generate the costs associated with compute time or the VM's IP address. Because the disks and data are still stored, and the resource is present in your Azure subscription, you're still billed for disk storage. Deallocating a VM when you don't plan on using it for some time is just one way to minimize costs. For example, you might deallocate the VMs you use for testing purposes on weekends when your testing team isn't using them. You'll learn more about ways to minimize cost later in this module.
- **Azure subscription types** - Some Azure subscription types also include usage allowances, which affect costs. For example, an Azure free trial subscription provides access to a number of Azure products that are free for 12 months. It also includes credit to spend within your first 30 days of sign-up. And you get access to more than 25 products that are always free (based on resource and region availability).
- **Azure Marketplace** - You can also purchase Azure-based solutions and services from third-party vendors through Azure Marketplace. Examples include managed network firewall appliances or connectors to third-party backup services. Billing structures are set by the vendor.

Does location or network traffic affect cost?

When you provision a resource in Azure, you need to define the location (known as the Azure region) of where it will be deployed. Let's see why this decision can have cost consequences.

Location - Azure infrastructure is distributed globally, which enables you to deploy your services centrally or provision your services closest to where your customers use them. Different regions can have different associated prices. Because geographic regions can impact where your network traffic flows, network traffic is a cost influence to consider as well. For example, say Tailwind Traders decides to provision its Azure resources in the Azure regions that offer the lowest prices. That decision would save the company some money. But, if they need to transfer data between those regions, or if their users are located in different parts of the world, any potential savings could be offset by the additional network usage costs of transferring data between those resources. Zones for billing of network traffic Billing zones are a factor in determining the cost of some Azure services. [Bandwidth](#) refers to data moving in and out of Azure datacenters. Some inbound data transfers (data going into Azure datacenters) are free. For outbound data transfers (data leaving Azure datacenters), data transfer pricing is based on *zones*.



A zone is a geographical grouping of Azure regions for billing purposes. The following zones include some of the regions as shown here:

- **Zone 1:** Australia Central, West US, East US, Canada West, West Europe, France Central, and others
- **Zone 2:** Australia East, Japan West, Central India, Korea South, and others
- **Zone 3:** Brazil South, South Africa North, South Africa West, UAE Central, UAE North
- **DE Zone 1:** Germany Central, Germany Northeast

Azure Zone -> Azure Region -> Availability Zone -> Datacentre -> Server

Note: Not all Azure regions support availability zones.

North America has several Azure regions, including West US, Central US, South Central US, East Us, and Canada East.

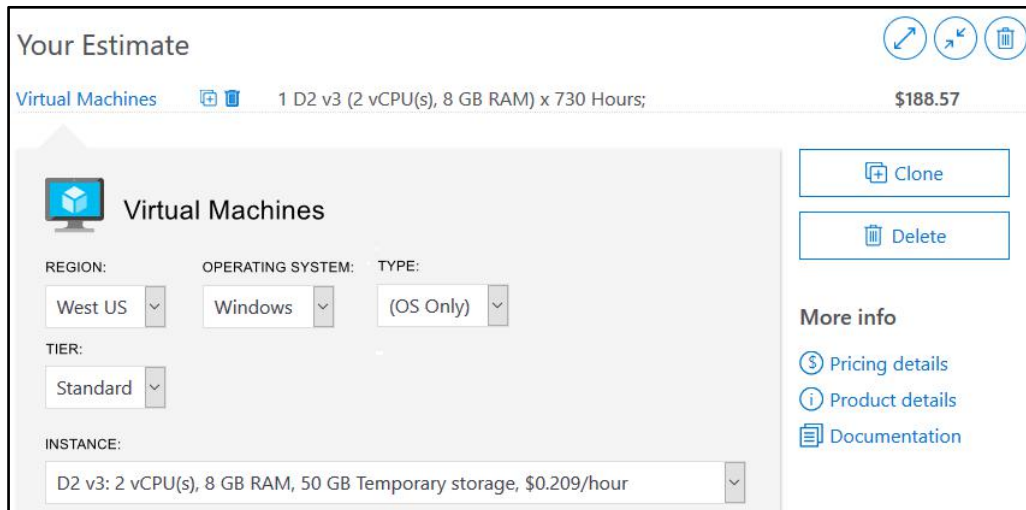
Availability Zones is a high-availability offering that protects your applications and data from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there are a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region protects applications and data from datacenter failures. Zone-redundant services replicate your applications and data across Availability Zones to protect from single-points-of-failure. With Availability Zones, Azure offers industry best 99.99% VM uptime SLA (You need a minimum of two virtual machines with each one located in a different availability zone).

How can I estimate the total cost?

As you've learned, an accurate cost estimate takes all of the preceding factors into account. Fortunately, the Azure Pricing calculator helps you with that process.

The Pricing calculator displays Azure products in categories. You add these categories to your estimate and configure according to your specific requirements. You then receive a consolidated estimated price, with a detailed breakdown of the costs associated with each resource you added to your solution. You can export or share that estimate or save it for later. You can load a saved estimate and modify it to match updated requirements.

You also can access pricing details, product details, and documentation for each product from within the Pricing calculator.



The screenshot shows the 'Your Estimate' interface for Virtual Machines. At the top, it displays 'Virtual Machines' with a plus icon, a trash icon, and the configuration '1 D2 v3 (2 vCPU(s), 8 GB RAM) x 730 Hours;' with a total cost of '\$188.57'. Below this, there are buttons for 'Clone' and 'Delete'. The main configuration area includes dropdowns for 'REGION:' (West US), 'OPERATING SYSTEM:' (Windows), 'TYPE:' ((OS Only)), 'TIER:' (Standard), and 'INSTANCE:' (D2 v3: 2 vCPU(s), 8 GB RAM, 50 GB Temporary storage, \$0.209/hour). On the right, there is a 'More info' section with links for 'Pricing details', 'Product details', and 'Documentation'.

The options that you can configure in the Pricing calculator vary between products, but they can include:

- **Region** - A region is the geographical location in which you can provision a service. Southeast Asia, Central Canada, Western United States, and Northern Europe are a few examples.
- **Tier** - Tiers, such as the Free tier or Basic tier, have different levels of availability or performance and different associated costs.
- **Billing options** - Billing options highlight the different ways you can pay for a service. Options can vary based on your customer type and subscription type and can include options to save costs.
- **Support options** - These options enable you to select additional support pricing options for certain services.
- **Programs and offers** - Your customer or subscription type might enable you to choose from specific licensing programs or other offers.
- **Azure Dev/Test pricing** - This option lists the available prices for development and test workloads. Dev/Test pricing applies when you run resources within an Azure subscription that's based on a Dev/Test offer.

Keep in mind that the Pricing calculator provides estimates and *not* actual price quotes. Actual prices can vary depending upon the date of purchase, the payment currency you're using, and the type of Azure customer you are.

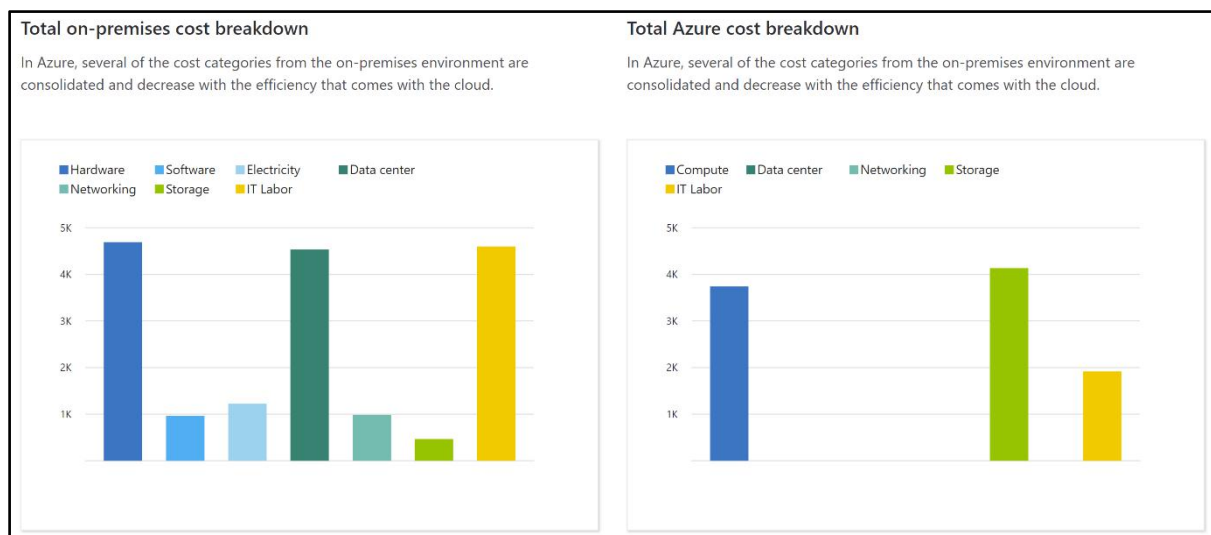
Total Cost of Ownership Calculator

Before Tailwind Traders takes its next steps toward migrating to the cloud, it wants to better understand what it spends today in its datacenter. Having a firm understanding of where the company is today will give it a greater sense of what cloud migration means in terms of cost. In this unit, you'll see how the Total Cost of Ownership (TCO) Calculator can help you compare the cost of running in the datacenter versus running on Azure.

The [TCO Calculator](#) helps you estimate the cost savings of operating your solution on Azure over time, instead of in your on-premises datacenter. The term *total cost of ownership* is commonly used in finance. It can be hard to see all the hidden costs related to operating a technology capability on-premises. Software licenses and hardware are additional costs.

With the TCO Calculator, you enter the details of your on-premises workloads. Then you review the suggested industry average cost (which you can adjust) for related operational costs. These costs include electricity, network maintenance, and IT labor. You're then presented with a side-by-side report. Using the report, you can compare those costs with the same workloads running on Azure.

The following image shows one example.

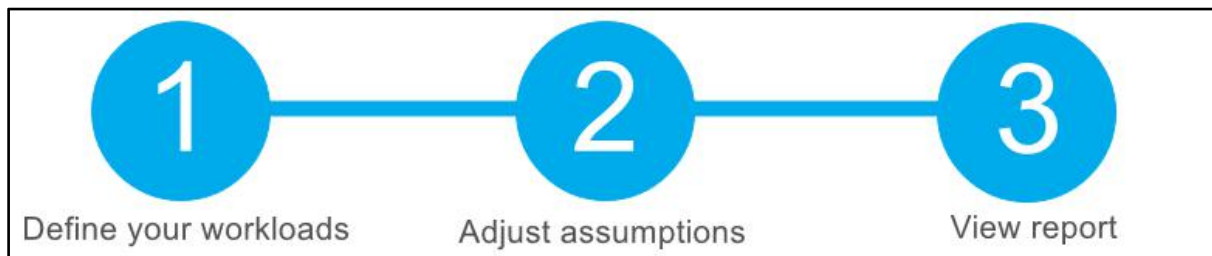


Note: You don't need an Azure subscription to work with the TCO Calculator.

How does the TCO Calculator work?

Working with the TCO Calculator involves three steps:

- Define your workloads.
- Adjust assumptions.
- View the report.



Let's take a closer look at each step.

Step 1: Define your workloads

First, you enter the specifications of your on-premises infrastructure into the TCO Calculator, based on these four categories:

- **Servers** - This category includes operating systems, virtualization methods, CPU cores, and memory (RAM).
- **Databases** - This category includes database types, server hardware, and the Azure service you want to use, which includes the expected maximum concurrent user sign-ins.
- **Storage** - This category includes storage type and capacity, which includes any backup or archive storage.
- **Networking** - This category includes the amount of network bandwidth you currently consume in your on-premises environment.

Step 2: Adjust assumptions

Next, you specify whether your current on-premises licenses are enrolled for [Software Assurance](#), which can save you money by reusing those licenses on Azure. You also specify whether you need to replicate your storage to another Azure region for greater redundancy.

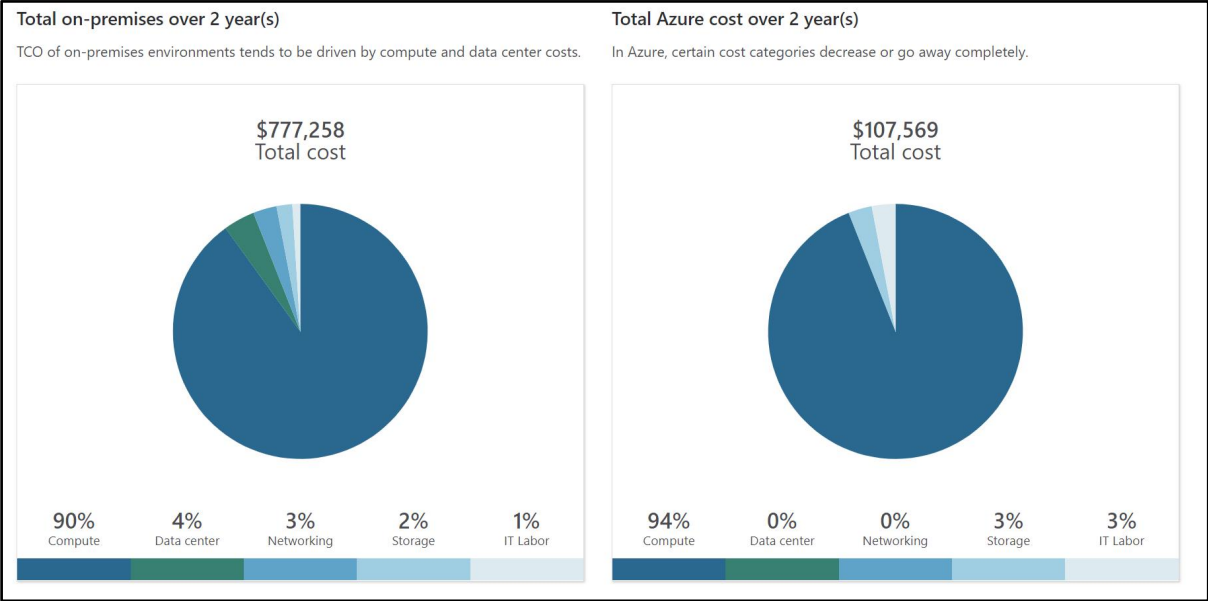
Then, you can see the key operating cost assumptions across several different areas, which vary among teams and organizations. These costs have been certified by Nucleus Research, an independent research company. For example, these costs include:

- Electricity price per kilowatt hour (KWh).
- Hourly pay rate for IT administration.
- Network maintenance cost as a percentage of network hardware and software costs.

To improve the accuracy of the TCO Calculator results, you adjust the values so that they match the costs of your current on-premises infrastructure.

Step 3: View the report

Choose a time frame between one and five years. the TCO Calculator generates a report that's based on the information you've entered. Here's an example:



For each category (compute, datacenter, networking, storage, and IT labor), you can also view a side-by-side comparison of the cost breakdown of operating those workloads on-premises versus operating them on Azure.

Here's an example:

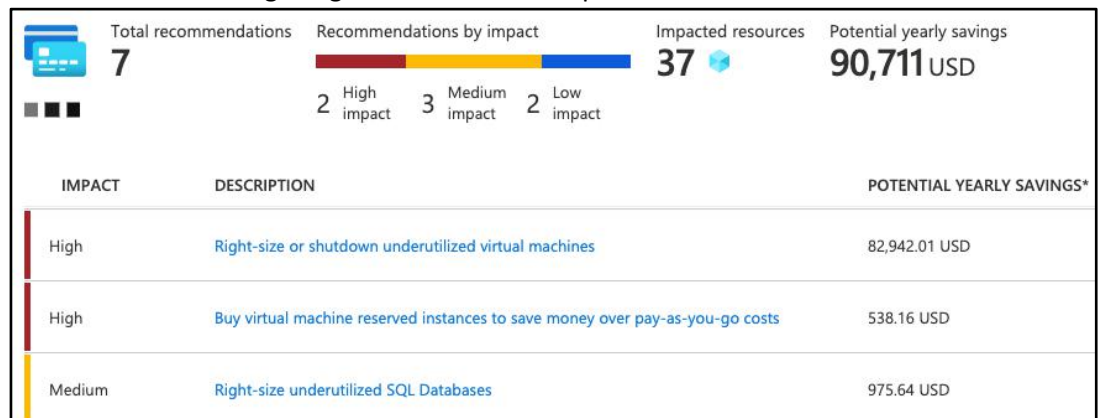
Estimated on-premises cost (2 year(s))		Estimated Azure cost (2 year(s))	
⌵ Compute cost		Azure compute cost	
⌵ Data center cost		Azure data center cost	
⌵ Networking cost		Azure networking cost	
⌶ Storage cost		Azure storage cost	
Hardware		Page Blob storage	
Local Disk/SAN-HDD		Usable storage volume in GB	
Cost per GB		Storage cost per GB/month	
Storage (RAID 10 configuration) volume in GB		Annual storage cost per usable volume	
Total storage procurement cost		Total Page Blob LRS storage maintenance cost over two year(s)	
\$5,191.68		\$1,105.92	

You can download, share, or save this report to review later.

Manage and minimize total cost on Azure

As a home improvement retailer, the proverb "measure twice, cut once" is fitting for the team at Tailwind Traders. Here are some recommended practices that can help you minimize your costs:

- Understand estimated costs before you deploy - To help you plan your solution on Azure, carefully consider the products, services, and resources you need. Read the relevant documentation to understand how each of your choices is metered and billed. Calculate your projected costs by using the Pricing calculator and the Total Cost of Ownership (TCO) Calculator. Only add the products, services, and resources that you need for your solution.
- Use Azure Advisor to monitor your usage - Ideally, you want your provisioned resources to match your actual usage. Azure Advisor identifies unused or underutilized resources and recommends unused resources that you can remove. This information helps you configure your resources to match your actual workload. The following image shows some example recommendations from Azure Advisor:



- Recommendations are sorted by impact: high, medium, or low. In some cases, Azure Advisor can automatically remediate, or fix, the underlying problem. Other issues, such as the two that are listed as high impact, require human intervention.
- Use spending limits to restrict your spending - If you have a free trial or a credit-based Azure subscription, you can use spending limits to prevent accidental overrun. For example, when you spend all the credit included with your Azure free account, Azure resources that you deployed are removed from production and your Azure virtual machines (VMs) are stopped and deallocated. The data in your storage accounts is available as read-only. At this point, you can upgrade your free trial subscription to a pay-as-you-go subscription. If you have a credit-based subscription and you reach your configured spending limit, Azure suspends your subscription until a new billing period begins. A related concept is *quotas*, or limits on the number of similar resources you can provision within your subscription. For example, you can allocate up to 25,000 VMs per region. These limits mainly help Microsoft plan its datacenter capacity.

- Use Azure Reservations to prepay - Azure Reservations offers discounted prices on certain Azure services. Azure Reservations can save you up to 72 percent as compared to pay-as-you-go prices. To receive a discount, you reserve services and resources by paying in advance. For example, you can prepay for one year or three years of use of VMs, database compute capacity, database throughput, and other Azure resources. The following example shows estimated savings on VMs. In this example, you save an estimated 72 percent by committing to a three-year term.

Select the product you want to purchase

Reserved VM Instances (RIs) provide a significant discount over pay-as-you-go VM prices by allowing you to pre-purchase the base costs of your VM usage for a period of 1 or 3 years. Reserved instance discount will automatically apply to matching VMs, you don't need to re-deploy resources to get reservation discount. The reservation applies only to hardware usage. Windows is charged separately. [Learn More](#)

Scope * Shared Billing subscription * Cost Management Research (1caaa5a3-2b66-438e-8...

Recommended All Products

Filter by name, region, or instance flexi... Region : East US Term : Three Years Billing frequency : Monthly + Add Filter Reset filters

Showing recommendations based on your usage over the last 30 d... [Learn more](#)

↑↓	Name ↑↓	Region ↑↓	Instance flexibility group ↑↓	vCPUs ↑↓	RAM (GB) ↑↓	Term ↑↓	Billing freque... ↑↓	Recommended Quantity ↑↓
	Standard_DS3_v2	East US	DSv2 Series	4	14	Three Years	Monthly	9 - See details
	Standard_DS2_v2	East US	DSv2 Series	2	7	Three Years	Monthly	5 - See details
	Standard_DS1_v2	East US	DSv2 Series	1	3.5	Three Years	Monthly	2 - See details
	Standard_D2s_v3	East US	DSv3 Series	2	8	Three Years	Monthly	2 - See details
	Standard_E16s_v3	East US	ESv3 Series	16	128	Three Years	Monthly	2 - See details
	Standard_F2s_v2	East US	FSv2 Series	2	4	Three Years	Monthly	2 - See details
	Standard_D2_v3	East US	Dv3 Series	2	8	Three Years	Monthly	1 - See details

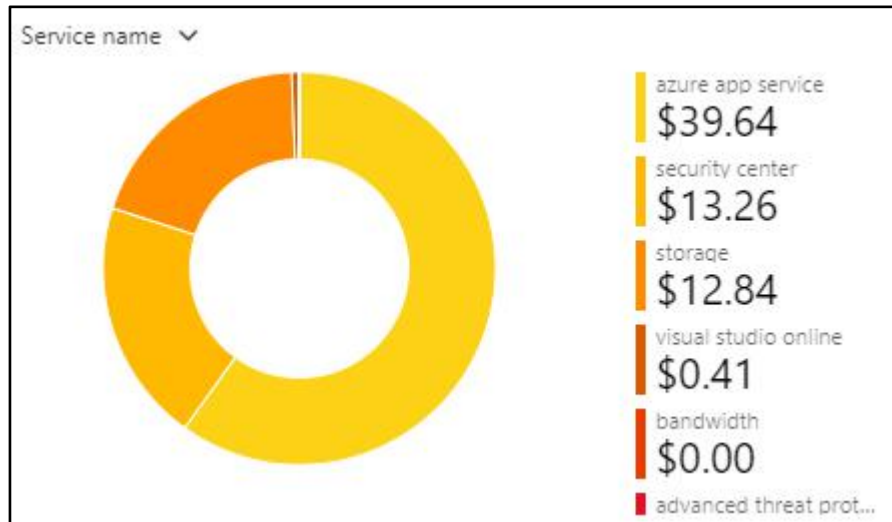
Not seeing what you want? [Browse all products.](#)

[Add to cart](#) [Close](#)

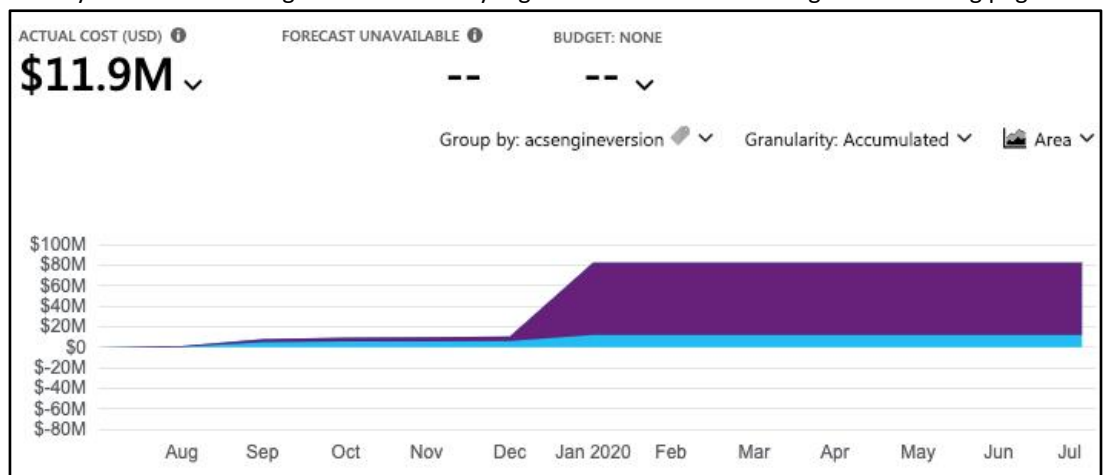
Monthly price per VM: 59.00 USD
72% Estimated savings

- Azure Reservations are available to customers with an Enterprise Agreement, Cloud Solution Providers, and pay-as-you-go subscriptions.
- Choose low-cost locations and regions - The cost of Azure products, services, and resources can vary across locations and regions. If possible, you should use them in those locations and regions where they cost less. But remember, some resources are metered and billed according to how much outgoing (egress) network bandwidth they consume. You should provision connected resources that are metered by bandwidth in the same Azure region to reduce egress traffic between them.
- Research available cost-saving offers - Keep up to date with the latest Azure customer and subscription offers, and switch to offers that provide the greatest cost-saving benefit.

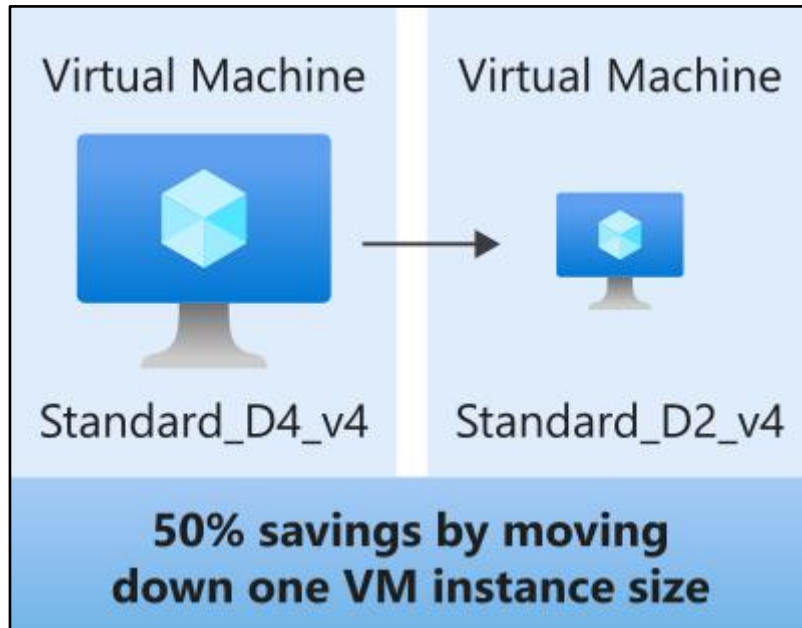
- Use Azure Cost Management + Billing to control spending - Azure Cost Management + Billing is a free service that helps you understand your Azure bill, manage your account and subscriptions, monitor and control Azure spending, and optimize resource use. The following image shows current usage broken down by service:



- In this example, Azure App Service, a web application hosting service, generates the greatest cost. Azure Cost Management + Billing features include:
 - **Reporting** - Use historical data to generate reports and forecast future usage and expenditure.
 - **Data enrichment** - Improve accountability by categorizing resources with tags that correspond to real-world business and organizational units.
 - **Budgets** - Create and manage cost and usage budgets by monitoring resource demand trends, consumption rates, and cost patterns.
 - **Alerting** - Get alerts based on your cost and usage budgets.
 - **Recommendations** - Receive recommendations to eliminate idle resources and to optimize the Azure resources you provision.
- Apply tags to identify cost owners - *Tags* help you manage costs associated with the different groups of Azure products and resources. You can apply tags to groups of Azure resources to organize billing data. For example, if you run several VMs for different teams, you can use tags to categorize costs by department, such as Human Resources, Marketing, or Finance, or by environment, such as Test or Production. Tags make it easier to identify groups that generate the biggest Azure costs, which can help you adjust your spending accordingly. The following image shows a year's worth of usage broken down by tags on the Azure Cost Management + Billing page



- Resize underutilized virtual machines - A common recommendation that you'll find from Azure Cost Management + Billing and Azure Advisor is to resize or shut down VMs that are underutilized or idle. As an example, say you have a VM whose size is **Standard_D4_v4**, a general-purpose VM type with four vCPUs and 16 GB of memory. You might discover that this VM is idle 90 percent of the time. Virtual machine costs are linear and double for each size larger in the same series. So in this case, if you reduce the VM's size from **Standard_D4_v4** to **Standard_D2_v4**, which is the next size lower, you reduce your compute cost by 50 percent. The following image shows this idea:

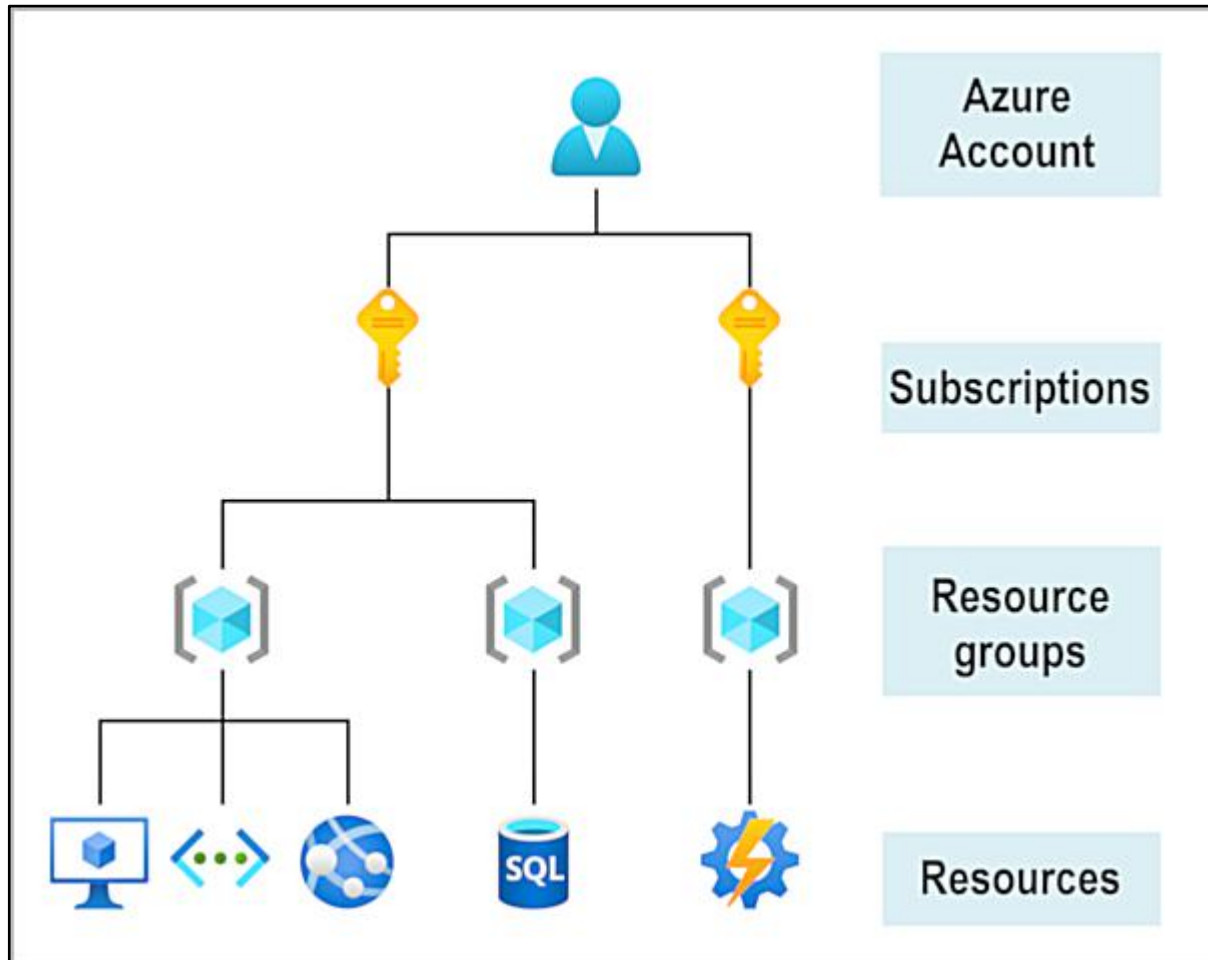


-
- Keep in mind that resizing a VM requires it to be stopped, resized, and then restarted. This process might take a few minutes depending on how significant the size change is. Be sure to properly plan for an outage, or shift your traffic to another instance while you perform resize operations.
- Deallocate virtual machines during off hours - Recall that to *deallocate* a VM means to no longer run the VM, but preserve the associated hard disks and data in Azure. If you have VM workloads that are only used during certain periods, but you're running them every hour of every day, you're wasting money. These VMs are great candidates to shut down when not in use and start back when you need them, saving you compute costs while the VM is deallocated. This approach is an excellent strategy for development and testing environments, where the VMs are needed only during business hours. Azure even provides a way to automatically start and stop your VMs on a schedule.

