**Difference between webapp and logic app**

**What is Azure?**

The Azure cloud platform is more than 200 products and cloud services designed to help you bring new solutions to life—to solve today’s challenges and create the future. Build, run and manage applications across multiple clouds, on-premises, and at the edge, with the tools and frameworks of your choice.

**Benefits of Azure**

* You don’t need to invest in hardware upfront.
* You can terminate the resources whenever you don’t require them.
* You only pay for how much you use.
* **Azure Account**
* **Subscriptions** –This is a way of billing how much you use on Azure.
* **ACD (Azure Active Directory)** – This is an identity store (You can create users that have access to resources that you define in Azure)
* **Resource** – Every resource needs to be part of a resource group.
* **Regions or locationsA picture containing graphical user interface

  Description automatically generated**
* **RG (Resource Groups)** – The RG is used to go out and provide a logical grouping of resources. A resource needs to be mapped onto a resource group.
* **VM (Virtual Machine)**
  + **A Series | Entry Level** 
    - **Dev Test**
    - **Web Servers**
    - **Databases**
    - **Code Repositories**
  + **B Series | Burstable VMs**
  + **D Series | General Purpose Compute** – Run on Intel Xeon Processors
    - **Offer the vcpu’s, memory**
    - **And temporary storage**
  + **Da Series** - Use AMD Epic Processors
  + **Dc Series** – If you need additional Security
  + **E Series | Memory Optimized VMs**
  + **M Series is ideal for extremely large databases**
  + **Constrained vCPUs** – To reduce the cost of S/W
  + **F Series | Compute Optimized** – High CPU to memory ratio and great for medium traffic web servers, Network Appliances, Batch Processes, and Application servers as well as video encoding and rendering, AI inferencing, and gaming applications.
    - **F series runs on the latest Xeon Scalable processors and can scale up to 72 vcpu**.
  + **L Series | Storage Optimized** 
    - **Run Big data**
    - **No-SQL databases**
    - **Large Data warehousing**
    - **Up to 19.2TB of local storage which yields up to 3.8 Million IOPS**

**Virtual Machines in Azure**

|  |  |  |
| --- | --- | --- |
| Type | Sizes | Description |
| [General-purpose](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-general) | B, Dsv3, Dv3, Dasv4, Dav4, DSv2, Dv2, Av2, DC, DCv2, Dv4, Dsv4, Ddv4, Ddsv4, Dv5, Dsv5, Ddv5, Ddsv5, Dasv5, Dadsv5 | Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers. |
| [Compute-optimized](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-compute) | F, Fs, Fsv2, FX | High CPU-to-memory ratio. Good for medium-traffic web servers, network appliances, batch processes, and application servers. |
| [Memory-optimized](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-memory) | Esv3, Ev3, Easv4, Eav4, Ebdsv5, Ebsv5, Ev4, Esv4, Edv4, Edsv4, Ev5, Esv5, Edv5, Edsv5, Easv5, Eadsv5, Mv2, M, DSv2, Dv2 | High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics. |
| [Storage optimized](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-storage) | Lsv2, Lsv3, Lasv3 | High disk throughput and IO are ideal for Big Data, SQL, NoSQL databases, data warehou,sing and large transactional databases. |
| [GPU](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-gpu) | NC, NCv2, NCv3, NCasT4\_v3, ND, NDv2, NV, NVv3, NVv4, NDasrA100\_v4, NDm\_A100\_v4 | Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs. |
| [High performance compute](https://docs.microsoft.com/en-us/azure/virtual-machines/sizes-hpc) | HB, HBv2, HBv3, HC, H | Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA). |

**VM Connect**

* **RDP** (Remote Desktop Protocol)
* **SSH**
* **Bastion**

# Bastion

* Fully managed service that helps secure remote access to your virtual machines.
* Azure Bastion is a fully managed service that provides more secure and seamless Remote Desktop Protocol (RDP) and Secure Shell Protocol (SSH) access to virtual machines (VMs) without any exposure through public IP addresses.
* Steps 🡪 create vnet (subnet should be named “AzureBastionSubnet”> create bastion >

**RDP Port = 3389**

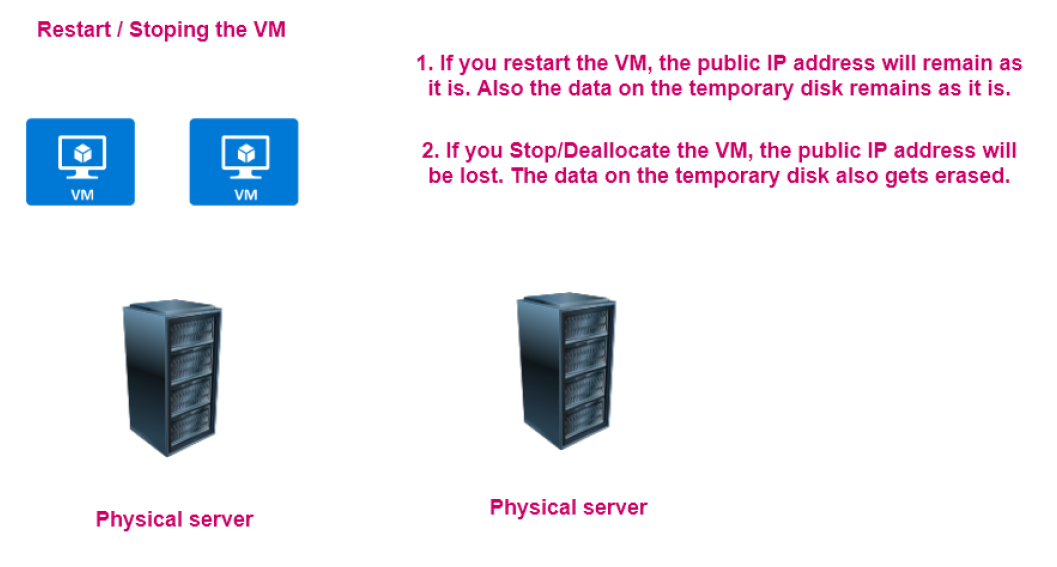
**Note: If we delete Resource Group then automatically all resources will be deleted, those that belong to that RG. And when you delete something then always check the deletion notification to confirm because sometimes some resources would not have been deleted because of some dependencies.**

**You can also create Budget alerts in Azure.**

**Graphical user interface, application

Description automatically generated**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

****

****

Graphical user interface, text

Description automatically generated

**Disk Types**

1. **Standard HDD**
   1. **This is ideal for backup environments and non-critical workloads.**
   2. **Max disk size –32,767 GiB**
   3. **Max throughput –500 MB/s**
   4. **Max IOPS -2000**
2. **Standard SSD**
   1. **This is ideal for Web Servers and Dev/Test Environments.**
   2. **Max disk size –32,767 GiB**
   3. **Max throughput –750 MB/s**
   4. **Max IOPS -6000.**
3. **Premium SSD**
   1. **This is ideal for Production environments.**
   2. **Max disk size –32,767 GiB**
   3. **Max throughput –950 MB/s**
   4. **Max IOPS –20,000.**
4. **Ultra Disk**
   1. **This is ideal for IO-intensive workloads –SQL, Oracle databases.**
   2. **Max disk size –32,767 GiB**
   3. **Max throughput –950 MB/s**
   4. **Max IOPS –20,000.**

**Azure provide 2 types of disk storage**

* Managed

Managed disk has some advantages over unmanaged disks in the sense that disks will be created and managed for you. This is a IaaS offering.

* Unmanaged

With unmanaged disks, you must manage it yourself. Basically, this means that your virtual hard disks are stored in a storage account as page blobs.

**Server Side Disk Encryption**

**Server-Side Encryption** (also referred to as encryption-at-rest or Azure Storage encryption) automatically encrypts data stored on Azure managed disks (OS and data disks) when persisting on the Storage Clusters. For full details, see [Server-side encryption of Azure Disk Storage](https://docs.microsoft.com/en-us/azure/virtual-machines/disk-encryption).

### **Customer-managed keys**

You can choose to manage encryption at the level of each managed disk, with your keys. When you specify a customer-managed key, that key is used to protect and control access to the key that encrypts your data. Customer-managed keys offer greater flexibility to manage access controls.

**VNET**

**Networking Basics**

IPv4 = 32 bit

IPv6 = 128 bit

**5 Different Classes of IP Address (Classful IP Addressing)**

Yhan kisi ki demand as it is fulfill nhi hoti. Jese Kisi ko 1000 IP addresses chahiye to isme use fixed number of IP addresses nhi milenge

|  |  |  |  |
| --- | --- | --- | --- |
| N | H | H | H |

1. A = 0-126 (Network Bit = 1, Host Bit = 0)

(127 is LoopBack Address (Self Testing, Local Host) 127.0.0.1 )

|  |  |  |  |
| --- | --- | --- | --- |
| N | N | H | H |

1. B = 128-191

|  |  |  |  |
| --- | --- | --- | --- |
| N | N | N | H |

1. C = 192-223
2. D = 224-239
3. E = 240-255

Note – Class A, B, C me network address find krne ke lie Host bit ko 0 se replace kr denge or HostAddress/BroadCast Address find krne ke lie host bit ki value maximum kr denge.

Network Address & Broadcast address never assign to any system. We can only assign IP address between Network address and Broadcast address.

**Ports = 65535**

1. C Class Network = 256

Usable IP = 256-2 = 254

1. B Class Network IP = 256\*256 = 65536

Usable IP = 65536-2 = 65534

1. A Class Network IP = 256\*256\*256 = 16,777,216

Usable IP = 16,777,216-2 = 16,777,214

**Subnetting**

* Network within a network.
* Logically division of IP addresses.
* The process of dividing a network into smaller network sections is called **subnetting**.
* Subnetting is the strategy use to partition a single physical network into more than one smaller logical sub networks/subnets.
* Router is a internetworking device.

Subnet is a segmented piece of large network. More specifically, subnets are a logical partition of an IP network into multiple, smaller network segments.

**CIDR Notations (Classless Inter-Domain Routing)** (Classless IP Addressing 1993)

* A method of assigning Internet Protocol (IP) addresses that **improves the efficiency of address distribution** and replaces the previous system based on Class A, Class B and Class C networks.
* Classless addressing me IP addresses ki westing bahut kam hoti h.
* Isme user ki demand ki size ka block diya jata h.
* IANA provide that block size that user demands.
* No Classes
* Only Blocks

**Rules**

* Address should be continuous.
* No of addresses in a block must be in power of 2.
* First Address of every block must be evenly divisible with size of block.

IQ – How many IP’s are reserved in a subnet?

A – 5 IP’s are reserved in a subnet for specific purpose.

Ex – 10.0.0.0/24 – Network IP

10.0.0.1/24 – Reserve for Default Gateway

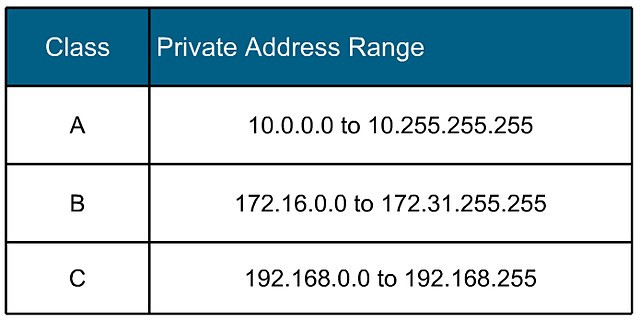
10.0.0.2/24 – Reserved for DNS

10.0.0.3/24 – Reserved for DNS

10.0.0.255/24 – Reserved for Broadcast

**Private IP Address**

* Used for communication within a private network. Cannot be directly contacted over the internet.
* Addresses can be reused per network.
* Assigned to a device by router.
* Has a small set range of possible addresses.
* A private IP address is **a range of non-internet facing IP addresses used in an internal network**.



**Public IP Address**

* Used to communicate over the public internet – outside a private network.
* Addresses are unique and cannot be reused.
* Assigned by an ISP.
* Addresses can be any combination of numbers not within the private IP range.
* Public is mainly for management traffic.
* A public IP address is **an IP address that can be accessed directly over the internet and is assigned to your network router by your internet service provider (ISP).**

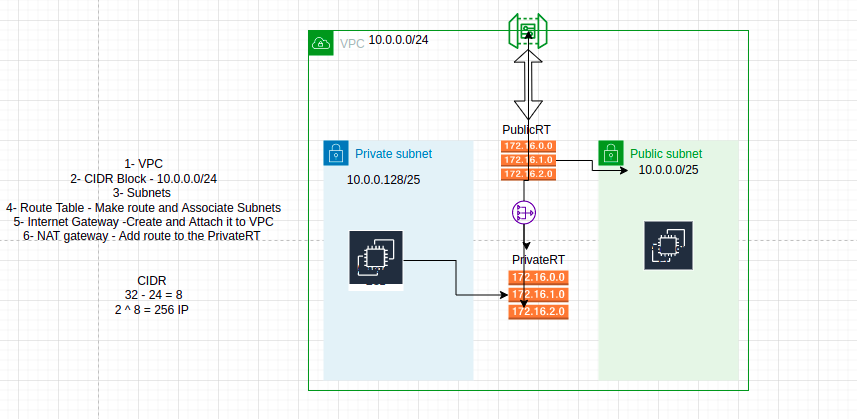
**IPv4**

* 32-bit logical address
* 4 Octet
* 0-255 Octet value
* IP Address = Network ID + Host ID

**Vnet**

Vnet enables you to launch Azure resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of Azure.

Azure Virtual Network (VNet) is the fundamental building block for your private network in Azure. VNet enables many types of Azure resources, such as Azure Virtual Machines (VM), to securely communicate with each other, the internet, and on-premises networks. VNet is similar to a traditional network that you'd operate in your own data center, but brings with it additional benefits of Azure's infrastructure such as scale, availability, and isolation.

****

1. Vnet
2. CIDR Block
3. Subnets (Ex- Private Subnet, Public Sebnet)
4. Rout Table
5. Internet Gateway
6. NAT Gateway

**Subnet**

A subnet is a range of IP addresses in the virtual network. You can divide a virtual network into multiple subnets for organization and security. Each NIC in a VM is connected to one subnet in one virtual network.

**Public Subnet**

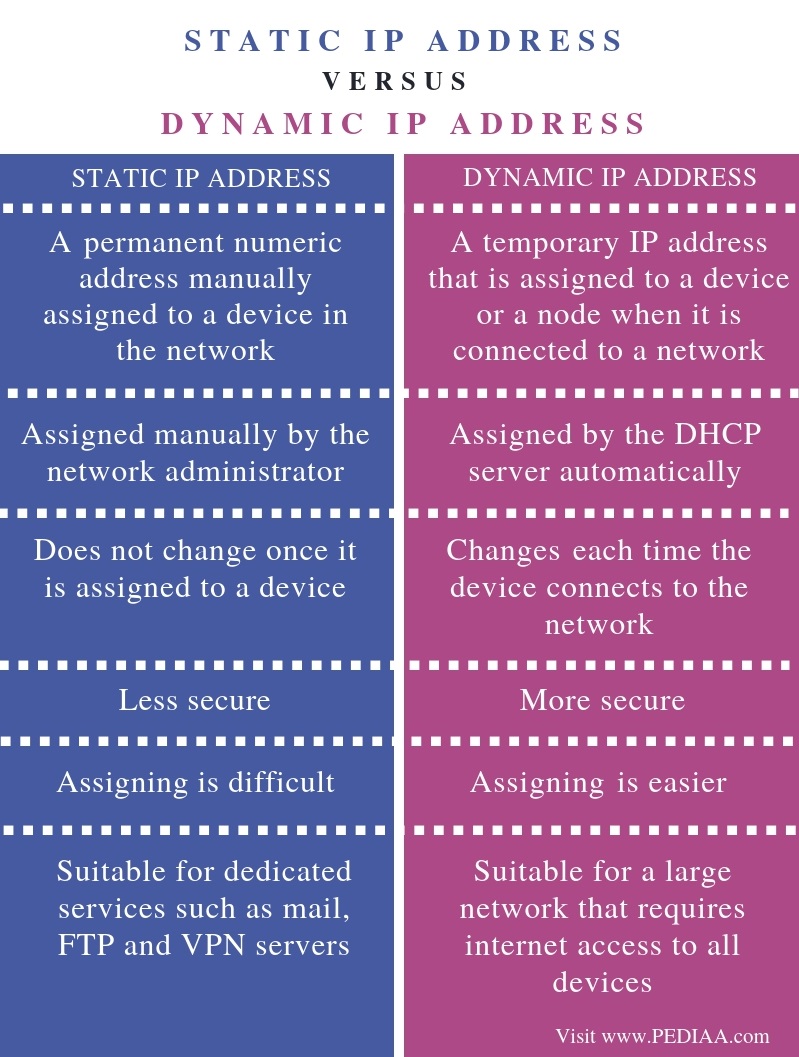
A public subnet is **a subnet that's associated with a route table that has a route to an internet gateway.**

**Private Subnet**

**If a subnet is associated with a route table that does not have a route to an internet gateway**, it's known as a private subnet.

**Static and Dynamic IP addressing**

When a device is assigned a static IP address, the address does not change. Most devices use dynamic IP addresses, which are assigned by the network when they connect and change over time.



**Attaching multiple network interface to single Virtual Machine**

****

**Note –** If you want to attach one or more NIC to the VM than you have to make a NIC first and then stop your VM because a new NIC will not connect to a VM in a running state. So that you have to stop and then attach a new NIC to VM.

**Note** – We can only attach NSG to the VM, NIC, or SUBNET, not to the VNet.

**Network Security Group & Rules** (Inbound Rules & Outbound Rules)

A network security group **contains security rules** (like an inbound rule, or outbound rule) **that allow or deny inbound network traffic to, or outbound network traffic from, several types of Azure resources.**

**Priority of Security Rules**

A number between 100 and 4096. Rules are processed in priority order, with lower numbers processed before higher numbers, because lower numbers have higher priority. Once traffic matches a rule, processing stops. As a result, any rules that exist with lower priorities (higher numbers) that have the same attributes as rules with higher priorities aren't processed.

