AZ-104 Azure Administrator: Comprehensive Study Notes

This document provides an updated and expanded guide for the AZ-104 Microsoft Azure Administrator certification, incorporating the latest Azure service information and best practices.

1. Azure Virtual Machines (VMs)

Azure Virtual Machines (VMs) are one of the most fundamental compute resources in Azure, allowing you to deploy and manage scalable computing on demand. They provide the flexibility of virtualization without the need to buy and maintain the underlying hardware.

1.1. What is an Azure VM?

- **Definition:** An on-demand, scalable computing resource that provides the flexibility of virtualization without the need to buy and maintain the underlying hardware.
- **Purpose:** Used for various workloads, including hosting applications, running development and test environments, and extending on-premises data centers.
- **Operating Systems:** Supports a wide range of operating systems, including Windows Server, Linux distributions (Ubuntu, RHEL, CentOS, etc.), and custom images.

1.2. Creating and Accessing VMs

- **Creation Methods:** VMs can be created using the Azure Portal, Azure CLI, Azure PowerShell, ARM templates, or SDKs.
- **Windows VM Access:** Typically accessed via Remote Desktop Protocol (RDP).
- **Linux VM Access:** Commonly accessed via Secure Shell (SSH).
- **Extensions:** Azure VM extensions are small applications that provide post-deployment configuration and automation tasks on Azure VMs. Examples include custom script extensions, anti-malware extensions, and Azure Monitor extensions.

1.3. Availability Options for Azure VMs

Azure offers several options to ensure the high availability of your virtual machines, protecting your applications and data from planned and unplanned downtime.

- **Availability Zones:**
  - **Concept:** Physically separate locations within an Azure region, each with independent power, cooling, and networking.

    - **SLA:** Provides an industry-leading 99.99% VM uptime SLA when using two or more instances across two or more Availability Zones in the same Azure region.
    - **Protection:** Protects against datacenter-level failures.
    - **Use Case:** Ideal for applications requiring the highest level of availability and resilience to regional outages.
- **Availability Sets:**
    - **Concept:** A logical grouping of VMs that allows Azure to understand how your application is built to provide for redundancy and availability.
    - **SLA:** Provides a 99.95% VM uptime SLA for two or more VMs within an Availability Set.
    - **Fault Domains (FD):** Physical separation of VMs across different racks, power, and network switches. Protects against hardware failures and power interruptions within a datacenter.
    - **Update Domains (UD):** Logical grouping of VMs that can be rebooted at the same time during planned maintenance. Ensures that not all VMs in your application are updated simultaneously.
    - **Use Case:** Suitable for applications requiring high availability within a single datacenter.
- **Virtual Machine Scale Sets (VMSS):**
    - **Concept:** A logical grouping of identical, load-balanced VMs. Allows you to create and manage a group of identical, auto-scaling VMs.
    - **Scalability:** Enables automatic scaling (scaling out by adding more instances, scaling in by removing instances) based on demand or a defined schedule.
    - **High Availability:** Inherently provides high availability by distributing instances across fault domains and update domains.
    - **Use Case:** Ideal for large-scale, stateless applications that require dynamic scaling and high availability, such as web servers or batch processing.

1.4. Scalability

Scalability refers to the ability of a system to handle a growing amount of work by adding resources.

- **Vertical Scaling (Scale Up/Down):**
    - **Concept:** Increasing or decreasing the resources (CPU, RAM, disk space) of a single VM instance.

  - **Scale Up:** Adding more power to an existing VM (e.g., upgrading from a Standard_B2s to a Standard_D4s_v3).
  - **Scale Down:** Reducing the resources of an existing VM.
  - **Use Case:** Suitable for applications that require more power for a single instance, but where horizontal scaling is not feasible or necessary.
- **Horizontal Scaling (Scale Out/In):**
  - **Concept:** Adding or removing VM instances to distribute the workload across multiple machines.
  - **Scale Out:** Adding more VM instances to handle increased load (e.g., adding more web servers to a web application).
  - **Scale In:** Removing VM instances when demand decreases.
  - **Mechanism:** Primarily achieved using Virtual Machine Scale Sets (VMSS).
  - **Use Case:** Ideal for stateless applications that can distribute their workload across multiple instances, providing high availability and elasticity (e.g., e-commerce websites during peak sales like 'Big Billion Day Sale' or 'The Great Indian Sale').

2. Managing Identity and Governance in Azure

Azure provides robust identity and access management capabilities, primarily through Microsoft Entra ID (formerly Azure Active Directory), to secure access to resources and applications.

2.1. What is a Directory Service?

A directory service is a software system that stores, organizes, and provides access to information in a directory. In the context of identity management, it stores information about users, groups, devices, and applications, and manages access to resources.

2.2. Microsoft Entra ID (formerly Azure Active Directory - AAD)

Microsoft Entra ID is Microsoft's cloud-based identity and access management service. It helps your employees and guest users sign in and access internal and external resources securely.

- **Key Features:**
  - **Authentication and Authorization:** Manages user authentication and authorizes access to applications and resources.
  - **Single Sign-On (SSO):** Enables users to access multiple applications and resources with a single set of credentials.