- Delete unused resources - This recommendation might sound obvious, but if you aren't using a resource, you should shut it down. It's not uncommon to find nonproduction or proof-of-concept systems that are no longer needed following the completion of a project. Regularly review your environment, and work to identify these systems. Shutting down these systems can have a dual benefit by saving you on infrastructure costs and potential savings on licensing and operating costs.
- Migrate from IaaS to PaaS services - As you move your workloads to the cloud, a natural evolution is to start with infrastructure as a service (IaaS) services because they map more directly to concepts and operations you're already familiar with. Over time, one way to reduce costs is to gradually move IaaS workloads to run on platform as a service (PaaS) services. While you can think of IaaS as direct access to compute infrastructure, PaaS provides ready-made development and deployment environments that are managed for you. As an example, say you run SQL Server on a VM running on Azure. This configuration requires you to manage the underlying operating system, set up a SQL Server license, manage software and security updates, and so on. You also pay for the VM whether or not the database is processing queries. One way to potentially save costs is to move your database from SQL Server on a VM to Azure SQL Database. Azure SQL Database is based on SQL Server. Not only are PaaS services such as Azure SQL Database often less expensive to run, but because they're managed for you, you don't need to worry about software updates, security patches, or optimizing physical storage for read and write operations.
- Save on licensing costs - Licensing is another area that can dramatically impact your cloud spending. Let's look at some ways you can reduce your licensing costs.
 - Choose cost-effective operating systems - Many Azure services provide a choice of running on Windows or Linux. In some cases, the cost depends on which you choose. When you have a choice, and your application doesn't depend on the underlying operating system, it's useful to compare pricing to see whether you can save money.
 - Use Azure Hybrid Benefit to repurpose software licenses on Azure - If you've purchased licenses for Windows Server or SQL Server, and your licenses are covered by [Software Assurance](#), you might be able to repurpose those licenses on VMs on Azure. Some of the details vary between Windows Server or SQL Server.

Azure Accounts

To create and use Azure services, you need an Azure subscription. When you're completing Learn modules, most of the time a temporary subscription is created for you, which runs in an environment called the Learn sandbox. When you're working with your own applications and business needs, you need to create an Azure account, and a subscription will be created for you. After you've created an Azure account, you're free to create additional subscriptions. For example, your company might use a single Azure account for your business and separate subscriptions for development, marketing, and sales departments. After you've created an Azure subscription, you can start creating Azure resources within each subscription.



If you're new to Azure, you can sign up for a free account on the Azure website to start exploring at no cost to you. When you're ready, you can choose to upgrade your free account. You can create a new subscription that enables you to start paying for Azure services you need to use that are beyond the limits of a free account.

You can purchase Azure access directly from Microsoft by signing up on the [Azure website](#) or through a Microsoft representative. You can also purchase Azure access through a Microsoft partner. Cloud Solution Provider partners offer a range of complete managed-cloud solutions for Azure.

An Azure AD tenant can have multiple subscriptions but an Azure subscription can only be associated with one Azure AD tenant. You can change the Azure AD tenant to which a subscription is associated. If your subscription expires, you lose access to all the other resources associated with the subscription. However, the Azure AD directory remains in Azure. You can associate and manage the directory using a different Azure subscription.

Azure Free Account

The Azure free account includes:

- Free access to popular Azure products for 12 months.
- A credit (150 GBP or 200 USD) to spend for the first 30 days.
- Access to more than 25 products that are always free.

The Azure free account is an excellent way for new users to get started and explore. To sign up, you need a phone number, a credit card, and a Microsoft or GitHub account. The credit card information is used for identity verification only. You won't be charged for any services until you upgrade to a paid subscription. You can only create one free Azure account per Microsoft account.

What is the Learn sandbox?

Many of the Learn exercises use a technology called the sandbox, which creates a temporary subscription that's added to your Azure account. This temporary subscription allows you to create Azure resources for the duration of a Learn module. Learn automatically cleans up the temporary resources for you after you've completed the module.

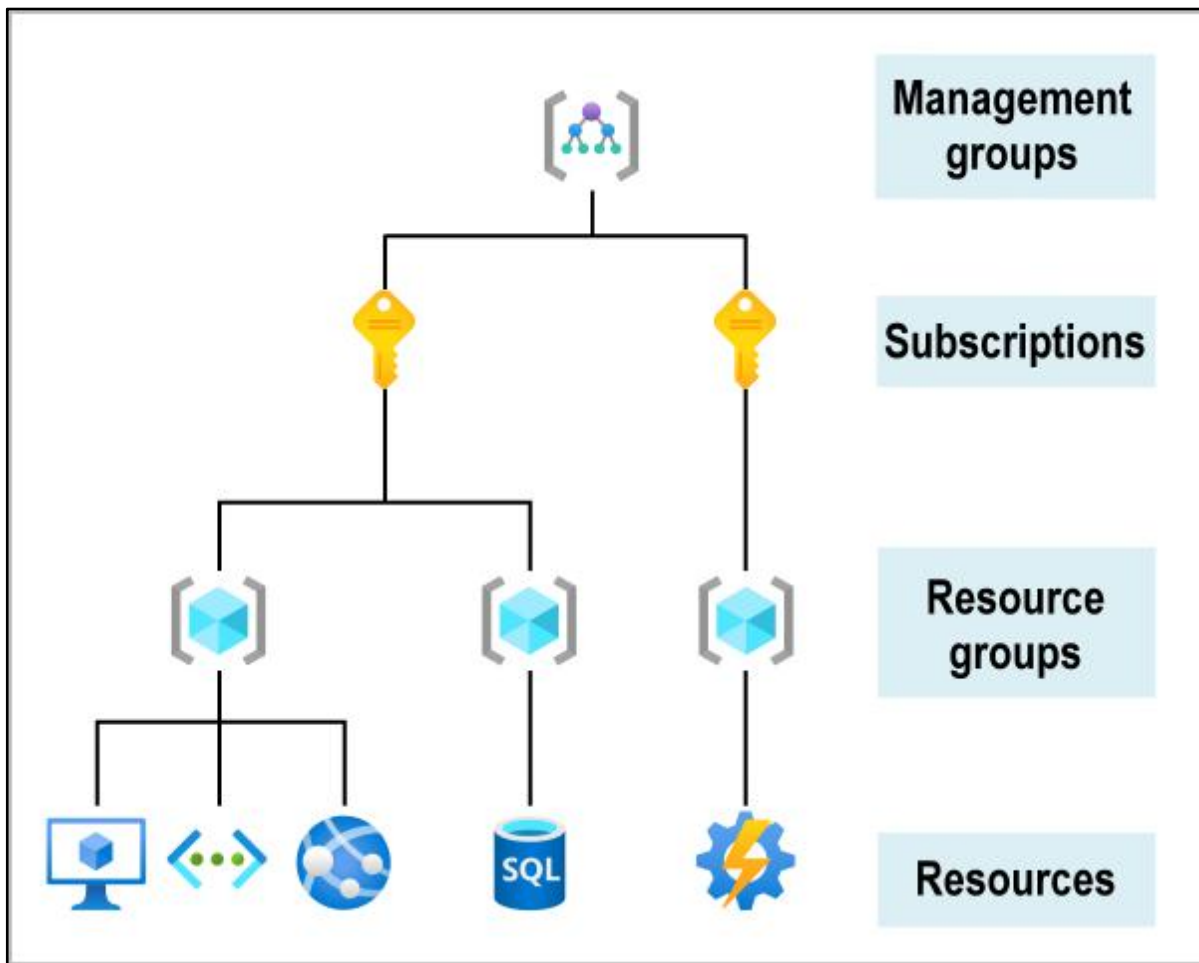
When you're completing a Learn module, you're welcome to use your personal subscription to complete the exercises in a module. The sandbox is the preferred method to use though, because it allows you to create and test Azure resources at no cost to you.

Azure free account has a limit of \$200 Credit. Azure free account has a 5 GB blob storage limit and a 5 GB file storage limit. Azure free account has a limit of 10 web, mobile or API apps

Organizing Structure for Resources

The organizing structure for resources in Azure has four levels: management groups, subscriptions, resource groups, and resources.

The following image shows the top-down hierarchy of organization for these levels.



Having seen the top-down hierarchy of organization, let's describe each of those levels from the bottom up:

- **Resources:** Resources are instances of services that you create, like virtual machines, storage, or SQL databases. Tags for Resources are not inherited by default from their Resource Group. A resource group can be used to scope access control for administrative actions. By default, permissions set at the resource level are inherited by the resources in the resource group.
- **Resource groups:** Resources are combined into resource groups, which act as a logical container into which Azure resources like web apps, databases, and storage accounts are deployed and managed. Azure resources deployed to a single resource group can be located in different regions. The resource group only contains metadata about the resources it contains. When creating a resource group, you need to provide a location for that resource group. You may be wondering, "Why does a resource group need a location? And, if the resources can have different locations than the resource group, why does the resource group location matter at all?" The resource group stores metadata about the resources. When you specify a location for the resource group, you're specifying where that metadata is stored. For compliance reasons, you may need to ensure that your data is stored in a particular region.

Subscriptions: A subscription groups together user accounts and the resources that have been created by those user accounts. The first thing you create in Azure is a subscription. You can think of an Azure subscription as an 'Azure account'. You get billed per subscription. A subscription is an agreement with Microsoft to use one or more Microsoft cloud platforms or services, for which charges accrue based on either a per-user license fee or on cloud-based resource consumption. For each subscription, there are limits or quotas on the amount of resources that you can create and use. Organizations can use subscriptions to manage costs and the resources that are created by users, teams, or projects. An Azure subscription is a boundary for permissions to resources and for billing. You are charged monthly for all resources in a subscription. You may want an additional subscription to avoid hitting subscription limits, to create separate environments for security, or to isolate data for compliance reasons. You cannot merge two subscriptions into a single

subscription. However, you can move some Azure resources from one subscription to another. You can also transfer ownership of a subscription and change the billing type for a subscription. A single Azure tenant (Azure Active Directory) can have multiple subscriptions and store resources in the different subscriptions. However, a resource instance can exist in only one subscription.

Management groups: These groups help you manage access, policy, and compliance for multiple subscriptions. All subscriptions in a management group automatically inherit the conditions applied to the management group.

You need an Azure Active Directory account to manage a subscription, not a Microsoft account. An account is created in the Azure Active Directory when you create the subscription. Further accounts can be created in the Azure Active Directory to manage the subscription. A subscription can have multiple administrators, you can assign additional account administrators in the Azure Portal.

RBAC - Role-based access control - has several Azure built-in roles that you can assign to users, groups, service principals, and managed identities. Role assignments are the way you control access to Azure resources. If the built-in roles don't meet the specific needs of your organization, you can create your own Azure custom roles. The Service Administrator and Co-Administrators are assigned the Owner role at the subscription scope; Co-Administrator - up to 200 per subscription.

Azure Policies

Azure policies can be used to define requirements for resource properties during deployment and for already existing resources. Azure Policy controls properties such as the types or locations of resources. Azure Policy is a service in Azure that you use to create, assign, and manage policies. These policies enforce different 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. All data stored by Azure Policy is encrypted at rest. For example, you can have a policy to allow only a certain SKU size of virtual machines in your environment. Once this policy is implemented, new and existing resources are evaluated for compliance. With the right type of policy, existing resources can be brought into compliance.

Azure Policy is a service in Azure that you use to create, assign, and manage policies. These policies enforce different rules and effects over your resources, so those resources stay compliant with your corporate standards and service level agreements. If there are any existing resources that aren't compliant with a new policy assignment, they appear under Non-compliant resources, they will still continue to function as normal during this time.

Azure policies can be used to define requirements for resource properties during deployment and for already existing resources. Azure Policy controls properties such as the types or locations of resources.

An Azure Policy initiative is a group of policy definitions.

Resource Limits

Many Azure resource have quote limits. The purpose of the quota limits is to help you control your Azure costs. However, it is common to require an increase to the default quota. You can request a quota limit increase by opening a support request. In the support request, select 'Service and subscription limits (quotas)' for the Issue type, select your subscription and the service you want to increase the quota for.

Resource Locks

As an administrator, you can lock a subscription, resource group, or resource to prevent other users in your organization from accidentally deleting or modifying critical resources. The lock overrides any permissions the user might have.

You can set the lock level to **CanNotDelete** or **ReadOnly**. In the portal, the locks are called **Delete** and **Read-only** respectively.

CanNotDelete means authorized users can still read and modify a resource, but they can't delete the resource.

ReadOnly means authorized users can read a resource, but they can't delete or update the resource. Applying this lock is similar to restricting all authorized users to the permissions granted by the **Reader** role.

Unlike role-based access control, you use management locks to apply a restriction across all users and roles. To learn about setting permissions for users and roles, see [Azure role-based access control \(Azure RBAC\)](#).

Lock inheritance - When you apply a lock at a parent scope, all resources within that scope inherit the same lock. Even resources you add later inherit the lock from the parent. The most restrictive lock in the inheritance takes precedence.

Understand scope of locks - Control plane operations are operations sent to <https://management.azure.com>. Data plane operations are operations sent to your instance of a service, such as <https://myaccount.blob.core.windows.net/>. For more information, see Azure control plane and data plane. To discover which operations use the control plane URL, see the Azure REST API. This distinction means locks prevent changes to a resource, but they don't restrict how resources perform their own functions. For example, a **ReadOnly** lock on a SQL Database logical server prevents you from deleting or modifying the server. It doesn't prevent you from creating, updating, or deleting data in the databases on that server. Data transactions are permitted because those operations aren't sent to <https://management.azure.com>.

SLAs and Service Lifecycle

A service-level agreement (SLA) is a formal agreement between a service company and the customer. For Azure, this agreement defines the performance standards that Microsoft commits to for you, the customer.

Why are SLAs important? - Understanding the SLA for each Azure service you use helps you understand what guarantees you can expect. When you build applications on Azure, the availability of the services that you use affect your application's performance. Understanding the SLAs involved can help you establish the SLA you set with your customers.

Where can I access SLAs for Azure services? - You can access SLAs from [Service Level Agreements](#). You don't need an Azure subscription to review service SLAs. Each Azure service defines its own SLA. Azure services are organized by category.

What's in a typical SLA? - A typical SLA breaks down into these sections:

- **Introduction** - This section explains what to expect in the SLA, including its scope and how subscription renewals can affect the terms.
- **General terms** - This section contains terms that are used throughout the SLA so that both parties (you and Microsoft) have a consistent vocabulary. For example, this section might define what's meant by downtime, incidents, and error codes. This section also defines the general terms of the agreement, including how to submit a claim, receive credit for any performance or availability issues, and limitations of the agreement.
- **SLA details** - This section defines the specific guarantees for the service. Performance commitments are commonly measured as a percentage. That percentage typically ranges from 99.9 percent ("three nines") to 99.99 percent ("four nines"). The primary performance commitment typically focuses on *uptime*, or the percentage of time that a product or service is successfully operational. Some SLAs focus on other factors as well, including *latency*, or how fast the service must respond to a request. This section also defines any additional terms that are specific to this service.

How do percentages relate to total downtime? - *Downtime* refers to the time duration that the service is unavailable. The difference between 99.9 percent and 99.99 percent might seem minor, but it's important to understand what these numbers mean in terms of total downtime. These amounts are cumulative, which means that the duration of multiple different service outages would be combined, or added together. Here's a table to give you a sense of how total downtime decreases as the SLA percentage increases from 99 percent to 99.999 percent:

HOW DO PERCENTAGES RELATE TO TOTAL DOWNTIME?			
SLA percentage	Downtime per week	Downtime per month	Downtime per year
99	1.68 hours	7.2 hours	3.65 days
99.9	10.1 minutes	43.2 minutes	8.76 hours
99.95	5 minutes	21.6 minutes	4.38 hours
99.99	1.01 minutes	4.32 minutes	52.56 minutes
99.999	6 seconds	25.9 seconds	5.26 minutes

What are service credits? - A *service credit* is the percentage of the fees you paid that are credited back to you according to the claim approval process. An SLA describes how Microsoft responds when an Azure service fails to perform to its specification. For example, you might receive a discount on your Azure bill as compensation when a service fails to perform according to its SLA. If the SLA for an Azure service is not met, you receive credits for that service and that service only. The credits are deducted from your monthly bill for that service. Credits typically increase as uptime decreases. Here's how credits are applied for Azure Database for MySQL according to uptime:

Monthly uptime percentage	Service credit percentage
< 99.99	10
< 99	25
< 95	100

What's the SLA for free services? - Free products typically don't have an SLA. For example, many Azure services provide a *free* or *shared* tier that provides more limited functionality. Services like Azure Advisor are always free. The [SLA for Azure Advisor](#) states that because it's free, it doesn't have a financially backed SLA.

How do I know when there's an outage? - [Azure status](#) provides a global view of the health of Azure services and regions. If you suspect there's an outage, this is often a good place to start your investigation. Azure status provides an RSS feed of changes to the health of Azure services that you can subscribe to. You can connect this feed to communication software such as Microsoft Teams or Slack. From the Azure status page, you can also access Azure Service Health. This provides a personalized view of the health of the Azure services and regions that you're using, directly from the Azure portal.

How can I request a service credit from Microsoft? - Typically, you need to file a claim with Microsoft to receive a service credit. If you purchase Azure services from a Cloud Solution Provider (CSP) partner, your CSP typically manages the claims process. Each SLA specifies the timeline by which you must submit your claim and when Microsoft processes your claim. For many services, you must submit your claim by the end of the calendar month following the month in which the incident occurred.

Your Application SLA

An application SLA defines the SLA requirements for a specific application. This term typically refers to an application that you build on Azure. Tailwind Traders runs an application that it built on Azure called "Special Orders." The application tracks special orders that customers have placed in the company's retail stores. A special order includes an item and any customizations the customer needs. For example, a folding door might include customizations such as dimension and hinge placement. Because customizations typically require special handling, the customized item needs to be ordered from the supplier when a customer needs it.

There are many design decisions you can make to improve the availability and resiliency of the applications and services you build on Azure. These decisions extend beyond just the SLA for a specific service. In this part, you'll explore a few of these considerations. A good place to start is to have a discussion with your team about how important the availability of each application is to your business. The following sections cover a few factors that Tailwind Traders might consider.

- **Business impact** - If the Special Orders application goes down, what would the business impact be? In this case, customers can't place new orders through the store and staff can't check the status of existing orders. Customers will either need to try again later or possibly go to a competitor.
- **Effect on other business operations** - The Special Orders application doesn't affect other operations. So the majority of the Tailwind Traders business will continue to function normally if the Special Orders application went down.
- **Usage patterns** - Usage patterns define when and how users access your application. One question to consider is whether the availability requirement differs between critical and non-critical time periods. For example, a tax-filing application can't fail during a filing deadline. For Tailwind Traders, retail stores aren't open 24 hours a day, so if the application were down in the middle of the night, the impact would be minimal. However, because Tailwind Traders has retail locations all over the world, it will need to ensure that each location has access to the service during its retail hours.
- **What does the team decide?** - Let's say that Tailwind Traders decides that an SLA of 99.9 percent is acceptable for the Special Orders application. This gives the company an estimated downtime of 10.1 minutes per week. But how will it ensure that its technology choices support its application SLA?

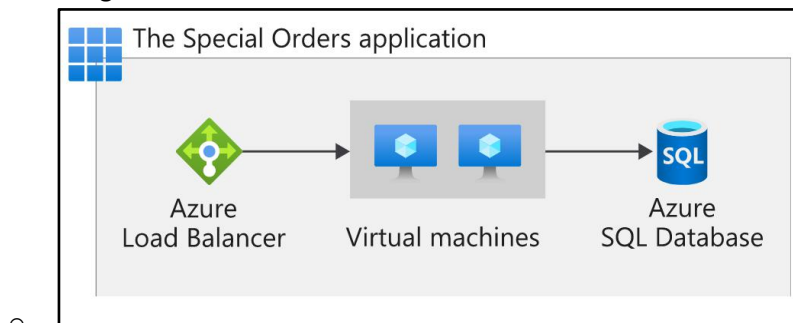
In the next part, you'll see how the team maps its application requirements to specific Azure services. You'll learn about some of the techniques you can use to help ensure that your technology choices meet your application SLA.

Design your application to meet your SLA

Tailwind Traders decides that an SLA of 99.9 percent is acceptable for the Special Orders application. Recall that this gives the company an estimated downtime of 10.1 minutes per week. Now you need to design an efficient and reliable solution for this application on Azure, keeping that application SLA in mind. You'll select the Azure products and services you need, and provision your cloud resources according to those requirements. In reality, failures will happen. Hardware can fail. The network can have intermittent timeout periods. While it's rare for an entire service or region to experience a disruption, you still need to plan for such events. Let's follow the process Tailwind Traders uses to ensure that its technology choices meet its application SLA.

Identify your workloads - A workload is a distinct capability or task that's logically separated from other tasks, in terms of business logic and data storage requirements. Each workload defines a set of requirements for availability, scalability, data consistency, and disaster recovery. On Azure, the Special Orders application will require:

- Two virtual machines.
- One instance of Azure SQL Database.
- One instance of Azure Load Balancer.
- Here's a diagram that shows the basic architecture:



Combine SLAs to compute the composite SLA

After you've identified the SLA for the individual workloads in the Special Orders application, you might notice that those SLAs are not all the same. How does this affect our overall application SLA requirement of 99.9 percent? To work that out, you'll need to do some math. The process of combining SLAs helps you compute the composite SLA for a set of services. Computing the composite SLA requires that you multiply the SLA of each individual service. From Service Level Agreements, you discover the SLA for each Azure service that you need. They are:

Service	SLA
Azure Virtual Machines	99.9 percent
Azure SQL Database	99.99 percent
Azure Load Balancer	99.99 percent

Therefore, for the Special Orders application, the composite SLA would be:

$$99.9\% \times 99.9\% \times 99.99\% \times 99.99\% = 0.999 \times 0.999 \times 0.9999 \times 0.9999 = 0.9978 = 99.78\%$$

Recall that you need two virtual machines. Therefore, you include the Virtual Machines SLA of 99.9 percent two times in the formula.

Note that even though all of the individual services have SLAs equal to or better than the application SLA, combining them results in an overall number that's lower than the 99.9 percent you need. Why? Because using multiple services adds an extra level of complexity and slightly increases the risk of failure.

You see here that the composite SLA of 99.78 percent doesn't meet the required SLA of 99.9 percent. You might go back to team and ask whether this is acceptable. Or you might implement some other strategies into your design to improve this SLA.

What happens when the composite SLA doesn't meet your needs?

For the Special Orders application, the composite SLA doesn't meet the required SLA of 99.9 percent. Let's look at a few strategies that Tailwind Traders might consider:

- Choose customization options that fit your required SLA - Each of the workloads defined previously has its own SLA, and the customization choices you make when you provision each workload affects that SLA. For example:
 - **Disks** - With Virtual Machines, you can choose from a Standard HDD Managed Disk, a Standard SSD Managed Disk, or a Premium SSD or Ultra Disk. The SLA for a single VM would be either 95 percent, 99.5 percent or 99.9 percent, depending on the disk choice.
 - **Tiers** - Some Azure services are offered as both a free tier product and as a standard paid service. For example, Azure Automation provides 500 minutes of job runtime in an Azure free account, but is not backed by an SLA. The standard tier SLA for Azure Automation is 99.9 percent. Make sure that your purchasing decisions take into account the impact on the SLA for the Azure services that you choose. Doing so ensures that the SLA supports your required application SLA. Here, Tailwind Traders might choose the Ultra Disk option for its virtual machines to help guarantee greater uptime.
 - **Build availability requirements into your design** - There are application design considerations you can use that relate to the underlying cloud infrastructure. For example, to improve the availability of the application, avoid having any single points of failure. So instead of adding more virtual machines, you can deploy one or more extra instances of the same virtual machine across the different availability zones in the same Azure region. An availability zone is a unique physical location within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. These zones use different schedules for maintenance, so if one zone is affected, your virtual machine instance in the other zone is unaffected. Deploying two or more instances of an Azure virtual machine across two or more availability zones raises the virtual machine SLA to 99.99 percent. Recalculating your composite SLA above with this Virtual Machines SLA gives you an application SLA of:
 - $99.9\% \times 99.9\% \times 99.99\% \times 99.99\% = 99.96\%$
 - This revised SLA of 99.96 percent exceeds your target of 99.9 percent.
 - To learn more about the SLA for Virtual Machines, visit [SLA for Virtual Machines](#).
 - **Include redundancy to increase availability** - To ensure high availability, you might plan for your application to have duplicate components across several regions, known as redundancy. Conversely, to minimize costs during non-critical periods, you might run your application only in a single region. Tailwind Traders might consider this if there's a trend that the special order rates are much higher during certain months or seasons. To achieve maximum availability in your application, add redundancy to every single part of the application. This redundancy includes the application itself, as well as the underlying services and infrastructure. Be aware, however, that doing so can be difficult and expensive, and often results in solutions that are more complex than they need to be. Consider how critical high availability is to your requirements before you add redundancy. There may be simpler ways to meet your application SLA.

Very high performance is difficult to achieve. Performance targets above 99.99 percent are very difficult to achieve. An SLA of 99.99 percent means 1 minute of downtime per week. It's difficult for humans to respond to failures quickly enough to meet SLA performance targets above 99.99 percent. Instead, your application must be able to self-diagnose and self-heal during an outage.