Protocol = TCP, UDP, ICMP, ESP, AH, or Any.

**Azure Application Gateway Service / LB (Load Balancer) / Web Traffic Load Balancer**

* This service is a web traffic load balancer that is used to distribute traffic to web applications.
* The web applications can reside on Virtual Machines, Virtual Machines Scale sets, or even on on-premises servers.
* The application gateway is on OSI Layer 7 load balancer.
* Secure Sockets Layer (SSL/TLS) termination.
* Here requests to the application gateway can be secure.
* And then the requests to the backend pool resource can go unencrypted.
* This can lift the burden of the backend pool for decrypting requests.
* The decryption of requests can be left to the Application gateway resource.
* You can also enable Autoscaling for your Application Gateway resource.
* This allows the Application Gateway to scale up or down based on traffic load patterns.
* You can also enable the web application firewall feature for the application gateway resource.
* Frontend components of the Application Gateway.
* Frontend IP address – Users will hit the Application Gateway via the frontend IP address.
* Listener – This is a logical entity that checks for incoming connection requests. There can be multiple listeners attached to an application gateway.

There are two types of Listeners Configurations

1. Basic – Here the listener listens to a single domain site.
2. Multi-site – Here the Listeners map to the multiple domain sites.

Different components of the application gateway.

**Routing Rules –** This is used to route the traffic from the listener to the backend pool.

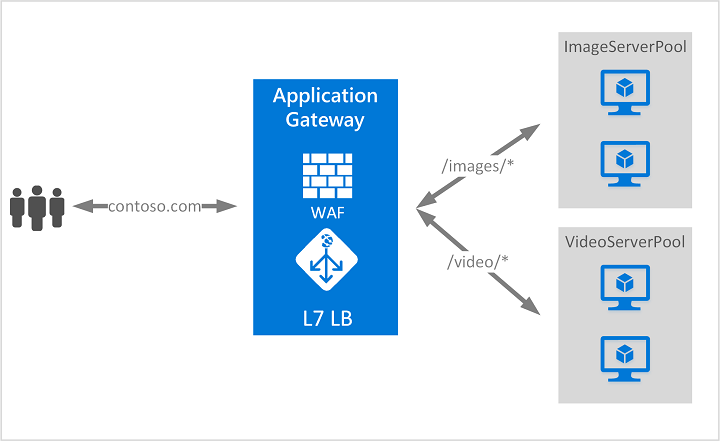
There are 2 types of routing rules.

1. **Basic -** Here all requests are routed to the backend pool directly.
2. **Path-Based –** Here requests are routed to the backend pool based on the URL in the request.

**Backend Pools –** These can be network interface cards, virtual machine scale sets, public or internal IP addresses, FQDN, or backends such as app service.

Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications. Traditional load balancers operate at the transport layer (OSI layer 4 - TCP and UDP) and route traffic based on the source IP address and port, to a destination IP address and port.

Application Gateway can make routing decisions based on additional attributes of an HTTP request, for example, URI path or host headers. For example, you can route traffic based on the incoming URL. So, if /images are in the incoming URL, you can route traffic to a specific set of servers (known as a pool) configured for images. If /video is in the URL, that traffic is routed to another pool that's optimized for videos.



**Vnet Peering**

* Virtual Network Peering is used to connect two Azure virtual networks together via the backbone network.
* Azure supports connecting two virtual networks located in the same region or networks located across regions.
* Once you enable virtual network peering between two virtual networks, the virtual machines can then communicate via their private IP addresses across the peering connection.
* You can also peer virtual networks that are located across different subscriptions.
* The virtual networks can't have overlapping CIDR blocks.

Virtual network peering enables you to seamlessly connect two or more [Virtual Networks](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview) in Azure. The virtual networks appear as one for connectivity purposes. The traffic between virtual machines in peered virtual networks uses the Microsoft backbone infrastructure. Like traffic between virtual machines in the same network, traffic is routed through Microsoft's private network only.

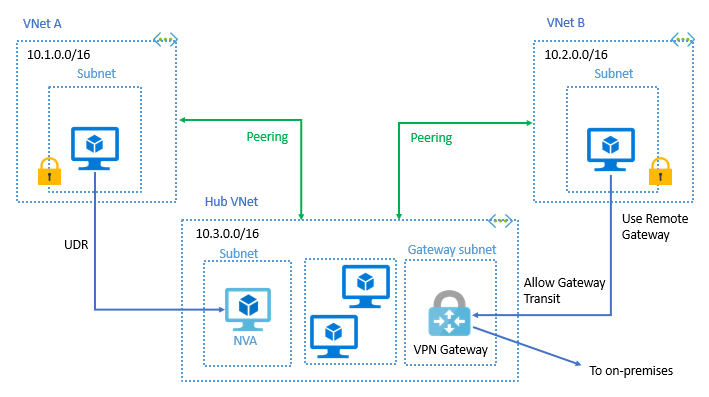
Azure supports the following 2 types of peering:

1. **Virtual network peering**: Connecting virtual networks within the same Azure region.

1. **Global virtual network peering**: Connecting virtual networks across Azure regions.

**Benefits of Vnet Peering whether local or global, include:**

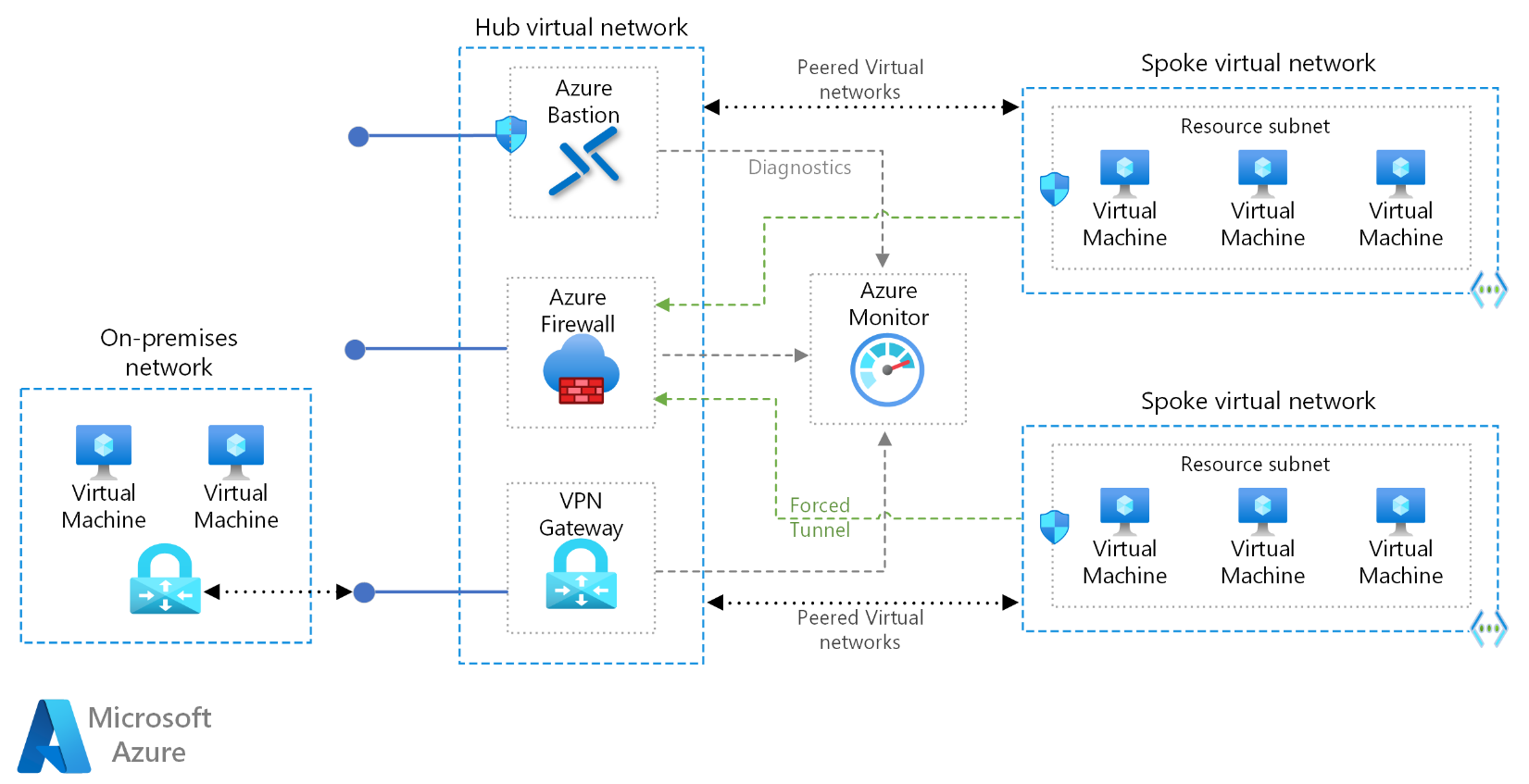
* Network traffic between peered virtual networks is private.
* Low-latency, high-bandwidth connection between resources in different virtual networks.
* Ability for resources in one virtual network to communicate with resources in a different virtual network.
* Ability to transfer data between virtual networks across Azure subscriptions.
* No downtime to resources in either virtual network when creating the peering, or after the peering is created.
* No public Internet, gateways, or encryption is required in the communication between the virtual networks.
* Peerings are not transitive.



**Hub and Spoke**

This reference architecture details a hub-spoke topology in Azure. The hub virtual network acts as a central point of connectivity to many spoke virtual networks. The hub can also be used as the connectivity point to your on-premises networks. The spoke virtual networks peer with the hub and can be used to isolate workloads.

## **Architecture**



### **Workflow**

The architecture consists of the following aspects:

**Hub virtual network:** The hub virtual network is the central point of connectivity to your on-premises network. It's a place to host services that can be consumed by the different workloads hosted in the spoke virtual networks.

**Spoke virtual networks:** Spoke virtual networks are used to isolate workloads in their own virtual networks, managed separately from other spokes. Each workload might include multiple tiers, with multiple subnets connected through Azure load balancers.

**Virtual network peering:** Two virtual networks can be connected using a [peering connection](https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-peering-overview). Peering connections are non-transitive, low latency connections between virtual networks. Once peered, the virtual networks exchange traffic by using the Azure backbone without the need for a router.

**Bastion Host:** Azure Bastion lets you securely connect to a virtual machine using your browser and the Azure portal. An Azure Bastion host is deployed inside an Azure Virtual Network and can access virtual machines in the virtual network (VNet), or virtual machines in peered VNets.

**Azure Firewall:** Azure Firewall is a managed firewall as a service. The Firewall instance is placed in its own subnet.

**VPN virtual network gateway or ExpressRoute gateway**. The virtual network gateway enables the virtual network to connect to the VPN device, or ExpressRoute circuit, used for connectivity with your on-premises network. For more information, see [Connect an on-premises network to a Microsoft Azure virtual network](https://docs.microsoft.com/en-us/microsoft-365/enterprise/connect-an-on-premises-network-to-a-microsoft-azure-virtual-network?view=o365-worldwide).

**VPN device**. A device or service that provides external connectivity to the on-premises network. The VPN device may be a hardware device or a software solution such as the Routing and Remote Access Service (RRAS) in Windows Server 2012.

**VPN**

* A VPN **provides an encrypted server and hides your IP address from corporations, government agencies and would-be hackers**.
* A VPN protects your identity even if you are using public or shared Wi-Fi, and your data will be kept private from any prying internet eyes.
* A VPN circumvents your personal ISP, instead of sending your internet connection to a hosted server. With servers located all over the world, users then have the ability to “relocate” themselves and access the internet from nearly anywhere. Encryption adds an extra layer of security, particularly for businesses that are frequently utilizing remote access. It can also be a helpful tool for travel, gaming, and streaming.

**3 Types of VPN Connections**

1. [Point-to-Site VPN connections](https://docs.microsoft.com/en-us/azure/vpn-gateway/design#P2S)
2. [Site-to-Site VPN connections](https://docs.microsoft.com/en-us/azure/vpn-gateway/design#s2smulti)
3. [VNet-to-VNet VPN connections](https://docs.microsoft.com/en-us/azure/vpn-gateway/design#V2V)
4. **Point-to-Site VPN connection**

A Point-to-Site VPN connection is used to establish a secure connection between multiple client machines and an Azure virtual network via the Internet.

Below is a diagram from the Microsoft documentation on a sample scenario

Diagram

Description automatically generated

* This sort of connection is based off certificates for authentication.
* You need to have a root certificate in place that needs to be uploaded to Azure for the point-to-site connection.
* A client certificate needs to be generated from the root certificate. This client certificate needs to be on each client computer that needs to connect to the Azure virtual network via the Point-to-Site connection.
* To generate the certificates, you can use a Certificate authority or generate a self-signed certificate using PowerShell. Some commands are given below

**// To generate the root certificate**

$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature `

-Subject "CN=RootCertificate" -KeyExportPolicy Exportable `

-HashAlgorithm sha256 -KeyLength 2048 `

-CertStoreLocation "Cert:\CurrentUser\My" -KeyUsageProperty Sign -KeyUsage CertSign

**// To generate the client certificate**

New-SelfSignedCertificate -Type Custom -DnsName P2SChildCert -KeySpec Signature `

-Subject "CN=ClientCertificate" -KeyExportPolicy Exportable `

-HashAlgorithm sha256 -KeyLength 2048 `

-CertStoreLocation "Cert:\CurrentUser\My" `

-Signer $cert -TextExtension @("2.5.29.37={text}1.3.6.1.5.5.7.3.2")

**Gateway Subnet:** The gateway subnet is part of the virtual network IP address range that you specify when configuring your virtual network. It contains the IP addresses that the virtual network gateway resources and services use.

**Virtual Network Gateway:** A VPN gateway is a specific type of virtual network gateway that is used to send encrypted traffic between an Azure virtual network and an on-premises location over the public Internet.

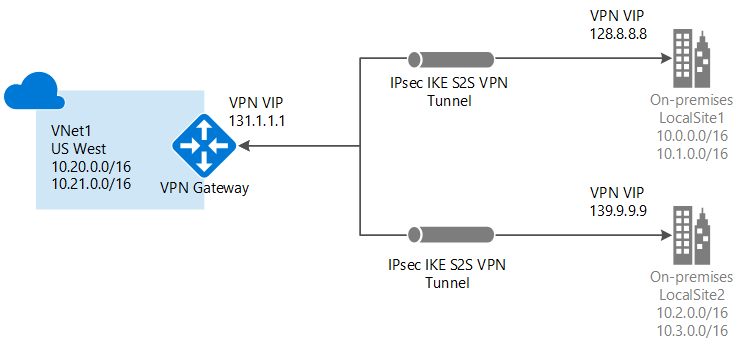
Point-to-site VPN can use one of the following protocols:

* **OpenVPN® Protocol**, an SSL/TLS based VPN protocol. A TLS VPN solution can penetrate firewalls, since most firewalls open TCP port 443 outbound, which TLS uses. OpenVPN can be used to connect from Android, iOS (versions 11.0 and above), Windows, Linux, and Mac devices (macOS versions 10.13 and above).
* **Secure Socket Tunneling Protocol (SSTP)**, a proprietary TLS-based VPN protocol. A TLS VPN solution can penetrate firewalls, since most firewalls open TCP port 443 outbound, which TLS uses. SSTP is only supported on Windows devices. Azure supports all versions of Windows that have SSTP and support TLS 1.2 (Windows 8.1 and later).
* **IKEv2 VPN**, a standards-based IPsec VPN solution. IKEv2 VPN can be used to connect from Mac devices (macOS versions 10.11 and above).

Note - IKEv2 and OpenVPN for P2S are available for the [**Resource Manager deployment model**](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/deployment-models) only. They are not available for the classic deployment model.

1. **Site-to-Site VPN connection**

A Site-to-Site (S2S) VPN gateway connection is a connection over IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. S2S connections can be used for cross-premises and hybrid configurations. A S2S connection requires a VPN device located on-premises that has a public IP address assigned to it.



VPN Gateway can be configured in active-standby mode using one public IP or in active-active mode using two public IPs. In active-standby mode, one IPsec tunnel is active and the other tunnel is in standby. In this setup, traffic flows through the active tunnel, and if some issue happens with this tunnel, the traffic switches over to the standby tunnel. Setting up VPN Gateway in active-active mode is *recommended* in which both the IPsec tunnels are simultaneously active, with data flowing through both tunnels at the same time. An additional advantage of active-active mode is that customers experience higher throughputs.

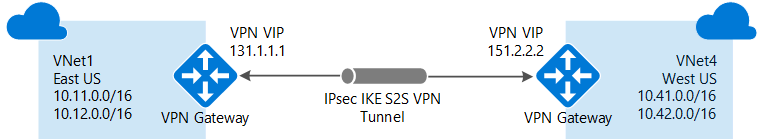
You can create more than one VPN connection from your virtual network gateway, typically connecting to multiple on-premises sites. When working with multiple connections, you must use a RouteBased VPN type (known as a dynamic gateway when working with classic VNets). Because each virtual network can only have one VPN gateway, all connections through the gateway share the available bandwidth. This type of connection is sometimes referred to as a "multi-site" connection.

**3. VNet-to-VNet VPN connections**

Connecting a virtual network to another virtual network (VNet-to-VNet) is similar to connecting a VNet to an on-premises site location. Both connectivity types use a VPN gateway to provide a secure tunnel using IPsec/IKE. You can even combine VNet-to-VNet communication with multi-site connection configurations. This lets you establish network topologies that combine cross-premises connectivity with inter-virtual network connectivity.