    - **User and Group Management:** Centralized management of user accounts, groups, and their attributes.
    - **Application Access:** Provides secure access to thousands of SaaS applications (e.g., Salesforce, Office 365) and custom applications.
    - **Hybrid Identity:** Synchronizes on-premises Active Directory identities to Microsoft Entra ID, enabling a unified identity experience.
    - **Multifactor Authentication (MFA):** Adds an extra layer of security by requiring more than one form of verification.
    - **Passwordless Authentication:** Supports various passwordless methods like FIDO2 security keys, Windows Hello for Business, and Microsoft Authenticator app.
    - **Conditional Access:** Enforces policies based on user, location, device, and application to grant or deny access.
- **Users in Microsoft Entra ID:**
    - **Member Users:** Users created directly in Microsoft Entra ID or synchronized from on-premises Active Directory.
    - **Guest Users:** External users invited to collaborate, typically from other Microsoft Entra tenants or consumer identity providers (e.g., Google, Microsoft accounts).
- **Groups in Microsoft Entra ID:**
    - **Security Groups:** Used to manage access to Azure resources, Microsoft Entra roles, and applications. Can contain users, other security groups, and device objects.
    - **Microsoft 365 Groups:** Designed for collaboration, providing a shared inbox, calendar, files, and other resources. Can be used for email distribution lists.
    - **Dynamic Groups:** Groups whose membership is automatically updated based on user or device attributes, defined by rules.
- **Self-Service Password Reset (SSPR):**
    - Allows users to reset their passwords without administrator intervention, reducing helpdesk calls.
    - Requires proper configuration and user registration of authentication methods.
- **Device Management and Join Types:**
    - **Microsoft Entra registered:** Devices registered with Microsoft Entra ID without requiring an organizational account to sign in to the device. Typically for personal devices (BYOD) to access corporate resources.
    - **Microsoft Entra joined:** Devices joined directly to Microsoft Entra ID, primarily for cloud-only organizations or mobile scenarios. Users sign in with their organizational accounts.

- **Hybrid Microsoft Entra joined:** Devices joined to both on-premises Active Directory and Microsoft Entra ID. Common in hybrid environments where organizations have on-premises AD and want to leverage Microsoft Entra ID for cloud services.

2.3. Microsoft Entra ID vs. Active Directory Domain Services (AD DS)

Feature	Microsoft Entra ID (Cloud-based)	Active Directory Domain Services (On-premises)
Deployment Model	SaaS (Software as a Service)	On-premises, managed by the organization
Management	Microsoft manages the platform, infrastructure, and domain controllers	Organization manages servers, domain controllers, and infrastructure
Protocols	HTTP/HTTPS, REST, SAML, OAuth 2.0, OpenID Connect, WS-Fed	LDAP, Kerberos, NTLM
Scalability	Highly available, secure, and scalable by design	Scalability depends on infrastructure design and management
Multi-tenancy	Multi-tenant architecture (isolated instances for each organization)	Single-tenant architecture (one organization per AD forest)
Hierarchy	Flat structure, no forests, trees, or parent-child domains	Hierarchical structure with forests, trees, and parent-child domains
Group Policy	Does not directly support Group Policy Objects (GPOs)	Supports GPOs for centralized management of user and computer settings
B2B/B2C	Built-in support for Business-to-Business (B2B) and Business-to-Consumer (B2C) collaboration	Requires additional components (e.g., ADFS) for external identity integration

2.4. Microsoft Entra ID Licenses and Premium Features

Microsoft Entra ID offers different licensing tiers, each providing a different set of features:

- **Free:** Basic user and group management, directory synchronization, self-service password change for cloud users, and single sign-on to a limited number of SaaS apps.
- **Microsoft 365 Apps:** Includes features for productivity apps like Exchange, SharePoint, and Teams, along with basic Microsoft Entra ID capabilities.
- **Premium P1:** Includes all Free features plus advanced administration, hybrid identity capabilities, self-service password reset for on-premises users, Conditional Access, MFA, Microsoft Identity Manager (MIM), and advanced usage reporting.
- **Premium P2:** Includes all Premium P1 features plus Microsoft Entra ID Protection (risk-based Conditional Access, identity risk detections) and Microsoft Entra ID Governance (Privileged Identity Management, access reviews, entitlement management).
- **Premium Features (P1/P2):**
    - **Identity Protection:** Detects and remediates identity-based risks, such as leaked credentials, suspicious sign-ins, and compromised users.
    - **Identity Governance:** Manages identity and access lifecycle, including:
        - **Privileged Identity Management (PIM):** Manages, controls, and monitors access to important resources in Microsoft Entra ID and Azure. Provides just-in-time and just-enough access.
        - **Access Reviews:** Enables organizations to manage group memberships, access to enterprise applications, and role assignments efficiently.
        - **Entitlement Management:** Automates access request workflows, access assignments, reviews, and expiration for users.
    - **Conditional Access Policy:** Granular control over access to resources based on conditions like user location, device compliance, and application sensitivity.
    - **Dynamic Queries (for Dynamic Groups):** Automatically updates group membership based on defined rules and user/device attributes.

3. Roles and Azure Hierarchy

Azure employs a robust role-based access control (RBAC) system to manage who has access to what resources. This is complemented by a hierarchical structure that helps organize and govern resources at scale.

3.1. Azure Role-Based Access Control (RBAC)

Azure RBAC is an authorization system that provides fine-grained access management of Azure resources. It allows you to manage who has access to Azure resources, what they can do with those resources, and what areas they can access.