Access preview services and preview features

Now that Tailwind Traders has its applications up and running, it wants to start looking into new capabilities. One option is to look at preview services. In this part, you'll learn how Azure services go from the preview phase to being generally available. For Tailwind Traders, migration from the datacenter to Azure is more about operational efficiency. The research and development team is looking into new, cloud-based features that will keep them ahead of the competition. Tailwind Traders is experimenting with a custom drone delivery system for customers in rural areas. The company needs the ability to use real-time storm tracking in the drone guidance system, but the feature isn't ready yet. There's a new AI Storm Analyzer service that has just entered the public preview phase. So Tailwind Traders has decided to incorporate it into the early stages of application testing. (AI Storm Analyzer is a fictitious Azure service, introduced here for illustration purposes only.) Before the team moves forward, it wants a better understanding of how preview services affect its SLA. Let's begin by defining the Azure service lifecycle.

What is the service lifecycle? - The *service lifecycle* defines how every Azure service is released for public use:

- Every Azure service starts in the development phase. In this phase, the Azure team collects and defines its requirements, and begins to build the service.
- Next, the service is released to the public preview phase. During this phase, the public can access and experiment with it so that it can provide feedback. Your feedback helps Microsoft improve services. More importantly, providing feedback gives you the opportunity to request new or different capabilities so that services better meet your needs.
- After a new Azure service is validated and tested, it's released to all customers as a production-ready service. This is known as *general availability* (GA).

What terms and conditions can I expect? - Each Azure preview defines its own terms and conditions. All preview-specific terms and conditions supplement your existing Azure service agreement. Some previews aren't covered by customer support. Therefore, previews are not recommended for business-critical workloads.

How can I access preview services? - You can access preview services from the Azure portal. Here's how to see what preview services are available. You can follow along if you have an Azure subscription:

- Go to the [Azure portal](#) and sign in.
- Select **Create a resource**.
- Enter *preview* in the search box, and select **Enter**.
- Select a service to learn more about it. You can also launch the service if you'd like to try it out.

How can I access new features for an existing service? - Some preview features relate to a specific area of an existing Azure service. For example, a compute or database service that you use daily might provide enhanced functionality. These preview features are accessible when you deploy, configure, and manage the service. Although you can use an Azure preview feature in production, make sure you're aware of any limitations around its use before you deploy it to production.

How can I access preview features for the Azure portal?

You can access preview features that are specific to the Azure portal from [Microsoft Azure \(Preview\)](#). Typical portal preview features provide performance, navigation, and accessibility improvements to the Azure portal interface. You see **Microsoft Azure (Preview)** near the menu bar to remind you that you're working with a preview version of the Azure portal.

How can I provide feedback on the Azure portal? - You can provide feedback:

- From the **Feedback** tab in the Azure portal.
- From the [Azure portal feedback forum](#).

How can I stay updated on the latest announcements? - The [Azure updates](#) page provides information about the latest updates to Azure products, services, and features, as well as product roadmaps and announcements. From the Azure updates page, you can:

- View details about all Azure updates.
- See which updates are in general availability, preview, or development.
- Browse updates by product category or update type.
- Search for updates by keyword.
- Subscribe to an RSS feed to receive notifications.
- Access the Microsoft Connect page to read Azure product news and announcements.

Modern Lifecycle Policy

The Modern Lifecycle Policy covers products and services that are serviced and supported continuously. Under this policy, the product or service remains in support if the following criteria are met:

- Customers must stay current as per the servicing and system requirements published for the product or service.
- Customers must be licensed to use the product or service.
- Microsoft must currently offer support for the product or service.

Changes for these products and services may be more frequent and require customers to be alert for forthcoming modifications to their product or service. For products and services governed by the Modern Lifecycle Policy, unless otherwise noted, Microsoft's policy is to provide a minimum 30 days' notification when customers are required to take action in order to avoid significant degradation to the normal use of the product or service.

For products governed by the Modern Lifecycle Policy, Microsoft will provide a minimum of 12 months' notification prior to ending support if no successor product or service is offered—excluding free services or preview releases. For products under existing lifecycle policies (such as the Fixed Lifecycle Policy), Microsoft will continue to provide servicing and support for a fixed amount of time. Products that have already launched with existing lifecycle policies will continue to be supported according to the [published end of support dates](#).

Azure Support

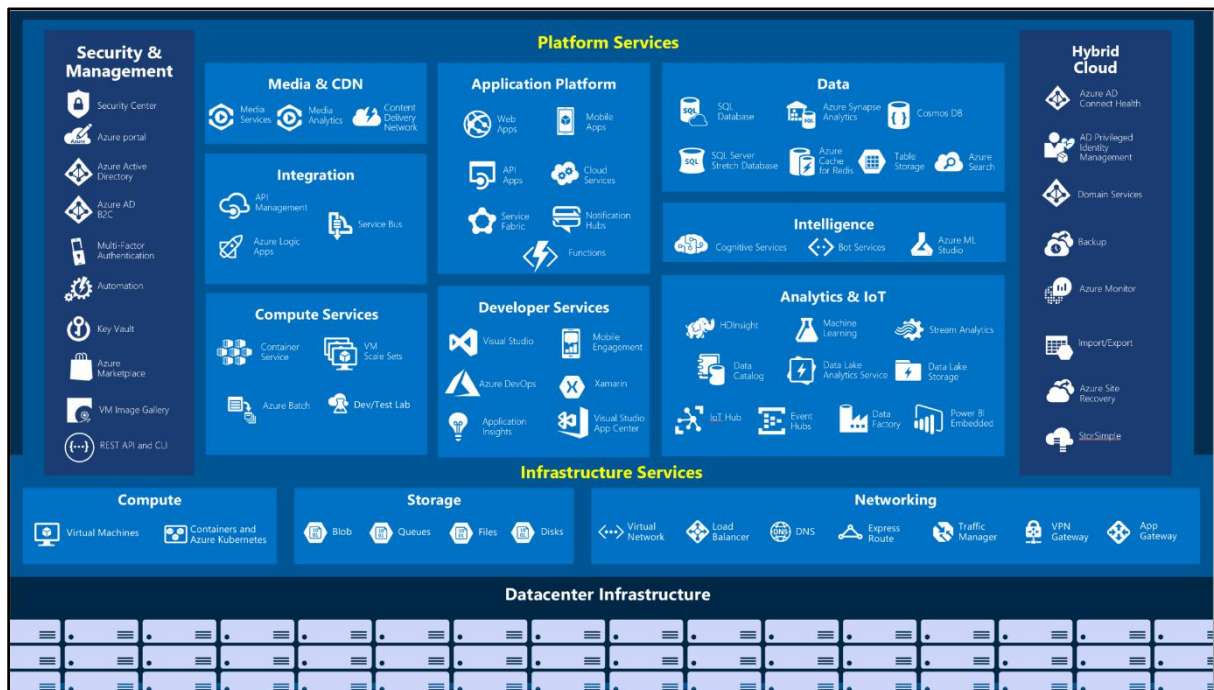
	DEVELOPER	STANDARD	PROFESSIONAL DIRECT	
Basic	Purchase support	Purchase support	Purchase support	
Request support				
Price	Included for all Azure customers	£21.62 per month	£74.54 per month	£745.31 per month
Scope	Included for all Azure customers	Trial and non-production environments	Production workload environments	Business-critical dependence
Billing and subscription management support	Available	Available	Available	Available
24/7 self-help resources, including Microsoft Learn, Azure portal how-to videos, documentation and community support	Available	Available	Available	Available
Ability to submit as many support tickets as you need	Available	Available	Available	Available
Azure Advisor – your free, personalised guide to Azure best practices	Available	Available	Available	Available
Azure health status and notifications	Available	Available	Available	Available
Third-party software support with interoperability and configuration guidance and troubleshooting	Not available	Available	Available	Available
24/7 access to technical support by email and phone after a support request has been submitted	Not available	Available during business hours by email only.	Available	Available
Case severity and response time	Not available	Minimal business impact (Sev C):	Minimal business impact (Sev C):	Minimal business impact (Sev C):
		Within eight business hours ¹	Within eight business hours ¹ Moderate business impact (Sev B):	Within four business hours ¹ Moderate business impact (Sev B): Within two hoursCritical business impact (Sev A):

			Within four hoursCritical business impact (Sev A):	Within one hour
			Within one hour	
Architecture Support	Not available	General guidance	General guidance	Guidance from a pool of ProDirect delivery managers
Support API (see details)	Not available	Not available	Not available	Create and manage Azure support tickets programmatically
Operations Support	Not available	Not available	Not available	Service reviews and advisory consultation from a pool of ProDirect delivery managers are non-transferable and limited to Azure ProDirect customers only
Training	Not available	Not available	Not available	Webinars led by Azure engineers
Proactive Guidance	Not available	Not available	Not available	From a pool of ProDirect delivery managers

Azure Services

- Compute
- Networking
- Storage
- Mobile
- Databases
- Web
- Internet of Things (IoT)
- Big data
- AI
- DevOps

Here's a big-picture view of the available services and features in Azure.



Security

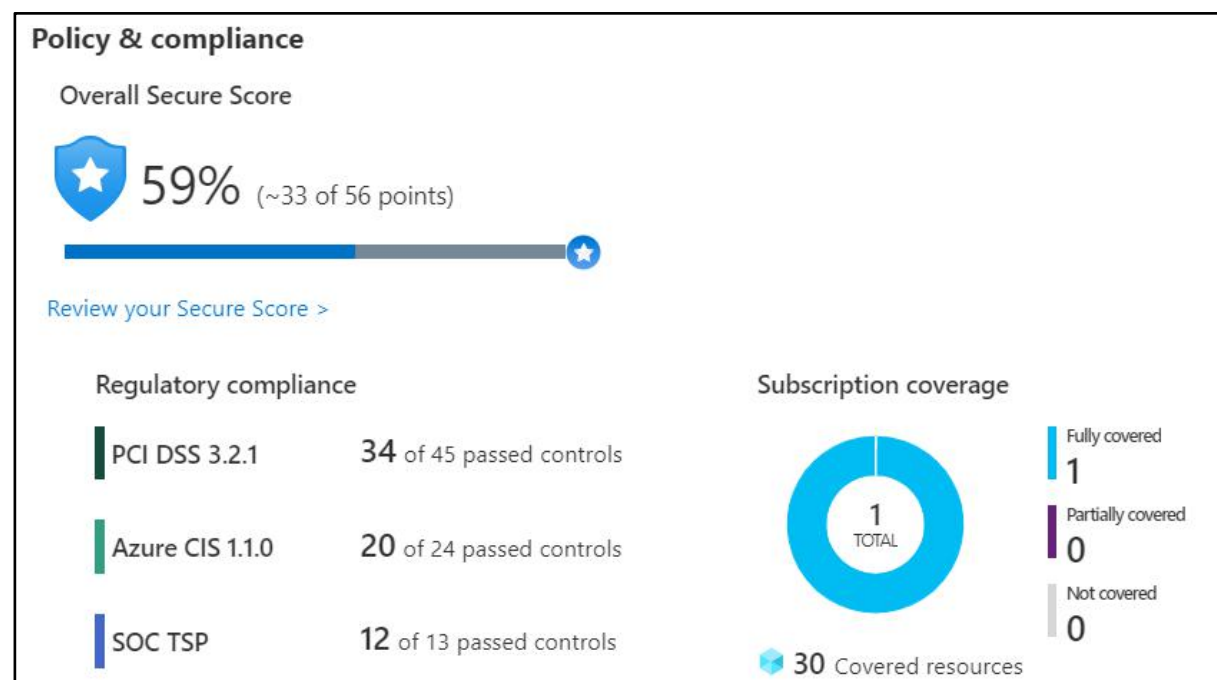
Azure Security Center

Azure Security Center is a monitoring service that provides visibility of your security posture across all of your services, both on Azure and on-premises. The term security posture refers to cybersecurity policies and controls, as well as how well you can predict, prevent, and respond to security threats.

Security Center can:

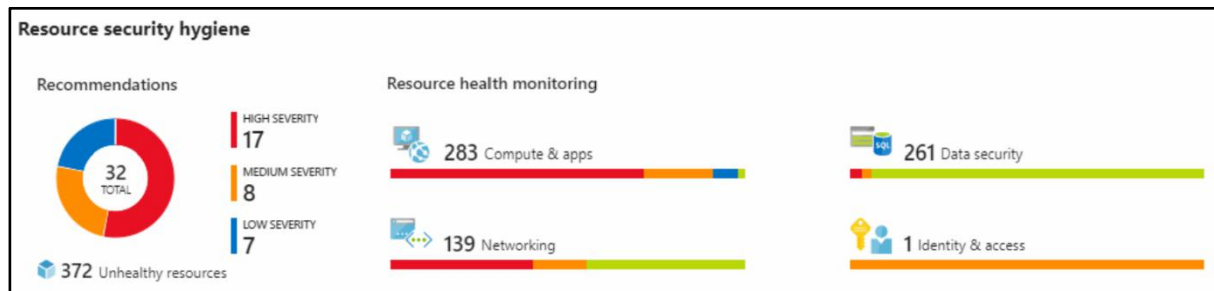
- Monitor security settings across on-premises and cloud workloads.
- Automatically apply required security settings to new resources as they come online.
- Provide security recommendations that are based on your current configurations, resources, and networks.
- Continuously monitor your resources and perform automatic security assessments to identify potential vulnerabilities before those vulnerabilities can be exploited.
- Use machine learning to detect and block malware from being installed on your virtual machines (VMs) and other resources. You can also use adaptive application controls to define rules that list allowed applications to ensure that only applications you allow can run.
- Detect and analyze potential inbound attacks and investigate threats and any post-breach activity that might have occurred.
- Provide just-in-time access control for network ports. Doing so reduces your attack surface by ensuring that the network only allows traffic that you require at the time that you need it to.

Understand your security posture - Tailwind Traders can use Security Center to get a detailed analysis of different components in its environment. Because the company's resources are analyzed against the security controls of any governance policies it has assigned, it can view its overall regulatory compliance from a security perspective all from one place. See the following example of what you might see in Azure Security Center:



Let's say that Tailwind Traders must comply with the Payment Card Industry's Data Security Standard (PCI DSS). This report shows that the company has resources that it needs to remediate.

In the Resource security hygiene section, Tailwind Traders can see the health of its resources from a security perspective. To help prioritize remediation actions, recommendations are categorized as low, medium, and high. Here's an example:



What's secure score? Secure score is a measurement of an organization's security posture. Secure score is based on security controls, or groups of related security recommendations. Your score is based on the percentage of security controls that you satisfy. The more security controls you satisfy, the higher the score you receive. Your score improves when you remediate all of the recommendations for a single resource within a control.

Following the secure score recommendations can help protect your organization from threats. From a centralized dashboard in Azure Security Center, organizations can monitor and work on the security of their Azure resources like identities, data, apps, devices, and infrastructure.

Secure score helps you:

- Report on the current state of your organization's security posture.
- Improve your security posture by providing discoverability, visibility, guidance, and control.
- Compare with benchmarks and establish key performance indicators (KPIs).
- Protect against threats

Security Center includes advanced cloud defense capabilities for VMs, network security, and file integrity. Let's look at how some of these capabilities apply to Tailwind Traders:

- Just-in-time VM access - Tailwind Traders will configure just-in-time access to VMs. This access blocks traffic by default to specific network ports of VMs, but allows traffic for a specified time when an admin requests and approves it.
- Adaptive application controls - Tailwind Traders can control which applications are allowed to run on its VMs. In the background, Security Center uses machine learning to look at the processes running on a VM. It creates exception rules for each resource group that holds the VMs and provides recommendations. This process provides alerts that inform the company about unauthorized applications that are running on its VMs.
- Adaptive network hardening - Security Center can monitor the internet traffic patterns of the VMs, and compare those patterns with the company's current network security group (NSG) settings. From there, Security Center can make recommendations about whether the NSGs should be locked down further and provide remediation steps.
- File integrity monitoring - Tailwind Traders can also configure the monitoring of changes to important files on both Windows and Linux, registry settings, applications, and other aspects that might indicate a security attack.

Respond to security alerts - Tailwind Traders can use Security Center to get a centralized view of all of its security alerts. From there, the company can dismiss false alerts, investigate them further, remediate alerts manually, or use an automated response with a workflow automation. Workflow automation uses Azure Logic Apps and Security Center connectors. The logic app can be triggered by a threat detection alert or by a Security Center recommendation, filtered by name or by severity. You can then configure the logic app to run an action, such as sending an email, or posting a message to a Microsoft Teams channel.

The advanced monitoring capabilities in Security Center lets you track and manage compliance and governance over time. The overall compliance provides you with a measure of how much your subscriptions are compliant with policies associated with your workload.

Azure Sentinel

Azure Sentinel is Microsoft's cloud-based SIEM system. It uses intelligent security analytics and threat analysis.

Security management on a large scale can benefit from a dedicated security information and event management (SIEM) system. A SIEM system aggregates security data from many different sources (as long as those sources support an open-standard logging format). It also provides capabilities for threat detection and response.

Azure Sentinel enables you to:

- Collect cloud data at scale - Collect data across all users, devices, applications, and infrastructure, both on-premises and from multiple clouds.
- Detect previously undetected threats - Minimize false positives by using Microsoft's comprehensive analytics and threat intelligence.
- Investigate threats with artificial intelligence - Examine suspicious activities at scale, tapping into years of cybersecurity experience from Microsoft.
- Respond to incidents rapidly - Use built-in orchestration and automation of common tasks.

Tailwind Traders decides to explore the capabilities of Azure Sentinel. First, the company identifies and connects its data sources. Azure Sentinel supports a number of data sources, which it can analyze for security events. These connections are handled by built-in connectors or industry-standard log formats and APIs.

- Connect Microsoft solutions - Connectors provide real-time integration for services like Microsoft Threat Protection solutions, Microsoft 365 sources (including Office 365), Azure Active Directory, and Windows Defender Firewall.
- Connect other services and solutions - Connectors are available for common non-Microsoft services and solutions, including AWS CloudTrail, Citrix Analytics (Security), Sophos XG Firewall, VMware Carbon Black Cloud, and Okta SSO.
- Connect industry-standard data sources - Azure Sentinel supports data from other sources that use the Common Event Format (CEF) messaging standard, Syslog, or REST API.

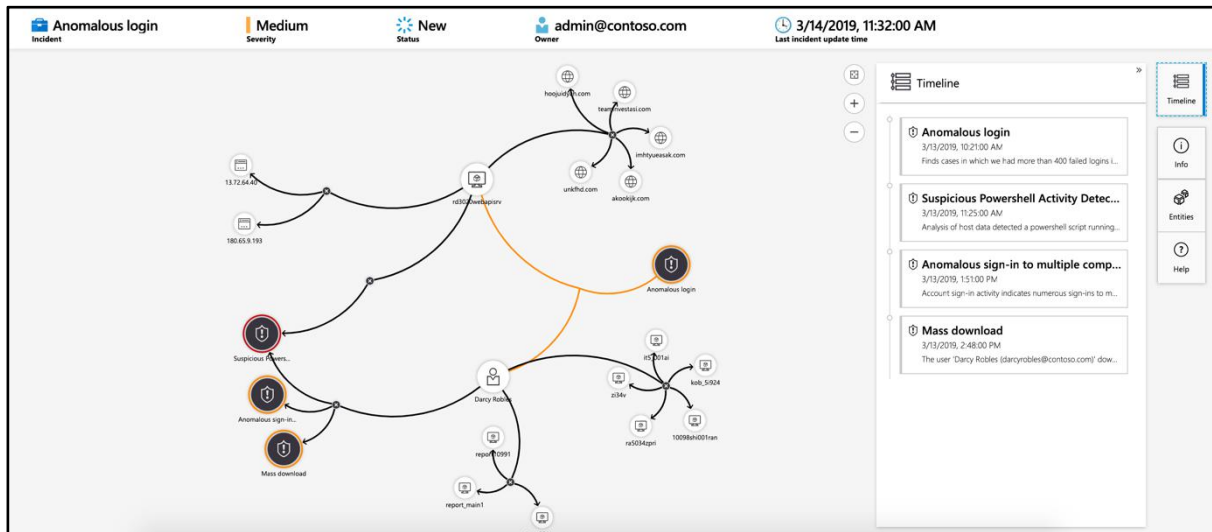
Detect threats - Tailwind Traders needs to be notified when something suspicious occurs. It decides to use both built-in analytics and custom rules to detect threats:

- Built in analytics use templates designed by Microsoft's team of security experts and analysts based on known threats, common attack vectors, and escalation chains for suspicious activity. These templates can be customized and search across the environment for any activity that looks suspicious. Some templates use machine learning behavioral analytics that are based on Microsoft proprietary algorithms.
- Custom analytics are rules that you create to search for specific criteria within your environment. You can preview the number of results that the query would generate (based on past log events) and set a schedule for the query to run. You can also set an alert threshold.

Investigate and respond

When Azure Sentinel detects suspicious events, Tailwind Traders can investigate specific alerts or incidents (a group of related alerts). With the investigation graph, the company can review information from entities directly connected to the alert, and see common exploration queries to help guide the investigation.

Here's an example that shows what an investigation graph looks like in Azure Sentinel.



The company will also use Azure Monitor Workbooks to automate responses to threats. For example, it can set an alert that looks for malicious IP addresses that access the network and create a workbook that does the following steps:

- When the alert is triggered, open a ticket in the IT ticketing system.
- Send a message to the security operations channel in Microsoft Teams or Slack to make sure the security analysts are aware of the incident.
- Send all of the information in the alert to the senior network admin and to the security admin. The email message includes two user option buttons: Block or Ignore.
- When an admin chooses Block, the IP address is blocked in the firewall, and the user is disabled in Azure Active Directory. When an admin chooses Ignore, the alert is closed in Azure Sentinel, and the incident is closed in the IT ticketing system.
- The workbook continues to run after it receives a response from the admins.

Workbooks can be run manually or automatically when a rule triggers an alert.

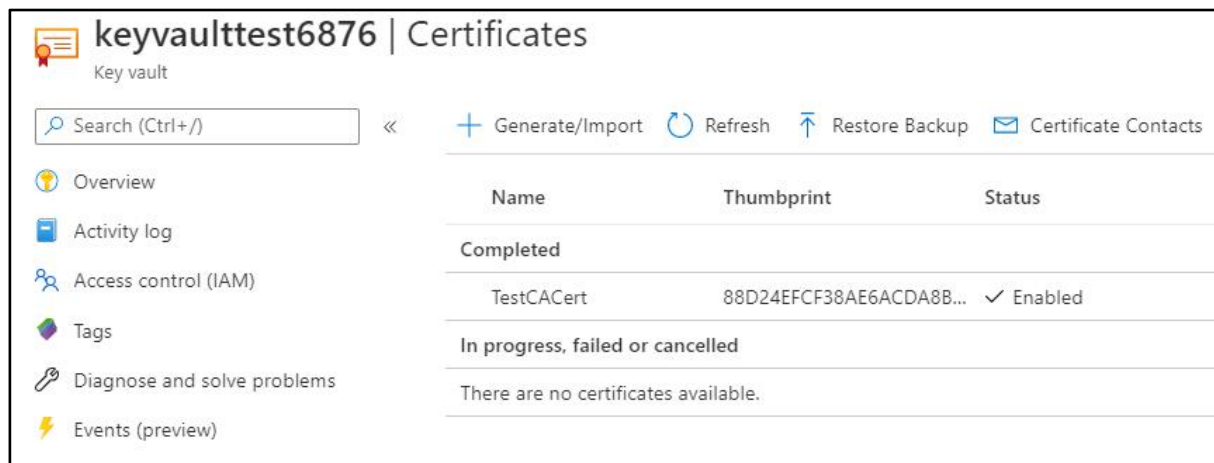
Azure Key Vault

Azure Key Vault is a centralized cloud service for storing an application's secrets in a single, central location. It provides secure access to sensitive information by providing access control and logging capabilities. As Tailwind Traders builds its workloads in the cloud, it needs to carefully handle sensitive information such as passwords, encryption keys, and certificates. This information needs to be available for an application to function, but it might allow an unauthorized person access to application data.

What can Azure Key Vault do? Azure Key Vault can help you:

- Manage secrets - You can use Key Vault to securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets.
- Manage encryption keys - You can use Key Vault as a key management solution. Key Vault makes it easier to create and control the encryption keys that are used to encrypt your data.
- Manage SSL/TLS certificates - Key Vault enables you to provision, manage, and deploy your public and private Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates for both your Azure resources and your internal resources.
- Store secrets backed by hardware security modules (HSMs) - These secrets and keys can be protected either by software or by FIPS 140-2 Level 2 validated HSMs.

Here's an example that shows a certificate used for testing in Key Vault.



What are the benefits of Azure Key Vault?

The benefits of using Key Vault include:

- Centralized application secrets – Centralizing the storage for your application secrets enables you to control their distribution, and reduces the chances that secrets are accidentally leaked.
- Securely stored secrets and keys - Azure uses industry-standard algorithms, key lengths, and HSMs. Access to Key Vault requires proper authentication and authorization.
- Access monitoring and access control - By using Key Vault, you can monitor and control access to your application secrets.
- Simplified administration of application secrets - Key Vault makes it easier to enroll and renew certificates from public certificate authorities (CAs). You can also scale up and replicate content within regions and use standard certificate management tools.
- Integration with other Azure services - You can integrate Key Vault with storage accounts, container registries, event hubs, and many more Azure services. These services can then securely reference the secrets stored in Key Vault.

Authentication with Key Vault works in conjunction with Azure Active Directory (Azure AD), which is responsible for authenticating the identity of any given security principal. Azure AD authenticates users and provides access tokens. An access token is a security token that is issued by an authorization server. It contains information about the user and the app for which the token is intended, which can be used to access Web APIs and other protected resources. Instead of creating apps that each maintain their own username and password information, which incurs a high administrative burden when you need to add or remove users across multiple apps, apps can delegate that responsibility to a centralized identity provider. Azure Active Directory (Azure AD) is a centralized identity provider in the cloud. Delegating authentication and authorization to it enables scenarios such as Conditional Access policies that require a user to be in a specific location, the use of multi-factor authentication, as well as enabling a user to sign in once and then be automatically signed in to all of the web apps that share the same centralized directory. This capability is referred to as Single Sign On (SSO).

Azure Dedicated Host

On Azure, virtual machines (VMs) run on shared hardware that Microsoft manages. Although the underlying hardware is shared, your VM workloads are isolated from workloads that other Azure customers run.

Some organizations must follow regulatory compliance that requires them to be the only customer using the physical machine that hosts their virtual machines. Azure Dedicated Host provides dedicated physical servers to host your Azure VMs for Windows and Linux.

Here's a diagram that shows how VMs relate to dedicated hosts and host groups. A dedicated host is mapped to a physical server in an Azure datacenter. A host group is a collection of dedicated hosts.



What are the benefits of Azure Dedicated Host?

- Gives you visibility into, and control over, the server infrastructure that's running your Azure VMs.
- Helps address compliance requirements by deploying your workloads on an isolated server.
- Lets you choose the number of processors, server capabilities, VM series, and VM sizes within the same host.

Availability considerations for Dedicated Host - After a dedicated host is provisioned, Azure assigns it to the physical server in Microsoft's cloud datacenter. For high availability, you can provision multiple hosts in a host group, and deploy your VMs across this group. VMs on dedicated hosts can also take advantage of maintenance control. This feature enables you to control when regular maintenance updates occur, within a 35-day rolling window.

Pricing considerations - You're charged per dedicated host, independent of how many VMs you deploy to it. The host price is based on the VM family, type (hardware size), and region. Software licensing, storage, and network usage are billed separately from the host and VMs. For more information see [Azure Dedicated Host pricing](#).

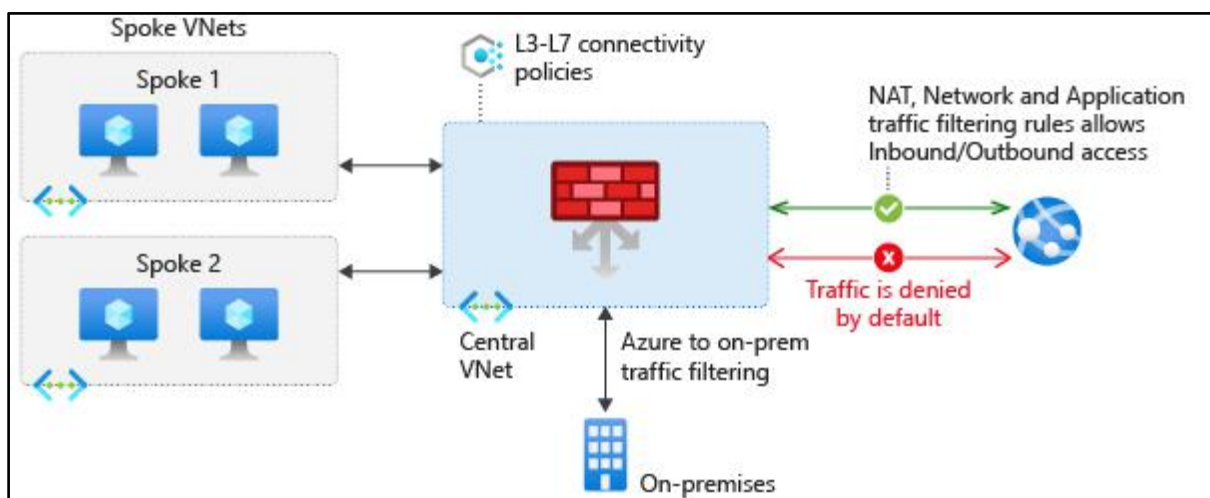
Azure Firewall

A firewall is a network security device that monitors incoming and outgoing network traffic and decides whether to allow or block specific traffic based on a defined set of security rules. You can create firewall rules that specify ranges of IP addresses. Only clients granted IP addresses from within those ranges are allowed to access the destination server. Firewall rules can also include specific network protocol and port information.