The VNets you connect can be:

* in the same or different regions
* in the same or different subscriptions
* in the same or different deployment models



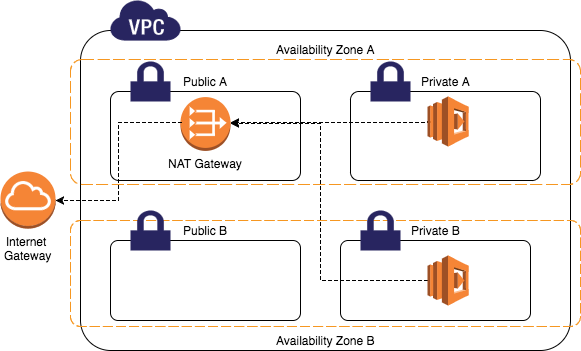
**VPN Gateway**

VPN Gateway sends encrypted traffic between an Azure virtual network and an on-premises location over the public Internet. You can also use VPN Gateway to send encrypted traffic between Azure virtual networks over the Microsoft network. A VPN gateway is a specific type of virtual network gateway. Each virtual network can have only one VPN gateway. However, you can create multiple connections to the same VPN gateway. When you create multiple connections to the same VPN gateway, all VPN tunnels share the available gateway bandwidth.

**NAT Gateway**

For private subnet to talk external world.

A NAT gateway is **a Network Address Translation (NAT) service**. You can use a NAT gateway so that instances in a private subnet can connect to services outside your VPC but external services cannot initiate a connection with those instances.



**What is NAT and Internet gateway?**

A NAT device forwards traffic from the instances in the private subnet to the internet or other AWS services, and then sends the response back to the instances while Internet Gateway is used to allow resources in your VPC to access internet.

**Route Table**

Azure Route Tables, or User Defined Routing, **allow you to create network routes so that your CloudGen Firewall VM can handle the traffic both between your subnets and to the Internet**.

A **Route Table** is used to direct traffic in/out of a subnet. It contains a number of CIDRs (IP address ranges) and where to direct the appropriate traffic.

How traffic will route.

**Express Route**

* The ExpressRoute connection does not go over the puclic internet.
* Your connections are more reliable, faster and you get less latency.
* You get two connections for each Express route circuit for redundancy.
* ExpressRoute is **a service that enables you to create private connections between Azure datacenters and infrastructure that's on your premises or in a colocation environment**.

Diagram

Description automatically generated

**Azure WAN**

Azure Virtual WAN is a networking service that brings many networking, security, and routing functionalities together to provide a single operational interface. Some of the main features include:

* Branch connectivity (via connectivity automation from Virtual WAN Partner devices such as SD-WAN or VPN CPE).
* Site-to-site VPN connectivity.
* Remote user VPN connectivity (point-to-site).
* Private connectivity (ExpressRoute).
* Intra-cloud connectivity (transitive connectivity for virtual networks).
* VPN ExpressRoute inter-connectivity.
* Routing, Azure Firewall, and encryption for private connectivity.

You don't have to have all of these use cases to start using Virtual WAN. You can get started with just one use case, and then adjust your network as it evolves.

The Virtual WAN architecture is a hub and spoke architecture with scale and performance built in for branches (VPN/SD-WAN devices), users (Azure VPN/OpenVPN/IKEv2 clients), ExpressRoute circuits, and virtual networks. It enables a [global transit network architecture](https://docs.microsoft.com/en-us/azure/virtual-wan/virtual-wan-global-transit-network-architecture), where the cloud hosted network 'hub' enables transitive connectivity between endpoints that may be distributed across different types of 'spokes'.

**Network Watcher**

Azure Network Watcher provides tools to monitor, diagnose, view metrics, and enable or disable logs for resources in an Azure virtual network. Network Watcher is designed to monitor and repair the network health of IaaS (Infrastructure-as-a-Service) products which includes Virtual Machines, Virtual Networks, Application Gateways, Load balancers, etc.

**Connection Monitor**

Check the network connectivity between machines. These can be in Azure or on your on-premises environments.

**Next Hop**

Here you can see the next route for a packet of data. This helps you understand whether the packet is being routed to the correct destination.

**IP Flow Verify**

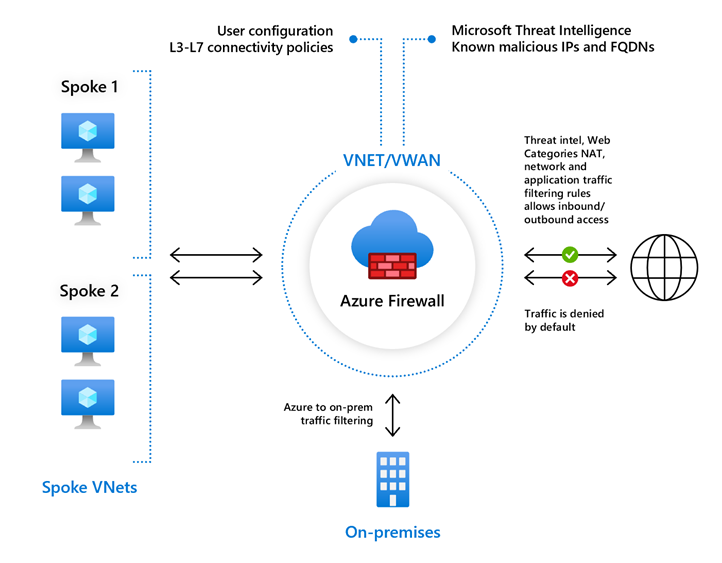
This can be used to check if a packet is allowed or denied to or from a virtual machine. If a packet is being denied by a security group, you can see which rule is denying the packet.

**Connection Troubleshoot**

Check the connection from a virtual machine to a virtual machine, fully qualified domain name, URI or IPv4 address.

**Azure Firewall**

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



**DNS**

Azure DNS allows you to host your DNS domain in Azure, so you can manage your DNS records using the same credentials, billing and support contract as your other Azure services. Zones can be either public or private, where Private DNS Zones (in Managed Preview) are only visible to VMs that are in your virtual network. Our global network of name servers uses Anycast routing to provide outstanding performance and availability.

**COMPUTE**

**Virtualization**

Virtualization is **the fundamental technology that powers cloud computing**. Virtualization is software that manipulates hardware, while cloud computing refers to a service that results from that manipulation. You can't have cloud computing without virtualization.

**Azure Storage Account**

An Azure storage account contains all of your Azure Storage data objects, including blobs, file shares, queues, tables, and disks. The 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 your storage account is durable and highly available, secure, and massively scalable.

**Cloud Computing**

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. Large clouds often have functions distributed over multiple locations, each location being a data center.

## **Difference between Cloud Computing and Virtualization**

| **Cloud Computing** | **Virtualization** |
| --- | --- |
| Cloud computing refers to the distribution of computer services such as servers, storage, networking, and intelligence via the Internet, or the cloud. | Virtualization is a technology that lets you use a single physical hardware system to create many virtual environments or dedicated resources. |
| Deliver a range of resources to users in groups for a variety of objectives. | Packaged resources are delivered to specified consumers for a specific purpose. |
| Scalable to a large extent | Low-scalability |
| Cloud computing entails the use of several machines in the event of a disaster recovery | It is reliant on a single peripheral. |
| Cloud computing has a greater total cost than virtualization. | Virtualization has a lower total cost than virtualization. |
| Many pieces of specialized hardware are required. | While a single dedicated piece of hardware can perform admirably. |

**Data Disk**

A data disk is **a managed disk that's attached to a virtual machine to store application data, or other data you need to keep**. Data disks are registered as SCSI drives and are labeled with a letter that you choose. Each data disk has a maximum capacity of 32,767 gibibytes (GiB).

**Managed Disks** = are managed by Microsoft Azure and you don't need any storage account while created new disk. Since the storage account is managed by Azure you do not have full control of the disks that are being created.

**Un-managed Disks** = is something which requires you to create a storage account before you create any new disk. Since, the storage account is created and owned by you, you have full control over all the data that is present on your storage account. Additionally, you also need to take care of encryption, data recovery plans etc.

**IOPS**

* IOPS measures the number of read and write operations per second
* Input/output operations per second is an input/output performance measurement used to characterize computer storage devices like hard disk drives, solid state drives, and storage area networks.

**Throughput**

* Throughput measures the number of bits read or written per second.

**Data Disk Snapshot**

A snapshot is **a full, read-only copy of a virtual hard disk (VHD)**. You can use a snapshot as a point-in-time backup, or to help troubleshoot virtual machine (VM) issues. You can take a snapshot of both operating system (OS) or data disk VHDs.

**Shared Disks**

Azure shared disks is a feature for Azure managed disks that allow you to attach a managed disk to multiple virtual machines (VMs) simultaneously. Attaching a managed disk to multiple VMs allows you to either deploy new or migrate existing clustered applications to Azure.

**Custom Script Extensions**

The Custom Script Extension downloads and runs scripts on Azure virtual machines (VMs). This extension is useful for post-deployment configuration, software installation, or any other configuration or management task. You can download scripts from Azure Storage or GitHub or provide them to the Azure portal at extension runtime.

Example for custom script. Save below script in .ps1 extension.

Script =

Install-WindowsFeature -name Web-Server – IncludeManagementTools

Install-WindowsFeature -name Print-Server -IncludeManagementTools

Save these two lines in .ps1 extension.

**Advantages**

* No local or domain credential needed to login to Azure VM.
* VM does not need to have a Public IP address to be able to remotely connect to VM, like Powershell remoting needs.
* Simple to implement, not many prerequisites needed.

**Disadvantages**

* The custom script extension needs to be enabled for each VM you want to run your (PowerShell) script on.
* The VM needs to have internet access to access the script location Azure storage or GitHub.
* Relatively slow.

The Custom Script Extension integrates with Azure Resource Manager templates. You can also run it by using the Azure CLI, PowerShell, the Azure portal, or the Azure Virtual Machines REST API.

* This tool can be used on Azure virtual machines to download and execute scripts.
* This is ideal when you want to deploy any custom configuration of any software installation on a VM.
* The scripts can be located in an Azure storage account or even in Github.
* A time duration of 90 minutes is allowed for the script to run. Any longer and the result will be a failed extension provision.
* It is ideal not to place reboots inside the script because the extension will not continue after the reboot. Hence if you have other commands that need to run via the extension after the reboot, they won’t run.
* If your script does need a reboot, then maybe you can look at other tools such as desired state configuration, Chef, or Puppet.
* The script will run only once.
* The custom script extension will run under the impersonation of the local system account.

# Azure Serial Console

The Serial Console in the Azure portal provides access to a text-based console for virtual machines (VMs) and virtual machine scale set instances running either Linux or Windows. This serial connection connects to the ttyS0 or COM1 serial port of the VM or virtual machine scale set instance, providing access independent of the network or operating system state. The serial console can only be accessed by using the Azure portal and is allowed only for those users who have an access role of Contributor or higher to the VM or virtual machine scale set.

Serial Console works in the same manner for VMs and virtual machine scale set instances. In this doc, all mentions to VMs will implicitly include virtual machine scale set instances unless otherwise stated.

Serial Console is generally available in global Azure regions and in public preview in Azure Government. It is not yet available in the Azure China cloud.

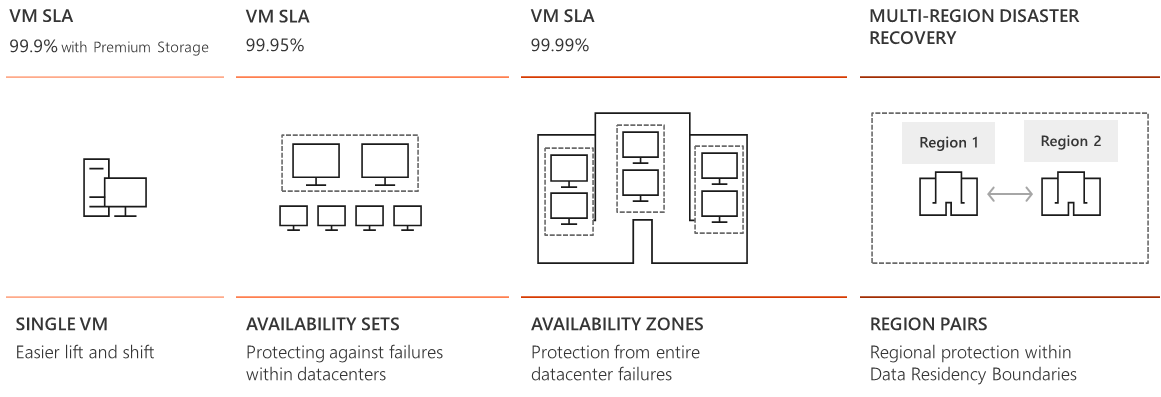
## Prerequisites to access the Azure Serial Console

To access the Serial Console on your VM or virtual machine scale set instance, you will need the following:

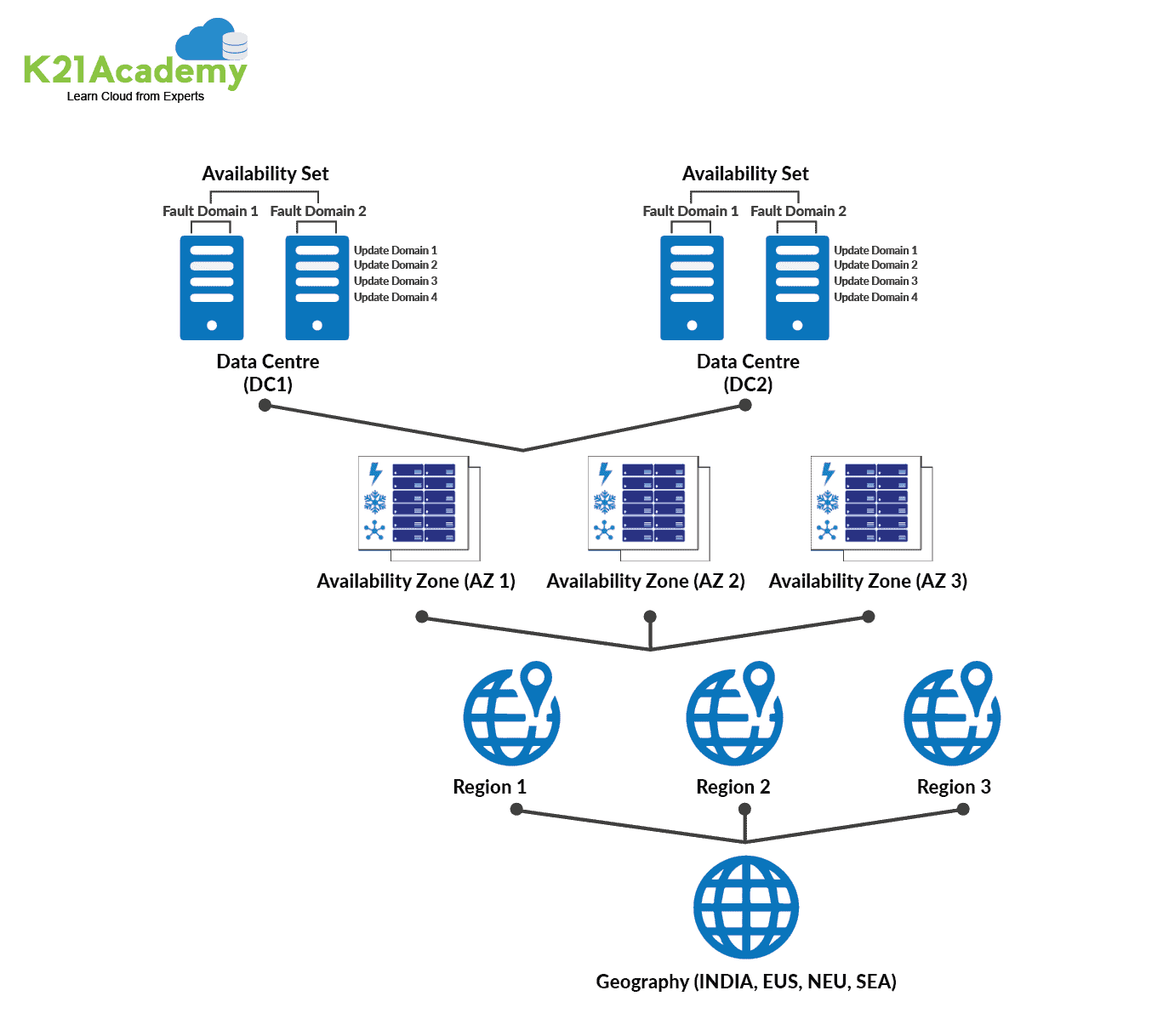
* Boot diagnostics must be enabled for the VM
* A user account that uses password authentication must exist within the VM. You can create a password-based user with the [reset password](https://docs.microsoft.com/en-us/azure/virtual-machines/extensions/vmaccess#reset-password) function of the VM access extension. Select **Reset password** from the **Help** section.
* The Azure account accessing Serial Console must have [Virtual Machine Contributor role](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#virtual-machine-contributor) for both the VM and the [boot diagnostics](https://docs.microsoft.com/en-us/troubleshoot/azure/virtual-machines/boot-diagnostics) storage account
* Classic deployments aren't supported. Your VM or virtual machine scale set instance must use the Azure Resource Manager deployment model.
* Serial Console is not supported when the storage account has firewall enabled.
* Serial Console is not supported when the storage account has **Allow storage account key access** disabled.

**Note -** Serial Console is currently incompatible with a managed boot diagnostics storage account. To use Serial Console, ensure that you are using a custom storage account that is in the same region as your VM and accessible from all networks. You can find the setting in the **Networking** section of the storage account **Overview** page.

**Availability set**



An availability set is a logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability. We recommended that two or more VMs are created within an availability set to provide for a highly available application and to meet the [99.95% Azure SLA](https://azure.microsoft.com/support/legal/sla/virtual-machines/). There is no cost for the Availability Set itself, you only pay for each VM instance that you create.



**Availability Zone**

Azure availability zones are physically separate locations within each Azure region that are tolerant to local failures. Failures can range from software and hardware failures to events such as earthquakes, floods, and fires. Tolerance to failures is achieved because of redundancy and logical isolation of Azure services. To ensure resiliency, a minimum of three separate availability zones are present in all availability zone-enabled regions.

**Placement Groups / Proximity Placement Groups**

Normally when you create multiple virtual machines or virtual machines that are part of a virtual machine scale set, these machines could be located in different data centers.