- **Key Principles:**
  - **Least Privilege:** Grant only the necessary permissions to perform a task.
  - **Separation of Duties:** Ensure that no single person has too much control.
- **How RBAC Works:**
  - **Security Principal:** The entity that requests access to an Azure resource (user, group, service principal, or managed identity).
  - **Role Definition:** A collection of permissions. Azure provides many built-in roles (e.g., Owner, Contributor, Reader, Virtual Machine Contributor) and allows for custom roles.
  - **Scope:** The level at which the access is applied. This can be a management group, subscription, resource group, or individual resource.
  - **Assignment:** The process of attaching a role definition to a security principal at a specific scope.
- **Types of Roles:**
  - **Microsoft Entra Roles (formerly AAD Roles):**
    - Govern access to Microsoft Entra ID resources (e.g., users, groups, applications) and administrative tasks within the directory.
    - Examples:
      - **Global Administrator:** Has all permissions in Microsoft Entra ID and can manage all administrative aspects of the directory.
      - **User Administrator:** Manages all aspects of users and groups, including creating and deleting users, and managing passwords.
      - **Application Administrator:** Manages all aspects of enterprise applications, application registrations, and application proxy settings.
      - **Global Reader:** Can read everything in Microsoft Entra ID but cannot make any changes.
  - **Azure Resource Roles (Subscription Roles):**
    - Govern access to Azure resources (e.g., VMs, storage accounts, virtual networks) within a subscription.
    - Examples:
      - **Owner:** Has full access to all resources and can delegate access to others.
      - **Contributor:** Can create and manage all types of Azure resources but cannot grant access to others.
      - **Reader:** Can view all Azure resources but cannot make any changes.

        - **Virtual Machine Contributor:** Can create and manage virtual machines but cannot access the virtual network or storage account they are connected to.
    - **Custom Roles:**
        - If built-in roles don't meet your specific access needs, you can create custom roles by combining a set of permissions.
        - Custom roles can be assigned at the management group, subscription, or resource group scope.

3.2. Azure Hierarchy

Azure organizes resources in a hierarchical structure to provide a clear scope for management, governance, and billing.

- **Hierarchy Levels (from top to bottom):**
    1. **Management Groups (MGs)**
    2. **Subscriptions**
    3. **Resource Groups**
    4. **Resources**
- **Management Groups (MGs):**
    - **Purpose:** Provide a level of scope above subscriptions. They allow you to organize subscriptions into containers to apply governance conditions (like policies and RBAC) to all subscriptions within the management group.
    - **Benefits:** Centralized management, simplified policy assignment, and consistent compliance across multiple subscriptions.
    - **Hierarchy:** Can be nested up to six levels deep, allowing for a flexible and scalable organizational structure.
- **Subscriptions:**
    - **Purpose:** A logical container for Azure services and resources. It links Azure resources to an Azure account, which is used for billing.
    - **Billing Boundary:** All resources deployed within a subscription are billed together.
    - **Access Boundary:** Access control policies (RBAC) are applied at the subscription level and inherited by resources within it.
    - **Trust Relationship:** Each Azure subscription has a trust relationship with a Microsoft Entra tenant, which authenticates users and services.
- **Resource Groups (RGs):**

    - **Purpose:** A logical container for Azure resources that share the same lifecycle, management, and permissions.
    - **Lifecycle Management:** Resources within a resource group can be deployed, updated, and deleted together.
    - **Location:** A resource group is associated with a region, but the resources within it can be in different regions.
    - **Best Practice:** Group resources that are part of the same application or solution into a single resource group.
- **Resources:**
    - **Definition:** Individual instances of Azure services (e.g., Virtual Machines, Storage Accounts, Virtual Networks, Web Apps).
    - **Deployment:** Resources are deployed within resource groups.

3.3. Governance Features

Azure provides several features to help you govern your resources and ensure compliance.

- **Azure Policy:**
    - **Purpose:** Enforces organizational standards and assesses compliance at scale. It ensures that resources adhere to corporate standards and service level agreements.
    - **Definition:** Specifies the conditions for compliance and the effect to take (e.g., Deny, Audit, DeployIfNotExists).
    - **Assignment:** Policies are assigned to a scope (management group, subscription, or resource group) and apply to all resources within that scope.
    - **Initiatives (Policy Sets):** A collection of policy definitions grouped together to achieve a larger goal. For example, a security initiative might include policies for encryption, network security, and access control.
- **Azure Resource Locks:**
    - **Purpose:** Prevent accidental deletion or modification of critical Azure resources.
    - **Types:**
        - **CanNotDelete:** Authorized users can still read and modify a resource, but they can't delete it.
        - **ReadOnly:** Authorized users can only read a resource. They can't delete or update it.
    - **Inheritance:** Locks applied at a higher scope (e.g., subscription or resource group) are inherited by resources within that scope.

- **Resource Movement:**
  - Azure allows you to move resources between resource groups and subscriptions. This can be useful for organizational changes, refactoring, or managing resource lifecycles.
  - **Considerations:** Understand dependencies and potential impacts before moving resources.
- **Resource Tagging:**
  - **Purpose:** Apply metadata to your Azure resources in the form of key-value pairs.
  - **Benefits:** Helps with resource organization, cost management, billing, operational management, and compliance.
  - **Examples:** `Environment: Production`, `CostCenter: Marketing`, `Owner: JohnDoe`.

3.4. Cost Management

Managing costs effectively in Azure is crucial for optimizing your cloud spend.

- **Cost Analysis:**
  - **Purpose:** Provides tools to monitor, allocate, and optimize your cloud costs.
  - **Features:** View historical costs, forecast future spending, and identify cost trends.
- **Budgets:**
  - **Purpose:** Set spending thresholds and receive alerts when your spending approaches or exceeds predefined limits.
  - **Actions:** Can trigger automated actions (e.g., stopping VMs) when budget thresholds are met.
- **Cost Saving Strategies:**
  - **Azure Reservations:** Commit to a one-year or three-year plan for certain Azure services (e.g., VMs, Azure SQL Database) to get significant discounts compared to pay-as-you-go rates.
  - **Azure Hybrid Benefit (BYOL - Bring Your Own License):** Use your existing on-premises Windows Server and SQL Server licenses with Software Assurance to save on Azure VMs and SQL Database.
  - **Region Selection:** Costs can vary significantly between Azure regions. Choose regions that offer the best pricing for your services.
  - **Credits:** Utilize any Azure credits you may have (e.g., from Visual Studio subscriptions or Azure sponsorships).