Tailwind Traders currently runs firewall appliances, which combine hardware and software, to protect its on-premises network. These firewall appliances require a monthly licensing fee to operate, and they require IT staff to perform routine maintenance. As Tailwind Traders moves to the cloud, the IT manager wants to know what Azure services can protect both the company's cloud networks and its on-premises networks.

Azure Firewall is a managed, cloud-based network security service that helps protect resources in your Azure virtual networks. A virtual network is similar to a traditional network that you'd operate in your own datacenter. It's a fundamental building block for your private network that enables virtual machines and other compute resources to securely communicate with each other, the internet, and on-premises networks.

Here's a diagram that shows a basic Azure Firewall implementation:



Azure Firewall is a stateful firewall. A stateful firewall analyzes the complete context of a network connection, not just an individual packet of network traffic. Azure Firewall features high availability and unrestricted cloud scalability.

Azure Firewall provides a central location to create, enforce, and log application and network connectivity policies across subscriptions and virtual networks. Azure Firewall uses a static (unchanging) public IP address for your virtual network resources, which enables outside firewalls to identify traffic coming from your virtual network. The service is integrated with Azure Monitor to enable logging and analytics.

Azure Firewall provides many features, including:

- Built-in high availability.
- Unrestricted cloud scalability.
- Inbound and outbound filtering rules.
- Inbound Destination Network Address Translation (DNAT) support.
- Azure Monitor logging.

You typically deploy Azure Firewall on a central virtual network to control general network access.

What can I configure with Azure Firewall? With Azure Firewall, you can configure:

- Application rules that define fully qualified domain names (FQDNs) that can be accessed from a subnet.
- Network rules that define source address, protocol, destination port, and destination address.
- Network Address Translation (NAT) rules that define destination IP addresses and ports to translate inbound requests.

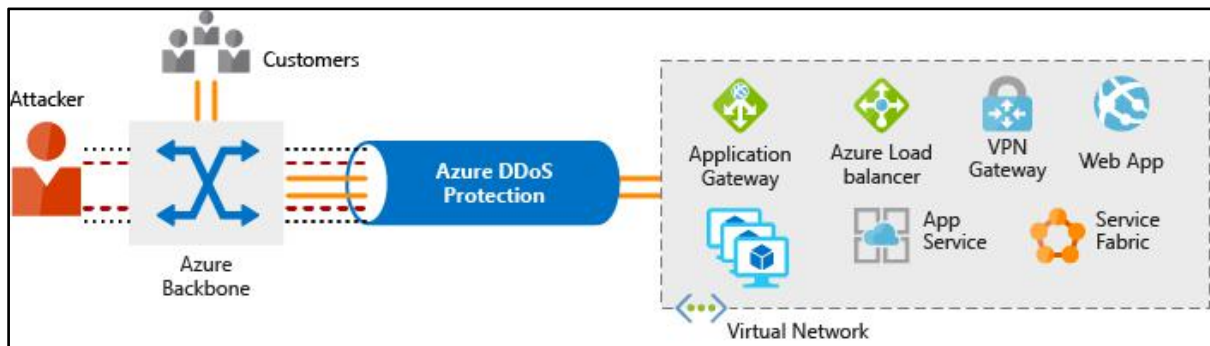
Azure Application Gateway also provides a firewall that's called the web application firewall (WAF). WAF provides centralized, inbound protection for your web applications against common exploits and vulnerabilities. Azure Front Door and Azure Content Delivery Network also provide WAF services.

Azure DDoS Protection

Any large company can be the target of a large-scale network attack. Tailwind Traders is no exception. Attackers might flood your network to make a statement or simply for the challenge. As Tailwind Traders moves to the cloud, it wants to understand how Azure can help prevent distributed denial of service (DDoS) and other attacks. In this part, you learn how Azure DDoS Protection (Standard service tier) helps protect your Azure resources from DDoS attacks. First, let's define what a DDoS attack is.

What are DDoS attacks? - A distributed denial of service attack attempts to overwhelm and exhaust an application's resources, making the application slow or unresponsive to legitimate users. DDoS attacks can target any resource that's publicly reachable through the internet, including websites.

What is Azure DDoS Protection? - Azure DDoS Protection (Standard) helps protect your Azure resources from DDoS attacks. When you combine DDoS Protection with recommended application design practices, you help provide a defense against DDoS attacks. DDoS Protection uses the scale and elasticity of Microsoft's global network to bring DDoS mitigation capacity to every Azure region. The DDoS Protection service helps protect your Azure applications by analyzing and discarding DDoS traffic at the Azure network edge, before it can affect your service's availability. This diagram shows network traffic flowing into Azure from both customers and an attacker:



DDoS Protection identifies the attacker's attempt to overwhelm the network and blocks further traffic from them, ensuring that traffic never reaches Azure resources. Legitimate traffic from customers still flows into Azure without any interruption of service.

DDoS Protection can also help you manage your cloud consumption. When you run on-premises, you have a fixed number of compute resources. But in the cloud, elastic computing means that you can automatically scale out your deployment to meet demand. A cleverly designed DDoS attack can cause you to increase your resource allocation, which incurs unneeded expense. DDoS Protection Standard helps ensure that the network load you process reflects customer usage. You can also receive credit for any costs accrued for scaled-out resources during a DDoS attack.

What service tiers are available to DDoS Protection? DDoS Protection provides these service tiers:

- **Basic** - The Basic service tier is automatically enabled for free as part of your Azure subscription. Always-on traffic monitoring and real-time mitigation of common network-level attacks provide the same defenses that Microsoft's online services use. The Basic service tier ensures that Azure infrastructure itself is not affected during a large-scale DDoS attack. The Azure global network is used to distribute and mitigate attack traffic across Azure regions.
- **Standard** - The Standard service tier provides additional mitigation capabilities that are tuned specifically to Azure Virtual Network resources. DDoS Protection Standard is relatively easy to enable and requires no changes to your applications. The Standard tier provides always-on traffic monitoring and real-time mitigation of common network-level attacks. It provides the same defenses that Microsoft's online services use. Protection policies are tuned through dedicated traffic monitoring and machine learning algorithms. Policies are applied to public IP addresses, which are associated with resources deployed in virtual networks such as Azure Load Balancer and Application Gateway. The Azure global network is used to distribute and mitigate attack traffic across Azure regions.

What kinds of attacks can DDoS Protection help prevent?

- The Standard service tier can help prevent:
 - Volumetric attacks - The goal of this attack is to flood the network layer with a substantial amount of seemingly legitimate traffic.
 - Protocol attacks - These attacks render a target inaccessible by exploiting a weakness in the layer 3 and layer 4 protocol stack.
 - Resource-layer (application-layer) attacks (only with web application firewall) - These attacks target web application packets to disrupt the transmission of data between hosts. You need a web application firewall (WAF) to protect against L7 attacks. DDoS Protection Standard protects the WAF from volumetric and protocol attacks.

Network Security Groups

Although Azure Firewall and Azure DDoS Protection can help control what traffic can come from outside sources, Tailwind Traders also wants to understand how to protect its internal networks on Azure. Doing so will give the company an extra layer of defense against attacks. In this part, you examine network security groups (NSGs).

What are network security groups? - A network security group enables you to filter network traffic to and from Azure resources within an Azure virtual network. You can think of NSGs like an internal firewall. An NSG can contain multiple inbound and outbound security rules that enable you to filter traffic to and from resources by source and destination IP address, port, and protocol.

How do I specify NSG rules?

A network security group can contain as many rules as you need, within Azure subscription limits. Each rule specifies these properties:

Property	Description
Name	A unique name for the NSG.
Priority	A number between 100 and 4096. Rules are processed in priority order, with lower numbers processed before higher numbers.
Source or Destination	A single IP address or IP address range, service tag, or application security group.
Protocol	TCP, UDP, or Any.
Direction	Whether the rule applies to inbound or outbound traffic.
Port Range	A single port or range of ports.
Action	Allow or Deny.

When you create a network security group, Azure creates a series of default rules to provide a baseline level of security. You can't remove the default rules, but you can override them by creating new rules with higher priorities.

Combine Azure services to create a complete network security solution

When you're considering an Azure security solution, consider all the elements of defense in depth.

Here are some recommendations on how to combine Azure services to create a complete network security solution.

Secure the perimeter layer

The perimeter layer is about protecting your organization's resources from network-based attacks. Identifying these attacks, alerting the appropriate security teams, and eliminating their impact are important to keeping your network secure. To do this:

- Use Azure DDoS Protection to filter large-scale attacks before they can cause a denial of service for users
- Use perimeter firewalls with Azure Firewall to identify and alert on malicious attacks against your network.

Secure the network layer

At this layer, the focus is on limiting network connectivity across all of your resources to allow only what's required. Segment your resources and use network-level controls to restrict communication to only what's needed.

By restricting connectivity, you reduce the risk of lateral movement throughout your network from an attack. Use network security groups to create rules that define allowed inbound and outbound communication at this layer. Here are some recommended practices:

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

Combine services

You can combine Azure networking and security services to manage your network security and provide increased layered protection. Here are two ways you can combine services:

- Network security groups and Azure Firewall - Azure Firewall complements the functionality of network security groups. Together, they provide better defense-in-depth network security. Azure Firewall is a fully stateful, centralized network firewall as a service. It provides network-level and application-level protection across different subscriptions and virtual networks. Network security groups provide distributed network-layer traffic filtering to limit traffic to resources within virtual networks in each subscription.
- Azure Application Gateway web application firewall and Azure Firewall - Combining them provides more layers of protection. Web application firewall (WAF) is a feature of Azure Application Gateway that provides your web applications with centralized, inbound protection against common exploits and vulnerabilities. Azure Firewall provides:
 - Inbound protection for non-HTTP/S protocols (for example, RDP, SSH, and FTP).
 - Outbound network-level protection for all ports and protocols.
 - Application-level protection for outbound HTTP/S.

Azure management tools

At a high level, there are two broad categories of management tools: visual tools and code-based tools.

Visual tools provide full, visually friendly access to all the functionality of Azure. However, visual tools might be less useful when you're trying to set up a large deployment of resources with interdependencies and configuration options.

When you're attempting to quickly set up and configure Azure resources, a code-based tool is usually the better choice. Although it might take time to understand the right commands and parameters at first, after they've been entered, they can be saved into files and used repeatedly as needed. Also, the code that performs setup and configuration can be stored, versioned, and maintained along with application source code in a source code-management tool such as Git. This approach to managing hardware and cloud resources, which developers use when they write application code, is referred to as infrastructure as code.

There are two approaches to infrastructure as code: imperative code and declarative code. Imperative code details each individual step that should be performed to achieve a desired outcome. By contrast, declarative code details only a desired outcome, and it allows an interpreter to decide how to best achieve that outcome. This distinction is important because tools that are based on declarative code can provide a more robust approach to deploying dozens or hundreds of resources simultaneously and reliably.

The Azure portal

By using the Azure portal, a web-based user interface, you can access virtually every feature of Azure. The Azure portal provides a friendly, graphical UI to view all the services you're using, create new services, configure your services, and view reports. The Azure portal is how most users first experience Azure. But, as your Azure usage grows, you'll likely choose a more repeatable code-centric approach to managing your Azure resources.

The Azure mobile app

The Azure mobile app provides iOS and Android access to your Azure resources when you're away from your computer. With it, you can:

- Monitor the health and status of your Azure resources.

- Check for alerts, quickly diagnose and fix issues, and restart a web app or virtual machine (VM).

- Run the Azure CLI or Azure PowerShell commands to manage your Azure resources.

Azure PowerShell

Azure PowerShell is a shell with which developers and DevOps and IT professionals can execute commands called cmdlets (pronounced command-lets). These commands call the Azure Rest API to perform every possible management task in Azure. A PowerShell script needs to be run in PowerShell. Cmdlets can be executed independently or combined into a script file and executed together to orchestrate:

- The routine setup, teardown, and maintenance of a single resource or multiple connected resources.

- The deployment of an entire infrastructure, which might contain dozens or hundreds of resources, from imperative code.

- Capturing the commands in a script makes the process repeatable and automatable.

Azure PowerShell is available for Windows, Linux, and Mac, and you can access it in a web browser via Azure Cloud Shell.

Windows PowerShell has helped Windows-centric IT organizations automate many of their on-premises operations for years, and these organizations have built up a large catalog of custom scripts and cmdlets, as well as expertise.

The Azure CLI

The Azure CLI command-line interface is an executable program with which a developer, DevOps professional, or IT professional can execute commands in Bash. The commands call the Azure Rest API to perform every possible management task in Azure. You can run the commands independently or combined into a script and executed together for the routine setup, teardown, and maintenance of a single resource or an entire environment.

In many respects, the Azure CLI is almost identical to Azure PowerShell in what you can do with it. Both run on Windows, Linux, and Mac, and can be accessed in a web browser via Cloud Shell. The primary difference is the syntax you use. If you're already proficient in PowerShell or Bash, you can use the tool you prefer.

ARM templates

Although it's possible to write imperative code in Azure PowerShell or the Azure CLI to set up and tear down one Azure resource or orchestrate an infrastructure comprising hundreds of resources, there's a better way to implement this functionality.

By using Azure Resource Manager templates (ARM templates), you can describe the resources you want to use in a declarative JSON format. The benefit is that the entire ARM template is verified before any code is executed to ensure that the resources will be created and connected correctly. The template then orchestrates the creation of those resources in parallel. That is, if you need 50 instances of the same resource, all 50 instances are created at the same time.

Ultimately, the developer, DevOps professional, or IT professional needs only to define the desired state and configuration of each resource in the ARM template, and the template does the rest. Templates can even execute PowerShell and Bash scripts before or after the resource has been set up.

Do you need to perform one-off management, administrative, or reporting actions? Azure PowerShell and the Azure CLI are Azure management tools that allow you to quickly obtain the IP address of a virtual machine (VM) you've deployed, reboot a VM, or scale an app. You might want to keep custom scripts for both tools handy on your local hard drive for certain operations that you need to perform multiple times.

By contrast to the Azure CLI and PowerShell, Azure Resource Manager templates (ARM templates) define the infrastructure requirements in your application for repeatable deployments. Although ARM templates aren't intended for one-off scenarios, it's possible to use them for this purpose. However, for one-off scenarios, you may prefer more agile tools like PowerShell, Azure CLI scripts, or the Azure portal.

Keep in mind that ARM templates can include both PowerShell and/or Azure CLI scripts, which will give you the ability to utilize scripts for tasks that may not be possible with the ARM template itself. The ability to combine Azure management tools gives flexibility in choosing the right tool(s) for your particular need.

The Azure portal can perform most, if not all, management and administrative actions. If you're just learning Azure and/or need to set up and manage resources infrequently (or prefer a visual interface for viewing reports), it makes sense to take advantage of the visual presentation that the Azure portal offers.

However, if you're in a cloud management or administrative role, it's less efficient to rely solely on visual scanning and clicking. To quickly find the settings and information you want to work with, the Azure CLI or PowerShell will give you the most flexibility for repeatable tasks.

The last management tool to discuss is the Azure mobile app, which you can access via an iOS or Android phone or tablet. Because it's full featured, it's likely the best choice when a laptop isn't readily available and you need to view and triage (determine priorities of) issues immediately.

Do you need a way to repeatedly set up one or more resources and ensure that all the dependencies are created in the proper order? ARM templates define your application's infrastructure requirements for a repeatable deployment that is done in a consistent manner. A validation step ensures that all resources can be created in the proper order based on dependencies, in parallel, and idempotent (the property that a deployment command always sets the target environment into the same configuration, regardless of the environment's starting state).

By contrast, it's entirely possible to use either PowerShell or the Azure CLI to set up all the resources for a deployment. However, there's no validation step in these tools. If a script encounters an error, the dependency resources can't be rolled back easily, deployments happen serially, and only some operations are idempotent.

When you're scripting, do you come from a Windows administration or Linux administration background? If you or your cloud administrators come from a Windows administration background, it's likely you'll prefer PowerShell. If you or your cloud administrators come from a Linux administration background, it's likely you'll prefer the Azure CLI. In practice, either tool can be used to perform most one-off administration tasks.

Compute Services

Compute services are often one of the primary reasons why companies move to the Azure platform. Azure provides a range of options for hosting applications and services.

Azure compute is an on-demand computing service for running cloud-based applications. It provides computing resources such as disks, processors, memory, networking, and operating systems. The resources are available on-demand and can typically be made available in minutes or even seconds. You pay only for the resources you use, and only for as long as you're using them.

Azure supports a wide range of computing solutions for development and testing, running applications, and extending your datacenter. The service supports Linux, Windows Server, SQL Server, Oracle, IBM, and SAP. Azure also has many services that can run virtual machines (VMs). Each service provides different options depending on your requirements.

Here are some examples of compute services in Azure:

Azure Virtual Machines	Windows or Linux virtual machines (VMs) hosted in Azure
Azure Virtual Machine Scale Sets	Scaling for Windows or Linux VMs hosted in Azure
Azure Kubernetes Service	Cluster management for VMs that run containerized services
Azure Service Fabric	Distributed systems platform that runs in Azure or on-premises
Azure Batch	Managed service for parallel and high-performance computing applications
Azure Container Instances	Containerized apps run on Azure without provisioning servers or VMs
Azure Functions	An event-driven, serverless compute service

Virtual Machines

Virtual machines are software emulations of physical computers. They include a virtual processor, memory, storage, and networking resources. VMs host an operating system, and you can install and run software just like a physical computer. When using a remote desktop client, you can use and control the VM as if you were sitting in front of it.

With Azure Virtual Machines, you can create and use VMs in the cloud. Virtual Machines provides infrastructure as a service (IaaS) and can be used in different ways. When you need total control over an operating system and environment, VMs are an ideal choice. Just like a physical computer, you can customize all the software running on the VM. This ability is helpful when you're running custom software or custom hosting configurations.

With Azure Virtual Machines, you can create and use VMs in the cloud. VMs provide infrastructure as a service (IaaS) in the form of a virtualized server and can be used in many ways. Just like a physical computer, you can customize all of the software running on the VM. VMs are an ideal choice when you need:

- Total control over the operating system (OS).
- The ability to run custom software.
- To use custom hosting configurations.

An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs the VM. You still need to configure, update, and maintain the software that runs on the VM.

You can create and provision a VM in minutes when you select a preconfigured VM image. Selecting an image is one of the most important decisions you'll make when you create a VM. An image is a template used to create a VM. These templates already include an OS and often other software, like development tools or web hosting environments.

In the Azure virtual machines page in the Azure portal, there is a column named Maintenance Status. This column will display service issues that could affect your virtual machine. A service failure is rare but host server maintenance that could affect your virtual machines is more common. Azure periodically updates its platform to improve the reliability, performance, and security of the host infrastructure for virtual machines. The purpose of these updates ranges from patching software components in the hosting environment to upgrading networking components or decommissioning hardware.

Examples of when to use VMs:

- During testing and development. VMs provide a quick and easy way to create different OS and application configurations. Test and development personnel can then easily delete the VMs when they no longer need them.
- When running applications in the cloud. The ability to run certain applications in the public cloud as opposed to creating a traditional infrastructure to run them can provide substantial economic benefits. For example, an application might need to handle fluctuations in demand. Shutting down VMs when you don't need them or quickly starting them up to meet a sudden increase in demand means you pay only for the resources you use.
- When extending your datacenter to the cloud. An organization can extend the capabilities of its own on-premises network by creating a virtual network in Azure and adding VMs to that virtual network. Applications like SharePoint can then run on an Azure VM instead of running locally. This arrangement makes it easier or less expensive to deploy than in an on-premises environment.
- During disaster recovery. As with running certain types of applications in the cloud and extending an on-premises network to the cloud, you can get significant cost savings by using an IaaS-based approach to disaster recovery. If a primary datacenter fails, you can create VMs running on Azure to run your critical applications and then shut them down when the primary datacenter becomes operational again.
- Move to the cloud with VMs: VMs are also an excellent choice when you move from a physical server to the cloud (also known as lift and shift). You can create an image of the physical server and host it within a VM with little or no changes. Just like a physical on-premises server, you must maintain the VM. You update the installed OS and the software it runs.

Scale VMs in Azure

You can run single VMs for testing, development, or minor tasks. Or you can group VMs together to provide high availability, scalability, and redundancy. No matter what your uptime requirements are, Azure has several features that can meet them. These features include:

- Virtual machine scale sets
- Azure Batch

Virtual machine scale sets

Virtual machine scale sets are an Azure compute resource that you can use to deploy and manage a set of identical VMs. With all VMs configured the same, virtual machine scale sets are designed to support true autoscale. No pre-provisioning of VMs is required. For this reason, it's easier to build large-scale services targeting big compute, big data, and containerized workloads. As demand goes up, more VM instances can be added. As demand goes down, VM instances can be removed. The process can be manual, automated, or a combination of both.

Virtual machine scale sets let you create and manage a group of identical, load-balanced VMs. Imagine you're running a website that enables scientists to upload astronomy images that need to be processed. If you duplicated the VM, you'd normally need to configure an additional service to route requests between multiple instances of the website. Virtual machine scale sets could do that work for you.

Scale sets allow you to centrally manage, configure, and update a large number of VMs in minutes to provide highly available applications. The number of VM instances can automatically increase or decrease in response to demand or a defined schedule. With virtual machine scale sets, you can build large-scale services for areas such as compute, big data, and container workloads.

Azure Batch

Azure Batch enables large-scale parallel and high-performance computing (HPC) batch jobs with the ability to scale to tens, hundreds, or thousands of VMs.

When you're ready to run a job, Batch does the following:

- Starts a pool of compute VMs for you.
- Installs applications and staging data.
- Runs jobs with as many tasks as you have.
- Identifies failures.
- Requeues work.
- Scales down the pool as work completes.

There might be situations in which you need raw computing power or supercomputer-level compute power. Azure provides these capabilities.

Containers and Kubernetes

While virtual machines are an excellent way to reduce costs versus the investments that are necessary for physical hardware, they're still limited to a single operating system per virtual machine. If you want to run multiple instances of an application on a single host machine, containers are an excellent choice.

A Container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another.

Container Instances and Azure Kubernetes Service are Azure compute resources that you can use to deploy and manage containers. Containers are lightweight, virtualized application environments. They're designed to be quickly created, scaled out, and stopped dynamically. You can run multiple instances of a containerized application on a single host machine.

What are containers? Containers are a virtualization environment. Much like running multiple virtual machines on a single physical host, you can run multiple containers on a single physical or virtual host. Unlike virtual machines, you don't manage the operating system for a container. Virtual machines appear to be an instance of an operating system that you can connect to and manage, but containers are lightweight and designed to be created, scaled out, and stopped dynamically. While it's possible to create and deploy virtual machines as application demand increases, containers are designed to allow you to respond to changes on demand. With containers, you can quickly restart in case of a crash or hardware interruption. One of the most popular container engines is Docker, which is supported by Azure.

VMs can only run one OS at a time, so you have multiple server apps that require different OS's it will mean that you will require multiple VMs. VM's also emulate an entire computer so tasks such as starting a VM or taking a snapshot are comparatively slow, often taking several minutes. Containers are lighter weight and resolves these limitations.

VM's virtualise the hardware, containers virtualise the OS. VM's offer complete control. Containers offer portability, performance and management capabilities. Containers offer cluster orchestration options. Containerised applications are often smaller than their non-containerised counterparts. Development of containerised applications is advantageous also, because the end-user runtime environment can be determined with much less variability between user setups.

Managing Containers

Containers are managed through a container orchestrator, which can start, stop, and scale out application instances as needed. There are two ways to manage both Docker and Microsoft-based containers in Azure:

- **Azure Container Instances:** Azure Container Instances offers the fastest and simplest way to run a container in Azure without having to manage any virtual machines or adopt any additional services. It's a platform as a service (PaaS) offering that allows you to upload your containers, which it runs for you.
- **Azure Kubernetes Service (AKS):** The task of automating, managing, and interacting with a large number of containers is known as orchestration. Azure Kubernetes Service is a complete orchestration service for containers with distributed architectures and large volumes of containers.

Use containers in your solutions: Containers are often used to create solutions by using a microservice architecture. This architecture is where you break solutions into smaller, independent pieces. For example, you might split a website into a container hosting your front end, another hosting your back end, and a third for storage. This split allows you to separate portions of your app into logical sections that can be maintained, scaled, or updated independently. Imagine your website back-end has reached capacity but the front end and storage aren't being stressed. You could:

- Scale the back end separately to improve performance.
- Decide to use a different storage service.
- Replace the storage container without affecting the rest of the application.

What is a microservice? An autonomous web service with a small well-defined scope, that is loosely coupled from any other web service. Generally, microservices run with other microservices to create a full architecture, with each microservice being self-contained and taking control of a separate business capability. Microservices can be developed independently, each with their separate code base which can be of any language. Microservices can be scaled independently. Microservices control their own data and external state. This approach gives a layer of fault isolation, meaning that if one microservice fails it will not affect the entire system and can be easily replaced with a new instance. They can communicate with other microservices using well-defined APIs, or by using an orchestration/management layer via a higher-level consuming application that coordinates calls to various lower level microservices and combines results. Benefits of microservices include a high release velocity for development, high scalability, more flexible/manageable development team sizes and ability to break-down complex domains of an application into more manageable sub-domains.



App Service

App Service is an HTTP-based service that enables you to build and host many types of web-based solutions without managing infrastructure. For example, you can host web apps, mobile back ends, and RESTful APIs in several supported programming languages. Applications developed in .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python can run in and scale with ease on both Windows-based and Linux-based environments.

With Azure App Service, you can quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. You can meet rigorous performance, scalability, security, and compliance requirements while using a fully managed platform to perform infrastructure maintenance. App Service is a platform as a service (PaaS) offering.

App Service enables you to build and host web apps, background jobs, mobile back-ends, and RESTful APIs in the programming language of your choice without managing infrastructure. It offers automatic scaling and high availability. App Service supports Windows and Linux and enables automated deployments from GitHub, Azure DevOps, or any Git repo to support a continuous deployment model.

This platform as a service (PaaS) environment allows you to focus on the website and API logic while Azure handles the infrastructure to run and scale your web applications.

Azure App Service costs: You pay for the Azure compute resources your app uses while it processes requests based on the App Service plan you choose. The App Service plan determines how much hardware is devoted to your host. For example, the plan determines whether it's dedicated or shared hardware and how much memory is reserved for it. There's even a free tier you can use to host small, low-traffic sites.

With App Service, you can host most common app service styles like:

- **Web apps** - App Service includes full support for hosting web apps by using ASP.NET, ASP.NET Core, Java, Ruby, Node.js, PHP, or Python. You can choose either Windows or Linux as the host operating system.
- **API apps** - Much like hosting a website, you can build REST-based web APIs by using your choice of language and framework. You get full Swagger support and the ability to package and publish your API in Azure Marketplace. The produced apps can be consumed from any HTTP- or HTTPS-based client.
- **WebJobs** - You can use the WebJobs feature to run a program (.exe, Java, PHP, Python, or Node.js) or script (.cmd, .bat, PowerShell, or Bash) in the same context as a web app, API app, or mobile app. They can be scheduled or run by a trigger. WebJobs are often used to run background tasks as part of your application logic.
- **Mobile apps** - Use the Mobile Apps feature of App Service to quickly build a back end for iOS and Android apps. With just a few clicks in the Azure portal, you can:
 - Store mobile app data in a cloud-based SQL database.
 - Authenticate customers against common social providers, such as MSA, Google, Twitter, and Facebook.
 - Send push notifications.
 - Execute custom back-end logic in C# or Node.js.
 - On the mobile app side, there's SDK support for native iOS and Android, Xamarin, and React native apps

App Service handles most of the infrastructure decisions you deal with in hosting web-accessible apps:

- Deployment and management are integrated into the platform.
- Endpoints can be secured.
- Sites can be scaled quickly to handle high traffic loads.
- The built-in load balancing and traffic manager provide high availability.

All of these app styles are hosted in the same infrastructure and share these benefits. This flexibility makes App Service the ideal choice to host web-oriented applications.

Azure Functions

This is an Azure serverless compute offering. Functions are ideal when you're concerned only about the code running your service and not the underlying platform or infrastructure. They're commonly used when you need to perform work in response to an event (often via a REST request), timer, or message from another Azure service, and when that work can be completed quickly, within seconds or less. Functions can execute code in almost any modern language.