Sometimes an application/system that uses multiple virtual machines , want the virtual machines to be located closer together to get least latency when it comes to communication between the virtual machines.

By placing the virtual machines as part of a proximity group, the virtual machines will be physically located close to each other.

When using proximity placement groups, ensure the virtual machines have accelerated networking enabled. This also helps to improve network performance.

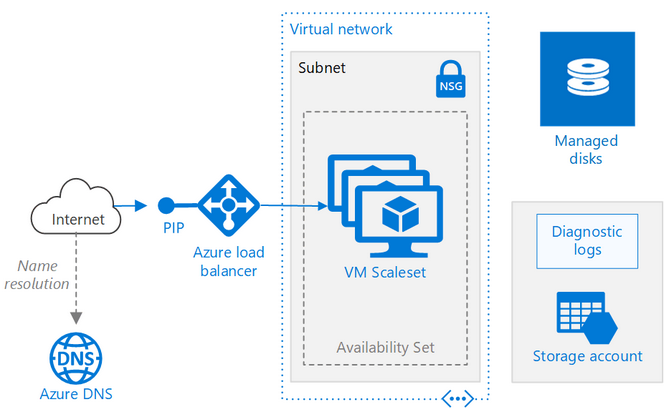
A proximity placement group is assigned to a data center when the first resource (VM) is deployed and released once the last resource is being deleted or stopped.

For example, when you place your Microsoft Azure Virtual Machines in a single Azure region, the physical distance between the virtual machines is reduced. Placing them within a single availability zone is another step you can take to deploy your virtual machines closer to each other. However, as the Azure footprint grows, a single availability zone may span multiple physical data centers resulting in network latency that can impact your overall application performance. If a region does not support availability zones or if your application does not use availability zones, the latency between the application tiers may increase as a result.

IQ – Difference b/w Availability Set and Proximity Placement Group

**Scale Sets**

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



**VM Images**

* This is a copy of the full VM which includes the data disks or just the OS disk.
* You can create an image and place as part of an Azure compute gallery.
* You can share the Azure compute gallery across your organization so that other users can create VM’s based on the images stored in the gallery.
* Image Definition – This is a groupting of image versions. Each image definition has information about why the image was created and other information related to the image.
* Image version – This is used to create the VM.

Two types of images that you can create.

* 1. Specialized VM Images – Here information about specific users and machine information is retained.

So new VM’s created out of the image will have the same computer name and admin user information.

* 1. Generalized VM Images – Here information about specific users and machine information is removed.

Here you have to perform the process of generalization. The original VM is unusable after you perform this process.

**Resizing a Virtual Machine**

After you create a virtual machine (VM), you can scale the VM up or down by changing the VM size. In some cases, you must deallocate the VM first. This can happen if the new size is not available on the hardware cluster that is currently hosting the VM.

If your VM uses Premium Storage, make sure that you choose an **s** version of the size to get Premium Storage support. For example, choose Standard\_E4**s**\_v3 instead of Standard\_E4\_v3.

**App Service Plan (ASP)**

* You don’t have to maintain the underlying compute infrastructure.
* It has feature such as autoscaling and security.
* It has DevOps capabilities which includes cantinuous deployment

In App Service (Web Apps, API Apps, or Mobile Apps), an app always runs in an App Service plan. In addition, [Azure Functions](https://docs.microsoft.com/en-us/azure/azure-functions/dedicated-plan) also has the option of running in an App Service plan. An App Service plan defines a set of compute resources for a web app to run. These compute resources are analogous to the [server farm](https://wikipedia.org/wiki/Server_farm) in conventional web hosting. One or more apps can be configured to run on the same computing resources (or in the same App Service plan).

When you create an App Service plan in a certain region (for example, West Europe), a set of computing resources is created for that plan in that region. Whatever apps you put into this App Service plan run on these compute resources as defined by your App Service plan. Each App Service plan defines:

* Operating System (Windows, Linux)
* Region (West US, East US, etc.)
* Number of VM instances
* Size of VM instances (Small, Medium, Large)
* Pricing tier (Free, Shared, Basic, Standard, Premium, PremiumV2, PremiumV3, Isolated, IsolatedV2)

**App Service Overview**

Azure App Service is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. Applications run and scale with ease on both Windows and [Linux](https://docs.microsoft.com/en-us/azure/app-service/overview#app-service-on-linux)-based environments.

App Service not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and TLS/SSL certificates.

With App Service, you pay for the Azure compute resources you use. The compute resources you use are determined by the App Service plan that you run your apps on.

**Azure Web Apps**

**Azure Web Apps** is a [cloud computing](https://en.wikipedia.org/wiki/Cloud_computing) based platform for hosting websites, created and operated by [Microsoft](https://en.wikipedia.org/wiki/Microsoft). It is a [platform as a service](https://en.wikipedia.org/wiki/Platform_as_a_service) (PaaS) that allows publishing Web apps running on multiple frameworks and written in different programming languages ([.NET](https://en.wikipedia.org/wiki/.NET_Framework), [node.js](https://en.wikipedia.org/wiki/Node.js), [PHP](https://en.wikipedia.org/wiki/PHP), [Python](https://en.wikipedia.org/wiki/Python_(programming_language)), and [Java](https://en.wikipedia.org/wiki/Java_(programming_language))), including Microsoft proprietary ones and 3rd party ones.

**Azure Web App Backups**

The backup feature that is available with Azure Web App can be used to create backups of your web app.

The backups are stored in an Azure storage account.

Here the App configuration, the file content and the database connected to the application get backed up.

To use the Backup and Restore feature, the App Service Plan needs to be in the Standard, Premium or Isolated tier.

Backups of the app + database can be up to a maximum of 10 GB.

**Azure Web App Logging**

You get a set of logging features that are available for the Azure Web App.

The different types of logging that are available are

Application Logging –This captures log messages that are generated by your application code.

Web server logging –This records raw HTTP request data.

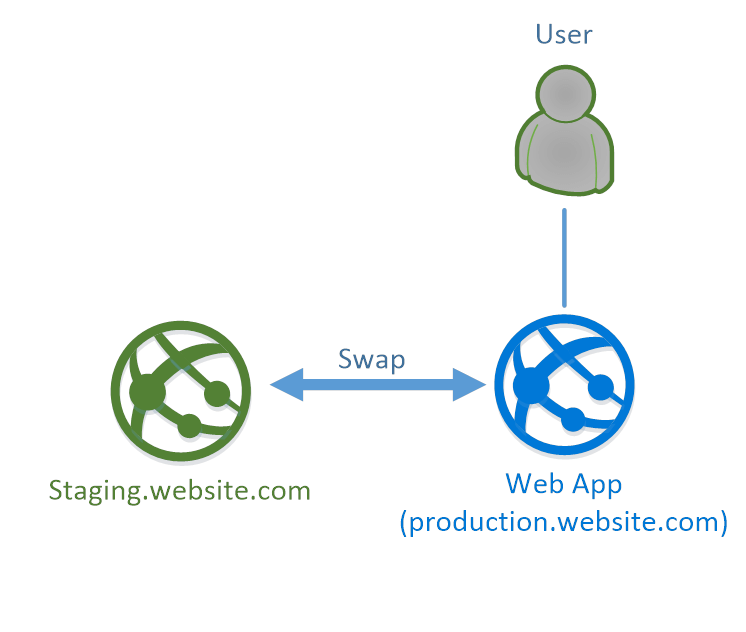
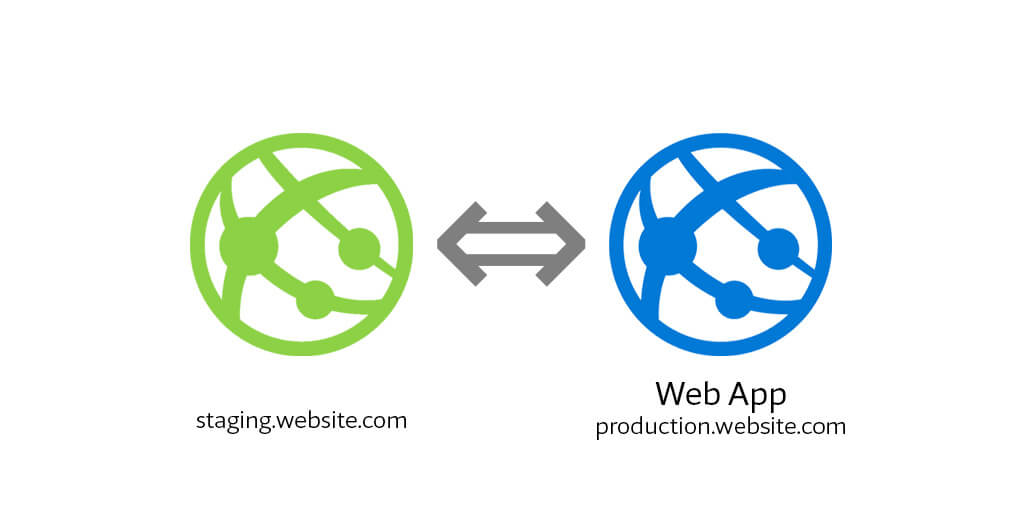
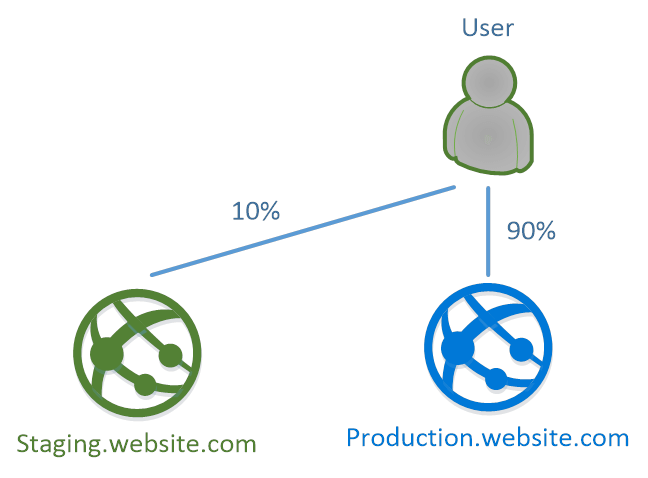
**Deployment Slots**

1. Production slot
2. Staging slot

Staging environment for web apps

* You have the chance to validate all application changes in the staging deployment slot.
* You can then swap the staging slot with the production slot.
* This helps eliminate the downtime for your application when new changes are deployed.
* You can also easily roll back the changes.

Azure Functions deployment slots allow your function app to run different instances called "slots". Slots are different environments exposed via a publicly available endpoint. One app instance is always mapped to the production slot, and you can swap instances assigned to a slot on demand.

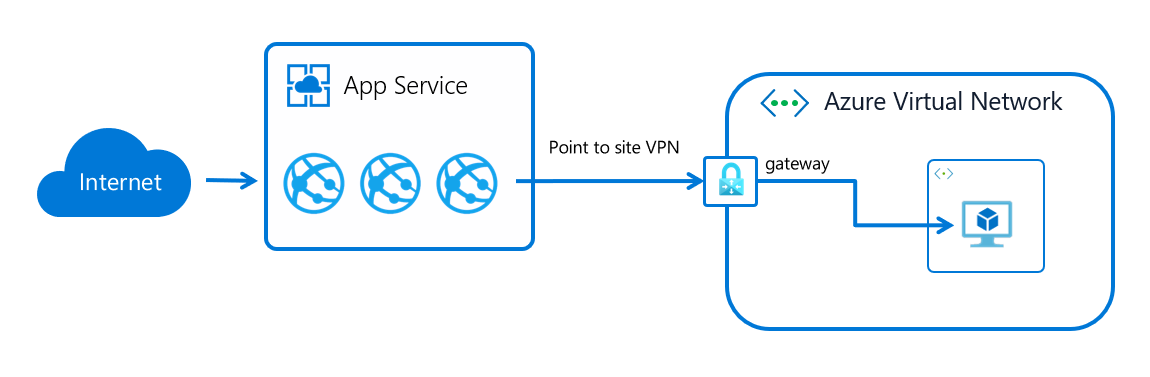


**Azure Web Apps Vnet Integration**

Azure Web App is a public service so the entire idea of the service is to host your web application that can be access by the people of the internet. Now lets say that this web app needs to access a database that is hosted on an azure virtual machine that is in a vnet. And let’s say that this vm does not have a public IP address it only has a private IP address. Nrmally, we don’t give public IP address to our database servers because of security threads. This database server could be in a vnet that has only a private IP address and then your web application needs to contact the database server that is hosted on a vnet but you can’t connect directly with that database server so than you have to enable **Vnet Integration.** This allows the app service to access the resources within the vnet using there private IP addresses. But it does not mean that the resources from your vnet can privately contact the azure webapp. It does’t work the other way arround. It is only to ensure that traffic from your azure web app can privately reach your azure virtual machines within a virtual network.

By integrating with an Azure virtual network (VNet) from your [Azure App Service app](https://docs.microsoft.com/en-us/azure/app-service/overview), you can reach private resources from your app within the virtual network. The VNet integration feature has two variations:

* **Regional virtual network integration**: Connect to Vnet in the same region. You must have a dedicated subnet in the virtual network you're integrating with.
* **Gateway-required virtual network integration**: When you connect directly to a virtual network in other regions or to a classic virtual network in the same region, you must use gateway-required virtual network integration.
* Allow the app service to access resources within the VNET.
* But it does not allow private inbound access to your web App from the Virtual network.



**Custom Domain**

Every new Azure AD tenant comes with an initial domain name, <domainname>.onmicrosoft.com. You can't change or delete the initial domain name, but you can add your organization's names. Adding custom domain names helps you to create user names that are familiar to your users, such as [alain@contoso.com](mailto:alain@contoso.com).

## Before you begin

Before you can add a custom domain name, create your domain name with a domain registrar. For an accredited domain registrar

**SSL Certificate**

An SSL certificate is a digital certificate that authenticates a website's identity and enables an encrypted connection. SSL stands for Secure Sockets Layer, a security protocol that creates an encrypted link between a web server and a web browser.

Companies and organizations need to add SSL certificates to their websites to secure online transactions and keep customer information private and secure.

In short: SSL keeps internet connections secure and prevents criminals from reading or modifying information transferred between two systems. When you see a padlock icon next to the URL in the address bar, that means SSL protects the website you are visiting.

Since its inception about 25 years ago, there have been several versions of SSL protocol, all of which at some point ran into security troubles. A revamped and renamed version followed — TLS (Transport Layer Security), which is still in use today. However, the initials SSL stuck, so the new version of the protocol is still usually called by the old name.

**Docker**

This is an open platform that is used for developing, shipping and running applications.

Docker has the ability to package and run an application in a loosely isolated environment called a container.

**Container**

This is a runnable instance of an image

**Image**

This is a read-only template with instruction that are required to create the docker container.

Create **Azure Container Registry** to store images

**Azure Container Instances**

Containers are becoming the preferred way to package, deploy, and manage cloud applications. Azure Container Instances offers the fastest and simplest way to run a container in Azure, without having to manage any virtual machines and without having to adopt a higher-level service.

Create **Container Instance** service before that you have to make **Container Registry** and enable admin user in container registry. **Steps = container registry> Access Keys> Enable Admin User**

Note – After creating a container you can’t change size of container.

**Azure Container Groups**

A container group is a collection of containers that get scheduled on the same host machine. The containers in a container group share a lifecycle, resources, local network, and storage volumes. It's similar in concept to a pod in [Kubernetes](https://kubernetes.io/docs/concepts/workloads/pods/).

**Lab - Azure Container Groups - Resources**

You can use the following resources as reference for Container Groups

1. You can use the below appdeployment.yml file for the deployment of the container in the container group/instance. Create a file, add the following content and save it as a appdeployment.yml file. Remember to change the Azure Container Registry changes accordingly.