  - **Spot Instances:** Leverage unused Azure compute capacity at a significant discount. Ideal for fault-tolerant, flexible applications that can handle interruptions.
  - **Right-sizing Resources:** Continuously monitor and adjust the size of your VMs and other resources to match actual usage, avoiding over-provisioning.
  - **Auto-scaling:** Implement auto-scaling for VM Scale Sets and App Services to automatically adjust capacity based on demand, minimizing costs during low usage periods.
  - **Deallocate Unused Resources:** Stop and deallocate VMs and other resources when not in use to avoid incurring compute charges.

4. Administrative Tools

Azure provides a variety of tools to manage and interact with your resources, catering to different preferences and automation needs.

4.1. Azure Portal

- **Description:** A web-based, unified console that provides a graphical user interface (GUI) to manage all your Azure resources.
- **Features:**
  - **Centralized Management:** Create, configure, monitor, and delete resources from a single interface.
  - **Customizable Dashboard:** Personalize your dashboard with widgets to quickly access frequently used resources and monitor key metrics.
  - **Cloud Shell Integration:** Provides direct access to Azure CLI and Azure PowerShell within the portal.
  - **Cost Management:** Built-in tools for cost analysis and budget management.
  - **Marketplace:** Discover and deploy a wide range of Azure services and third-party solutions.
  - **Accessibility:** Accessible from any web browser on any device.

4.2. Azure Resource Manager (ARM) Templates

- **Description:** A JSON-based declarative language used to define and deploy your Azure infrastructure as code (IaC).
- **Key Concepts:**
  - **Declarative Syntax:** You describe the desired state of your infrastructure, and ARM takes care of deploying it.

    - **Idempotent:** Deploying the same template multiple times results in the same resource state, without creating duplicate resources.
    - **Components:**
        - **Schema:** Specifies the ARM template language version.
        - **ContentVersion:** Version of the template (e.g., 1.0.0.0).
        - **Parameters:** Values that are provided at deployment time to customize the deployment (e.g., VM size, resource names).
        - **Variables:** Values that are constructed within the template to simplify expressions.
        - **Resources:** The actual Azure resources to be deployed (e.g., `Microsoft.Compute/virtualMachines`).
        - **Outputs:** Values returned from the deployment (e.g., public IP address of a VM).
        - **Functions:** Built-in functions to perform operations like string manipulation, array creation, and resource ID retrieval.
    - **Mandatory Components:** `schema`, `contentVersion`, and `resources` are mandatory.
- **Benefits:**
    - **Automation:** Automate the deployment of complex infrastructure.
    - **Consistency:** Ensure consistent deployments across environments (dev, test, prod).
    - **Version Control:** Store templates in version control systems (e.g., Git) for tracking changes and collaboration.
    - **Error Reduction:** Reduce human error associated with manual deployments.
- **Deployment Methods:** Can be deployed via Azure Portal, Azure CLI, Azure PowerShell, Visual Studio, Visual Studio Code, and Azure DevOps pipelines.

4.3. Azure CLI (Command-Line Interface)

- **Description:** A cross-platform command-line tool for connecting to Azure and executing administrative commands.
- **Features:**
    - **Cross-Platform:** Runs on Windows, macOS, and Linux.
    - **Scripting:** Ideal for automating tasks through scripts.
    - **Syntax:** Uses a `az <command group> <command> <subcommand>` structure (e.g., `az vm create`, `az storage account list`).
    - **Integration:** Integrates with Azure Cloud Shell and local terminals.

    - **Output Formats:** Supports various output formats like JSON, JSONC, TSV, Table, and YAML.

4.4. Azure PowerShell

- **Description:** A set of cmdlets (command-lets) for managing Azure resources using PowerShell.
- **Features:**
    - **PowerShell Integration:** Leverages the full power of PowerShell scripting.
    - **Cmdlet Naming Convention:** Follows a `Verb-Noun` naming convention (e.g., `New-AzVM`, `Get-AzStorageAccount`).
    - **Modules:** Organized into modules (e.g., `Az.Compute`, `Az.Storage`). The `Az` module is the recommended replacement for the deprecated `AzureRM` module.
    - **Scripting:** Excellent for complex automation and administrative tasks.
    - **Integration:** Integrates with Azure Cloud Shell, PowerShell ISE, and Visual Studio Code.

4.5. REST Client and SDKs

- **REST Client:**
    - **Description:** Azure services are built on a RESTful API architecture. You can interact with Azure directly using HTTP requests.
    - **Use Case:** Advanced scenarios where you need direct control over API calls or when no SDK is available for a specific language/platform.
- **SDKs (Software Development Kits):**
    - **Description:** Libraries available for various programming languages (e.g., Python, .NET, Java, Node.js) that simplify interaction with Azure services.
    - **Benefits:** Abstract away the complexities of REST API calls, provide object-oriented interfaces, and handle authentication, error handling, and retry logic.
    - **Use Case:** Developing custom applications that integrate with Azure, automating complex workflows, or building management tools.

5. Networking in 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 (VMs), to securely communicate with each other, the internet, and on-premises networks.