Functions scale automatically based on demand, so they're a solid choice when demand is variable. For example, you might receive messages from an IoT solution that's used to monitor a fleet of delivery vehicles. You'll likely have more data arriving during business hours.

Using a virtual machine-based approach, you'd incur costs even when the virtual machine is idle. With functions, Azure runs your code when it's triggered and automatically deallocates resources when the function is finished. In this model, you're only charged for the CPU time used while your function runs.

Functions can be either stateless or stateful. When they're stateless (the default), they behave as if they're restarted every time they respond to an event. When they're stateful (called Durable Functions), a context is passed through the function to track prior activity.

Functions are a key component of serverless computing. They're also a general compute platform for running any type of code. If the needs of the developer's app change, you can deploy the project in an environment that isn't serverless. This flexibility allows you to manage scaling, run on virtual networks, and even completely isolate the functions.

With the Azure Functions service, you can host a single method or function by using a popular programming language in the cloud that runs in response to an event. An example of an event might be an HTTP request, a new message on a queue, or a message on a timer.

Because of its atomic nature, Azure Functions can serve many purposes in an application's design. Functions can be written in many common programming languages, such as C#, Python, JavaScript, Typescript, Java, and PowerShell.

Azure Functions scales automatically, and charges accrue only when a function is triggered. These qualities make Azure Functions a solid choice when demand is variable. For example, you might be receiving messages from an IoT solution that monitors a fleet of delivery vehicles. You'll likely have more data arriving during business hours. Azure Functions can scale out to accommodate these busier times.

An Azure function is a stateless environment. A function behaves as if it's restarted every time it responds to an event. This feature is ideal for processing incoming data. And if state is required, the function can be connected to an Azure storage account.

Azure Functions can perform orchestration tasks by using an extension called Durable Functions, which allow developers to chain functions together while maintaining state.

The Azure Functions solution is ideal when you're concerned only with the code that's running your service and not the underlying platform or infrastructure. You use Azure Functions most commonly when you need to perform work in response to an event. You do this often via a REST request, timer, or message from another Azure service, and when that work can be completed quickly, within seconds or less.

When you're using the Consumption plan, instances of the Azure Functions host are dynamically added and removed based on the number of incoming events. The Consumption plan is the fully serverless hosting option for Azure Functions. The Consumption plan scales automatically, even during periods of high load. When running functions in a Consumption plan, you're charged for compute resources only when your functions are running. On a Consumption plan, a function execution times out after a configurable period of time. Billing is based on number of executions, execution time, and memory used. Usage is aggregated across all functions within a function app.

Azure Logic Apps

Logic apps are Azure serverless compute offering, they are similar to Functions. Both enable you to trigger logic based on an event. Where functions execute code, logic apps execute workflows that are designed to automate business scenarios and are built from predefined logic blocks.

Every Azure logic app workflow starts with a trigger, which fires when a specific event happens or when newly available data meets specific criteria. Many triggers include basic scheduling capabilities, so developers can specify how regularly their workloads will run. Each time the trigger fires, the Logic Apps engine creates a logic app instance that runs the actions in the workflow. These actions can also include data conversions and flow controls, such as conditional statements, switch statements, loops, and branching.

You create logic app workflows by using a visual designer on the Azure portal or in Visual Studio. The workflows are persisted as a JSON file with a known workflow schema.

Azure provides more than 200 different connectors and processing blocks to interact with different services. These resources include the most popular enterprise apps. You can also build custom connectors and workflow steps if the service you need to interact with isn't covered. You then use the visual designer to link connectors and blocks together. You pass data through the workflow to do custom processing, often all without writing any code.

As an example, let's say a ticket arrives in Zendesk. You could:

- Detect the intent of the message with cognitive services.
- Create an item in SharePoint to track the issue.
- Add the customer to your Dynamics 365 CRM system if they aren't already in your database.
- Send a follow-up email to acknowledge their request.

All of those actions could be designed in a visual designer, which makes it easy to see the logic flow. For this reason, it's ideal for a business analyst role.

Logic Apps is a low-code/no-code development platform hosted as a cloud service. The service helps you automate and orchestrate tasks, business processes, and workflows when you need to integrate apps, data, systems, and services across enterprises or organizations. Logic Apps simplifies how you design and build scalable solutions, whether in the cloud, on-premises, or both. This solution covers app integration, data integration, system integration, enterprise application integration (EAI), and business-to-business (B2B) integration.

Azure Logic Apps is designed in a web-based designer and can execute logic that's triggered by Azure services without writing any code. You build an app by linking triggers to actions with connectors. A trigger is an event (such as a timer) that causes an app to execute, then a new message to be sent to a queue, or an HTTP request. An action is a task or step that can execute. There are logic actions such as those you would find in most programming languages. Examples of actions include working with variables, decision statements and loops, and tasks that parse and modify data.

To build enterprise integration solutions with Azure Logic Apps, you can choose from a growing gallery of over 200 connectors. The gallery includes services such as Salesforce, SAP, Oracle DB, and file shares. If you can't find the action or connector you need, you can build your own by using custom code.

What are the differences between Azure Functions and Azure Logic Apps

You can call Azure Functions from Azure Logic Apps, and vice versa. The primary difference between the two services is their intent. Azure Functions is a serverless compute service, and Azure Logic Apps is intended to be a serverless orchestration service. Although you can use Azure Functions to orchestrate a long-running business process that involves various connections, this was not its primary use case when it was designed. Additionally, the two services are priced differently. Azure Functions pricing is based on the number of executions and the running time of each execution. Logic Apps pricing is based on the number of executions and the type of connectors that it utilizes.

Analyze the decision criteria

With two viable serverless options, it can be difficult to know which is the best one for the job. In this unit, we'll analyze the criteria that experts employ when they're choosing a serverless service to use for a given business need.

Understanding the criteria can also help you better understand the nuanced differences between the products:

- Do you need to perform an orchestration across well-known APIs? As we noted previously, Azure Logic Apps was designed with orchestration in mind, from the web-based visual configurator to the pricing model. Logic Apps excels at connecting a large array of disparate services via their APIs to pass and process data through many steps in a workflow. It's possible to create the same workflow by using Azure Functions, but it might take a considerable amount of time to research which APIs to call and how to call them. Azure Logic Apps has already componentized these API calls so that you supply only a few details and the details of calling the necessary APIs is abstracted away.
- Do you need to execute custom algorithms or perform specialized data parsing and data lookups? With Azure Functions, you can use the full expressiveness of a programming language in a compact form. This lets you concisely build complex algorithms, or data lookup and parsing operations. You would be responsible for maintaining the code, handling exceptions resiliently, and so on. Although Azure Logic Apps can perform logic (loops, decisions, and so on), if you have a logic-intensive orchestration that requires a complex algorithm, implementing that algorithm might be more verbose and visually overwhelming.
- Do you have existing automated tasks written in an imperative programming language? If you already have your orchestration or business logic expressed in C#, Java, Python, or another popular programming language, it might be easier to port your code into the body of an Azure Functions function app than to re-create it by using Azure Logic Apps.
- Do you prefer a visual (declarative) workflow or writing (imperative) code? Ultimately, your choice comes down to whether you prefer to work in a declarative environment or an imperative environment. Developers who have expertise in an imperative programming language might prefer to think about automation and orchestration from an imperative mindset. IT professionals and business analysts might prefer to work in a more visual low-code/no-code (declarative) environment.

Functions and Logic Apps can both create complex orchestrations. An orchestration is a collection of functions or steps that are executed to accomplish a complex task. With Functions, you write code to complete each step. With Logic Apps, you use a GUI to define the actions and how they relate to one another. You can mix and match services when you build an orchestration, calling functions from logic apps and calling logic apps from functions. Here are some common differences between the two:

	Functions	Logic Apps
State	Normally stateless, but Durable Functions provide state.	Stateful.
Development	Code-first (imperative).	Designer-first (declarative).
Connectivity	About a dozen built-in binding types. Write code for custom bindings.	Large collection of connectors. Enterprise Integration Pack for B2B scenarios. Build custom connectors.
Actions	Each activity is an Azure function. Write code for activity functions.	Large collection of ready-made actions.
Monitoring	Azure Application Insights.	Azure portal, Log Analytics.
Management	REST API, Visual Studio.	Azure portal, REST API, PowerShell, Visual Studio.
Execution context	Can run locally or in the cloud.	Runs only in the cloud.

Azure Virtual Desktop

Azure Virtual Desktop on Azure is a desktop and application virtualization service that runs on the cloud. It enables your users to use a cloud-hosted version of Windows from any location. Azure Virtual Desktop works across devices like Windows, Mac, iOS, Android, and Linux. It works with apps that you can use to access remote desktops and apps. You can also use most modern browsers to access Azure Virtual Desktop-hosted experiences.

Why should you use Azure Virtual Desktop?

Provide the best user experience:

Users have the freedom to connect to Azure Virtual Desktop with any device over the internet. They use a Azure Virtual Desktop client to connect to their published Windows desktop and applications. This client could either be a native application on the device or the Azure Virtual Desktop HTML5 web client.

You can make sure your session host virtual machines (VMs) run near apps and services that connect to your datacenter or the cloud. This way your users stay productive and don't encounter long load times.

User sign-in to Azure Virtual Desktop is fast because user profiles are containerized by using FSLogix. At sign-in, the user profile container is dynamically attached to the computing environment. The user profile is immediately available and appears in the system exactly like a native user profile.

You can provide individual ownership through personal (persistent) desktops. For example, you might want to provide personal remote desktops for members of an engineering team. Then they can add or remove programs without impacting other users on that remote desktop.

Enhance security:

Azure Virtual Desktop provides centralized security management for users' desktops with Azure Active Directory (Azure AD). You can enable multifactor authentication to secure user sign-ins. You can also secure access to data by assigning granular role-based access controls (RBACs) to users.

With Azure Virtual Desktop, the data and apps are separated from the local hardware. Azure Virtual Desktop runs them instead on a remote server. The risk of confidential data being left on a personal device is reduced.

User sessions are isolated in both single and multi-session environments.

Azure Virtual Desktop also improves security by using reverse connect technology. This connection type is more secure than the Remote Desktop Protocol. We don't open inbound ports to the session host VMs.

RemoteApp - is a virtual application solution that allows users to run windows-based applications regardless of what operating system they are using. It allows users to launch virtual applications from a server that appear on their computer as if it is installed locally, but in reality are running on a remote server. The user is able to open and save files locally, or open or save to their network drives, which are accessible from the remoteapps.

Host pools are a collection of one or more identical virtual machines (VMs) within Azure Virtual Desktop environments. Each host pool can contain an app group that users can interact with as they would on a physical desktop. (Max session limit = the maximum number of users you want load-balanced to a single session host).

What are some key features of Azure Virtual Desktop?

- **Simplified management** - Azure Virtual Desktop is an Azure service, so it will be familiar to Azure administrators. You use Azure AD and RBACs to manage access to resources. With Azure, you also get tools to automate VM deployments, manage VM updates, and provide disaster recovery. As with other Azure services, Azure Virtual Desktop uses Azure Monitor for monitoring and alerts. This standardization lets admins identify issues through a single interface.
- **Performance management**: Azure Virtual Desktop gives you options to load balance users on your VM host pools. Host pools are collections of VMs with the same configuration assigned to multiple users. For the best performance, you can configure load balancing to occur as users sign in (breadth mode). With breadth mode, users are sequentially allocated across the host pool for your workload. To save costs, you can configure your VMs for depth mode load balancing where users are fully allocated on one VM before moving to the next. Azure Virtual Desktop provides tools to automatically provision additional VMs when incoming demand exceeds a specified threshold.
- **Multi-session Windows 10 deployment**: Azure Virtual Desktop lets you use Windows 10 Enterprise multi-session, the only Windows client-based operating system that enables multiple concurrent users on a single VM. Azure Virtual Desktop also provides a more consistent experience with broader application support compared to Windows Server-based operating systems.

How can you reduce costs with Azure Virtual Desktop?

- **Bring your own licenses** - Azure Virtual Desktop is available to you at no additional cost if you have an eligible Microsoft 365 license. Just pay for the Azure resources used by Azure Virtual Desktop.
 - Bring your eligible Windows or Microsoft 365 license to get Windows 10 Enterprise and Windows 7 Enterprise desktops and apps at no additional cost.
 - If you're an eligible Microsoft Remote Desktop Services Client Access License customer, Windows Server Remote Desktop Services desktops and apps are available at no additional cost.
- **Save on compute costs** - Buy one-year or three-year Azure Reserved Virtual Machine Instances to save you up to 72 percent versus pay-as-you-go pricing. You can pay for a reservation up front or monthly. Reservations provide a billing discount and don't affect the runtime state of your resources.

Networking

Linking compute resources and providing access to applications is the key function of Azure networking. Networking functionality in Azure includes a range of options to connect the outside world to services and features in the global Azure datacenters.

Here are some examples of networking services in Azure:

Azure Virtual Network	Connects VMs to incoming virtual private network (VPN) connections
Azure Load Balancer	Balances inbound and outbound connections to applications or service endpoints
Azure Application Gateway	Optimizes app server farm delivery while increasing application security
Azure VPN Gateway	Accesses Azure Virtual Networks through high-performance VPN gateways
Azure DNS	Provides ultra-fast DNS responses and ultra-high domain availability
Azure Content Delivery Network	Delivers high-bandwidth content to customers globally
Azure DDoS Protection	Protects Azure-hosted applications from distributed denial of service (DDoS) attacks
Azure Traffic Manager	Distributes network traffic across Azure regions worldwide. Azure Traffic Manager is a DNS-based load balancing solution.
Azure ExpressRoute	Connects to Azure over high-bandwidth dedicated secure connections
Azure Network Watcher	Monitors and diagnoses network issues by using scenario-based analysis.
Azure Firewall	Implements high-security, high-availability firewall with unlimited scalability.
Azure Virtual WAN	Creates a unified wide area network (WAN) that connects local and remote sites.

Azure Virtual Networks

Azure virtual networks enable Azure resources, such as VMs, web apps, and databases, to communicate with each other, with users on the internet, and with your on-premises client computers. You can think of an Azure network as a set of resources that links other Azure resources.

Networks in Azure are known as virtual networks. A virtual network can have multiple IP address spaces and multiple subnets. Azure automatically routes traffic between different subnets within a virtual network. Each virtual network is its own network.

Azure virtual networks provide the following key networking capabilities:

- Isolation and segmentation
- Internet communications
- Communicate between Azure resources
- Communicate with on-premises resources
- Route network traffic
- Filter network traffic
- Connect virtual networks

Isolation and segmentation: Virtual Network allows you to create multiple isolated virtual networks. When you set up a virtual network, you define a private IP address space by using either public or private IP address ranges. You can divide that IP address space into subnets and allocate part of the defined address space to each named subnet. For name resolution, you can use the name resolution service that's built in to Azure. You also can configure the virtual network to use either an internal or an external DNS server.

Internet communications: A VM in Azure can connect to the internet by default. You can enable incoming connections from the internet by defining a public IP address or a public load balancer. For VM management, you can connect via the Azure CLI, Remote Desktop Protocol, or Secure Shell.

You'll want to enable Azure resources to communicate securely with each other. You can do that in one of two ways:

- **Virtual networks:** These can connect not only VMs but other Azure resources, such as the App Service Environment for Power Apps, Azure Kubernetes Service, and Azure virtual machine scale sets.
- **Service endpoints** You can use service endpoints to connect to other Azure resource types, such as Azure SQL databases and storage accounts. This approach enables you to link multiple Azure resources to virtual networks to improve security and provide optimal routing between resources.

Azure virtual networks enable you to link resources together in your on-premises environment and within your Azure subscription. In effect, you can create a network that spans both your local and cloud environments. There are three mechanisms for you to achieve this connectivity:

- **Point-to-site virtual private networks** - The typical approach to a virtual private network (VPN) connection is from a computer outside your organization, back into your corporate network. In this case, the client computer initiates an encrypted VPN connection to connect that computer to the Azure virtual network.
- **Site-to-site virtual private networks** - A site-to-site VPN links your on-premises VPN device or gateway to the Azure VPN gateway in a virtual network. In effect, the devices in Azure can appear as being on the local network. The connection is encrypted and works over the internet.
- **Azure ExpressRoute** - For environments where you need greater bandwidth and even higher levels of security, Azure ExpressRoute is the best approach. ExpressRoute provides dedicated private connectivity to Azure that doesn't travel over the internet.

By default, Azure routes traffic between subnets on any connected virtual networks, on-premises networks, and the internet. You also can control routing and override those settings, as follows:

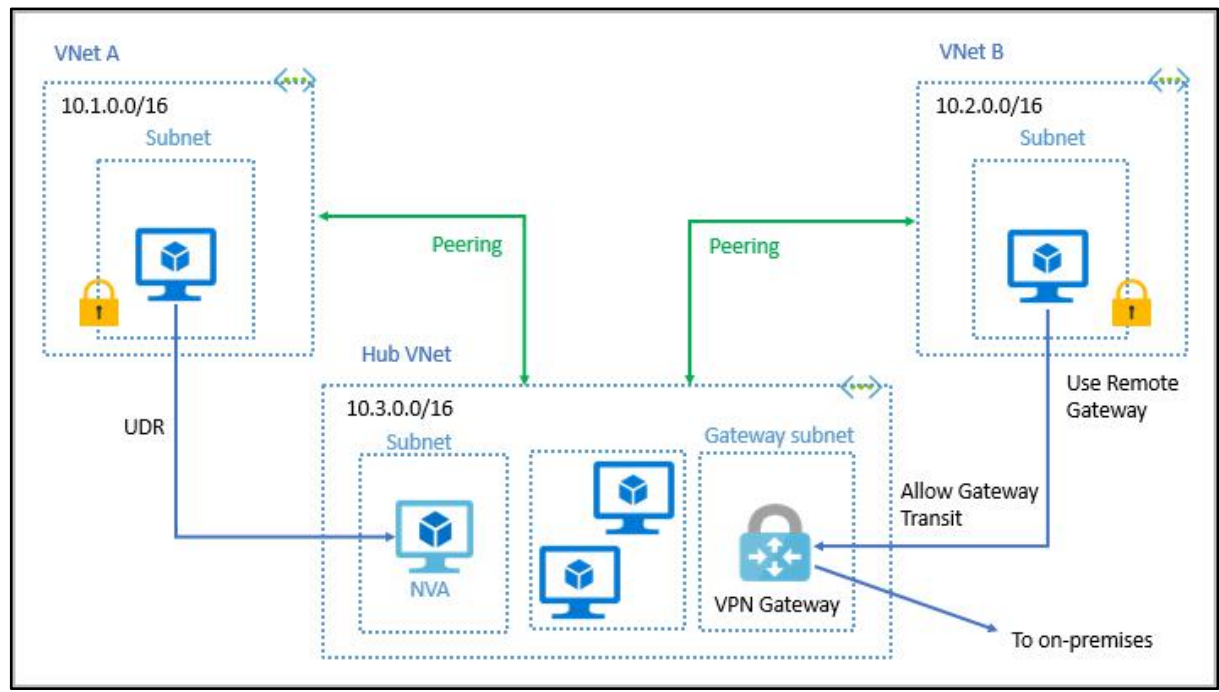
- **Route tables** - A route table allows you to define rules about how traffic should be directed. You can create custom route tables that control how packets are routed between subnets.
- **Border Gateway Protocol (BGP)** – This works with Azure VPN gateways or ExpressRoute to propagate on-premises BGP routes to Azure virtual networks.

Azure virtual networks enable you to filter traffic between subnets by using the following approaches:

- **Network security groups** - This is an Azure resource that can contain multiple inbound and outbound security rules. You can define these rules to allow or block traffic, based on factors such as source and destination IP address, port, and protocol.
- **Network virtual appliances** – This is a specialized VM that can be compared to a hardened network appliance. A network virtual appliance carries out a particular network function, such as running a firewall or performing wide area network (WAN) optimization.

You can link virtual networks together by using virtual network peering. Peering enables resources in each virtual network to communicate with each other. These virtual networks can be in separate regions, which allows you to create a global interconnected network through Azure.

UDR is user-defined Routing. UDR is a significant update to Azure's Virtual Networks as this allows network admins to control the routing tables between subnets within a VNet, as well as between VNets, thereby allowing for greater control over network traffic flow.



Creating a Network - You'll configure the following settings for a basic virtual network:

- **Network name** - must be unique in your subscription, but it doesn't need to be globally unique. Make the name a descriptive one that's easy to remember and identified from other virtual networks.
- **Address space** - When you set up a virtual network, you define the internal address space in Classless Interdomain Routing (CIDR) format. This address space needs to be unique within your subscription and any other networks that you connect to. Let's assume you choose an address space of 10.0.0.0/24 for your first virtual network. The addresses defined in this address space range from 10.0.0.1 to 10.0.0.254. You then create a second virtual network and choose an address space of 10.0.0.0/8. The addresses in this address space range from 10.0.0.1 to 10.255.255.254. Some of the addresses overlap and can't be used for the two virtual networks. But you can use 10.0.0.0/16, with addresses that range from 10.0.0.1 to 10.0.255.254, and 10.1.0.0/16, with addresses that range from 10.1.0.1 to 10.1.255.254. You can assign these address spaces to your virtual networks because there's no address overlap. **Note** - You can add address spaces after you create the virtual network.
- **Subscription** - This option only applies if you have multiple subscriptions to choose from.
- **Resource group** - Like any other Azure resource, a virtual network needs to exist in a resource group. You can either select an existing resource group or create a new one.
- **Location** - Select the location where you want the virtual network to exist.
- **Subnet** - Within each virtual network address range, you can create one or more subnets that partition the virtual network's address space. Routing between subnets will then depend on the default traffic routes. You also can define custom routes. Alternatively, you can define one subnet that encompasses all the virtual networks' address ranges. **Note** -Subnet names must begin with a letter or number and end with a letter, number, or underscore. They may contain only letters, numbers, underscores, periods, or hyphens.
- **DDoS protection** - You can select either Basic or Standard DDoS protection. Standard DDoS protection is a premium service. For more information on Standard DDoS protection, see [Azure DDoS protection Standard overview](#).
- **Service endpoints** - Here, you enable service endpoints. Then you select from the list which Azure service endpoints you want to enable. Options include Azure Cosmos DB, Azure Service Bus, Azure Key Vault, and so on.
- **Network security group** – These have security rules that enable you to filter the type of network traffic that can flow in and out of virtual network subnets and network interfaces. You create the network security group separately. Then you associate it with the virtual network.
- **Route table** -Azure automatically creates a route table for each subnet within an Azure virtual network and adds system default routes to the table. You can add custom route tables to modify traffic between virtual networks.
- You can also amend the service endpoints.
- **Connected devices**: Use the virtual network to connect machines.
- **Peerings**: Link virtual networks in peering arrangements.

You can also monitor and troubleshoot virtual networks. Or, you can create an automation script to generate the current virtual network. Virtual networks are powerful and highly configurable mechanisms for connecting entities in Azure. You can connect Azure resources to one another or to resources you have on-premises. You can isolate, filter, and route your network traffic. Azure allows you to increase security where you feel you need it.

VPN gateways

A VPN gateway is a type of virtual network gateway. Azure VPN Gateway instances are deployed in Azure Virtual Network instances and enable the following connectivity:

- Connect on-premises datacenters to virtual networks through a site-to-site connection.
- Connect individual devices to virtual networks through a point-to-site connection.
- Connect virtual networks to other virtual networks through a network-to-network connection.

All transferred data is encrypted in a private tunnel as it crosses the internet. You can deploy only one VPN gateway in each virtual network, but you can use one gateway to connect to multiple locations, which includes other virtual networks or on-premises datacenters.

When you deploy a VPN gateway, you specify the VPN type: either policy-based or route-based. The main difference between these two types of VPNs is how traffic to be encrypted is specified. In Azure, both types of VPN gateways use a pre-shared key as the only method of authentication. Both types also rely on Internet Key Exchange (IKE) in either version 1 or version 2 and Internet Protocol Security (IPSec). IKE is used to set up a security association (an agreement of the encryption) between two endpoints. This association is then passed to the IPSec suite, which encrypts and decrypts data packets encapsulated in the VPN tunnel.

Policy-based VPN gateways specify statically the IP address of packets that should be encrypted through each tunnel. This type of device evaluates every data packet against those sets of IP addresses to choose the tunnel where that packet is going to be sent through.

Key features of policy-based VPN gateways in Azure include:

- Support for IKEv1 only.
- Use of static routing, where combinations of address prefixes from both networks control how traffic is encrypted and decrypted through the VPN tunnel. The source and destination of the tunneled networks are declared in the policy and don't need to be declared in routing tables.
- Policy-based VPNs must be used in specific scenarios that require them, such as for compatibility with legacy on-premises VPN devices.

Route-based VPNs - If defining which IP addresses are behind each tunnel is too cumbersome, route-based gateways can be used. With route-based gateways, IPSec tunnels are modeled as a network interface or virtual tunnel interface. IP routing (either static routes or dynamic routing protocols) decides which one of these tunnel interfaces to use when sending each packet. Route-based VPNs are the preferred connection method for on-premises devices. They're more resilient to topology changes such as the creation of new subnets. Use a route-based VPN gateway if you need any of the following types of connectivity:

- Connections between virtual networks
- Point-to-site connections
- Multisite connections
- Coexistence with an Azure ExpressRoute gateway

Key features of route-based VPN gateways in Azure include:

- Supports IKEv2
- Uses any-to-any (wildcard) traffic selectors
- Can use dynamic routing protocols, where routing/forwarding tables direct traffic to different IPSec tunnels. In this case, the source and destination networks aren't statically defined as they are in policy-based VPNs or even in route-based VPNs with static routing. Instead, data packets are encrypted based on network routing tables that are created dynamically using routing protocols such as Border Gateway Protocol (BGP).

VPN gateway sizes

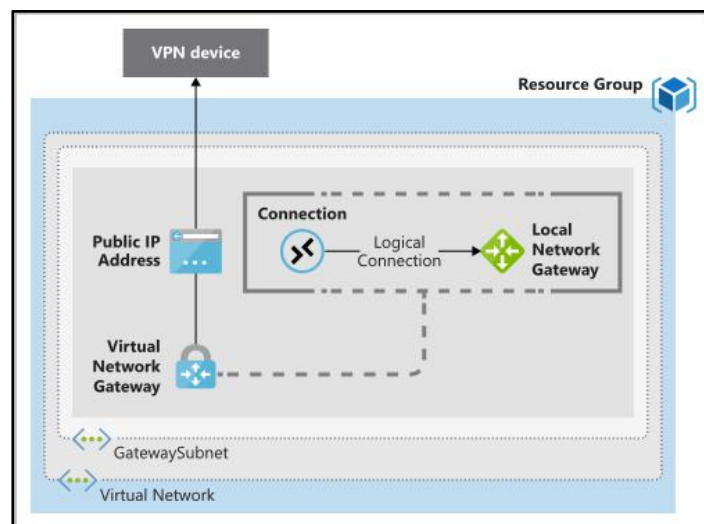
The capabilities of your VPN gateway are determined by the SKU (Stock Keeping Unit – meaning pricing tier) or size that you deploy. A Basic VPN gateway should only be used for Dev/Test workloads. In addition, it's unsupported to migrate from Basic to the VpnGW1/2/3/Az SKUs at a later time without having to remove the gateway and redeploy.

Before you can deploy a VPN gateway, you'll need some Azure and on-premises resources.

Required Azure resources:

- **Virtual network.** Deploy a virtual network with enough address space for the additional subnet that you'll need for the VPN gateway. The address space for this virtual network must not overlap with the on-premises network that you'll be connecting to. You can deploy only one VPN gateway within a virtual network.
- **Gateway Subnet.** Deploy a subnet called GatewaySubnet for the VPN gateway. Use at least a /27 address mask to make sure you have enough IP addresses in the subnet for future growth. You can't use this subnet for any other services.
- **Public IP address.** Create a Basic-SKU dynamic public IP address if you're using a non-zone-aware gateway. This address provides a public-routable IP address as the target for your on-premises VPN device. This IP address is dynamic, but it won't change unless you delete and re-create the VPN gateway.
- **Local network gateway.** Create a local network gateway to define the on-premises network's configuration, such as where the VPN gateway will connect and what it will connect to. This configuration includes the on-premises VPN device's public IPv4 address and the on-premises routable networks. This information is used by the VPN gateway to route packets that are destined for on-premises networks through the IPsec tunnel.
- **Virtual network gateway.** Create the virtual network gateway to route traffic between the virtual network and the on-premises datacenter or other virtual networks. The virtual network gateway can be either a VPN or ExpressRoute gateway, but this unit only deals with VPN virtual network gateways.
- **Connection.** Create a connection resource to create a logical connection between the VPN gateway and the local network gateway.
 - The connection is made to the on-premises VPN device's IPv4 address as defined by the local network gateway.
 - The connection is made from the virtual network gateway and its associated public IP address.
 - You can create multiple connections.

The following diagram shows this combination of resources and their relationships to help you better understand what's required to deploy a VPN gateway.