1. apiVersion: 2019-12-01
2. location: northeurope
3. name: AppGroup
4. properties:
5. containers:
6. - name: app
7. properties:
8. image: appregistry10002313.azurecr.io/myapp:latest
9. resources:
10. requests:
11. cpu: 1
12. memoryInGb: 1.5
13. ports:
14. - port: 80
15. osType: Linux
16. ipAddress:
17. type: Public
18. ports:
19. - protocol: tcp
20. port: 80
21. imageRegistryCredentials:
22. - server: appregistry10002313.azurecr.io
23. username: appregistry10002313
24. password: RGhhcWieDFffTCZ2DPYe=QEDqKr4NGbI
25. type: Microsoft.ContainerInstance/containerGroups

2. Then execute the following command in Azure cloud shell to deploy the container group.

az container create --resource-group container-grp --file appdeployment.yml

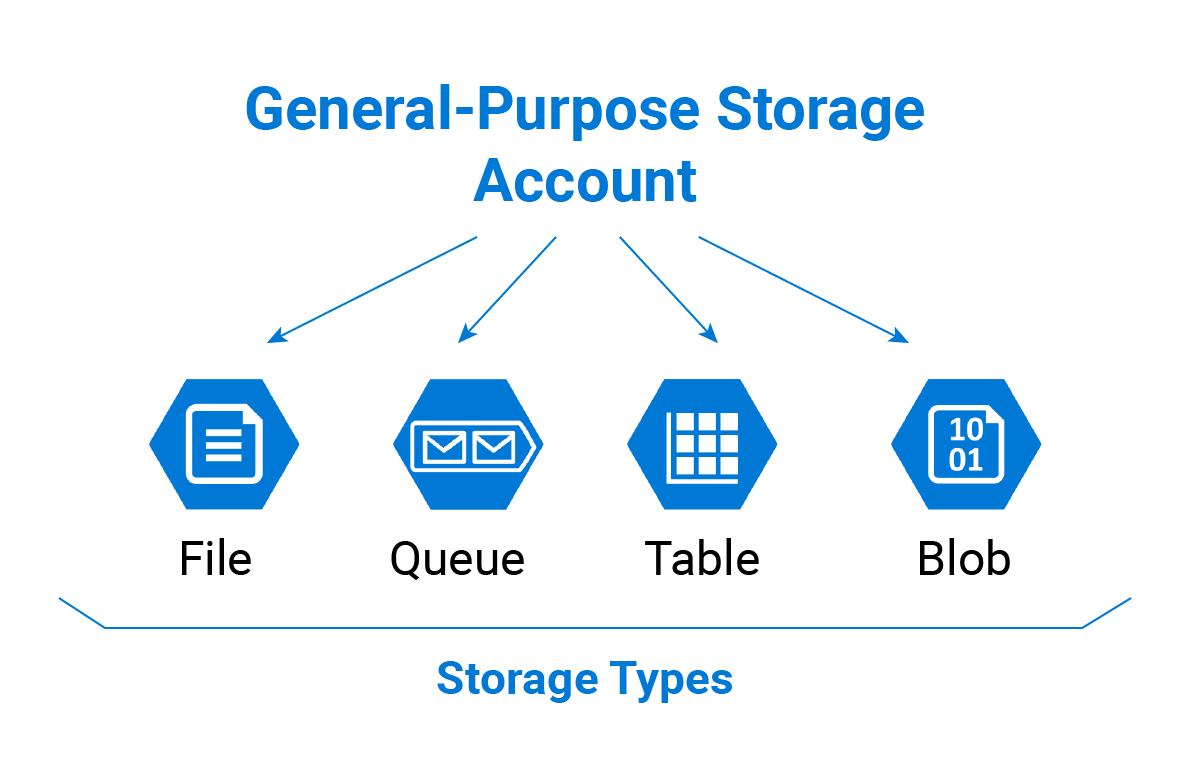
**STORAGE**

**Storage Account Overview**

An Azure storage account contains all of your Azure Storage data objects, including blobs, file shares, queues, tables, and disks. The 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 your storage account is durable and highly available, secure, and massively scalable. (BLOB = Binary Large Object)

**Azure Storage Service**

* 1. **Azure Blob –** For Unstructured Data
  2. **Azure File –** Sharing of files among multiple VM.
  3. **Azure Queue –** Each file is 64kb, Data persist for 7 Days max.
  4. **Azure Table –** Structured No-SQL Data



**Types of Storage Accounts**

* 1. **Standard General Purpose V2: -** This is standard storage for blobs, file shares, queues and tables.
  2. **Premium block blobs: -** This is supported for block and append blobs. This is when you want fast access to your blobs, high transaction rates.
  3. **Premium file shares: -** This is supported for file shares. This is when you want fast access to your files, high transaction rates.
  4. **Premium Page blobs: -** This is supported for page blobs. This is when you want fast access to your blobs, high transaction rates.

**Features of Azure Storage Account**

* **Durable & Highly Available** –Safe in case of transient hardware failure or natural disaster.
* **Secure** – Data is encrypted.
* **Scalable**
* **Managed** – Azure handles hardware maintainence, updates and critical issue for you.
* **Accessible** – Data is accessible from anywhere in the world over HTTP or HTTPS.

**Blob Storage Service**

The word blob is an acronym which stands for binary large object. Blobs typically include large files that are unstructured, such as images, video, music files, backup files etc.

Blob storage can be divided into two access tiers, a hot access tier for data which is accessed frequently and a cold access tier for data which is not accessed very often.

The cold access tier is cheaper than the hot access tier and as such you can store more data at a lower cost, it is also slightly less available, like 99% as opposed to the 99.9% of the hot storage tier.

**Azure storage support three types of Blobs**

1. **Block Blobs –**

* As suggested by the name, block blobs are made of blocks. Each block has a block ID. Blocks can be different sizes, potentially up to a maximum of 4.75TB. Block clients are set to 32MB size block by default, but this can be configured using the SingleBlobUploadThresholdInBytes property.
* Block blobs are optimized for data streaming, and has some features which helps you to manage blobs such as an MD5 hash for verification or parallel uploads.
* Store text and binary data, upto about 47 TB. Block Blobs are made up of block of data that can be managed individually.
* This is made up of blocks of data that can managed individually.

1. **Append Blobs –**

* Append blobs are used to append data. Basically, this means that each time you make changes to the blob, the data will be appended at the end of the blob. Data cannot be changed or deleted, only appended.
* Made up of blocks like block blobs but are optimized for append operation. Append Blobs are ideal for scenarios such as logging data from virtual machines.
* These are block blobs that are optimized for append operations – good for logging.

1. **Page Blobs –**

* A page blob consists out of pages. A page is 512 bytes, and the blob can go up to 1 TB in size.
* This is typically used for fast read and write operations.
* It stores random access files upto 8 TB in size Page Blobs store the virtual hard drive (VHD) file, serve as disk for azure virtual machine.
* Page Blobs are collection of 512-Bytes Page, which provide the ability to read/write arbitrary range of bytes hence, page blobs are ideal for storing index-based and sparse data structure like OS and data disk for VM and Databases.
* This is used for virtual hard drive (VHD) files for azure virtual machines.
* Eg – Azure SQL DB uses page blob.

Blob storage is optimized for storing massive amounts of unstructured data, Such as text or Binary data.

**Blob Storage Used for**

* Serving images or documents directly to a browser.
* Storing files for distributed access.
* Streaming video and audio.
* Storing data for backup and restore discster revcovery and archiving.
* Storing data for analysis by on On-premises or Azure hosted service.

Object in blob storage can be accessed from anywhere in the world via HTTP or HTTPS.

# Authorize access to data in Azure Storage

Each time you access data in your storage account, your client application makes a request over HTTP/HTTPS to Azure Storage. By default, every resource in Azure Storage is secured, and every request to a secure resource must be authorized. Authorization ensures that the client application has the appropriate permissions to access a particular resource in your storage account.

**Authorization Techniques in Storage Account**

1. Access Keys (Primary Key, Secondary Key)
2. Shared Access Signature (SAS)
3. Azure Active Directory

**1. Access Keys**

When you create a storage account, Azure generates two 512-bit storage account access keys for that account. These keys can be used to authorize access to data in your storage account via Shared Key authorization.

Microsoft recommends that you use Azure Key Vault to manage your access keys, and that you regularly rotate and regenerate your keys. Using Azure Key Vault makes it easy to rotate your keys without interruption to your applications. You can also manually rotate your keys.

## **Protect your access keys**

Your storage account access keys are similar to a root password for your storage account. Always be careful to protect your access keys. Use Azure Key Vault to manage and rotate your keys securely. Avoid distributing access keys to other users, hard-coding them, or saving them anywhere in plain text that is accessible to others. Rotate your keys if you believe they may have been compromised.

**2. Shared Access Signature**

A shared access signature (SAS) provides secure delegated access to resources in your storage account. With a SAS, you have granular control over how a client can access your data. For example:

* What resources the client may access.
* What permissions they have to those resources.
* How long the SAS is valid.

## Types of shared access signatures

Azure Storage supports three types of shared access signatures:

1. User delegation SAS
2. Service SAS
3. Account SAS

### **User delegation SAS**

A user delegation SAS is secured with Azure Active Directory (Azure AD) credentials and also by the permissions specified for the SAS. A user delegation SAS applies to Blob storage only.

For more information about the user delegation SAS, see [Create a user delegation SAS (REST API)](https://docs.microsoft.com/en-us/rest/api/storageservices/create-user-delegation-sas).

### **Service SAS**

A service SAS is secured with the storage account key. A service SAS delegates access to a resource in only one of the Azure Storage services: Blob storage, Queue storage, Table storage, or Azure Files.

For more information about the service SAS, see [Create a service SAS (REST API)](https://docs.microsoft.com/en-us/rest/api/storageservices/create-service-sas).

### **Account SAS**

An account SAS is secured with the storage account key. An account SAS delegates access to resources in one or more of the storage services. All of the operations available via a service or user delegation SAS are also available via an account SAS.

You can also delegate access to the following:

* Service-level operations (For example, the **Get/Set Service Properties** and **Get Service Stats** operations).
* Read, write, and delete operations that aren't permitted with a service SAS.

**3. Azure Active Directory**

Azure Active Directory (Azure AD) is a cloud-based identity and access management service. This service helps your employees access external resources, such as Microsoft 365, the Azure portal, and thousands of other SaaS applications. Azure Active Directory also helps them access internal resources like apps on your corporate intranet network, along with any cloud apps developed for your own organization.

## **Who uses Azure AD?**

* IT Admins
* App Developers
* Microsoft 365, Office 365, Azure,

## **What are the Azure AD licenses?**

* **Azure Active Directory Free.**
* **Azure Active Directory Premium P1**
* **Azure Active Directory Premium P2**
* **"Pay as you go" feature licenses**

**Azure Storage Explorer**

Microsoft Azure Storage Explorer is a standalone app that makes it easy to work with Azure Storage data on Windows, macOS, and Linux.

**Stored Access Policy**

A stored access policy provides an additional level of control over service-level shared access signatures (SAS) on the server side. Establishing a stored access policy serves to group shared access signatures and to provide additional restrictions for signatures that are bound by the policy. You can use a stored access policy to change the start time, expiry time, or permissions for a signature, or to revoke it after it has been issued.

The following storage resources support stored access policies:

* Blob containers
* File shares
* Queues
* Tables

**AD Authentication**

One of the main features of an identity platform is to verify, or *authenticate*, credentials when a user signs in to a device, application, or service. In Azure Active Directory (Azure AD), authentication involves more than just the verification of a username and password. To improve security and reduce the need for help desk assistance, Azure AD authentication includes the following components:

* Self-service password reset
* Azure AD Multi-Factor Authentication
* Hybrid integration to write password changes back to on-premises environment
* Hybrid integration to enforce password protection policies for an on-premises environment
* Passwordless authentication

**Storage Redundancy**

Azure Storage always stores multiple copies of your data so that it's protected from planned and unplanned events, including transient hardware failures, network or power outages, and massive natural disasters. Redundancy ensures that your storage account meets its availability and durability targets even in the face of failures.

When deciding which redundancy option is best for your scenario, consider the tradeoffs between lower costs and higher availability. The factors that help determine which redundancy option you should choose include:

* How your data is replicated in the primary region.
* Whether your data is replicated to a second region that is geographically distant to the primary region, to protect against regional disasters (geo-replication).
* Whether your application requires read access to the replicated data in the secondary region if the primary region becomes unavailable for any reason (geo-replication with read access).

# Access Tiers

Data stored in the cloud grows at an exponential pace. To manage costs for your expanding storage needs, it can be helpful to organize your data based on how frequently it will be accessed and how long it will be retained. Azure storage offers different access tiers so that you can store your blob data in the most cost-effective manner based on how it's being used. Azure Storage access tiers include:

* **Hot tier** - An online tier optimized for storing data that is accessed or modified frequently. The Hot tier has the highest storage costs, but the lowest access costs.
* **Cool tier** - An online tier optimized for storing data that is infrequently accessed or modified. Data in the Cool tier should be stored for a minimum of 30 days. The Cool tier has lower storage costs and higher access costs compared to the Hot tier.
* **Archive tier** - An offline tier optimized for storing data that is rarely accessed, and that has flexible latency requirements, on the order of hours. Data in the Archive tier should be stored for a minimum of 180 days.

Azure storage capacity limits are set at the account level, rather than according to access tier. You can choose to maximize your capacity usage in one tier, or to distribute capacity across two or more tiers.

## **Online access tiers**

When your data is stored in an online access tier (either Hot or Cool), users can access it immediately. The Hot tier is the best choice for data that is in active use, while the Cool tier is ideal for data that is accessed less frequently, but that still must be available for reading and writing.

Example usage scenarios for the Hot tier include:

* Data that's in active use or is expected to be read from and written to frequently.
* Data that's staged for processing and eventual migration to the Cool access tier.

Usage scenarios for the Cool access tier include:

* Short-term data backup and disaster recovery.
* Older data sets that aren't used frequently, but are expected to be available for immediate access.
* Large data sets that need to be stored in a cost-effective way while additional data is being gathered for processing.

To learn how to move a blob to the Hot or Cool tier, see [Set a blob's access tier](https://docs.microsoft.com/en-us/azure/storage/blobs/access-tiers-online-manage).

Data in the Cool tier has slightly lower availability, but offers the same high durability, retrieval latency, and throughput characteristics as the Hot tier. For data in the Cool tier, slightly lower availability and higher access costs may be acceptable trade-offs for lower overall storage costs, as compared to the Hot tier. For more information, see [SLA for storage](https://azure.microsoft.com/support/legal/sla/storage/v1_5/).

A blob in the Cool tier in a general-purpose v2 account is subject to an early deletion penalty if it's deleted or moved to a different tier before 30 days has elapsed. This charge is prorated. For example, if a blob is moved to the Cool tier and then deleted after 21 days, you'll be charged an early deletion fee equivalent to 9 (30 minus 21) days of storing that blob in the Cool tier.

The Hot and Cool tiers support all redundancy configurations. For more information about data redundancy options in Azure Storage, see [Azure Storage redundancy](https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy).

## **Archive access tier**

The Archive tier is an offline tier for storing data that is rarely accessed. The Archive access tier has the lowest storage cost, but higher data retrieval costs and latency compared to the Hot and Cool tiers. Example usage scenarios for the Archive access tier include:

* Long-term backup, secondary backup, and archival datasets
* Original (raw) data that must be preserved, even after it has been processed into final usable form
* Compliance and archival data that needs to be stored for a long time and is hardly ever accessed

To learn how to move a blob to the Archive tier, see [Archive a blob](https://docs.microsoft.com/en-us/azure/storage/blobs/archive-blob).

Data must remain in the Archive tier for at least 180 days or be subject to an early deletion charge. For example, if a blob is moved to the Archive tier and then deleted or moved to the Hot tier after 45 days, you'll be charged an early deletion fee equivalent to 135 (180 minus 45) days of storing that blob in the Archive tier.

While a blob is in the Archive tier, it can't be read or modified. To read or download a blob in the Archive tier, you must first rehydrate it to an online tier, either Hot or Cool. Data in the Archive tier can take up to 15 hours to rehydrate, depending on the priority you specify for the rehydration operation. For more information about blob rehydration, see [Overview of blob rehydration from the Archive tier](https://docs.microsoft.com/en-us/azure/storage/blobs/archive-rehydrate-overview).

**Storage Lifecycle Policy**

Data sets have unique lifecycles. Early in the lifecycle, people access some data often. But the need for access often drops drastically as the data ages. Some data remains idle in the cloud and is rarely accessed once stored. Some data sets expire days or months after creation, while other data sets are actively read and modified throughout their lifetimes. Azure Storage lifecycle management offers a rule-based policy that you can use to transition blob data to the appropriate access tiers or to expire data at the end of the data lifecycle.

With the lifecycle management policy, you can:

* Transition blobs from cool to hot immediately when they're accessed, to optimize for performance.
* Transition current versions of a blob, previous versions of a blob, or blob snapshots to a cooler storage tier if these objects haven't been accessed or modified for a period of time, to optimize for cost. In this scenario, the lifecycle management policy can move objects from hot to cool, from hot to archive, or from cool to archive.
* Delete current versions of a blob, previous versions of a blob, or blob snapshots at the end of their lifecycles.
* Define rules to be run once per day at the storage account level.
* Apply rules to containers or to a subset of blobs, using name prefixes or [blob index tags](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-manage-find-blobs) as filters.

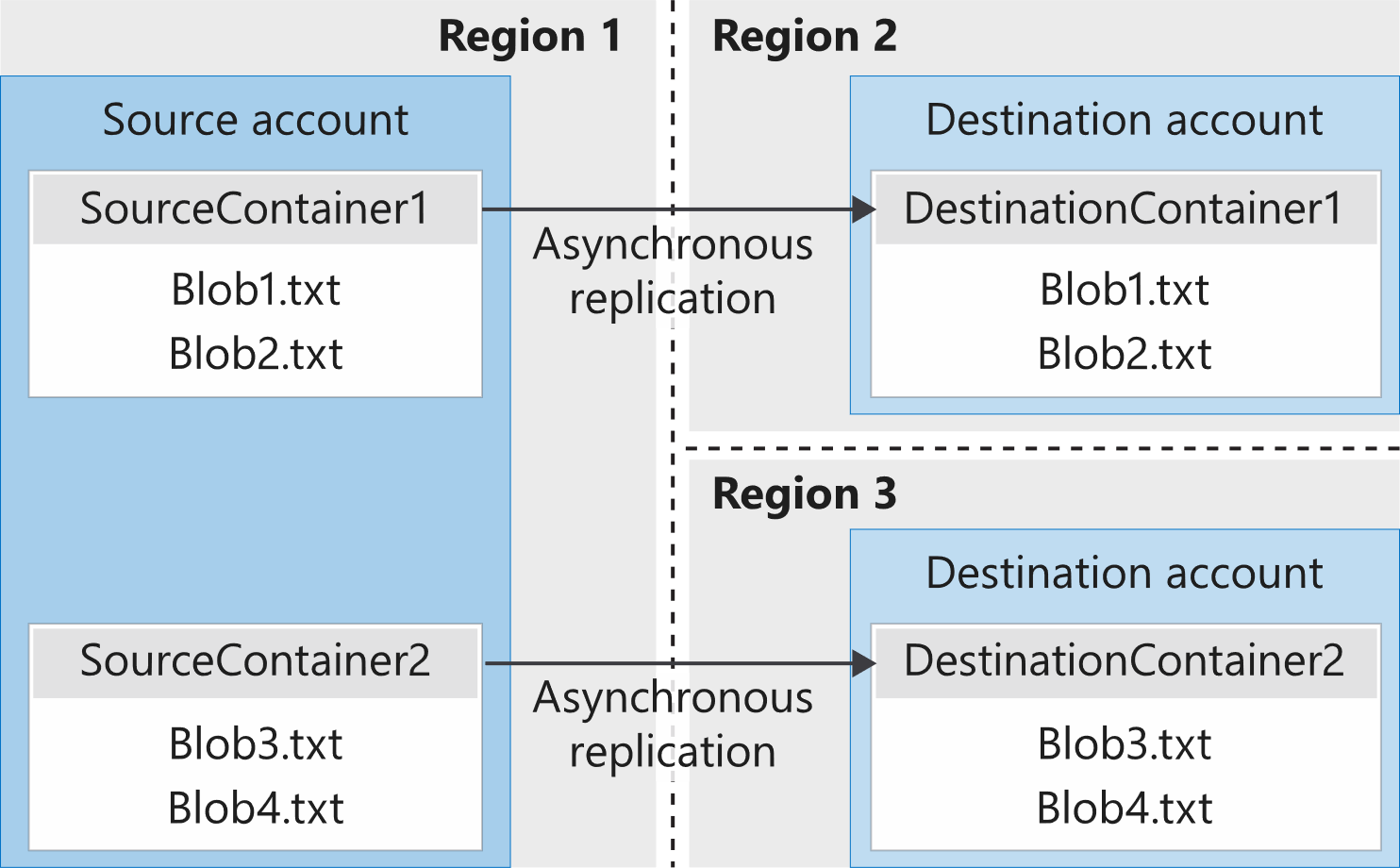
Consider a scenario where data is frequently accessed during the early stages of the lifecycle, but only occasionally after two weeks. Beyond the first month, the data set is rarely accessed. In this scenario, hot storage is best during the early stages. Cool storage is most appropriate for occasional access. Archive storage is the best tier option after the data ages over a month. By moving data to the appropriate storage tier based on its age with lifecycle management policy rules, you can design the least expensive solution for your needs.