5.1. Azure Virtual Networks (VNets)

- **Purpose:** Provides isolated and highly secure network environments in Azure.
- **Key Components:**
  - **Address Space:** A private IP address range (e.g., 10.0.0.0/16) that you define for your VNet. This range must be unique within your organization if you plan to connect to on-premises networks or other Azure VNets.
  - **Subnets:** VNets can be segmented into one or more subnets. Each subnet is a range of IP addresses within the VNet’ s address space. Resources are deployed into subnets.
  - **IP Addresses:**
    - **Private IP:** Assigned to resources within a VNet, used for internal communication. Can be static (remains the same) or dynamic (may change).
    - **Public IP:** Assigned to resources to enable inbound and outbound communication with the internet. Can be static or dynamic. Public IPs come in Basic and Standard SKUs, with Standard offering more features like availability zones support and security by default.
  - **Network Interface Card (NIC):** Enables an Azure VM to communicate with internet, Azure, and on-premises resources. A VM can have one or more NICs, each with its own IP configuration.
- **Private IP Ranges (RFC 1918):**
  - **Class A:** 10.0.0.0 - 10.255.255.255 (CIDR /8)
  - **Class B:** 172.16.0.0 - 172.31.255.255 (CIDR /16)
  - **Class C:** 192.168.0.0 - 192.168.255.255 (CIDR /24)
- **Azure Bastion:**
  - **Purpose:** A fully managed PaaS service that provides secure and seamless RDP/SSH connectivity to your virtual machines directly from the Azure portal over SSL. It eliminates the need for public IP addresses on your VMs.
  - **Benefits:** Enhanced security, simplified connectivity, and no exposure of VMs to the public internet.

5.2. Network Security Groups (NSGs)

- **Description:** A basic, stateful packet filtering firewall that allows or denies inbound and outbound network traffic to Azure resources.
- **Rules:** Consist of 5-tuple rules (Source IP, Destination IP, Source Port, Destination Port, Protocol).

- **Priority:** Rules are processed in order of priority (lower number = higher priority).
- **Default Rules:** Azure automatically creates default inbound and outbound rules.
- **Scope:** Can be associated with subnets or individual network interfaces (NICs).
- **Cost:** Free of charge.
- **Service Tags:** Represent a group of IP address prefixes for a given Azure service (e.g., VirtualNetwork, AzureLoadBalancer, Internet). Simplifies NSG rule creation.

5.3. Azure Firewall

- **Description:** A managed, cloud-based network security service that protects your Azure Virtual Network resources. It is a fully stateful firewall as a service.
- **Features:**
  - **Managed Service (PaaS):** Azure manages the underlying infrastructure, providing built-in high availability and unrestricted cloud scalability.
  - **Layer 7 Filtering:** Supports application-level filtering (HTTP/HTTPS) in addition to network-level filtering (Layer 3/4).
  - **Threat Intelligence:** Integrates with Microsoft Cyber Security to provide real-time threat intelligence feeds.
  - **Rules:**
    - **NAT Rules:** Network Address Translation rules (DNAT for inbound, SNAT for outbound).
    - **Network Rules:** Similar to NSG rules, based on IP address, port, and protocol.
    - **Application Rules:** Allow or deny FQDNs (Fully Qualified Domain Names) for HTTP/HTTPS traffic.
  - **FQDN Tagging:** Simplifies rule creation for well-known Azure and Windows services.
  - **Forced Tunneling:** Routes all internet-bound traffic from Azure to your on-premises network via a VPN or ExpressRoute connection.
  - **Deployment:** Requires its own dedicated subnet (`AzureFirewallSubnet`) within a VNet.
  - **Cost:** A paid service.

5.4. VNet Peering

- **Purpose:** Connects two Azure Virtual Networks seamlessly, allowing resources in both VNets to communicate with each other as if they were in the same network.
- **Types:**
  - **Regional VNet Peering:** Connects VNets within the same Azure region.

    - **Global VNet Peering:** Connects VNets across different Azure regions.
- **Benefits:** Low latency, high bandwidth connectivity, and no public IP addresses required for communication between peered VNets.

5.5. User-Defined Routes (UDRs)

- **Purpose:** Allows you to override Azure’ s default system routes and define custom routes for traffic within your VNet.
- **Use Cases:**
    - **Forcing Traffic through a Network Virtual Appliance (NVA):** Directing all traffic from a subnet through a firewall or other virtual appliance.
    - **Custom Routing Scenarios:** Implementing complex network topologies.
- **Components:**
    - **Route Table:** A collection of routes.
    - **Routes:** Define the destination IP address range and the next hop type (e.g., Virtual Appliance, Virtual Network Gateway, Internet).
- **Configuration Steps (Example for NVA):**
    1. Create a dedicated subnet for the NVA (e.g., AzureFirewallSubnet).
    2. Deploy the NVA (e.g., Azure Firewall) into this subnet.
    3. Create a Route Table.
    4. Add a route to the Route Table: Destination (e.g., 0.0.0.0/0 for all traffic), Next Hop Type (Virtual Appliance), Next Hop Address (NVA’ s private IP).
    5. Associate the Route Table with the subnets from which you want to force traffic through the NVA.