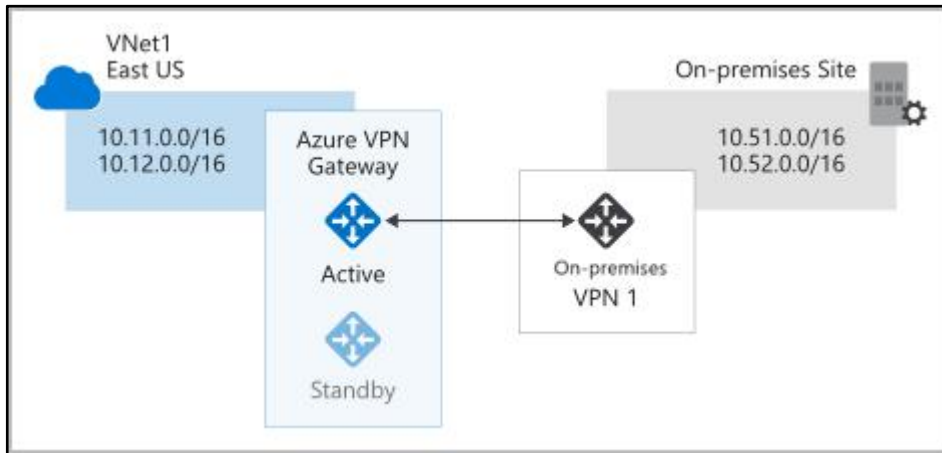


Required on-premises resources:

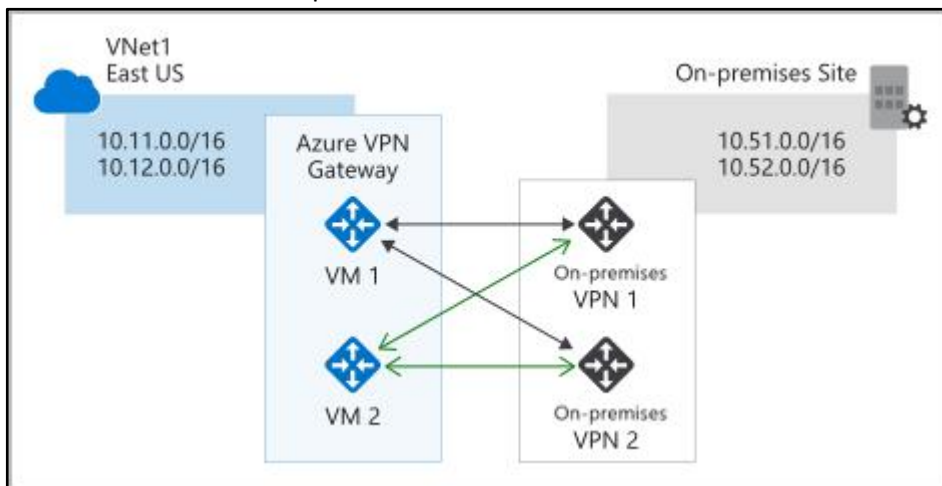
- A VPN device that supports policy-based or route-based VPN gateways
- A public-facing (internet-routable) IPv4 address

There are several ways to ensure you have a fault-tolerant configuration:

- **Active/standby** - By default, VPN gateways are deployed as two instances in an active/standby configuration, even if you only see one VPN gateway resource in Azure. When planned maintenance or unplanned disruption affects the active instance, the standby instance automatically assumes responsibility for connections without any user intervention. Connections are interrupted during this failover, but they're typically restored within a few seconds for planned maintenance and within 90 seconds for unplanned disruptions.



- **Active/active** - With the introduction of support for the BGP routing protocol, you can also deploy VPN gateways in an active/active configuration. In this configuration, you assign a unique public IP address to each instance. You then create separate tunnels from the on-premises device to each IP address. You can extend the high availability by deploying an additional VPN device on-premises.



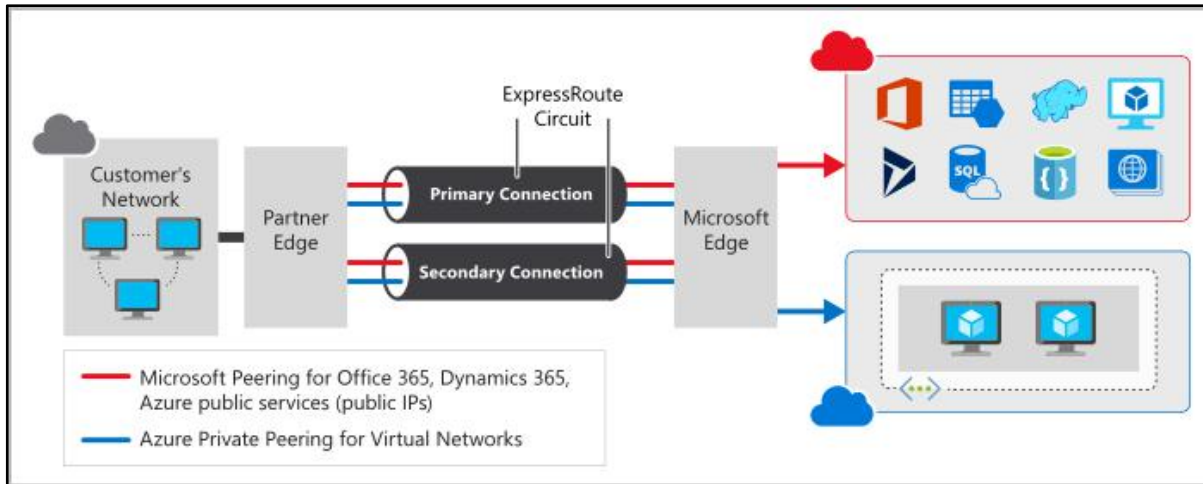
ExpressRoute failover - Another high-availability option is to configure a VPN gateway as a secure failover path for ExpressRoute connections. ExpressRoute circuits have resiliency built in. But they aren't immune to physical problems that affect the cables delivering connectivity or outages that affect the complete ExpressRoute location. In high-availability scenarios, where there's risk associated with an outage of an ExpressRoute circuit, you can also provision a VPN gateway that uses the internet as an alternative method of connectivity. In this way, you can ensure there's always a connection to the virtual networks.

Zone-redundant gateways - In regions that support availability zones, VPN gateways and ExpressRoute gateways can be deployed in a zone-redundant configuration. This configuration brings resiliency, scalability, and higher availability to virtual network gateways. Deploying gateways in Azure availability zones physically and logically separates gateways within a region while protecting your on-premises network connectivity to Azure from zone-level failures. These gateways require different gateway SKUs and use Standard public IP addresses instead of Basic public IP addresses.

Azure ExpressRoute

ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection with the help of 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. This allows ExpressRoute connections to offer more reliability, faster speeds, consistent latencies, and higher security than typical connections over the Internet.



We'll focus on two different layers of the Open Systems Interconnection (OSI) model:

- Layer 2 (L2): This layer is the Data Link Layer, which provides node-to-node communication between two nodes on the same network.
- Layer 3 (L3): This layer is the Network Layer, which provides addressing and routing between nodes on a multi-node network.

There are several benefits to using ExpressRoute as the connection service between Azure and on-premises networks:

- Layer 3 connectivity between your on-premises network and the Microsoft Cloud through a connectivity provider. Connectivity can be from an any-to-any (IPVPN) network, a point-to-point Ethernet connection, or through a virtual cross-connection via an Ethernet exchange.
- Connectivity to Microsoft cloud services across all regions in the geopolitical region.
- Global connectivity to Microsoft services across all regions with the ExpressRoute premium add-on.
- Dynamic routing between your network and Microsoft via BGP.
- Built-in redundancy in every peering location for higher reliability.
- Connection uptime SLA.
- QoS (Quality of Service) support for Skype for Business.

Layer 3 connectivity - ExpressRoute provides Layer 3 (address-level) connectivity between your on-premises network and the Microsoft cloud through connectivity partners. These connections can be from a point-to-point or any-to-any network. They can also be virtual cross-connections through an exchange.

Built-in redundancy - Each connectivity provider uses redundant devices to ensure that connections established with Microsoft are highly available. You can configure multiple circuits to complement this feature. All redundant connections are configured with Layer 3 connectivity to meet service-level agreements.

ExpressRoute enables direct access to the following services in all regions:

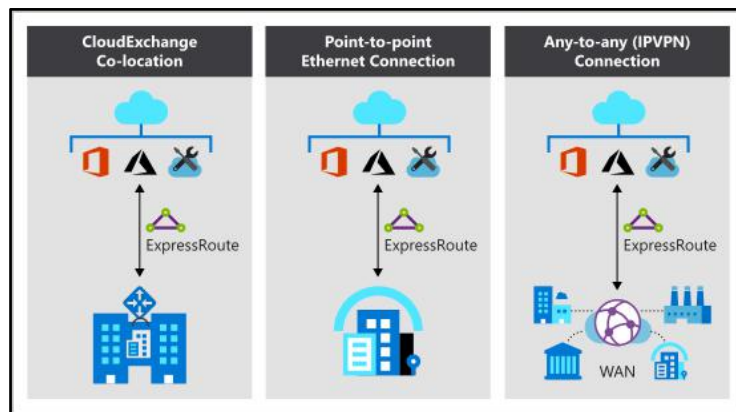
- Microsoft Office 365
- Microsoft Dynamics 365
- Azure compute services, such as Azure Virtual Machines
- Azure cloud services, such as Azure Cosmos DB and Azure Storage

Across on-premises connectivity with ExpressRoute Global Reach - You can enable ExpressRoute Global Reach to exchange data across your on-premises sites by connecting your ExpressRoute circuits. For example, assume that you have a private datacenter in California connected to ExpressRoute in Silicon Valley. You have another private datacenter in Texas connected to ExpressRoute in Dallas. With ExpressRoute Global Reach, you can connect your private datacenters through two ExpressRoute circuits. Your cross-datacenter traffic will travel through the Microsoft network.

Dynamic routing - ExpressRoute uses the Border Gateway Protocol (BGP) routing protocol. BGP is used to exchange routes between on-premises networks and resources running in Azure. This protocol enables dynamic routing between your on-premises network and services running in the Microsoft cloud.

ExpressRoute supports three models that you can use to connect your on-premises network to the Microsoft cloud:

- **CloudExchange colocation** - Colocated providers can normally offer both Layer 2 and Layer 3 connections between your infrastructure, which might be located in the colocation facility, and the Microsoft cloud. For example, if your datacenter is colocated at a cloud exchange such as an ISP, you can request a virtual cross-connection to the Microsoft cloud.
- **Point-to-point Ethernet connection** - Provide Layer 2 and Layer 3 connectivity between your on-premises site and Azure. You can connect your offices or datacenters to Azure by using the point-to-point links. For example, if you have an on-premises datacenter, you can use a point-to-point Ethernet link to connect to Microsoft.
- **Any-to-any connection** - Integrate your wide area network (WAN) with Azure by providing connections to your offices and datacenters. Azure integrates with your WAN connection to provide a connection like you would have between your datacenter and any branch offices. With any-to-any connections, all WAN providers offer Layer 3 connectivity. For example, if you already use Multiprotocol Label Switching to connect to your branch offices or other sites in your organization, an ExpressRoute connection to Microsoft behaves like any other location on your private WAN.



Security considerations - With ExpressRoute, your data doesn't travel over the public internet, so it's not exposed to the potential risks associated with internet communications. ExpressRoute is a private connection from your on-premises infrastructure to your Azure infrastructure. Even if you have an ExpressRoute connection, DNS queries, certificate revocation list checking, and Azure Content Delivery Network requests are still sent over the public internet.

Storage

Azure provides four main types of storage services.

Azure Blob storage	Storage service for very large objects, such as video files or bitmaps
Azure File storage	File shares that can be accessed and managed like a file server
Azure Queue storage	A data store for queuing and reliably delivering messages between applications
Azure Table storage	Table storage is a service that stores non-relational structured data (also known as structured NoSQL data) in the cloud, providing a key/attribute store with a schemaless design

These services all share several common characteristics:

- Durable and highly available with redundancy and replication.
- Secure through automatic encryption and role-based access control.
- Scalable with virtually unlimited storage.
- Managed, handling maintenance and any critical problems for you.
- Accessible from anywhere in the world over HTTP or HTTPS.

To begin using Azure Storage, you first create an Azure Storage account to store your data objects. You can create an Azure Storage account by using the Azure portal, PowerShell, or the Azure CLI. Your storage account will contain all of your Azure Storage data objects, such as blobs, files, and disks.

Azure VMs use Azure Disk Storage to store virtual disks. However, you can't use Azure Disk Storage to store a disk outside of a virtual machine.

A storage account provides a unique namespace for your Azure Storage data, that's accessible from anywhere in the world over HTTP or HTTPS. Data in this account is secure, highly available, durable, and massively scalable.

Data is not backed up automatically to another Azure Data Center although it can be depending on the replication option configured for the account. There are different replication options available with a storage account. The 'minimum' replication option is Locally Redundant Storage (LRS). Locally Redundant Storage (LRS) is the default which maintains three copies of the data in the data center. Geo-redundant storage (GRS) has cross-regional replication to protect against regional outages. Data is replicated synchronously three times in the primary region, then replicated asynchronously to the secondary region.

The current storage limit is 2 PB for US and Europe, and 500 TB for all other regions (including the UK) with no limit on the number of files.

Azure Blob Storage

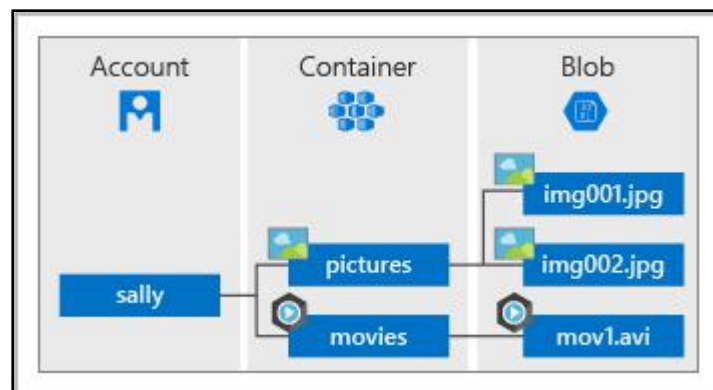
Azure Blob Storage is an object storage solution for the cloud. It can store massive amounts of data, such as text or binary data. Azure Blob Storage is unstructured, meaning that there are no restrictions on the kinds of data it can hold. Blob Storage can manage thousands of simultaneous uploads, massive amounts of video data, constantly growing log files, and can be reached from anywhere with an internet connection.

Blobs aren't limited to common file formats. A blob could contain gigabytes of binary data streamed from a scientific instrument, an encrypted message for another application, or data in a custom format for an app you're developing. One advantage of blob storage over disk storage is that it does not require developers to think about or manage disks; data is uploaded as blobs, and Azure takes care of the physical storage needs.

Blob Storage is ideal for:

- Serving images or documents directly to a browser.
- Storing files for distributed access.
- Streaming video and audio.
- Storing data for backup and restore, disaster recovery, and archiving.
- Storing data for analysis by an on-premises or Azure-hosted service.
- Storing up to 8 TB of data for virtual machines.

You store blobs in containers, which helps you organize your blobs depending on your business needs. The following diagram illustrates how you might use Azure accounts, containers, and blobs.



Data stored in the cloud can grow at an exponential pace. To manage costs for your expanding storage needs, it's helpful to organize your data based on attributes like frequency of access and planned retention period. Data stored in the cloud can be different based on how it's generated, processed, and accessed over its lifetime. Some data is actively accessed and modified throughout its lifetime. Some data is accessed frequently early in its lifetime with access dropping drastically as the data ages. Some data remains idle in the cloud and is rarely, if ever, accessed after it's stored. To accommodate these different access needs, Azure provides several access tiers, which you can use to balance your storage costs with your access needs.



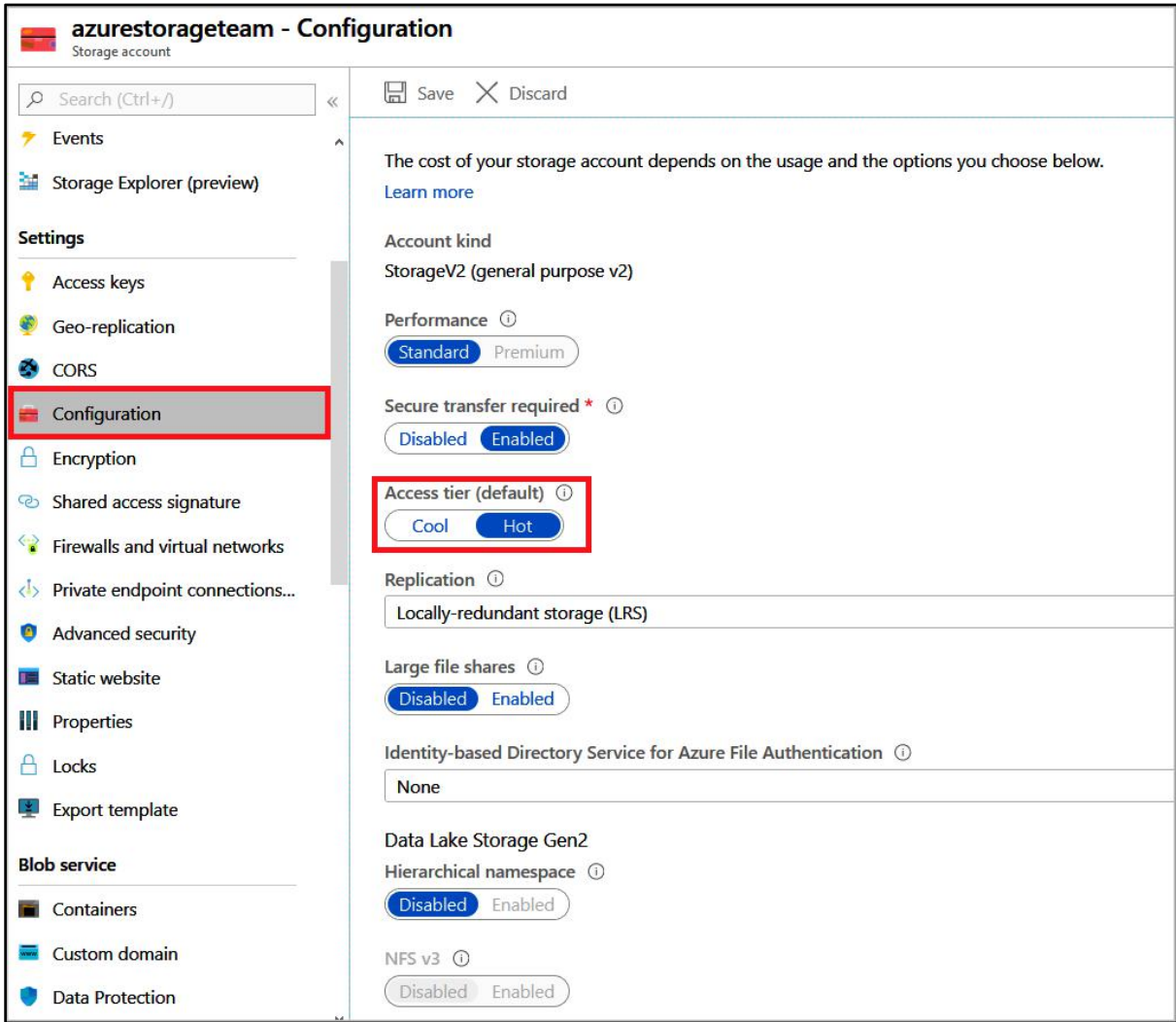
Azure Storage offers different access tiers for your blob storage, helping you store object data in the most cost-effective manner. The available access tiers include:

- Hot access tier: Optimized for storing data that is accessed frequently (for example, images for your website).
- Cool access tier: Optimized for data that is infrequently accessed and stored for at least 30 days (for example, invoices for your customers).
- Archive access tier: Appropriate for data that is rarely accessed and stored for at least 180 days, with flexible latency requirements (for example, long-term backups).

The following considerations apply to the different access tiers:

- Only the hot and cool access tiers can be set at the account level. The archive access tier isn't available at the account level.
- Hot, cool, and archive tiers can be set at the blob level, during upload or after upload.
- Data in the cool access tier can tolerate slightly lower availability, but still requires high durability, retrieval latency, and throughput characteristics similar to hot data. For cool data, a slightly lower availability service-level agreement (SLA) and higher access costs compared to hot data are acceptable trade-offs for lower storage costs.
- Archive storage stores data offline and offers the lowest storage costs, but also the highest costs to rehydrate and access data.

The following illustration demonstrates choosing between the hot and cool access tiers on a general purpose storage account.

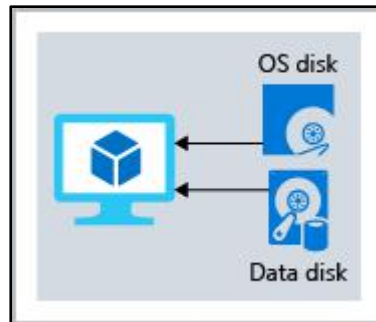


Azure Disk Storage

Disk Storage provides disks for Azure virtual machines. Applications and other services can access and use these disks as needed, similar to how they would in on-premises scenarios. Disk Storage allows data to be persistently stored and accessed from an attached virtual hard disk.

Disks come in many different sizes and performance levels, from solid-state drives (SSDs) to traditional spinning hard disk drives (HDDs), with varying performance tiers. You can use standard SSD and HDD disks for less critical workloads, premium SSD disks for mission-critical production applications, and ultra disks for data-intensive workloads such as SAP HANA, top tier databases, and transaction-heavy workloads. Azure has consistently delivered enterprise-grade durability for infrastructure as a service (IaaS) disks, with an industry-leading ZERO% annualized failure rate.

The following illustration shows an Azure virtual machine that uses separate disks to store different data.



Azure Files

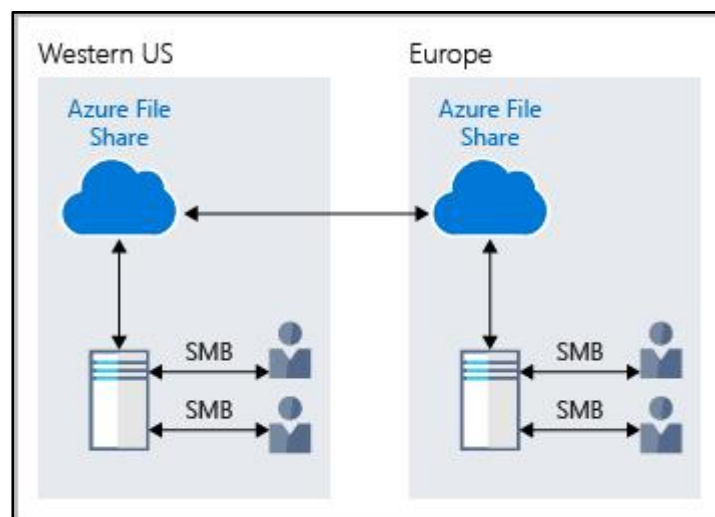
Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard SMB (Server Message Block) and Network File System (preview) protocols. Azure file shares can be mounted concurrently by cloud or on-premises deployments of Windows, Linux, and macOS. Applications running in Azure virtual machines or cloud services can mount a file storage share to access file data, just as a desktop application would mount a typical SMB share. Any number of Azure virtual machines or roles can mount and access the file storage share simultaneously. Typical usage scenarios would be to share files anywhere in the world, diagnostic data, or application data sharing.



Use Azure Files for the following situations:

- Many on-premises applications use file shares. Azure Files makes it easier to migrate those applications that share data to Azure. If you mount the Azure file share to the same drive letter that the on-premises application uses, the part of your application that accesses the file share should work with minimal changes, if any.
- Store configuration files on a file share and access them from multiple VMs. Tools and utilities used by multiple developers in a group can be stored on a file share, ensuring that everybody can find them, and that they use the same version.
- Write data to a file share, and process or analyze the data later. For example, you might want to do this with diagnostic logs, metrics, and crash dumps.

The following illustration shows Azure Files being used to share data between two geographical locations. Azure Files ensures the data is encrypted at rest, and the SMB protocol ensures the data is encrypted in transit.



One thing that distinguishes Azure Files from files on a corporate file share is that you can access the files from anywhere in the world, by using a URL that points to the file. You can also use Shared Access Signature (SAS) tokens to allow access to a private asset for a specific amount of time.

Here's an example of a service SAS URI, showing the resource URI and the SAS token:



Azure file shares do not currently support Kerberos authentication with your Active Directory (AD) or Azure Active Directory (AAD) identity, although this is a feature we are working on. Instead, you must access your Azure file share with the storage account key for the storage account containing your Azure file share. A storage account key is an administrator key for a storage account, including administrator permissions to all files and folders within the file share you're accessing, and for all file shares and other storage resources (blobs, queues, tables, etc) contained within your storage account.

Databases

Azure provides multiple database services to store a wide variety of data types and volumes. And with global connectivity, this data is available to users instantly.

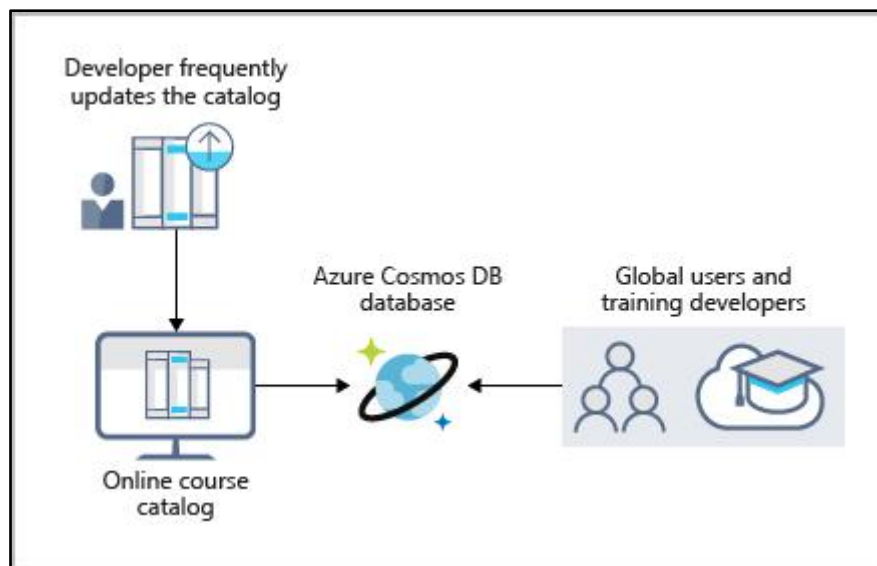
Azure Cosmos DB	Globally distributed database that supports NoSQL options
Azure SQL Database	Fully managed relational database with auto-scale, integral intelligence, and robust security
Azure Database for MySQL	Fully managed and scalable MySQL relational database with high availability and security
Azure Database for PostgreSQL	Fully managed and scalable PostgreSQL relational database with high availability and security
SQL Server on Azure Virtual Machines	Service that hosts enterprise SQL Server apps in the cloud
Azure Synapse Analytics	Fully managed data warehouse with integral security at every level of scale at no extra cost
Azure Database Migration Service	Service that migrates databases to the cloud with no application code changes
Azure Cache for Redis	Fully managed service caches frequently used and static data to reduce data and application latency
Azure Database for MariaDB	Fully managed and scalable MariaDB relational database with high availability and security.

Azure Cosmos DB

Azure Cosmos DB is a globally distributed, multi-model database service. You can elastically and independently scale throughput and storage across any number of Azure regions worldwide. You can take advantage of fast, single-digit-millisecond data access by using any one of several popular APIs. Azure Cosmos DB provides comprehensive service level agreements for throughput, latency, availability, and consistency guarantees.

Azure Cosmos DB supports schema-less data, which lets you build highly responsive and "Always On" applications to support constantly changing data. You can use this feature to store data that's updated and maintained by users around the world.

For example, Tailwind Traders provides a public training portal that is used by customers across the globe to learn about the different tools that Tailwind Traders creates. Tailwind Traders developers maintain and update the data. The following illustration shows a sample Azure Cosmos DB database that's used to store data for the Tailwind Traders training portal website.



Azure Cosmos DB is flexible. At the lowest level, Azure Cosmos DB stores data in atom-record-sequence (ARS) format. The data is then abstracted and projected as an API, which you specify when you're creating your database. Your choices include SQL, MongoDB, Cassandra, Tables, and Gremlin. This level of flexibility means that as you migrate your company's databases to Azure Cosmos DB, your developers can stick with the API that they're the most comfortable with.

Azure Cosmos DB is a great way to store unstructured and JSON data. Combined with Azure Functions, Cosmos DB makes storing data quick and easy with much less code than required for storing data in a relational database.