**Object Replication**

Object replication asynchronously copies block blobs between a source storage account and a destination account. Some scenarios supported by object replication include:

* **Minimizing latency.** Object replication can reduce latency for read requests by enabling clients to consume data from a region that is in closer physical proximity.
* **Increase efficiency for compute workloads.** With object replication, compute workloads can process the same sets of block blobs in different regions.
* **Optimizing data distribution.** You can process or analyze data in a single location and then replicate just the results to additional regions.
* **Optimizing costs.** After your data has been replicated, you can reduce costs by moving it to the archive tier using life cycle management policies.

The following diagram shows how object replication replicates block blobs from a source storage account in one region to destination accounts in two different regions.



To learn how to configure object replication, see [Configure object replication](https://docs.microsoft.com/en-us/azure/storage/blobs/object-replication-configure).

## **Prerequisites and caveats for object replication**

Object replication requires that the following Azure Storage features are also enabled:

* [Change feed](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-change-feed): Must be enabled on the source account. To learn how to enable change feed, see [Enable and disable the change feed](https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-change-feed#enable-and-disable-the-change-feed).
* [Blob versioning](https://docs.microsoft.com/en-us/azure/storage/blobs/versioning-overview): Must be enabled on both the source and destination accounts. To learn how to enable versioning, see [Enable and manage blob versioning](https://docs.microsoft.com/en-us/azure/storage/blobs/versioning-enable).

**Azure File Share**

Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard [Server Message Block (SMB) protocol](https://docs.microsoft.com/en-us/windows/win32/fileio/microsoft-smb-protocol-and-cifs-protocol-overview) or [Network File System (NFS) protocol](https://en.wikipedia.org/wiki/Network_File_System). Azure Files file shares can be mounted concurrently by cloud or on-premises deployments. SMB Azure file shares are accessible from Windows, Linux, and macOS clients. NFS Azure Files shares are accessible from Linux or macOS clients. Additionally, SMB Azure file shares can be cached on Windows Servers with [Azure File Sync](https://docs.microsoft.com/en-us/azure/storage/file-sync/file-sync-introduction) for fast access near where the data is being used.

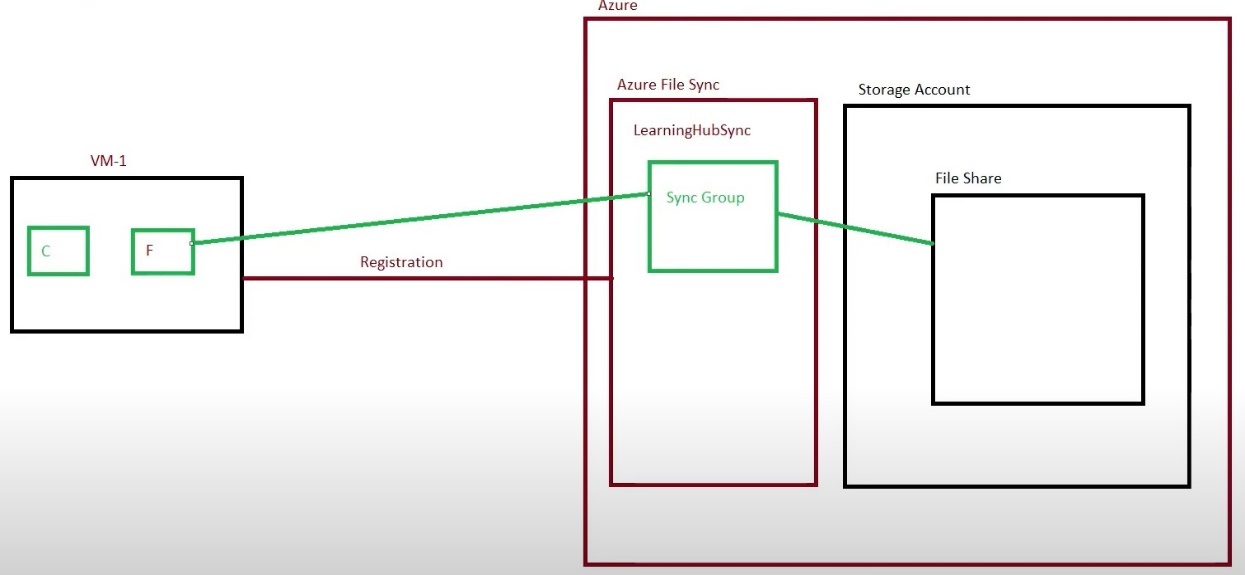
## **Why Azure Files is useful**

* **Replace or supplement on-premises file servers\**
* **"Lift and shift" applications**
* **Simplify cloud development**

## **Key benefits**

* **Shared access**
* **Fully managed**
* **Scripting and tooling**
* **Resiliency**
* **Familiar programmability**

**Azure File Sync**

Azure File Sync enables centralizing your organization's file shares in Azure Files, while keeping the flexibility, performance, and compatibility of a Windows file server. While some users may opt to keep a full copy of their data locally, Azure File Sync additionally has the ability to transform Windows Server into a quick cache of your Azure file share. You can use any protocol that's available on Windows Server to access your data locally, including SMB, NFS, and FTPS. You can have as many caches as you need across the world.

**AzCopy**

AzCopy is a command-line tool that moves data into and out of Azure Storage. See the [Get started with AzCopy](https://docs.microsoft.com/en-us/azure/storage/common/storage-use-azcopy-v10) article to download AzCopy and learn about the ways that you can provide authorization credentials to the storage service.

Copies source data to a destination location.

AzCopy is a command-line utility that you can use to copy blobs or files to or from a storage account.

**IAM**

**Azure 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.

An Azure subscription is **a logical container used to provision resources in Azure**. It holds the details of all your resources like virtual machines (VMs), databases, and more. When you create an Azure resource like a VM, you identify the subscription it belongs to.

**Azure Subscription Types**

* 1. **Free**
  2. **Pay-as-you-go**
  3. **member offers**

**Azure Tenant**

This is a dedicated and trusted instance of Azure AD.

**Azure Active Directory**

Each azure tenant has a dedicated and trusted azure AD directly. This includes the tenant’s users, groups and applications and is used for performing identity and access management onto resources.

Azure Active Directory (Azure AD) is a cloud-based identity and access management service. This service helps your employees access external resources, such as Microsoft 365, the Azure portal, and thousands of other SaaS applications. Azure Active Directory also helps them access internal resources like apps on your corporate intranet network, along with any cloud apps developed for your own organization. For more information about creating a tenant for your organization, see [Quickstart: Create a new tenant in Azure Active Directory](https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-access-create-new-tenant).

To learn the differences between Active Directory and Azure Active Directory, see [Compare Active Directory to Azure Active Directory](https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-compare-azure-ad-to-ad). You can also refer [Microsoft Cloud for Enterprise Architects Series](https://docs.microsoft.com/en-us/microsoft-365/solutions/cloud-architecture-models) posters to better understand the core identity services in Azure like Azure AD and Microsoft-365.

**Azure User**

Users can access assigned resources but cannot manage most tenant resources.

An IAM user interacts with your azure resources from the azure console and the azure CLI. By default, a new IAM user has no access to any azure resource.

Every user who needs access to Azure resources needs an Azure user account. A user account contains all the information needed to authenticate you during the sign-in process. Once authenticated, Azure AD builds an access token to authorize you, and determine what resources you can access, and what you can do with those resources.

**IAM Groups**

Azure AD Security Groups are analogous to Security Groups in on-prem Windows Active Directory. They are Security Principals, which means they can be used to secure objects in Azure AD. They can be created natively in Azure AD, or synced from Windows AD with [Azure AD Connect](https://docs.microsoft.com/en-us/azure/active-directory/hybrid/whatis-azure-ad-connect). Their membership can be static, or it can be [generated dynamically](https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/groups-create-rule) with rules.

An IAM group consists of IAM users and permissions assigned to those users.

# Azure role-based access control (Azure RBAC)

Access management for cloud resources is a critical function for any organization that is using the cloud. Azure role-based access control (Azure RBAC) helps you manage who has access to Azure resources, what they can do with those resources, and what areas they have access to.

Azure RBAC is an authorization system built on [Azure Resource Manager](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/overview) that provides fine-grained access management to Azure resources.

| **Built-in role** | **Description** | **ID** |
| --- | --- | --- |
| **General** |  |  |
| [Contributor](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#contributor) | Grants full access to manage all resources, but does not allow you to assign roles in Azure RBAC, manage assignments in Azure Blueprints, or share image galleries. | b24988ac-6180-42a0-ab88-20f7382dd24c |
| [Owner](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#owner) | Grants full access to manage all resources, including the ability to assign roles in Azure RBAC. | 8e3af657-a8ff-443c-a75c-2fe8c4bcb635 |
| [Reader](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#reader) | View all resources, but does not allow you to make any changes. | acdd72a7-3385-48ef-bd42-f606fba81ae7 |
| [User Access Administrator](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#user-access-administrator) | Lets you manage user access to Azure resources. | 18d7d88d-d35e-4fb5-a5c3-7773c20a72d9 |

## **How Azure RBAC works**

The way you control access to resources using Azure RBAC is to assign Azure roles. This is a key concept to understand – it's how permissions are enforced. A role assignment consists of three elements: **security principal, role definition, and scope.**

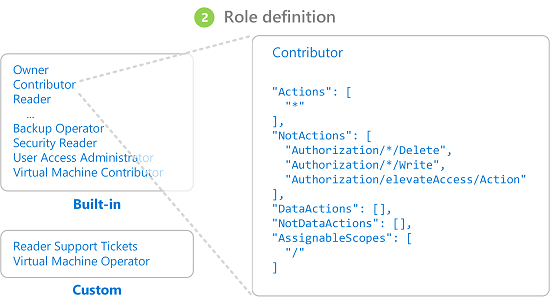
### **Security principal**

A security principal is an object that represents a user, group, service principal, or managed identity that is requesting access to Azure resources. You can assign a role to any of these security principals.



### **Role definition**

A role definition is a collection of permissions. It's typically just called a role. A role definition lists the actions that can be performed, such as read, write, and delete. Roles can be high-level, like owner, or specific, like virtual machine reader.

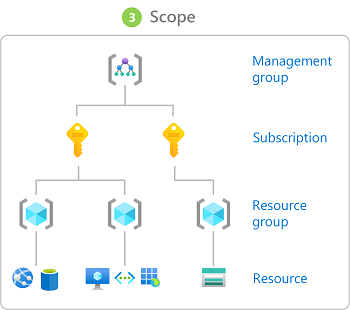


Azure includes several [built-in roles](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles) that you can use. For example, the [Virtual Machine Contributor](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#virtual-machine-contributor) role allows a user to create and manage virtual machines. If the built-in roles don't meet the specific needs of your organization, you can create your own [Azure custom roles](https://docs.microsoft.com/en-us/azure/role-based-access-control/custom-roles).

### **Scope**

Scope is the set of resources that the access applies to. When you assign a role, you can further limit the actions allowed by defining a scope. This is helpful if you want to make someone a [Website Contributor](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles#website-contributor), but only for one resource group.

In Azure, you can specify a scope at four levels: [management group](https://docs.microsoft.com/en-us/azure/governance/management-groups/overview), subscription, [resource group](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/overview#resource-groups), or resource. Scopes are structured in a parent-child relationship. You can assign roles at any of these levels of scope.



### **Role assignments**

A role assignment is the process of attaching a role definition to a user, group, service principal, or managed identity at a particular scope for the purpose of granting access. Access is granted by creating a role assignment, and access is revoked by removing a role assignment.

# Azure custom roles (Resource Group > Access Control IAM > ADD)

If the [Azure built-in roles](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles) don't meet the specific needs of your organization, you can create your own custom roles. Just like built-in roles, you can assign custom roles to users, groups, and service principals at management group (in preview only), subscription, and resource group scopes.

Custom roles can be shared between subscriptions that trust the same Azure AD directory. There is a limit of **5,000** custom roles per directory. (For Azure Germany and Azure China 21Vianet, the limit is 2,000 custom roles.) Custom roles can be created using the Azure portal, Azure PowerShell, Azure CLI, or the REST API.

**Dynamic Groups** (Azure AD > All Groups > New Group > Membership Type > Dynamic User)

Note – For Dynamic Groups you should have “Azure AD Premium P2” License.

A dynamic group is **one whose membership changes based on a defined set of criteria**. Until now dynamic groups were only possible in the Exchange environment, but now they can also be created in the Active Directory setting.

You can create attribute-based rules to enable dynamic membership for a group in Azure Active Directory (Azure AD), part of Microsoft Entra. Dynamic group membership adds and removes group members automatically using membership rules based on member attributes. This article details the properties and syntax to create dynamic membership rules for users or devices. You can set up a rule for dynamic membership on security groups or Microsoft 365 groups.

When the attributes of a user or a device change, the system evaluates all dynamic group rules in a directory to see if the change would trigger any group adds or removes. If a user or device satisfies a rule on a group, they're added as a member of that group. If they no longer satisfy the rule, they're removed. You can't manually add or remove a member of a dynamic group.

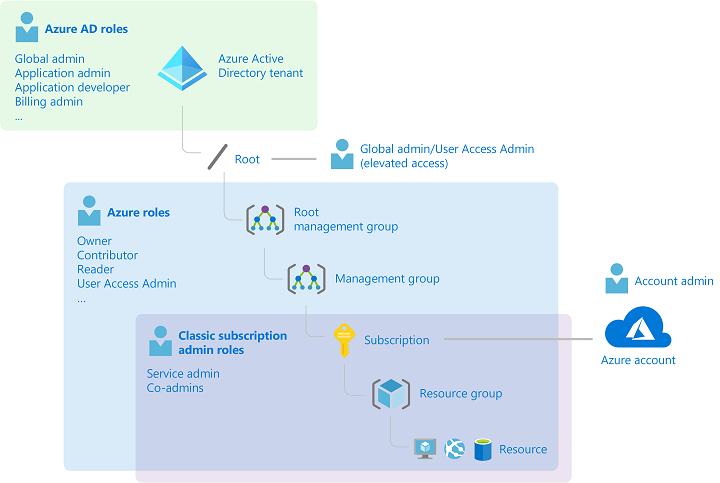
* You can create a dynamic group for devices or for users, but you can't create a rule that contains both users and devices.
* You can't create a device group based on the user attributes of the device owner. Device membership rules can reference only device attributes.

**Dynamic Device Group**

Similar like Dynamic Group but in dynamic group membership changes based on a defined set of criteria whereas in Dynamic Device Group device membership changes based on a defined set of criteria.

**AD Roles**

There are about 60 Azure Active Directory (Azure AD) built-in roles, which are roles with a fixed set of role permissions. To supplement the built-in roles, Azure AD also supports custom roles. Use custom roles to select the role permissions that you want. For example, you could create one to manage particular Azure AD resources such as applications or service principals.



## **Categories of Azure AD roles**

Azure AD built-in roles differ in where they can be used, which fall into the following three broad categories.

* **Azure AD-specific roles**: These roles grant permissions to manage resources within Azure AD only. For example, User Administrator, Application Administrator, Groups Administrator all grant permissions to manage resources that live in Azure AD.
* **Service-specific roles**: For major Microsoft 365 services (non-Azure AD), we have built service-specific roles that grant permissions to manage all features within the service. For example, Exchange Administrator, Intune Administrator, SharePoint Administrator, and Teams Administrator roles can manage features with their respective services. Exchange Administrator can manage mailboxes, Intune Administrator can manage device policies, SharePoint Administrator can manage site collections, Teams Administrator can manage call qualities and so on.
* **Cross-service roles**: There are some roles that span services. We have two global roles - Global Administrator and Global Reader. All Microsoft 365 services honor these two roles. Also, there are some security-related roles like Security Administrator and Security Reader that grant access across multiple security services within Microsoft 365. For example, using Security Administrator roles in Azure AD, you can manage Microsoft 365 Defender portal, Microsoft Defender Advanced Threat Protection, and Microsoft Defender for Cloud Apps. Similarly, in the Compliance Administrator role you can manage Compliance-related settings in Compliance portal, Exchange, and so on.

**Custom Domain**

A domain name is an important part of the identifier for resources in many Azure Active Directory (Azure AD), part of Microsoft Entra: it's part of a user name or email address for a user, part of the address for a group, and is sometimes part of the app ID URI for an application. A resource in Azure AD can include a domain name that's owned by the Azure AD organization (sometimes called a tenant) that contains the resource. Only a Global Administrator can manage domains in Azure AD.

**Self Service Password Reset**

Azure Active Directory (Azure AD) self-service password reset (SSPR) gives users the ability to change or reset their password, with no administrator or help desk involvement. If Azure AD locks a user's account or they forget their password, they can follow prompts to unblock themselves and get back to work. This ability reduces help desk calls and loss of productivity when a user can't sign in to their device or an application.