5.6. Azure DNS

- **Purpose:** A hosting service for DNS domains that provides name resolution using Microsoft Azure infrastructure.
- **Types:**
    - **Public DNS:** Hosts your public-facing domains (e.g., yourwebsite.com) and resolves public DNS queries.
    - **Private DNS:** Provides a reliable and secure DNS service for your virtual networks. It allows you to use custom domain names within your private network without the need for custom DNS solutions.
- **DNS Record Types:**

- **A Record:** Maps a domain name to an IPv4 address.
- **AAAA Record:** Maps a domain name to an IPv6 address.
- **CNAME Record:** Creates an alias for a domain name.
- **MX Record:** Specifies mail exchange servers for a domain.
- **PTR Record:** Performs reverse DNS lookup, mapping an IP address to a domain name.
- **NS Record:** Specifies the name servers for a domain.
- **SOA Record:** Start of Authority record, provides authoritative information about a DNS zone.
- **TXT Record:** Stores text information about a domain.
- **CAA Record:** Specifies which certificate authorities are allowed to issue certificates for a domain.

5.7. Azure Load Balancers

Azure offers various load balancing solutions to distribute network traffic across multiple backend resources, ensuring high availability and scalability.

- **Azure Load Balancer (ALB):**
  - **Description:** A Layer 4 (TCP/UDP) load balancer that distributes incoming network traffic across healthy instances within a backend pool.
  - **SKUs:** Basic and Standard. Basic is being retired, and Standard is recommended for production workloads.
  - **Components:**
    - **Frontend IP Configuration:** The IP address of the load balancer.
    - **Backend Pool:** A group of VMs or instances that will receive the load-balanced traffic.
    - **Health Probes:** Monitor the health of instances in the backend pool.
    - **Load Balancing Rules:** Define how incoming traffic is distributed to the backend pool.
    - **Inbound NAT Rules:** Direct specific inbound traffic to a particular VM or instance.
  - **Use Case:** Ideal for basic load balancing of TCP/UDP traffic, often used for internal applications or public-facing services where advanced application-layer features are not required.
- **Azure Application Gateway (AGW):**

  - **Description:** A Layer 7 (HTTP/HTTPS) load balancer that provides application-level routing and web application firewall (WAF) capabilities.
  - **Features:**
    - **URL-based Routing:** Routes traffic based on URL paths.
    - **Multi-site Hosting:** Hosts multiple web applications on a single Application Gateway instance.
    - **SSL Termination:** Decrypts SSL traffic at the gateway, offloading the encryption/decryption burden from backend servers.
    - **End-to-End SSL:** Re-encrypts traffic before sending it to backend servers.
    - **Web Application Firewall (WAF):** Protects web applications from common web vulnerabilities (e.g., SQL injection, cross-site scripting).
    - **Session Affinity:** Directs subsequent requests from a user to the same backend server.
    - **HTTP/2 Support:** Supports the HTTP/2 protocol.
    - **Redirection:** Redirects HTTP traffic to HTTPS.
    - **Custom Error Pages:** Configures custom error pages.
  - **Use Case:** Ideal for web applications requiring advanced routing, SSL termination, and WAF protection.
- **Azure Traffic Manager:**
  - **Description:** A DNS-based traffic load balancer that distributes traffic globally across multiple endpoints (e.g., Azure regions, external endpoints).
  - **Purpose:** Provides high availability and responsiveness by directing users to the closest or best-performing endpoint.
  - **Routing Methods:** Priority, Weighted, Performance, Geographic, Multivalue, Subnet.
  - **Use Case:** Global load balancing for applications deployed in multiple regions, ensuring disaster recovery and optimal user experience.
- **Azure Front Door:**
  - **Description:** A scalable, secure, and highly available entry point for global web applications. It combines CDN capabilities with advanced routing and security features.
  - **Features:**
    - **Global HTTP/HTTPS Load Balancing:** Routes traffic to the fastest and most available backend.

  - **Web Application Firewall (WAF):** Provides centralized protection for web applications.
  - **URL-based Routing and Path-based Routing:** Advanced routing capabilities.
  - **SSL Offloading and End-to-End SSL:** Similar to Application Gateway.
  - **Caching:** Improves performance by caching content at edge locations.
  - **DDoS Protection:** Integrated DDoS protection.
- **Use Case:** Ideal for global web applications requiring fast, secure, and highly available access with advanced routing and security features.

5.8. ALB vs. AGW (Comparison)

Feature	Azure Load Balancer (ALB)	Azure Application Gateway (AGW)
Layer	Layer 4 (Transport Layer)	Layer 7 (Application Layer)
Protocols	TCP, UDP	HTTP, HTTPS, HTTP/2, WebSocket
Features	Basic load balancing, Inbound NAT, Health Probes	URL-based routing, Multi-site hosting, SSL termination, WAF, Session affinity, Redirection, Custom error pages
SSL	Does not support SSL termination	Supports SSL termination and end-to-end SSL
Cost	Generally lower cost (Basic SKU is free, Standard has cost)	Higher cost due to advanced features
Use Case	Internal applications, basic public-facing services	Web applications, APIs, microservices requiring advanced routing and security
Intelligence	Less intelligent, operates at packet level	More intelligent, operates at application request level
FQDN Support	No direct FQDN support in rules	Supports FQDNs in application rules

Subnet	Can be deployed in any subnet	Requires a dedicated subnet

6. Azure Storage

Azure Storage is a Microsoft-managed cloud service that provides highly available, scalable, durable, and redundant storage for a variety of data objects.

6.1. Azure Storage Accounts

- **Purpose:** A storage account is a unique namespace in Azure for your data objects. All data objects in Azure Storage are stored in a storage account.
- **Account Kinds:**
  - **General-purpose v2 (GPv2):** The recommended storage account type for most scenarios. It supports all Azure Storage features and data services (Blobs, Files, Queues, Tables, Data Lake Storage Gen2).
  - **General-purpose v1 (GPv1):** An older type of storage account that supports all Azure Storage features but may not have the latest features or lowest per-gigabyte pricing. It is recommended to upgrade GPv1 accounts to GPv2.
  - **BlockBlobStorage:** A specialized storage account for block blobs with premium performance characteristics. Ideal for workloads requiring high transaction rates or low storage latency.
  - **FileStorage:** A specialized storage account for premium file shares. Optimized for high-performance applications requiring shared storage.