Azure SQL Database

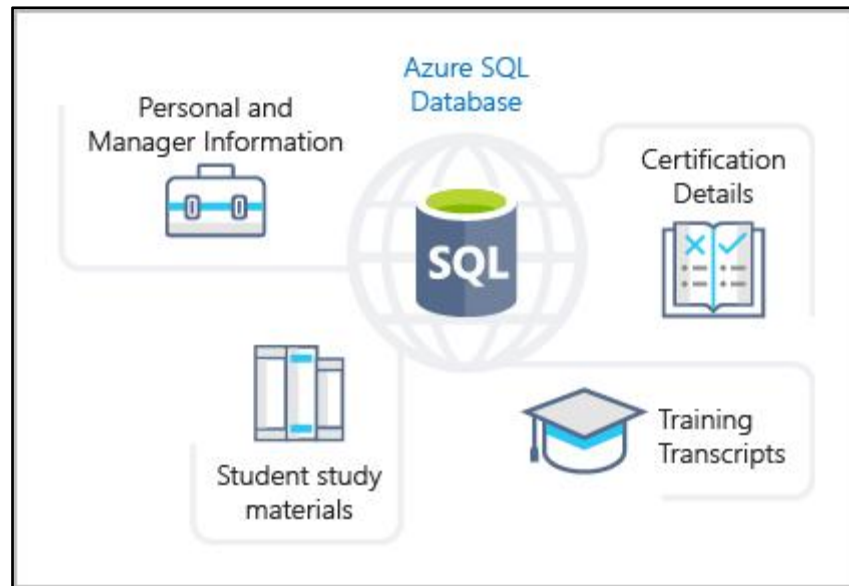
Azure SQL Database is a relational database based on the latest stable version of the Microsoft SQL Server database engine. SQL Database is a high-performance, reliable, fully managed, and secure database. You can use it to build data-driven applications and websites in the programming language of your choice, without needing to manage infrastructure.

Features:

- Azure SQL Database is a platform as a service (PaaS) database engine. It handles most of the database management functions, such as upgrading, patching, backups, and monitoring, without user involvement. SQL Database provides 99.99 percent availability. PaaS capabilities that are built into SQL Database enable you to focus on the domain-specific database administration and optimization activities that are critical for your business. SQL Database is a fully managed service that has built-in high availability, backups, and other common maintenance operations. Microsoft handles all updates to the SQL and operating system code. You don't have to manage the underlying infrastructure.
- You can create a highly available and high-performance data storage layer for the applications and solutions in Azure. SQL Database can be the right choice for a variety of modern cloud applications because it enables you to process both relational data and non-relational structures, such as graphs, JSON, spatial, and XML.
- You can use advanced query processing features, such as high-performance, in-memory technologies and intelligent query processing. In fact, the newest capabilities of SQL Server are released first to SQL Database, and then to SQL Server itself. You get the newest SQL Server capabilities, with no overhead for updates or upgrades, tested across millions of databases.

Migration:

- Tailwind Traders currently uses several on-premises servers running SQL Server, which provide data storage for your public-facing website (for example, customer data, order history, and product catalogs). In addition, your on-premises servers running SQL Server also provide data storage for your internal-only training portal website. Tailwind Traders uses the website for new employee training materials (such as study materials, certification details, and training transcripts). The following illustration shows the types of data that your company might store in the Azure SQL Database training portal website.



You can migrate your existing SQL Server databases with minimal downtime by using the Azure Database Migration Service. The Microsoft Data Migration Assistant can generate assessment reports that provide recommendations to help guide you through required changes prior to performing a migration. After you assess and resolve any remediation required, you're ready to begin the migration process. The Azure Database Migration Service performs all of the required steps. You just change the connection string in your apps.

Enable Azure Defender for SQL

Server Firewall

Azure Database for MySQL

Tailwind Traders currently manages several websites on-premises that use the LAMP stack (Linux, Apache, MySQL, PHP). As part of your planning for your migration strategy, the different teams at Tailwind Traders have been researching the available service offerings that Azure provides. You've already discovered that the Web Apps feature of Azure App Service provides built-in functionality to create web applications that use PHP on a Linux server running Apache. You've been tasked with investigating whether the database requirements for the web development team will continue to be met after the migration to Azure.

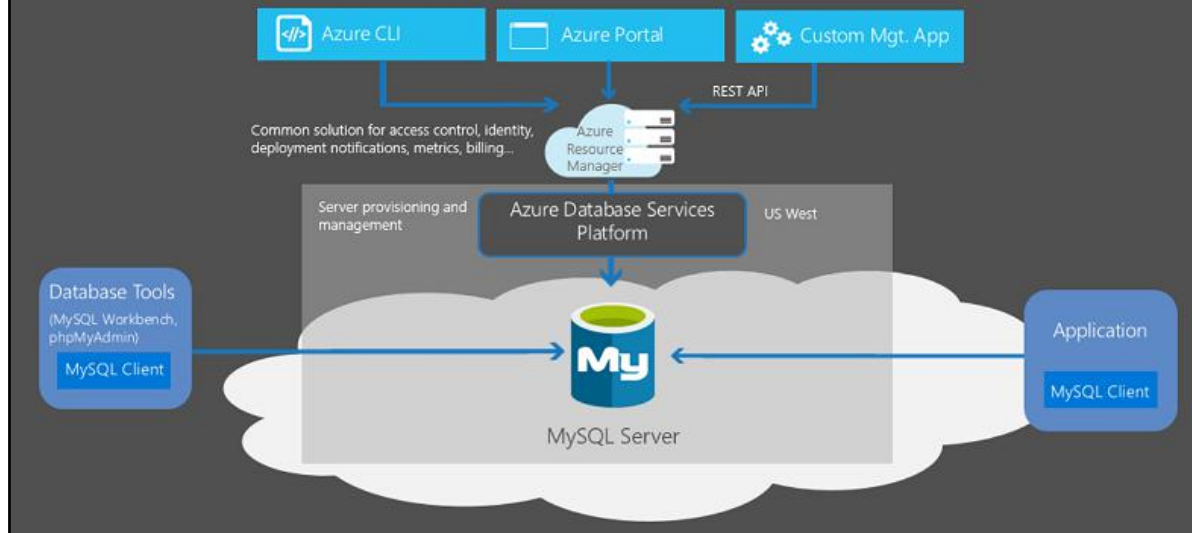
Azure Database for MySQL is a relational database service in the cloud, and it's based on the MySQL Community Edition database engine, versions 5.6, 5.7, and 8.0. With it, you have a 99.99 percent availability service level agreement from Azure, powered by a global network of Microsoft-managed datacenters. This helps keep your app running 24/7. With every Azure Database for MySQL server, you take advantage of built-in security, fault tolerance, and data protection that you would otherwise have to buy or design, build, and manage. With Azure Database for MySQL, you can use point-in-time restore to recover a server to an earlier state, as far back as 35 days.

Azure Database for MySQL delivers:

- Built-in high availability with no additional cost.
- Predictable performance and inclusive, pay-as-you-go pricing.
- Scale as needed, within seconds.
- Ability to protect sensitive data at-rest and in-motion.
- Automatic backups.
- Enterprise-grade security and compliance.

These capabilities require almost no administration, and all are provided at no additional cost. They allow you to focus on rapid app development and accelerating your time-to-market, rather than having to manage virtual machines and infrastructure. In addition, you can migrate your existing MySQL databases with minimal downtime by using the Azure Database Migration Service. After you have completed your migration, you can continue to develop your application with the open-source tools and platform of your choice. You don't have to learn new skills.

Create, Connect and Manage



Azure Database for MySQL offers several service tiers, and each tier provides different performance and capabilities to support lightweight to heavyweight database workloads. You can build your first app on a small database for a few dollars a month, and then adjust the scale to meet the needs of your solution. Dynamic scalability enables your database to transparently respond to rapidly changing resource requirements. You only pay for the resources you need, and only when you need them.

Azure Database for PostgreSQL

As part of its overall data strategy, Tailwind Traders have been using PostgreSQL for several years. You and your team probably already know the benefits of PostgreSQL. Part of your migration is to use Azure Database for PostgreSQL, and you want to make sure that you'll have access to the same benefits as your on-premises server before moving to the cloud.

Azure Database for PostgreSQL is a relational database service in the cloud. The server software is based on the community version of the open-source PostgreSQL database engine. Your familiarity with tools and expertise with PostgreSQL is applicable when you're using Azure Database for PostgreSQL.

Moreover, Azure Database for PostgreSQL delivers the following benefits:

- Built-in high availability compared to on-premises resources. There's no additional configuration, replication, or cost required to make sure your applications are always available.
- Simple and flexible pricing. You have predictable performance based on a selected pricing tier choice that includes software patching, automatic backups, monitoring, and security.
- Scale up or down as needed, within seconds. You can scale compute or storage independently as needed, to make sure you adapt your service to match usage.
- Adjustable automatic backups and point-in-time-restore for up to 35 days.
- Enterprise-grade security and compliance to protect sensitive data at-rest and in-motion. This security covers data encryption on disk and SSL encryption between client and server communication.

Azure Database for PostgreSQL is available in two deployment options: **Single Server** and **Hyperscale (Citus)**.

Single Server

The Single Server deployment option delivers:

- Built-in high availability with no additional cost (99.99 percent SLA).
- Predictable performance and inclusive, pay-as-you-go pricing.
- Vertical scale as needed, within seconds.
- Monitoring and alerting to assess your server.
- Enterprise-grade security and compliance.
- Ability to protect sensitive data at-rest and in-motion.
- Automatic backups and point-in-time-restore for up to 35 days.

All those capabilities require almost no administration, and all are provided at no additional cost. You can focus on rapid application development and accelerating your time to market, rather than having to manage virtual machines and infrastructure. You can continue to develop your application with the open-source tools and platform of your choice, without having to learn new skills.

The Single Server deployment option offers three pricing tiers: Basic, General Purpose, and Memory Optimized. Each tier offers different resource capabilities to support your database workloads. You can build your first app on a small database for a few dollars a month, and then adjust the scale to meet the needs of your solution. Dynamic scalability enables your database to transparently respond to rapidly changing resource requirements. You only pay for the resources you need, and only when you need them.

Hyperscale (Citius)

The Hyperscale (Citius) option horizontally scales queries across multiple machines by using sharding (explained below). Its query engine parallelizes incoming SQL queries across these servers for faster responses on large datasets. It serves applications that require greater scale and performance, generally workloads that are approaching, or already exceed, 100 GB of data.

The Hyperscale (Citius) deployment option supports multi-tenant applications, real-time operational analytics, and high throughput transactional workloads. Applications built for PostgreSQL can run distributed queries on Hyperscale (Citius) with standard connection libraries and minimal changes.

Sharding Explained

A database shard, or simply a shard, is a horizontal partition of data in a database or search engine. Each shard is held on a separate database server instance, to spread load. Some data within a database remains present in all shards, but some appears only in a single shard. Each shard (or server) acts as the single source for this subset of data.

Azure SQL Managed Instance

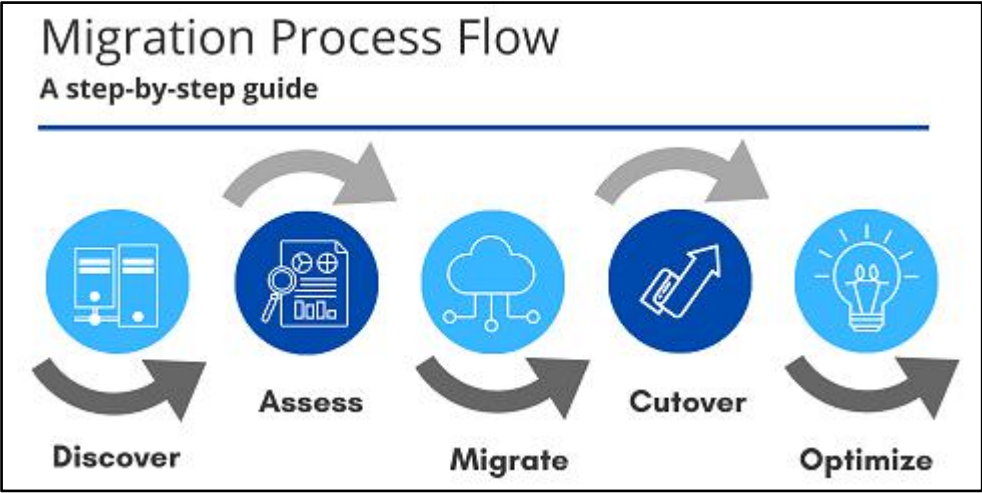
Azure SQL Managed Instance is a scalable cloud data service that provides the broadest SQL Server database engine compatibility with all the benefits of a fully managed platform as a service. Depending on your scenario, Azure SQL Managed Instance might offer more options for your database needs.

Features:

- Like Azure SQL Database, Azure SQL Managed Instance is a platform as a service (PaaS) database engine, which means that your company will be able to take advantage of the best features of moving your data to the cloud in a fully-managed environment. For example, your company will no longer need to purchase and manage expensive hardware, and you won't have to maintain the additional overhead of managing your on-premises infrastructure. On the other hand, your company will benefit from the quick provisioning and service scaling features of Azure, together with automated patching and version upgrades. In addition, you'll be able to rest assured that your data will always be there when you need it through built-in high availability features and a 99.99% uptime service level agreement (SLA). You'll also be able to protect your data with automated backups and a configurable backup retention period.
- Azure SQL Database and Azure SQL Managed Instance offer many of the same features; however, Azure SQL Managed Instance provides several options that might not be available to Azure SQL Database. For example, Tailwind Traders currently uses several on-premises servers running SQL Server, and they would like to migrate their existing databases to a SQL database running in the cloud. However, several of their databases use Cyrillic characters for collation. In this scenario, Tailwind Traders should migrate their databases to an Azure SQL Managed Instance, since Azure SQL Database only uses the default SQL_Latin1_General_CP1_CI_AS server collation.

Note - For a detailed list of the differences between Azure SQL Database and Azure SQL Managed Instance, see [Features comparison: Azure SQL Database and Azure SQL Managed Instance](#).

Migration: Azure SQL Managed Instance makes it easy to migrate your on-premises data on SQL Server to the cloud using the Azure Database Migration Service (DMS) or native backup and restore. After you have discovered all of the features that your company uses, you need to assess which on-premises SQL Server instances you can migrate to Azure SQL Managed Instance to see if you have any blocking issues. Once you have resolved any issues, you can migrate your data, then cutover from your on-premises SQL Server to your Azure SQL Managed Instance by changing the connection string in your applications.



Monitoring

Several basic questions or concerns face all companies that use the cloud:

- Are we using the cloud correctly? Can we squeeze more performance out of our cloud spend?
- Are we spending more than we need to?
- Do we have our systems properly secured?
- How resilient are our resources? If we experience a regional outage, could we fail over to another region?
- How can we diagnose and fix issues that occur intermittently?
- How can we quickly determine the cause of an outage?
- How can we learn about planned downtime?

Fortunately, by using a combination of monitoring solutions on Azure, you can:

- Gain answers, insights, and alerts to help ensure that you've optimized your cloud usage.
- Ascertain the root cause of unplanned issues.
- Prepare ahead of time for planned outages.

Azure Advisor

Azure Advisor evaluates your Azure resources and makes recommendations to help improve reliability, security, and performance, achieve operational excellence, and reduce costs. Advisor is designed to help you save time on cloud optimization. The recommendation service includes suggested actions you can take right away, postpone, or dismiss.

The recommendations are available via the Azure portal and the API, and you can set up notifications to alert you to new recommendations.

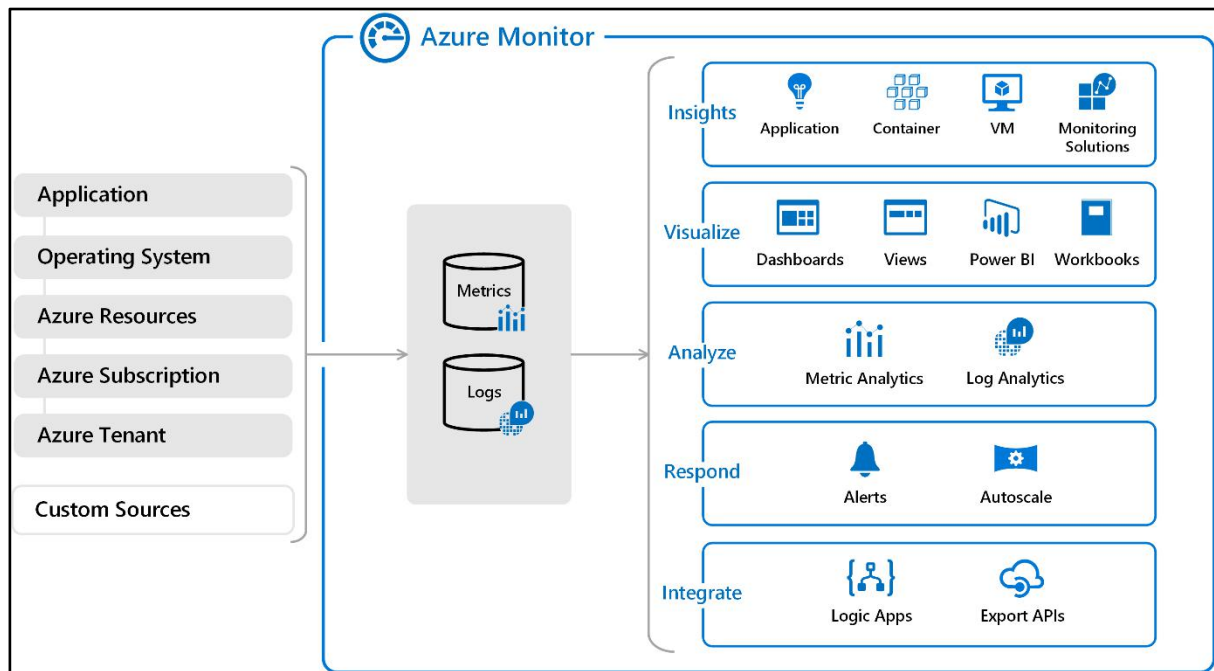
When you're in the Azure portal, the Advisor dashboard displays personalized recommendations for all your subscriptions, and you can use filters to select recommendations for specific subscriptions, resource groups, or services. The recommendations are divided into five categories:

- Reliability: Used to ensure and improve the continuity of your business-critical applications.
- Security: Used to detect threats and vulnerabilities that might lead to security breaches.
- Performance: Used to improve the speed of your applications.
- Cost: Used to optimize and reduce your overall Azure spending.
- Operational Excellence: Used to help you achieve process and workflow efficiency, resource manageability, and deployment best practices.

Azure Monitor

Azure Monitor is a platform for collecting, analyzing, visualizing, and potentially taking action based on the metric and logging data from your entire Azure and on-premises environment. All data collected by Azure Monitor fits into one of two fundamental types, metrics and logs.

The following diagram illustrates just how comprehensive Azure Monitor is.



- On the left is a list of the sources of logging and metric data that can be collected at every layer in your application architecture, from application to operating system and network.
- In the center, you can see how the logging and metric data is stored in central repositories.
- On the right, the data is used in a number of ways. You can view real-time and historical performance across each layer of your architecture, or aggregated and detailed information. The data is displayed at different levels for different audiences. You can view high-level reports on the Azure Monitor Dashboard or create custom views by using Power BI and Kusto queries.

Additionally, you can use the data to help you react to critical events in real time, through alerts delivered to teams via SMS, email, and so on. Or you can use thresholds to trigger autoscaling functionality to scale up or down to meet the demand.

Some popular products such as Azure Application Insights, a service for sending telemetry information from application source code to Azure, uses Azure Monitor under the hood. With Application Insights, your application developers can take advantage of the powerful data-analysis platform in Azure Monitor to gain deep insights into an application's operations and diagnose errors without having to wait for users to report them. It is an extensible Application Performance Management (APM) service.

Log Analytics is a tool in the Azure portal used to edit and run log queries with data in Azure Monitor Logs. You may write a simple query that returns a set of records and then use features of Log Analytics to sort, filter, and analyze them. Or you may write a more advanced query to perform statistical analysis and visualize the results in a chart to identify a particular trend. Whether you work with the results of your queries interactively or use them with other Azure Monitor features such as log query alerts or workbooks, Log Analytics is the tool that you're going to use write and test them.

Azure Service Health

Azure Service Health provides a personalized view of the health of the Azure services, regions, and resources you rely on. The status.azure.com website, which displays only major issues that broadly affect Azure customers, doesn't provide the full picture. But Azure Service Health displays both major and smaller, localized issues that affect you. Service issues are rare, but it's important to be prepared for the unexpected. You can set up alerts that help you triage outages and planned maintenance. After an outage, Service Health provides official incident reports, called root cause analyses (RCAs), which you can share with stakeholders.

Service Health helps you keep an eye on several event types:

- Service issues are problems in Azure, such as outages, that affect you right now. You can drill down to the affected services, regions, updates from your engineering teams, and find ways to share and track the latest information.
- Planned maintenance events can affect your availability. You can drill down to the affected services, regions, and details to show how an event will affect you and what you need to do. Most of these events occur without any impact to you and aren't shown here. In the rare case that a reboot is required, Service Health allows you to choose when to perform the maintenance to minimize the downtime.
- Health advisories are issues that require you to act to avoid service interruption, including service retirements and breaking changes. Health advisories are announced far in advance to allow you to plan.

Help and Support > Service Health > Planned Maintenance

Azure Event Hubs

Azure Event Hubs is a big data streaming platform and event ingestion service. It can receive and process millions of events per second. Data sent to an event hub can be transformed and stored by using any real-time analytics provider or batching/storage adapters. Azure Event Hubs can be used to ingest, buffer, store, and process your stream in real time to get actionable insights. Event Hubs uses a partitioned consumer model, enabling multiple applications to process the stream concurrently and letting you control the speed of processing. Azure Event Hubs can be used to capture your data in near-real time in an Azure Blob storage or Azure Data Lake Storage for long-term retention or micro-batch processing. The following scenarios are some of the scenarios where you can use Event Hubs:

- Anomaly detection (fraud/outliers)
- Application logging
- Analytics pipelines, such as clickstreams
- Live dashboarding
- Archiving data
- Transaction processing
- User telemetry processing
- Device telemetry streaming

Analyze the decision criteria

In this unit, you'll analyze the criteria that experts employ when they choose an Azure monitoring service for a specified business need. By understanding the criteria, you can better assess the nuanced differences among the products.

Do you need to analyze how you're using Azure to reduce costs? Improve resilience? Harden your security? Choose Azure Advisor when you're looking for an analysis of your deployed resources. Azure Advisor analyzes the configuration and usage of your resources and provides suggestions on how to optimize for reliability, security, performance, costs, and operations based on experts' best practices.

Do you want to monitor Azure services or your usage of Azure? If you want to keep tabs on Azure itself, especially the services and regions you depend on, you want to choose Azure Service Health. You can view the current status of the Azure services you rely on, upcoming planned outages, and services that will be sunset. You can set up alerts that help you stay on top of incidents and upcoming downtime without having to visit the dashboard regularly. However, if you want to keep track of the performance or issues related to your specific VM or container instances, databases, your applications, and so on, you want to visit Azure Monitor and create reports and notifications to help you understand how your services are performing or diagnose issues related to your Azure usage.

Do you want to measure custom events alongside other usage metrics? Choose Azure Monitor when you want to measure custom events alongside other collected telemetry data. Custom events, such as those added in the source code of your software applications, could help identify and diagnose why your application is behaving a certain way.

Do you need to set up alerts for outages or when autoscaling is about to deploy new instances? Here again, you would use Azure Monitor to set up alerts for key events that are related to your specific resources.

IoT

People are able to access more information than ever before. Personal digital assistants led to smartphones, and now there are smart watches, smart thermostats, and even smart refrigerators. Personal computers used to be the norm. Now the internet allows any item that's online-capable to access valuable information. This ability for devices to garner and then relay information for data analysis is referred to as IoT (Internet of Things).

IoT bridges the physical and digital worlds by enabling devices with sensors and an internet connection to communicate with cloud-based systems via the internet. IoT enables devices to gather and then relay information for data analysis. Smart devices are equipped with sensors that collect data. A few common sensors that measure attributes of the physical world include

- Environmental sensors that capture temperature and humidity levels.
- Barcode, QR code, or optical character recognition (OCR) scanners.
- Geo-location and proximity sensors.
- Light, color, and infrared sensors.
- Sound and ultrasonic sensors.
- Motion and touch sensors.
- Accelerometer and tilt sensors.
- Smoke, gas, and alcohol sensors.
- Error sensors to detect when there's a problem with the device.
- Mechanical sensors that detect anomalies or deformations.
- Flow, level, and pressure sensors for measuring gasses and liquids.

By using Azure IoT services, devices that are equipped with these kinds of sensors and that can connect to the internet could send their sensor readings to a specific endpoint in Azure via a message. The message's data is then collected and aggregated, and it can be converted into reports and alerts. Alternately, all devices could be updated with new firmware to fix issues or add new functionality by sending software updates from Azure IoT services to each device.

Let's suppose your company manufactures and operates smart refrigerated vending machines. What kinds of information would you want to monitor? You might want to ensure that:

- Each machine is operating without any errors.
- The machines haven't been compromised.
- The machines' refrigeration systems are keeping their contents within a certain temperature range.
- You're notified when products reach a certain inventory level so you can restock the machines.

If the hardware of your vending machines can collect and send this information in a standard message, the messages each machine sends can be received, stored, organized, and displayed by using Azure IoT services. The data that's collected from these devices could be combined with Azure AI services to help you predict:

- When machines need proactive maintenance.
- When inventories will need to be replenished and new product ordered from vendors.

Many services can assist and drive end-to-end solutions for IoT on Azure, these are described in the following pages.

Analyze the decision criteria

In this unit, we'll analyze the criteria that experts employ when they decide which IoT service to use for a given business need. Understanding the criteria can also help you better understand the nuanced differences between each product.

Is it critical to ensure that the device is not compromised?

No manufacturers or customers want their devices to be maliciously compromised and used for nefarious purposes, but it's more critical to ensure the integrity of an ATM than, say, a washing machine. When security is a critical consideration in your product's design, the best product option is Azure Sphere, which provides a comprehensive end-to-end solution for IoT devices.

As we mentioned in the previous unit, Azure Sphere ensures a secure channel of communication between the device and Azure by controlling everything from the hardware to the operating system and the authentication process. This ensures that the integrity of the device is uncompromised. After a secure channel is established, messages can be received from the device securely, and messages or software updates can be sent to the device remotely.

Do I need a dashboard for reporting and management?

Your next decision will be the level of services you require from your IoT solution. If you merely want to connect to your remote devices to receive telemetry and occasionally push updates, and you don't need any reporting capabilities, you might prefer to implement Azure IoT Hub by itself. Your programmers can still create a customized set of management tools and reports by using the IoT Hub RESTful API.

However, if you want a pre-built customizable user interface with which you can view and control your devices remotely, you might prefer to start with IoT Central. With this solution, you can control a single device or all devices at once, and you can set up alerts for certain conditions, such as a device failure.

IoT Central integrates with many different Azure products, including IoT Hub, to create a dashboard with reports and management features. The dashboard is based on starter templates for common industry and usage scenarios. You can use the dashboard that's generated by the starter template as is or customize it to suit your needs. You can have multiple dashboards and target them at a variety of users.

Azure IoT Hub

Messaging hub that provides secure communications between and monitoring of millions of IoT devices.

IoT Hub is a managed service that's hosted in the cloud and that acts as a central message hub for bi-directional communication between your IoT application and the devices it manages. You can use Azure IoT Hub to build IoT solutions with reliable and secure communications between millions of IoT devices and a cloud-hosted solution back end. You can connect virtually any device to your IoT hub.

The IoT Hub service supports communications both from the device to the cloud and from the cloud to the device. It also supports multiple messaging patterns, such as device-to-cloud telemetry, file upload from devices, and request-reply methods to control your devices from the cloud. After an IoT hub receives messages from a device, it can route that message to other Azure services.

From a cloud-to-device perspective, IoT Hub allows for command and control. That is, you can have either manual or automated remote control of connected devices, so you can instruct the device to open valves, set target temperatures, restart stuck devices, and so on.

IoT Hub monitoring helps you maintain the health of your solution by tracking events such as device creation, device failures, and device connections.

Azure IoT Central

Fully managed global IoT software as a service (SaaS) solution that makes it easy to connect, monitor, and manage IoT assets at scale.

Azure IoT Central builds on top of IoT Hub by adding a dashboard that allows you to connect, monitor, and manage your IoT devices. The visual user interface (UI) makes it easy to quickly connect new devices and watch as they begin sending telemetry or error messages. You can watch the overall performance across all devices in aggregate, and you can set up alerts that send notifications when a specific device needs maintenance. Finally, you can push firmware updates to the device.`

To help you get up and running quickly, IoT Central provides starter templates for common scenarios across various industries, such as retail, energy, healthcare, and government. You then customize the design starter templates directly in the UI by choosing from existing themes or creating your own custom theme, setting the logo, and so on. With IoT Central, you can tailor the starter templates for the specific data that's sent from your devices, the reports you want to see, and the alerts you want to send.

You can use the UI to control your devices remotely. This feature allows you to push a software update or modify a property of the device. You can adjust the desired temperature for one or all of your refrigerated vending machines from directly inside of IoT Central.