This feature helps users to reset their password withojut the need of contacting the IT help desk staff.

**License** – Password reset needs azure AD Premium P1 or P2 license for users.

**Password Writeback –** If there is a hybrid environment, the changed password can be written back to the on-premises Active Directory.

**Authentication Methods –** You can define authentication methods to reset the password.

**Number of methods –** Define the number of authentication methods required to reset the password.

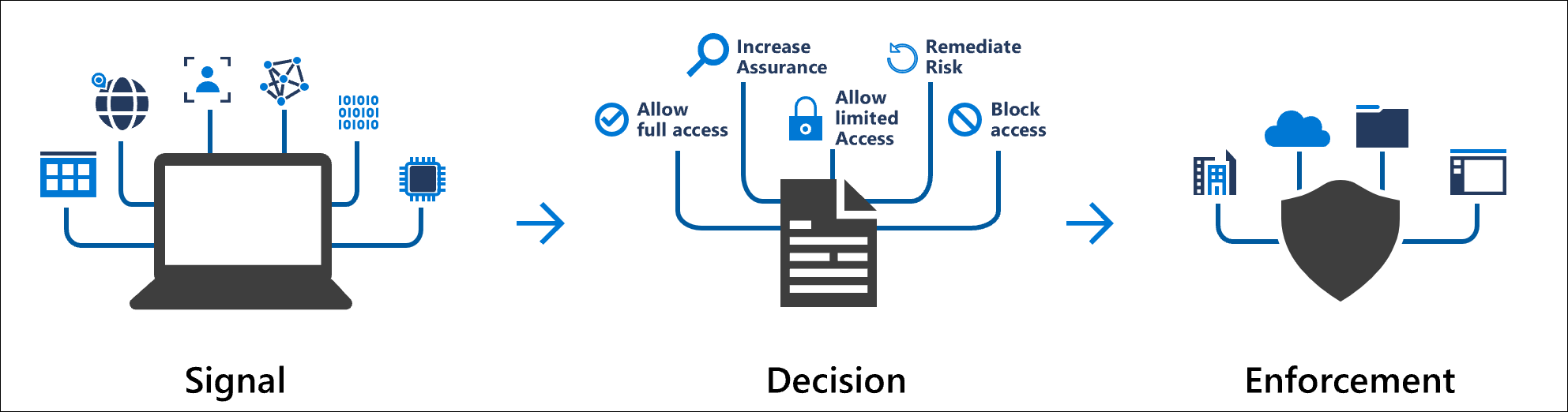
**Number of days –** Number of days before users need to reconfirm their authentication information.

**Notification –** Notify users when password is reset.

**Conditional Access Policy** (AD > Security > Conditional Access > Create new Conditional Access Policy)

The modern security perimeter now extends beyond an organization's network to include user and device identity. Organizations can use identity-driven signals as part of their access control decisions.

Conditional Access brings signals together, to make decisions, and enforce organizational policies. Azure AD Conditional Access is at the heart of the new identity-driven control plane.

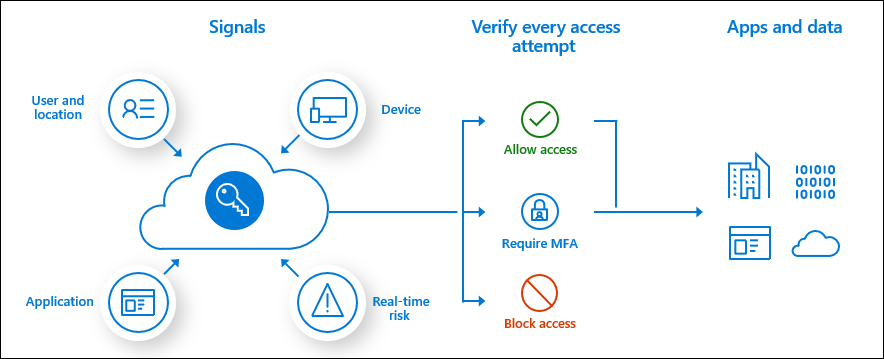


Conditional Access policies at their simplest are if-then statements, if a user wants to access a resource, then they must complete an action. Example: A payroll manager wants to access the payroll application and is required to do multi-factor authentication to access it.

Administrators are faced with two primary goals:

* Empower users to be productive wherever and whenever
* Protect the organization's assets

Use Conditional Access policies to apply the right access controls when needed to keep your organization secure.



**Administrative Units**

Administrative units in Azure Active Directory (Azure AD). An administrative unit is an Azure AD resource that can be a container for other Azure AD resources. An administrative unit can contain only users, groups, or devices.

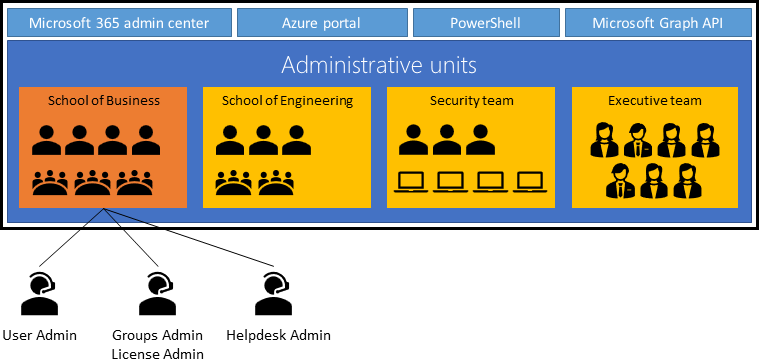
Administrative units restrict permissions in a role to any portion of your organization that you define. You could, for example, use administrative units to delegate the [Helpdesk Administrator](https://docs.microsoft.com/en-us/azure/active-directory/roles/permissions-reference#helpdesk-administrator) role to regional support specialists, so they can manage users only in the region that they support.

## **Deployment scenario**

It can be useful to restrict administrative scope by using administrative units in organizations that are made up of independent divisions of any kind. Consider the example of a large university that's made up of many autonomous schools (School of Business, School of Engineering, and so on). Each school has a team of IT admins who control access, manage users, and set policies for their school.

**A central administrator could:**

* Create an administrative unit for the School of Business.
* Populate the administrative unit with only students and staff within the School of Business.
* Create a role with administrative permissions over only Azure AD users in the School of Business administrative unit.
* Add the business school IT team to the role, along with its scope.



## **Constraints**

Here are some of the constraints for administrative units.

* Administrative units can't be nested.
* Administrative unit-scoped user account administrators can't create or delete users.
* Administrative units are currently not available in [Azure AD Identity Governance](https://docs.microsoft.com/en-us/azure/active-directory/governance/identity-governance-overview).

**Resource Tags**

This can be used to organize your resources.

Each tag consists of a name and a value pair.

For example, if you want to tag resources to a specific department, you can make use of resurce tags.

Tags are metadata elements that you apply to your Azure resources. They're key-value pairs that help you identify resources based on settings that are relevant to your organization. If you want to track the deployment environment for your resources, add a key named Environment. To identify the resources deployed to production, give them a value of Production. Fully formed, the key-value pair becomes, Environment = Production.

You can apply tags to your Azure resources, resource groups, and subscriptions.

**Resource Locks**

As an administrator, you can lock an Azure subscription, resource group, or resource to protect them from accidental user deletions and modifications. The lock overrides any user permissions.

You can set locks that prevent either deletions or modifications. In the portal, these locks are called **Delete** and **Read-only**. In the command line, these locks are called **CanNotDelete** and **ReadOnly**. In the left navigation panel, the subscription lock feature's name is **Resource locks**, while the resource group lock feature's name is **Locks**.

* **CanNotDelete** means authorized users can read and modify a resource, but they can't delete it.
* **ReadOnly** means authorized users can read a resource, but they can't delete or update it. Applying this lock is similar to restricting all authorized users to the permissions that the **Reader** role provides.

Unlike role-based access control (RBAC), you use management locks to apply a restriction across all users and roles. To learn about setting permissions for users and roles, see [Azure RBAC](https://docs.microsoft.com/en-us/azure/role-based-access-control/role-assignments-portal).

**Azure Policy**

Azure Policy helps to enforce organizational standards and to assess compliance at-scale. Through its compliance dashboard, it provides an aggregated view to evaluate the overall state of the environment, with the ability to drill down to the per-resource, per-policy granularity. It also helps to bring your resources to compliance through bulk remediation for existing resources and automatic remediation for new resources.

Common use cases for Azure Policy include implementing governance for resource consistency, regulatory compliance, security, cost, and management. Policy definitions for these common use cases are already available in your Azure environment as built-ins to help you get started.

All Azure Policy data and objects are encrypted at rest.

Azure Policy evaluates resources in Azure by comparing the properties of those resources to business rules. These business rules, described in [JSON format](https://docs.microsoft.com/en-us/azure/governance/policy/concepts/definition-structure), are known as [policy definitions](https://docs.microsoft.com/en-us/azure/governance/policy/overview#policy-definition). To simplify management, several business rules can be grouped together to form a [policy initiative](https://docs.microsoft.com/en-us/azure/governance/policy/overview#initiative-definition) (sometimes called a policySet). Once your business rules have been formed, the policy definition or initiative is [assigned](https://docs.microsoft.com/en-us/azure/governance/policy/overview#assignments) to any scope of resources that Azure supports, such as [management groups](https://docs.microsoft.com/en-us/azure/governance/management-groups/overview), subscriptions, [resource groups](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/overview#resource-groups), or individual resources. The assignment applies to all resources within the [Resource Manager scope](https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/overview#understand-scope) of that assignment. Subscopes can be excluded, if necessary. For more information, see [Scope in Azure Policy](https://docs.microsoft.com/en-us/azure/governance/policy/concepts/scope).

Azure Policy uses a [JSON format](https://docs.microsoft.com/en-us/azure/governance/policy/concepts/definition-structure) to form the logic the evaluation uses to determine whether a resource is compliant or not. Definitions include metadata and the policy rule. The defined rule can use functions, parameters, logical operators, conditions, and property [aliases](https://docs.microsoft.com/en-us/azure/governance/policy/concepts/definition-structure#aliases) to match exactly the scenario you want. The policy rule determines which resources in the scope of the assignment get evaluated.

**Management Groups**

If your organization has many Azure subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. *Management groups* provide a governance scope above subscriptions. You organize subscriptions into management groups the governance conditions you apply cascade by inheritance to all associated subscriptions.

Management groups give you enterprise-grade management at scale no matter what type of subscriptions you might have. However, all subscriptions within a single management group must trust the same Azure Active Directory (Azure AD) tenant.

For example, you can apply policies to a management group that limits the regions available for virtual machine (VM) creation. This policy would be applied to all nested management groups, subscriptions, and resources, and allow VM creation only in authorized regions.

**Monitor and Backup**

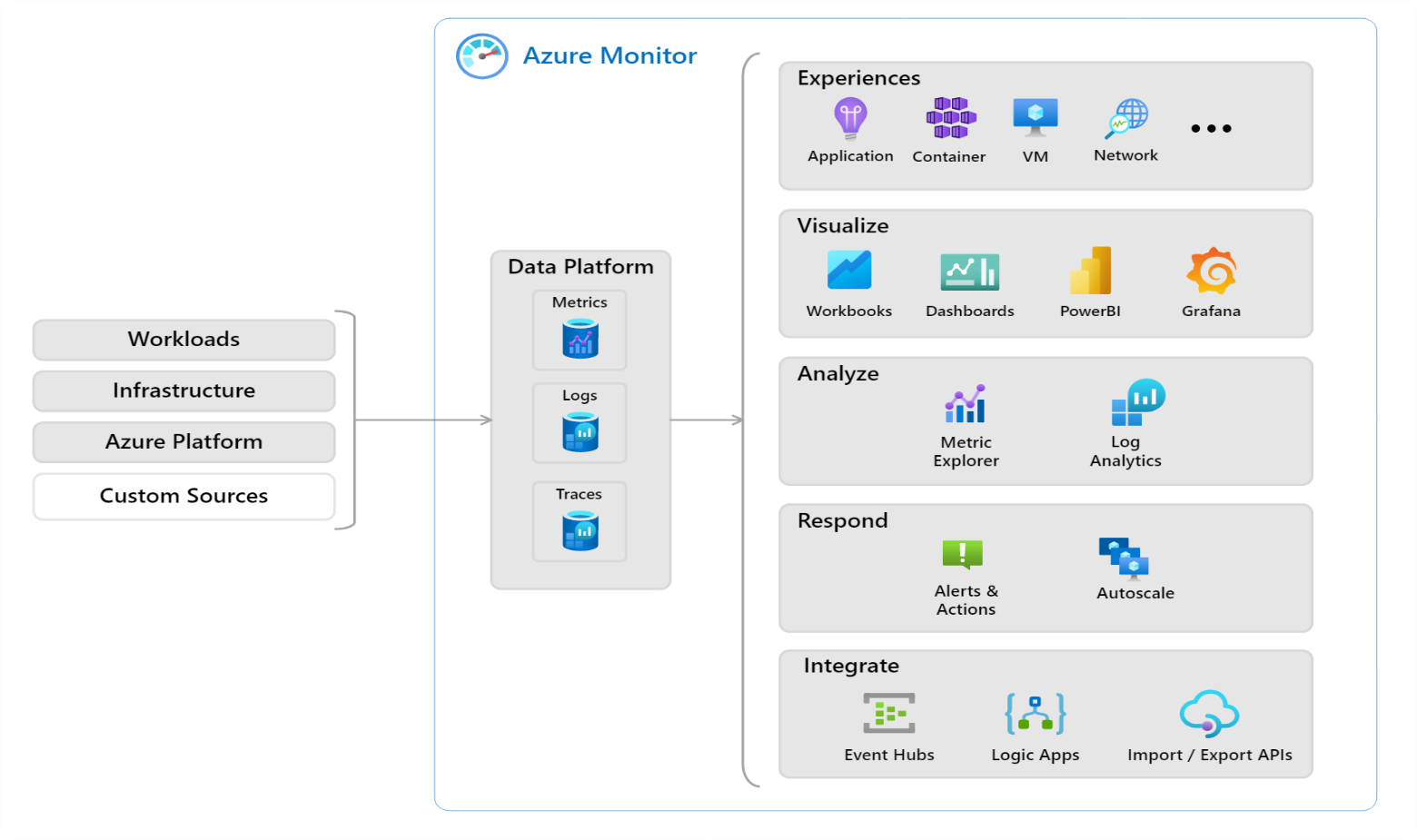
**Azure Monitor Service**

Azure Monitor helps you maximize the availability and performance of your applications and services. It delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. This information helps you understand how your applications are performing and proactively identify issues that affect them and the resources they depend on.

A few examples of what you can do with Azure Monitor include:

* Detect and diagnose issues across applications and dependencies with [Application Insights](https://docs.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview).
* Correlate infrastructure issues with [VM insights](https://docs.microsoft.com/en-us/azure/azure-monitor/vm/vminsights-overview) and [Container insights](https://docs.microsoft.com/en-us/azure/azure-monitor/containers/container-insights-overview).
* Drill into your monitoring data with [Log Analytics](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/log-query-overview) for troubleshooting and deep diagnostics.
* Support operations at scale with [automated actions](https://docs.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-action-rules).
* Create visualizations with Azure [dashboards](https://docs.microsoft.com/en-us/azure/azure-monitor/visualize/tutorial-logs-dashboards) and [workbooks](https://docs.microsoft.com/en-us/azure/azure-monitor/visualize/workbooks-overview).
* Collect data from [monitored resources](https://docs.microsoft.com/en-us/azure/azure-monitor/monitor-reference) by using [Azure Monitor Metrics](https://docs.microsoft.com/en-us/azure/azure-monitor/essentials/data-platform-metrics).
* Investigate change data for routine monitoring or for triaging incidents by using [Change Analysis](https://docs.microsoft.com/en-us/azure/azure-monitor/change/change-analysis).

The following diagram gives a high-level view of Azure Monitor. At the center of the diagram are the data stores for metrics and logs, which are the two fundamental types of data used by Azure Monitor. On the left are the [sources of monitoring data](https://docs.microsoft.com/en-us/azure/azure-monitor/agents/data-sources) that populate these [data stores](https://docs.microsoft.com/en-us/azure/azure-monitor/data-platform). On the right are the different functions that Azure Monitor performs with this collected data. Actions include analysis, alerting, and streaming to external systems.

****

**Azure Monitor Alerts**

Alerts help you detect and address issues before users notice them by proactively notifying you when Azure Monitor data indicates that there may be a problem with your infrastructure or application.

You can alert on any metric or log data source in the Azure Monitor data platform.

An **alert rule** monitors your telemetry and captures a signal that indicates that something is happening on a specified target. The alert rule captures the signal and checks to see if the signal meets the criteria of the condition. If the conditions are met, an alert is triggered, which initiates the associated action group and updates the state of the alert.

You create an alert rule by combining:

* The resource(s) to be monitored.
* The signal or telemetry from the resource
* Conditions

If you're monitoring more than one resource, the condition is evaluated separately for each of the resources and alerts are fired for each resource separately.

**Log Analytic Workspace**

Log Analytics Workspace Insights provides comprehensive monitoring of your workspaces through a unified view of your workspace usage, performance, health, agent, queries, and change log. This article will help you understand how to onboard and use Log Analytics Workspace Insights.

## Overview your Log Analytics workspaces

When accessing Log Analytics Workspace Insights through the Azure Monitor Insights, the 'At scale' perspective is shown. Here you can see how your workspaces are spread across the globe, review their retention, capping and license details (color coded), and choose a workspace to see its insights.

To launch Log Analytics Workspace Insights at scale, perform the following steps:

1. Sign into the [Azure portal](https://portal.azure.com/)
2. Select **Monitor** from the left-hand pane in the Azure portal, and under the Insights Hub section, select **Log Analytics Workspace Insights**.

**Log Analytics Queries**

Azure Monitor Logs is based on Azure Data Explorer, and log queries are written using the same Kusto query language (KQL). This is a rich language designed to be easy to read and author, so you should be able to start writing queries with some basic guidance.