6.2. Types of Azure Storage

Azure Storage offers different types of storage services, each designed for specific scenarios:

- **Blob Storage:**
  - **Purpose:** Stores massive amounts of unstructured data, such as text or binary data. Ideal for images, videos, documents, backup data, and data for analytics.
  - **Blob Types:**
    - **Block Blobs:** Optimized for uploading large amounts of data efficiently. Ideal for streaming and cloud-native workloads (e.g., documents, media files).
    - **Append Blobs:** Optimized for append operations, making them suitable for logging data from VMs or IoT devices.

        - **Page Blobs:** Optimized for random read/write operations. Used for virtual hard drive (VHD) files for Azure VMs.
    - **Access Tiers:**
        - **Hot:** Optimized for frequently accessed data. Higher storage costs, lower access costs.
        - **Cool:** Optimized for infrequently accessed data (accessed at least once every 30 days). Lower storage costs, higher access costs.
        - **Archive:** Optimized for rarely accessed data (accessed less than once every 180 days) with flexible latency requirements. Lowest storage costs, highest access costs.
- **Azure Files:**
    - **Purpose:** Provides fully managed file shares in the cloud that can be accessed via the industry-standard Server Message Block (SMB) protocol or Network File System (NFS) protocol.
    - **Use Case:** Can be used to replace or supplement on-premises file servers, lift-and-shift applications that rely on file shares, or for shared application settings.
    - **Azure File Sync:** Synchronizes Azure file shares with on-premises Windows Servers, providing a hybrid cloud solution.
- **Azure Table Storage:**
    - **Purpose:** A NoSQL key-value store for large datasets of structured, non-relational data.
    - **Use Case:** Ideal for flexible datasets like web applications, address books, device information, or other types of metadata.
- **Azure Queue Storage:**
    - **Purpose:** A service for storing large numbers of messages that can be accessed from anywhere in the world via authenticated HTTP or HTTPS calls.
    - **Use Case:** Decoupling components of a cloud application, building asynchronous workflows.
- **Azure Data Lake Storage Gen2:**
    - **Purpose:** A set of capabilities built on Azure Blob Storage, dedicated to big data analytics. It combines the scalability of Blob Storage with the features of a data lake.
    - **Use Case:** Big data analytics workloads, data warehousing.

6.3. Performance Tiers

- **Standard:** Uses Hard Disk Drives (HDDs) for cost-effective storage. Suitable for most workloads.
- **Premium:** Uses Solid State Drives (SSDs) for high-performance, low-latency storage. Ideal for I/O-intensive workloads like databases.

6.4. Replication Options

Azure Storage offers various replication strategies to ensure data durability and high availability.

- **Locally Redundant Storage (LRS):**
  - **Description:** Replicates your data three times within a single data center in the primary region.
  - **Durability:** Protects against drive and rack failures.
  - **SLA:** 99.999999999% (11 nines) durability over a given year.
- **Zone-Redundant Storage (ZRS):**
  - **Description:** Replicates your data three times across three Azure availability zones in the primary region.
  - **Durability:** Protects against data center-level failures within a region.
  - **SLA:** 99.9999999999% (12 nines) durability over a given year.
- **Geo-Redundant Storage (GRS):**
  - **Description:** Replicates your data three times within the primary region (LRS) and also replicates it three times to a secondary, distant region.
  - **Durability:** Protects against regional outages.
  - **SLA:** 99.99999999999999% (16 nines) durability over a given year.
- **Read-Access Geo-Redundant Storage (RA-GRS):**
  - **Description:** Similar to GRS, but also provides read access to the data in the secondary region.
  - **Use Case:** Ideal for applications that need to be highly available even during a regional outage, allowing read access to data from the secondary region.
- **Geo-Zone-Redundant Storage (GZRS):**
  - **Description:** Combines the high availability of ZRS with the disaster recovery capabilities of geo-replication. Data is replicated across three availability zones in the primary region and also replicated to a secondary, distant region.
  - **Durability:** Protects against both data center-level failures and regional outages.

- **Read-Access Geo-Zone-Redundant Storage (RA-GZRS):**
  - **Description:** Similar to GZRS, but also provides read access to the data in the secondary region.

6.5. Security of Azure Storage

Azure Storage offers multiple layers of security to protect your data.

- **Access Keys:**
  - **Description:** Two 512-bit storage account access keys (primary and secondary) that provide full access to the storage account.
  - **Best Practice:** Use Shared Access Signatures (SAS) or Microsoft Entra ID authentication instead of access keys for granular control and reduced risk.
- **Shared Access Signatures (SAS):**
  - **Description:** A URI that grants restricted access rights to your Azure Storage resources for a specified period.
  - **Types:**
    - **Service SAS:** Grants access to a specific resource (blob, file, queue, table).
    - **Account SAS:** Grants access to resources in one or more storage services.
    - **User Delegation SAS:** Secured with Microsoft Entra credentials and provides superior security.
  - **Benefits:** Granular control over permissions, start/expiry times, and IP addresses.
- **Encryption:**
  - **Encryption at Rest:** Data is automatically encrypted when written to Azure Storage using Microsoft-managed keys or customer-managed keys (Azure Key Vault integration).
  - **Encryption in Transit:** Data is encrypted when transferred between Azure services or to/from on-premises using HTTPS.
  - **Dual Layer Encryption:** Provides two layers of encryption for data at rest, enhancing security.
- **Secure Transfer Required:** Enforces that all requests to the storage account must be made over HTTPS.
- **Microsoft Entra ID Authentication:**
  - **Description:** Use Microsoft Entra ID to authorize requests to Blob and Queue storage, providing role-based access control (RBAC) at the container and blob level.
  - **Benefits:** Centralized identity management, auditing, and fine-grained permissions.