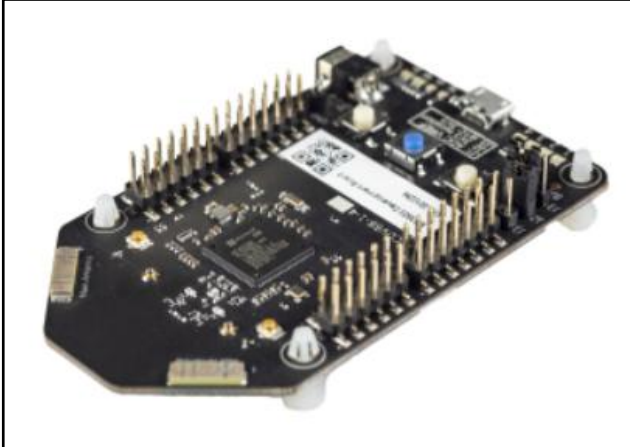
A key part of IoT Central is the use of device templates. By using a device template, you can connect a device without any service-side coding. IoT Central uses the templates to construct the dashboards, alerts, and so on. Device developers still need to create code to run on the devices, and that code must match the device template specification.

Azure Sphere

Azure Sphere creates an end-to-end, highly secure IoT solution for customers that encompasses everything from the hardware and operating system on the device to the secure method of sending messages from the device to the message hub. Azure Sphere has built-in communication and security features for internet-connected devices.

Azure Sphere comes in three parts:

- The first part is the Azure Sphere micro-controller unit (MCU), which is responsible for processing the operating system and signals from attached sensors. The following image displays the Seeed Azure Sphere MT3620 Development Kit MCU, one of several different starter kits that are available for prototyping and developing Azure Sphere applications.



- The second part is a customized Linux operating system (OS) that handles communication with the security service and can run the vendor's software.
- The third part is Azure Sphere Security Service, also known as AS3. Its job is to make sure that the device has not been maliciously compromised. When the device attempts to connect to Azure, it first must authenticate itself, per device, which it does by using certificate-based authentication. If it authenticates successfully, AS3 checks to ensure that the device hasn't been tampered with. After it has established a secure channel of communication, AS3 pushes any OS or approved customer-developed software updates to the device.

After the Azure Sphere system has validated the authenticity of the device and authenticated it, the device can interact with other Azure IoT services by sending telemetry (in situ collection of measurements or other data at remote points) and error information.

IoT Edge

Fully managed service that allows data analysis models to be pushed directly onto IoT devices, which allows them to react quickly to state changes without needing to consult cloud-based AI models.

Big data

Several years ago, Tailwind Traders rolled out a new GPS tracking system for all of its delivery vehicles. The new system provides real-time tracking data to your primary datacenter. Your CTO wants your team to look at several years of tracking data in order to determine trends. For example, an important trend might be a spike in deliveries around the holidays that would require hiring additional staff. Through an in-depth analysis of the tracking data that you've recorded, your CTO seeks to predict when changes are necessary, and then proactively take the necessary steps to appropriately manage spikes.

Data comes in all types of forms and formats. When we talk about big data, we're referring to large volumes of data. In this Tailwind Traders scenario, data is collected from the GPS sensors, which includes location information, data from weather systems, and many other sources that generate large amounts of data. This amount of data becomes increasingly hard to make sense of and to base decisions on. The volumes are so large that traditional forms of processing and analysis are no longer appropriate.

Open-source cluster technologies have been developed, over time, to try to deal with these large datasets. Azure supports a broad range of technologies and services to provide big data and analytic solutions.

Azure Service	Description
Azure Synapse Analytics	Run analytics at a massive scale by using a cloud-based enterprise data warehouse that takes advantage of massively parallel processing to run complex queries quickly across petabytes of data. Azure Synapse Analytics (formerly Azure SQL Data Warehouse) is a limitless analytics service that brings together enterprise data warehousing and big data analytics. You can query data on your terms by using either serverless or provisioned resources at scale. You have a unified experience to ingest, prepare, manage, and serve data for immediate BI (Business Intelligence) and machine learning needs.
Azure HDInsight	Process massive amounts of data with managed clusters in the cloud. Azure HDInsight is a fully managed, open-source analytics service for enterprises. It's a cloud service that makes it easier, faster, and more cost-effective to process massive amounts of data. You can run popular open-source frameworks and create cluster types such as Apache Spark, Apache Hadoop, Apache Kafka, Apache HBase, Apache Storm, and Machine Learning Services. HDInsight also supports a broad range of scenarios such as extraction, transformation, and loading (ETL), data warehousing, machine learning, and IoT.
Azure Databricks	Integrate this collaborative Apache Spark-based analytics service with other big data services in Azure. Azure Databricks helps you unlock insights from all your data and build artificial intelligence solutions. You can set up your Apache Spark environment in minutes, and then autoscale and collaborate on shared projects in an interactive workspace. Azure Databricks supports Python, Scala, R, Java, and SQL, as well as data science frameworks and libraries including TensorFlow, PyTorch, and scikit-learn.
Azure Data Lake Analytics	Azure Data Lake Analytics is an on-demand analytics job service that simplifies big data. Instead of deploying, configuring, and tuning hardware, you write queries to transform your data and extract valuable insights. The analytics service can handle jobs of any scale instantly by setting the dial for how much power you need. You only pay for your job when it's running, making it more cost-effective.

AI

AI, in the context of cloud computing, is based around a broad range of services, the core of which is machine learning. Machine learning is a data science technique that allows computers to use existing data to forecast future behaviors, outcomes, and trends. Using machine learning, computers learn without being explicitly programmed.

Forecasts or predictions from machine learning can make apps and devices smarter. For example, when you shop online, machine learning helps recommend other products you might like based on what you've purchased. Or when your credit card is swiped, machine learning compares the transaction to a database of transactions and helps detect fraud. And when your robot vacuum cleaner vacuums a room, machine learning helps it decide whether the job is done.

AI is a broad classification of computing that allows a software system to perceive its environment and take action that maximizes its chance of successfully achieving its goals. A goal of AI is to create a software system that's able to adapt, or learn something on its own without being explicitly programmed to do it.

There are two basic approaches to AI. The first is to employ a deep learning system that's modeled on the neural network of the human mind, enabling it to discover, learn, and grow through experience.

The second approach is machine learning, a data science technique that uses existing data to train a model, test it, and then apply the model to new data to forecast future behaviors, outcomes, and trends.

Forecasts or predictions from machine learning can make apps and devices smarter. For example, when you shop online, machine learning powers product recommendation systems that offer additional products based on what you've bought and what other shoppers have bought who have purchased similar items in the past.

Machine learning is also used to detect credit card fraud by analyzing each new transaction and using what it has learned from analyzing millions of fraudulent transactions.

Virtually every device or software system that collects textual, visual, and audio data could feed a machine learning model that makes that device or software system smarter about how it functions in the future.

At a high level, there are three primary product offerings from Microsoft, each of which is designed for a specific audience and use case. Each option provides a diverse set of tools, services, and programmatic APIs. In this module, we'll merely scratch the surface of the options' capabilities.

Language Understanding (LUIS) - A machine learning-based service to build natural language into apps, bots, and IoT devices. Quickly create enterprise-ready, custom models that continuously improve.

Azure Machine Learning

Azure Machine Learning is a platform for making predictions. It consists of tools and services that allow you to connect to data to train and test models to find one that will most accurately predict a future result. After you've run experiments to test the model, you can deploy and use it in real time via a web API endpoint.

With Azure Machine Learning, you can:

- Create a process that defines how to obtain data, how to handle missing or bad data, how to split the data into either a training set or test set, and deliver the data to the training process.
- Train and evaluate predictive models by using tools and programming languages familiar to data scientists.
- Create pipelines that define where and when to run the compute-intensive experiments that are required to score the algorithms based on the training and test data.
- Deploy the best-performing algorithm as an API to an endpoint so it can be consumed in real time by other applications.

Choose Azure Machine Learning when your data scientists need complete control over the design and training of an algorithm using your own data.

Azure Machine Learning Studio - Collaborative visual workspace where you can build, test, and deploy machine learning solutions by using prebuilt machine learning algorithms and data-handling modules.

Azure Cognitive Services

Azure Cognitive Services provides prebuilt machine learning models that enable applications to see, hear, speak, understand, and even begin to reason. Use Azure Cognitive Services to solve general problems, such as analyzing text for emotional sentiment or analyzing images to recognize objects or faces. You don't need special machine learning or data science knowledge to use these services. Developers access Azure Cognitive Services via APIs and can easily include these features in just a few lines of code.

While Azure Machine Learning requires you to bring your own data and train models over that data, Azure Cognitive Services, for the most part, provides pretrained models so that you can bring in your live data to get predictions on.

Azure Cognitive Services can be divided into the following categories:

- **Language services:** Allow your apps to process natural language with prebuilt scripts, evaluate sentiment, and learn how to recognize what users want.
- **Speech services:** Convert speech into text and text into natural-sounding speech. Translate from one language to another and enable speaker verification and recognition.
- **Vision services:** Add recognition and identification capabilities when you're analyzing pictures, videos, and other visual content.
- **Decision services / Knowledge mapping:** Map complex information and data to solve tasks such as intelligent recommendations and semantic search. Add personalized recommendations for each user that automatically improve each time they're used, moderate content to monitor and remove offensive or risky content, and detect abnormalities in your time series data.
- **Bing Search -** Add Bing Search APIs to your apps and harness the ability to comb billions of webpages, images, videos, and news with a single API call.

Azure Bot Service

Azure Bot Service and Bot Framework are platforms for creating virtual agents that understand and reply to questions just like a human. Azure Bot Service is a bit different from Azure Machine Learning and Azure Cognitive Services in that it has a specific use case. Namely, it creates a virtual agent that can intelligently communicate with humans. Behind the scenes, the bot you build uses other Azure services, such as Azure Cognitive Services, to understand what their human counterparts are asking for.

Bots can be used to shift simple, repetitive tasks, such as taking a dinner reservation or gathering profile information, on to automated systems that might no longer require direct human intervention. Users converse with a bot by using text, interactive cards, and speech. A bot interaction can be a quick question and answer, or it can be a sophisticated conversation that intelligently provides access to services.

Analyze the decision criteria

- Are you building a virtual agent that interfaces with humans via natural language?
 - Use Azure Bot Service when you need to create a virtual agent to interact with humans by using natural language. Bot Service integrates knowledge sources, natural language processing, and form factors to allow interaction across different channels. Bot Service solutions usually rely on other AI services for such things as natural language understanding or even translation for localizing replies into a customer's preferred language.
 - Before you jump in to build a custom chat experience by using Bot Service, it might make sense to search for prebuilt, no-code solutions that cover common scenarios. For example, you can use QnA Maker, which is available from Azure Marketplace, to build, train, and publish a sophisticated bot that uses FAQ pages, support websites, product manuals, SharePoint documents, or editorial content through an easy-to-use UI or via REST APIs.
 - Likewise, Power Virtual Agents integrates with Microsoft Power Platform so that you can use hundreds of prebuilt connectors for data input. You can extend Power Virtual Agents by building custom workflows with Power Automate, and if you feel that the out-of-the-box experience is too limiting, you can still build more complex interactions with Microsoft Bot Framework.
- Do you need a service that can understand the content and meaning of images, video, or audio, or that can translate text into a different language?
 - Use Azure Cognitive Services when it comes to general purpose tasks, such as performing speech to text, integrating with search, or identifying the objects in an image. Azure Cognitive Services is general purpose, meaning that many different kinds of customers can benefit from the work that Microsoft has already done to train and test these models and offer them inexpensively at scale.
- Do you need to predict user behavior or provide users with personalized recommendations in your app?
 - The Azure Cognitive Services Personalizer service watches your users' actions within an application. You can use Personalizer to predict their behavior and provide relevant experiences as it identifies usage patterns. Here again, you could capture and store user behavior and create your own custom Azure Machine Learning solution to do these things, but this approach would require much effort and expense.
- Will your app predict future outcomes based on private historical data?
 - Choose Azure Machine Learning when you need to analyze data to predict future outcomes. For example, suppose you need to analyze years' worth of financial transactions to discover new patterns that could help you create new products and services for your company's clients and then offer those new services during routine customer service calls. When you're working with proprietary data, you'll likely need to build a more custom-tailored machine learning model.
- Do you need to build a model by using your own data or perform a different task than those listed above?
 - Use Azure Machine Learning for maximum flexibility. Data scientists and AI engineers can use the tools they're familiar with and the data you provide to develop deep learning and machine learning models that are tuned for your particular requirements.

DevOps

DevOps brings together people, processes, and technology by automating software delivery to provide continuous value to your users. With Azure DevOps, you can create build and release pipelines that provide continuous integration, delivery, and deployment for your applications. You can integrate repositories and application tests, perform application monitoring, and work with build artifacts. You can also work with and backlog items for tracking, automate infrastructure deployment, and integrate a range of third-party tools and services such as Jenkins and Chef. All of these functions and many more are closely integrated with Azure to allow for consistent, repeatable deployments for your applications to provide streamlined build and release processes.

Azure DevOps Services

Azure DevOps Services is a suite of services that address every stage of the software development lifecycle.

- Azure Repos is a centralized source-code repository where software development, DevOps engineering, and documentation professionals can publish their code for review and collaboration.
- Azure Boards is an agile project management suite that includes Kanban boards, reporting, and tracking ideas and work from high-level epics to work items and issues.
- Azure Pipelines is a CI/CD pipeline automation tool.
- Azure Artifacts is a repository for hosting artifacts, such as compiled source code, which can be fed into testing or deployment pipeline steps.
- Azure Test Plans is an automated test tool that can be used in a CI/CD pipeline to ensure quality before a software release.

Azure DevOps is a mature tool with a large feature set that began as on-premises server software and evolved into a software as a service (SaaS) offering from Microsoft.

GitHub and GitHub Actions

GitHub is arguably the world's most popular code repository for open-source software. Git is a decentralized source-code management tool, and GitHub is a hosted version of Git that serves as the primary remote. GitHub builds on top of Git to provide related services for coordinating work, reporting and discussing issues, providing documentation, and more. It offers the following functionality:

- It's a shared source-code repository, including tools that enable developers to perform code reviews by adding comments and questions in a web view of the source code before it can be merged into the main code base.
- It facilitates project management, including Kanban boards.
- It supports issue reporting, discussion, and tracking.
- It features CI/CD pipeline automation tooling.
- It includes a wiki for collaborative documentation.
- It can be run from the cloud or on-premises

Most relevant for this module, GitHub Actions enables workflow automation with triggers for many lifecycle events. One such example would be automating a CI/CD toolchain.

A toolchain is a combination of software tools that aid in the delivery, development, and management of software applications throughout a system's development lifecycle. The output of one tool in the toolchain is the input of the next tool in the toolchain. Typical tool functions range from performing automated dependency updates to building and configuring the software, delivering the build artifacts to various locations, testing, and so on.

With such similarity between many GitHub and Azure DevOps features, you might wonder which product to choose for your organization. Unfortunately, the answer might not be straightforward.

Although both Azure DevOps and GitHub allow public and private code repositories, GitHub has a long history with public repositories and is trusted by tens of thousands of open-source project owners. GitHub is a lighter-weight tool than Azure DevOps, with a focus on individual developers contributing to the open-source code. Azure DevOps, on the other hand, is more focused on enterprise development, with heavier project-management and planning tools, and finer-grained access control.

Note: Your choices are not limited to Azure DevOps Services or GitHub and GitHub Actions. In practice, you can mix and match these services as needed. For example, you can use GitHub repos with Azure Boards for work item tracking.

Azure DevTest Labs

Azure DevTest Labs provides an automated means of managing the process of building, setting up, and tearing down virtual machines (VMs) that contain builds of your software projects. This way, developers and testers can perform tests across a variety of environments and builds. And this capability isn't limited to VMs. Anything you can deploy in Azure via an ARM template can be provisioned through DevTest Labs. Provisioning pre-created lab environments with their required configurations and tools already installed is a huge time saver for quality assurance professionals and developers.

Suppose you need to test a new feature on an old version of an operating system. Azure DevTest Labs can set up everything automatically upon request. After the testing is complete, DevTest Labs can shut down and deprovision the VM, which saves money when it's not in use. To control costs, the management team can restrict how many labs can be created, how long they run, and so on.

Analyze the decision criteria

In this unit, you'll analyze the criteria that experts employ when they choose DevOps tools or services to address specific business needs. Understanding the criteria can also help you better understand the nuanced differences between each product.

Do you need to automate and manage test-lab creation? If your aim is to automate the creation and management of a test lab environment, consider choosing Azure DevTest Labs. Among the three tools and services we've described, it's the only one that offers this functionality. However, you can automate the provisioning of new labs as part of a toolchain by using Azure Pipelines or GitHub Actions.

Are you building open-source software? Although Azure DevOps can publish public code repositories, GitHub has long been the preferred host for open-source software. If you're building open-source software, you would likely choose GitHub if for no other reasons than its visibility and general acceptance by the open-source development community.

The remaining decision criteria are specific to choosing between either Azure DevOps or GitHub.

Note: Your choices aren't limited to Azure DevOps Services or GitHub and GitHub Actions. In practice, you can mix and match these services as needed. For example, you can use GitHub repos with Azure Boards for work-item tracking.

Regarding source-code management and DevOps tools, what level of granularity do you need for permissions? GitHub works on a simple model of read/write permissions to every feature. Meanwhile, Azure DevOps has a much more granular set of permissions that allow organizations to refine who is able to perform most operations across the entire toolset.

Regarding source-code management and DevOps tools, how sophisticated does your project management and reporting need to be? Although GitHub has work items, issues, and a Kanban board, project management and reporting is the area where Azure DevOps excels. Azure DevOps is highly customizable, which allows an administrator to add custom fields to capture metadata and other information alongside each work item. By contrast, the GitHub Issues feature uses tags as its primary means of helping a team categorize issues.

Regarding source-code management and DevOps tools, how tightly do you need to integrate with third-party tools? Although we make no specific recommendations about third-party tools, it's important for you to understand your organization's existing investments in tools and services and to evaluate how these dependencies might affect your choice. It's likely that most vendors that create DevOps tools create hooks or APIs that can be used by both Azure Pipelines and GitHub Actions. Even so, it's probably worth the effort to validate that assumption.

Glossary

A

- (An) Active Directory Forest - is the collection of more than one domain trees having different name spaces or roots. This means that the forest contains a number of domain trees that do not share a common name space, or more so, do not have the same parent domain.
- (Azure) AD – (Azure) Active Directory
 - The Azure Active Directory Connect synchronization services (Azure AD Connect sync) is a main component of Azure AD Connect. It takes care of all the operations that are related to synchronize identity data between your on-premises environment and Azure AD.
- (Azure) AD Domain Services - provides managed domain services such as domain join, group policy, LDAP, and Kerberos/NTLM authentication. You can use Azure AD Domain Services without needing to manage, patch, or service domain controllers in the cloud. For ease and simplicity, defaults have been specified to provide a one-click deployment.
- (Azure) AD Identity Protection
 - Each day, Microsoft collects and analyses trillions of anonymized signals as part of user sign-in attempts. These signals help build patterns of good user sign-in behavior, and identify potential risky sign-in attempts. Azure AD Identity Protection can review user sign-in attempts and take additional action if there's suspicious behavior. Some of the following actions may trigger Azure AD Identity Protection risk detection:
 - Users with leaked credentials.
 - Sign-ins from anonymous IP addresses.
 - Impossible travel to atypical locations.
 - Sign-ins from infected devices.
 - Sign-ins from IP addresses with suspicious activity.
 - Sign-ins from unfamiliar locations.
 - The following three policies are available in Azure AD Identity Protection to protect users and respond to suspicious activity. You can choose to turn the policy enforcement on or off, select users or groups for the policy to apply to, and decide if you want to block access at sign-in or prompt for additional action.
 - User risk policy - Identifies and responds to user accounts that may have compromised credentials. Can prompt the user to create a new password.
 - Sign in risk policy - Identifies and responds to suspicious sign-in attempts. Can prompt the user to provide additional forms of verification using Azure AD Multi-Factor Authentication.
 - MFA registration policy - Makes sure users are registered for Azure AD Multi-Factor Authentication. If a sign-in risk policy prompts for MFA, the user must already be registered for Azure AD Multi-Factor Authentication.
 - When you enable a policy user or sign in risk policy, you can also choose the threshold for risk level - *low and above, medium and above, or high*. This flexibility lets you decide how aggressive you want to be in enforcing any controls for suspicious sign-in events.
- (Azure) Advanced Threat Protection (ATP) is a cloud-based security solution that leverages your on-premises Active Directory signals to identify, detect, and investigate advanced threats, compromised identities, and malicious insider actions directed at your organization. Sensors are software packages you install on your servers to upload information to Azure ATP.

B

Just as a blueprint allows an engineer or an architect to sketch a project's design parameters, Azure Blueprints 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 trust they're building within organizational compliance with a set of built-in components, such as networking, to speed up development and delivery. Blueprints are a declarative way to orchestrate the deployment of various resource templates and other artifacts such as:

- Role Assignments
- Policy Assignments
- Azure Resource Manager templates (ARM templates)
- Resource Groups

The Azure Blueprints service is backed by the globally distributed Azure Cosmos DB. Blueprint objects are replicated to multiple Azure regions. This replication provides low latency, high availability, and consistent access to your blueprint objects, regardless of which region Azure Blueprints deploys your resources to.

C

(Azure) China - Microsoft Azure operated by 21Vianet (Azure China) is a physically separated instance of cloud services located in China. It's independently operated and transacted by Shanghai Blue Cloud Technology Co., Ltd. ("21Vianet"), a wholly owned subsidiary of Beijing 21Vianet Broadband Data Center Co., Ltd.

Cloud Shell - Your Microsoft-managed admin machine in Azure, for Azure. Shell access from virtually anywhere. Connect to Azure using an authenticated, browser-based shell experience that's hosted in the cloud and accessible from virtually anywhere. Azure Cloud Shell is assigned per unique user account, and is automatically authenticated with each session. Get a modern command-line experience from multiple access points, including the Azure portal, shell.azure.com, Azure mobile app, Azure docs (e.g. Azure CLI, Azure PowerShell) and VS Code Azure Account extension.

Compliance Manager - in the Service Trust Portal is a workflow-based risk assessment tool that helps you track, assign, and verify your organization's regulatory compliance activities related to Microsoft Cloud services, such as Microsoft 365, Dynamics 365, and Azure.

Azure Cost Management - Azure customers with an Azure Enterprise Agreement (EA), Microsoft Customer Agreement (MCA), or Microsoft Partner Agreement (MPA) can use Azure Cost Management. Cost management is the process of effectively planning and controlling costs involved in your business. Cost management tasks are normally performed by finance, management, and app teams. Azure Cost Management + Billing helps organizations plan with cost in mind. It also helps to analyze costs effectively and take action to optimize cloud spending.

D

Distributed system - A system whose components are located on different networked computers, which communicate and coordinate their actions by passing messages to one another from any system.

E

Hadoop - Apache Hadoop is a collection of open-source software utilities that facilitates using a network of many computers to solve problems involving massive amounts of data and computation.

F

Federation is a collection of domains that have established trust, e.g. Authorization to access Azure resources can be provided by other identity providers by using federation. A commonly used example of this is to federate your on-premises Active Directory environment with Azure AD and use this federation for authentication and authorization.

G

GDPR is the General Data Protection Regulations. This standard was adopted across Europe in May 2018 and replaces the now deprecated Data Protection Directive. The General Data Protection Regulation (EU) (GDPR) is a regulation in EU law on data protection and privacy in the European Union (EU) and the European Economic Area (EEA). It also addresses the transfer of personal data outside the EU and EEA areas. The GDPR aims primarily to give control to individuals over their personal data and to simplify the regulatory environment for international business by unifying the regulation within the EU.

(Azure) Germany - is available to eligible customers and partners globally who intend to do business in the EU/EFTA, including the United Kingdom. Azure Germany offers a separate instance of Microsoft Azure services from within German datacenters. The datacenters are in two locations, Frankfurt/Main and Magdeburg. This placement ensures that customer data remains in Germany and that the datacenters connect to each other through a private network. All customer data is exclusively stored in those datacenters. A designated German company--the German data trustee--controls access to customer data and the systems and infrastructure that hold customer data.

(Azure) Government is the mission-critical cloud, delivering breakthrough innovation to US government customers and their partners. Only US federal, state, local and tribal governments and their partners have access to this dedicated instance, with operations controlled by screened US citizens. The key difference between Microsoft Azure and Microsoft Azure Government is that Azure Government is a sovereign cloud. It's a physically separated instance of Azure, dedicated to U.S. government workloads only. It's built exclusively for government agencies and their solution providers. Azure Government services handle data that is subject to certain government regulations and requirements, such as FedRAMP, NIST 800.171 (DIB), ITAR, IRS 1075, DoD L4, and CJIS. In order to provide you with the highest level of security and compliance, Azure Government uses physically isolated datacenters and networks (located in U.S. only).

H

(Apache) Hadoop - is a collection of open-source software utilities that facilitates using a network of many computers to solve problems involving massive amounts of data and computation. It provides a software framework for distributed storage and processing of big data using the MapReduce programming model.

(Azure) Hybrid Benefit - is a licensing benefit that helps you to significantly reduce the costs of running your workloads in the cloud. It works by letting you use your on-premises Software Assurance-enabled Windows Server and SQL Server licenses on Azure.

I

(Azure) Information Protection - is a cloud-based solution that helps an organization to classify and optionally, protect its documents and emails by applying labels. Labels can be applied automatically by administrators who define rules and conditions, manually by users, or a combination where users are given recommendations. The protection technology uses Azure Rights Management (often abbreviated to Azure RMS). This technology is integrated with other Microsoft cloud services and applications, such as Office 365 and Azure Active Directory. This protection technology uses encryption, identity, and authorization policies. Similarly to the labels that are applied, protection that is applied by using Rights Management stays with the documents and emails, independently of the location, inside or outside your organization, networks, file servers, and applications. Also, Azure Information Protection is used to automatically add a watermark to Microsoft Word documents that contain credit card information.

ISO is the International Organization for Standardization. Companies can be certified to ISO standards, for example ISO 9001 or 27001 are commonly used in IT companies.

J

Just-In-Time (JIT) - virtual machine (VM) access feature in Azure Security Center allows you to lock down inbound traffic to your Azure Virtual Machines. This reduces exposure to attacks while providing easy access when you need to connect to a VM.

K
L
M

The Microsoft Enterprise Agreement offers the best value to organizations with 500 or more users or devices that want a manageable volume licensing program that gives them the flexibility to buy cloud services and software licenses under one agreement.

The Microsoft Privacy Statement - explains what personal data Microsoft processes, how Microsoft processes the data, and the purpose of processing the data

N

NIST - The National Institute of Standards and Technology is a physical sciences laboratory, and a non-regulatory agency of the United States Department of Commerce

O
P

PowerApps - PowerApps lets you quickly build business applications with little or no code. PowerApps Portals allow organizations to create websites which can be shared with users external to their organization either anonymously or through the login provider of their choice like LinkedIn, Microsoft Account, other commercial login providers.

Q
R
S

Semantic search - refers to the ability of search engines to consider the intent and contextual meaning of search phrases when serving content to users on the web.

(Azure) Service Bus is a messaging service on cloud used to connect any applications, devices, and services running in the cloud to any other applications or services. As a result, it acts as a messaging backbone for applications available in the cloud or across any devices.

SKU - Stock Keeping Unit (meaning pricing tier)

SN architecture - Shared-Nothing architecture is a distributed computing architecture in which each update request is satisfied by a single node (processor/memory/storage unit). The intent is to eliminate contention among nodes

Spark - Apache Spark is an open-source unified analytics engine for large-scale data processing.

T

Telemetry - is the in situ collection of measurements or other data at remote points and their automatic transmission to receiving equipment (telecommunication) for monitoring

(Microsoft) Trust Center is an important part of the Microsoft Trusted Cloud Initiative and provides support and resources for the legal and compliance community. The Trust Center provides: In-depth information about security, privacy, compliance offerings, policies, features, and practices across Microsoft cloud products.

U

UNC Path - Universal Naming Convention Path (or **Uniform** Naming Convention), it is a PC format for specifying the location of resources on a local-area network (LAN). UNC uses the following format: \\server-name\shared-resource-pathname

V

W

X

Y

Z

To Add

Azure Functions

Azure Container Instances

Azure Kubernetes Service

Azure Cosmos DB

azure blobs

Add quiz questions from azure course

API REST vs SOAP

Acronyms extract

service-level agreements (SLAs)

Ensure bullet list indents and tables are fixed after document resizing

Uniform Resource Identifier (URI)

Regression testing (rarely non-regression testing[1]) is re-running functional and non-functional tests to ensure that previously developed and tested software still performs after a change

Authentication vs authorisation - The difference between authentication and authorization is:

- Authentication is proving your identity, proving that you are who you say you are. The most common example of this is logging in to a system by providing credentials such as a username and password.
- Authorization is what you're allowed to do once you've been authenticated. For example, what resources you're allowed to access and what you can do with those resources.