#### **Log Analytics Queries**

Below is a reference for the queries for Log Analytics

1. This can be used for search for a keyword in the event table

Event | search "demovm"

2. This can used to pick up 5 events taken in no specific order

Event | top 10 by TimeGenerated

3. This is used to filter based on a particular property of an event

Event | where EventLevel == 4

4. This can be used to check for the events generated in the previous 5 minutes

Event | where TimeGenerated > ago(5m)

5. This can be used to project certain properties

Event | where TimeGenerated > ago(5m) | project EventLog, Computer

6. Here you can summarize the events

Event |  where TimeGenerated > ago(1d) | summarize count() by Computer,Source

7. Here you can render a bar chart based on the data

Event |  where TimeGenerated > ago(1d) | summarize count() by Computer,Source | render barchart

**Custom Logs**

[Custom logs](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/custom-logs-overview) in Azure Monitor allow you to send external data to a Log Analytics workspace with a REST API.

The Custom Logs data source for the Log Analytics agent in Azure Monitor **allows you to collect events from text files on both Windows and Linux computers**. Many applications log information to text files instead of standard logging services such as Windows Event log or Syslog.

With the DCR based custom logs API in Azure Monitor, you can send data to a Log Analytics workspace from any REST API client. This allows you to send data from virtually any source to [supported built-in tables](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/custom-logs-overview#tables) or to custom tables that you create. You can even extend the schema of built-in tables with custom columns.

**Application Insights**

* Application performance management service for web developers.
* You can use this tool to monitor your application.
* It help developers detect anomalies in the application.
* It can help diagnose issues.
* It can also help understand how users use your application.

**What Gets Monitored**

* Request rates, the response times and failure rates –This is done at the page level.
* Exception recorded by your application.
* Page views and their load performance as reported from the user’s browser.
* User and session counts.
* Performance counters of the underlying Windows or Linux Machines.
* Diagnostic trace logs from your application.

Application Insights is a feature of [Azure Monitor](https://docs.microsoft.com/en-us/azure/azure-monitor/overview) that provides extensible application performance management (APM) and monitoring for live web apps. Developers and DevOps professionals can use Application Insights to:

* Automatically detect performance anomalies.
* Help diagnose issues by using powerful analytics tools.
* See what users actually do with apps.
* Help continuously improve app performance and usability.

Application Insights:

* Supports a wide variety of platforms, including .NET, Node.js, Java, and Python.
* Works for apps hosted on-premises, hybrid, or on any public cloud.
* Integrates with DevOps processes.
* Has connection points to many development tools.
* Can monitor and analyze telemetry from mobile apps by integrating with Visual Studio [App Center](https://appcenter.ms/).

## How Application Insights works

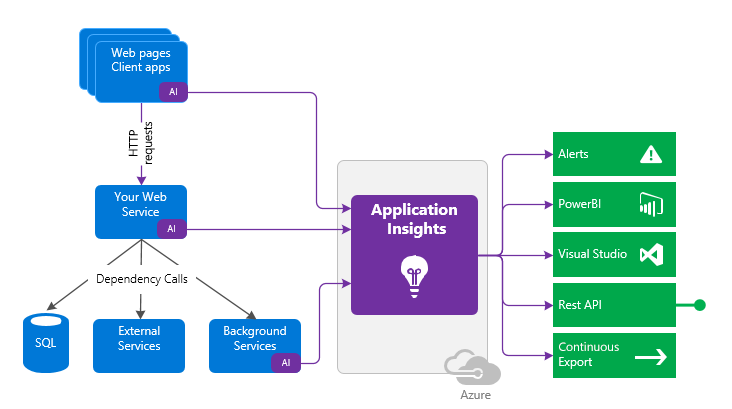
To use Application Insights, you either install a small instrumentation package (SDK) in your app, or enable Application Insights by using the Application Insights agent. For languages and platforms that support the Application Insights agent, see [Supported languages](https://docs.microsoft.com/en-us/azure/azure-monitor/app/platforms).

You can instrument the web app, any background components, and the JavaScript in the web pages themselves. The app and its components don't have to be hosted in Azure.

The instrumentation monitors your app and directs the telemetry data to an Application Insights resource by using a unique instrumentation key. The impact on your app's performance is small. Tracking calls are non-blocking, and are batched and sent in a separate thread.

You can pull in telemetry like performance counters, Azure diagnostics, or Docker logs from host environments. You can also set up web tests that periodically send synthetic requests to your web service. All these telemetry streams are integrated into Azure Monitor. In the Azure portal, you can apply powerful analytics and search tools to the raw data.

The following diagram shows how Application Insights instrumentation in an app sends telemetry to an Application Insights resource.

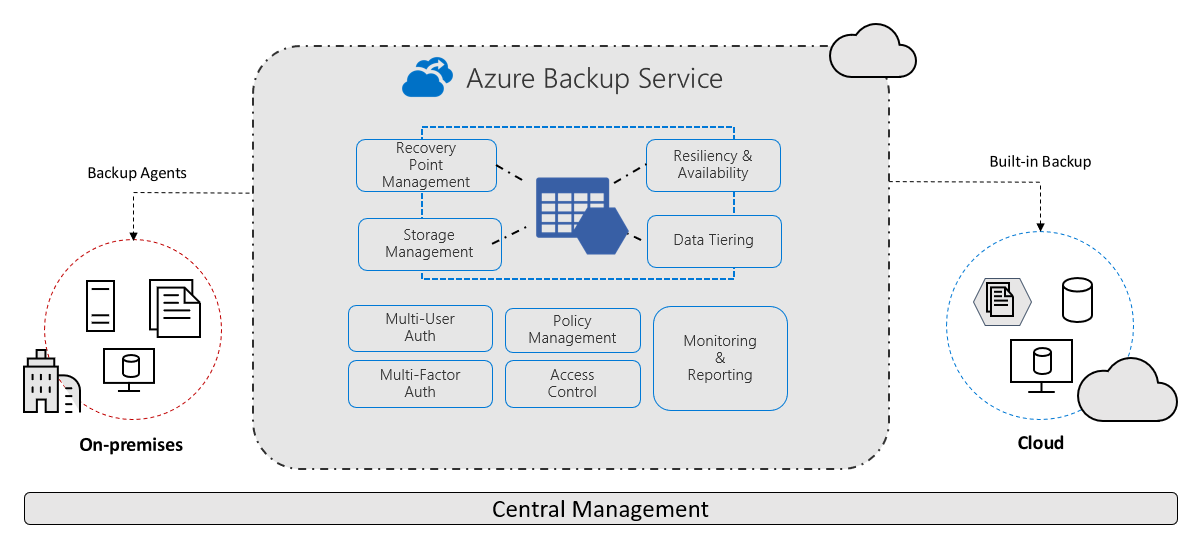


**Azure Backup Service**

The Azure Backup service provides simple, secure, and cost-effective solutions to back up your data and recover it from the Microsoft Azure cloud.

## What can I back up?

* **On-premises** - Back up files, folders, system state using the [Microsoft Azure Recovery Services (MARS) agent](https://docs.microsoft.com/en-us/azure/backup/backup-support-matrix-mars-agent). Or use the DPM or Azure Backup Server (MABS) agent to protect on-premises VMs ([Hyper-V](https://docs.microsoft.com/en-us/azure/backup/back-up-hyper-v-virtual-machines-mabs) and [VMware](https://docs.microsoft.com/en-us/azure/backup/backup-azure-backup-server-vmware)) and other [on-premises workloads](https://docs.microsoft.com/en-us/azure/backup/backup-mabs-protection-matrix)
* **Azure VMs** - [Back up entire Windows/Linux VMs](https://docs.microsoft.com/en-us/azure/backup/backup-azure-vms-introduction) (using backup extensions) or back up files, folders, and system state using the [MARS agent](https://docs.microsoft.com/en-us/azure/backup/backup-azure-manage-mars).
* **Azure Managed Disks** - [Back up Azure Managed Disks](https://docs.microsoft.com/en-us/azure/backup/backup-managed-disks)
* **Azure Files shares** - [Back up Azure File shares to a storage account](https://docs.microsoft.com/en-us/azure/backup/backup-afs)
* **SQL Server in Azure VMs** - [Back up SQL Server databases running on Azure VMs](https://docs.microsoft.com/en-us/azure/backup/backup-azure-sql-database)
* **SAP HANA databases in Azure VMs** - [Backup SAP HANA databases running on Azure VMs](https://docs.microsoft.com/en-us/azure/backup/backup-azure-sap-hana-database)
* **Azure Database for PostgreSQL servers** - [Back up Azure PostgreSQL databases and retain the backups for up to 10 years](https://docs.microsoft.com/en-us/azure/backup/backup-azure-database-postgresql)
* **Azure Blobs** - [Overview of operational backup for Azure Blobs](https://docs.microsoft.com/en-us/azure/backup/blob-backup-overview)



## **How Azure Backup protects from ransomware?**

Azure Backup helps protect your critical business systems and backup data against a ransomware attack by implementing preventive measures and providing tools that protect your organization from every step that attackers take to infiltrate your systems. It provides security to your backup environment, both when your data is in transit and at rest.

**Recovery Service Vault**

Recovery Services vaults make it easy to organize your backup data

This article describes the features of a Recovery Services vault. A Recovery Services vault is a storage entity in Azure that houses data. The data is typically copies of data, or configuration information for virtual machines (VMs), workloads, servers, or workstations. You can use Recovery Services vaults to hold backup data for various Azure services such as IaaS VMs (Linux or Windows) and SQL Server in Azure VMs. Recovery Services vaults support System Center DPM, Windows Server, Azure Backup Server, and more. Recovery Services vaults make it easy to organize your backup data, while minimizing management overhead. Recovery Services vaults are based on the Azure Resource Manager model of Azure, which provides features such as:

* **Enhanced capabilities to help secure backup data**: With Recovery Services vaults, Azure Backup provides security capabilities to protect cloud backups. The security features ensure you can secure your backups, and safely recover data, even if production and backup servers are compromised. [Learn more](https://docs.microsoft.com/en-us/azure/backup/backup-azure-security-feature)
* **Central monitoring for your hybrid IT environment**: With Recovery Services vaults, you can monitor not only your [Azure IaaS VMs](https://docs.microsoft.com/en-us/azure/backup/backup-azure-manage-vms) but also your [on-premises assets](https://docs.microsoft.com/en-us/azure/backup/backup-azure-manage-windows-server#manage-backup-items) from a central portal. [Learn more](https://docs.microsoft.com/en-us/azure/backup/backup-azure-monitoring-built-in-monitor)
* **Azure role-based access control (Azure RBAC)**: Azure RBAC provides fine-grained access management control in Azure. [Azure provides various built-in roles](https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles), and Azure Backup has three [built-in roles to manage recovery points](https://docs.microsoft.com/en-us/azure/backup/backup-rbac-rs-vault). Recovery Services vaults are compatible with Azure RBAC, which restricts backup and restore access to the defined set of user roles. [Learn more](https://docs.microsoft.com/en-us/azure/backup/backup-rbac-rs-vault)
* **Soft Delete**: With soft delete, even if a malicious actor deletes a backup (or backup data is accidentally deleted), the backup data is retained for 14 additional days, allowing the recovery of that backup item with no data loss. The additional 14 days of retention for backup data in the "soft delete" state don't incur any cost to you. [Learn more](https://docs.microsoft.com/en-us/azure/backup/backup-azure-security-feature-cloud).
* **Cross Region Restore**: Cross Region Restore (CRR) allows you to restore Azure VMs in a secondary region, which is an Azure paired region. By enabling this feature at the [vault level](https://docs.microsoft.com/en-us/azure/backup/backup-create-rs-vault#set-cross-region-restore), you can restore the replicated data in the secondary region any time, when you choose. This enables you to restore the secondary region data for audit-compliance, and during outage scenarios, without waiting for Azure to declare a disaster (unlike the GRS settings of the vault).

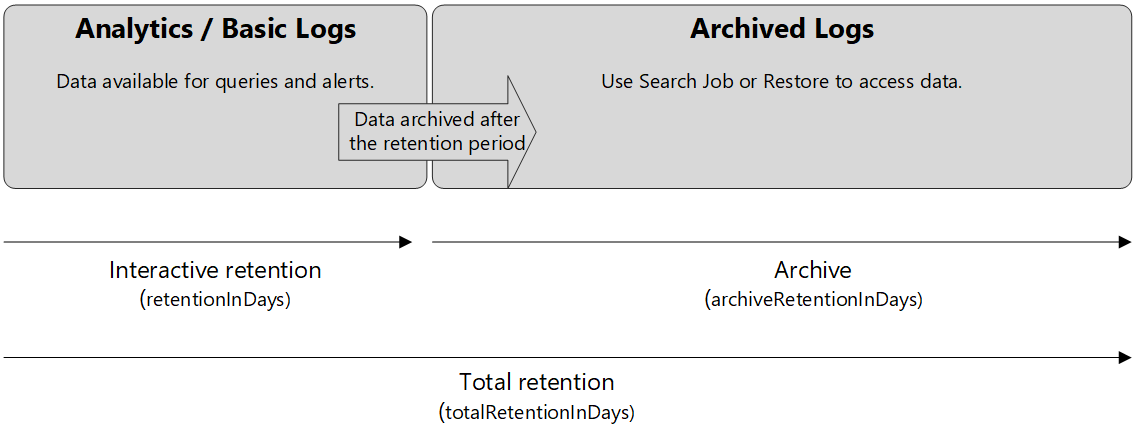
# Data retention and archive policies in Azure Monitor Logs

Retention policies define when to remove or archive data in a [Log Analytics workspace](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/log-analytics-workspace-overview). Archiving lets you keep older, less used data in your workspace at a reduced cost.

This article describes how to configure data retention and archiving.

## How retention and archiving work

Each workspace has a default retention policy that's applied to all tables. You can set a different retention policy on individual tables.



During the interactive retention period, data is available for monitoring, troubleshooting and analytics. When you no longer use the logs, but still need to keep the data for compliance or occasional investigation, archive the logs to save costs. You can access archived data by [running a search job](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/search-jobs) or [restoring archived logs](https://docs.microsoft.com/en-us/azure/azure-monitor/logs/restore).

**Azure Site Recovery**

As an organization, you need to adopt a business continuity and disaster recovery (BCDR) strategy that keeps your data safe, and your apps and workloads online, when planned and unplanned outages occur.

Azure Recovery Services contributes to your BCDR strategy:

* **Site Recovery service**: Site Recovery helps ensure business continuity by keeping business apps and workloads running during outages. Site Recovery [replicates](https://docs.microsoft.com/en-us/azure/site-recovery/azure-to-azure-quickstart) workloads running on physical and virtual machines (VMs) from a primary site to a secondary location. When an outage occurs at your primary site, you fail over to a secondary location, and access apps from there. After the primary location is running again, you can fail back to it.
* **Backup service**: The [Azure Backup](https://docs.microsoft.com/en-us/azure/backup/) service keeps your data safe and recoverable.

Site Recovery can manage replication for:

* Azure VMs replicating between Azure regions.
* On-premises VMs, Azure Stack VMs, and physical servers.

## **What does Site Recovery provide?**

| **Feature** | **Details** |
| --- | --- |
| **Simple BCDR solution** | Using Site Recovery, you can set up and manage replication, failover, and failback from a single location in the Azure portal. |
| **Azure VM replication** | You can set up disaster recovery of Azure VMs from a primary region to a secondary region. |
| **VMware VM replication** | You can replicate VMware VMs to Azure using the improved Azure Site Recovery replication appliance that offers better security and resilience than the configuration server. For more information, see [Disaster recovery of VMware VMs](https://docs.microsoft.com/en-us/azure/site-recovery/vmware-azure-about-disaster-recovery). |
| **On-premises VM replication** | You can replicate on-premises VMs and physical servers to Azure, or to a secondary on-premises datacenter. Replication to Azure eliminates the cost and complexity of maintaining a secondary datacenter. |
| **Workload replication** | Replicate any workload running on supported Azure VMs, on-premises Hyper-V and VMware VMs, and Windows/Linux physical servers. |

**Azure Powershell, CLI & ARM**

**Different Methods of Deploying Resources on Azure**

* 1. **Azure Portal**
  2. **Azure CLI**
  3. **Powershell**
  4. **ARM Template** (Json – Java Script Object Notation)
  5. **Biceps**

**Powershell**

* Powershell is a command line shell.
* It gives you features of scripting language.
* It gives you feature of configuration framework.
* Powershell can run on Linux, Windows, and MacOS.
* It has the ability to work with .NET objects.

Cmdlets – these are powershell commands

Modules – This is a collection of Powershell commands

**Powershell basic commands**

$PSVersionTable

Get-Service

Get-Service -Name “App\*”

Get-Service | Where-Object {$\_.Status -eq “Running”}

**Steps to Install Azure Powershell Module**

* Install Powershell latest version then
* Run as Administrator
* Run these below commands
* Set-ExecutionPolicy -ExecutionPolicy RemoteSigned -Scope CurrentUser
* Install-Module -Name Az -Scope CurrentUser -Repository PSGallery -Force -AllowClobber
* To start working with Azure PowerShell, sign in with your Azure credentials. > Connect-AzAccount

**Some Basic commands of Powershell on wondows**

Connect-AzAccount

Select-AzSubscription -SubscriptionName "XXXXX"

**ARM Overview**

* You define your infrastructure as code.
* Create an Azure Resource Manager template.
* This is a JavaScript Object Notation file that actully contains the definition of the infrastructure.
* You can store the ARM templates in your source code repository along with your application code.

With the move to the cloud, many teams have adopted agile development methods. These teams iterate quickly. They need to repeatedly deploy their solutions to the cloud, and know their infrastructure is in a reliable state. As infrastructure has become part of the iterative process, the division between operations and development has disappeared. Teams need to manage infrastructure and application code through a unified process.

## **Template format**

In its simplest structure, a template has the following elements:

JSONCopy

{

"$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",

"contentVersion": "",

"apiProfile": "",

"parameters": { },

"variables": { },

"functions": [ ],

"resources": [ ],

"outputs": { }

}