- **Firewall and Virtual Networks:**
    - **Description:** Configure storage account firewalls to restrict access to specific IP addresses or virtual networks.
    - **Private Endpoints:** Allows secure access to Azure Storage from your VNet via a private link, eliminating exposure to the public internet.
- **Container and Blob Security:**
    - **Public Access Levels:** Containers can be configured for private (no anonymous access), blob (anonymous read access for blobs), or container (anonymous read access for containers and blobs).
- **Immutable Storage:**
    - **Purpose:** Stores data in a write-once, read-many (WORM) state, preventing modification or deletion for a specified retention period.
    - **Use Case:** Regulatory compliance, legal hold, data retention policies.
- **Delete Protection (Soft Delete):**
    - **Purpose:** Enables recovery of accidentally deleted blobs or containers for a specified retention period.
    - **Benefits:** Prevents permanent data loss due to accidental deletions.

6.6. Storage Management Tools

- **Azure Storage Explorer:** A standalone graphical tool that allows you to easily work with Azure Storage data across Windows, macOS, and Linux.
- **AzCopy:** A command-line utility designed for high-performance copying of data to and from Azure Blob, File, and Table storage.
- **Azure Data Box:** A portfolio of products (Box, Disk, Heavy) used for transferring large amounts of data to Azure when network transfer is not feasible or too slow.
- **Azure Import/Export Service:** A service that allows you to securely import large amounts of data to Azure Blob storage or Azure Files, or export data from Azure Blob storage, by shipping disk drives to an Azure datacenter.

7. Platform as a Service (PaaS) and Container Services

Azure offers a variety of Platform as a Service (PaaS) offerings that abstract away the underlying infrastructure, allowing developers to focus on building and deploying applications. Containerization has also become a key deployment strategy, and Azure provides robust services to manage containerized workloads.

7.1. Azure App Service

- **Description:** A fully managed platform for building, deploying, and scaling web apps, mobile backends, and RESTful APIs. It supports various programming languages and frameworks.
- **App Service Plans:**
    - **Concept:** Defines a set of compute resources (VMs) for your App Service apps to run on. You pay for the compute resources defined by the App Service plan, not for the individual apps.
    - **Tiers:** Offers various pricing tiers (e.g., Free, Shared, Basic, Standard, Premium, Isolated) with different features, scalability, and performance levels.
    - **Scaling:** Supports both manual and auto-scaling of instances within the plan.
- **Deployment Slots (for Zero Downtime Blue-Green Deployment):**
    - **Concept:** Live apps running in separate environments. You can deploy to a staging slot, test it, and then swap it with the production slot without downtime.
    - **Benefits:** Enables blue-green deployments, A/B testing, and easy rollback.
    - **Configuration:** Each slot has its own hostname, and you can swap content and configurations between slots.

7.2. Azure Container Instances (ACI)

- **Description:** The fastest and simplest way to run a container in Azure without having to manage any virtual machines or learn a container orchestration platform.
- **Features:**
    - **Serverless Containers:** Provides a serverless platform for running containers, eliminating the need to provision and manage underlying infrastructure.
    - **Per-Second Billing:** You pay only for the compute resources consumed by your containers, billed by the second.
    - **Fast Startup:** Rapid deployment and startup times for containers.
    - **Public IP Connectivity:** Can expose containers directly to the internet with a public IP address.
    - **Persistent Storage:** Can mount Azure File shares for persistent storage.
    - **Use Cases:** Simple applications, task automation, batch processing, development and test environments.

7.3. Azure Container Registry (ACR)

- **Description:** A managed, private Docker registry service in Azure for storing and managing your private Docker container images and other OCI (Open Container Initiative) artifacts.
- **Features:**
  - **Private Registry:** Securely store your container images.
  - **Geo-Replication:** Replicate your registry to multiple Azure regions for global distribution and improved performance.
  - **Security:** Integrated with Microsoft Entra ID for authentication and authorization.
  - **ACR Tasks:** Automate OS and framework patching for your Docker images, and build images in the cloud.
  - **Use Cases:** Centralized repository for container images used by Azure Kubernetes Service (AKS), Azure Container Instances (ACI), and other container platforms.

7.4. Azure Kubernetes Service (AKS)

- **Description:** A fully managed Kubernetes service that simplifies deploying, managing, and scaling containerized applications using Kubernetes.
- **Architecture:**
  - **Master Node (Control Plane):** Managed by Azure, responsible for managing the Kubernetes cluster (scheduling, scaling, updates). You don't have direct access to the master node.
  - **Worker Nodes:** Virtual machines that run your containerized applications (pods). You manage and pay for these nodes.
- **Key Concepts:**
  - **Pods:** The smallest deployable units in Kubernetes, representing a single instance of a running process in your cluster. A pod can contain one or more containers.
  - **Deployments:** Define how your application is deployed and updated. They manage the desired state of your pods.
  - **Services:** Abstract the network access to a set of pods. They provide a stable IP address and DNS name for your application.
    - **LoadBalancer Service:** Exposes your application to the internet via an Azure Load Balancer.
    - **Ingress:** Manages external access to services in a cluster, typically HTTP/HTTPS. Can be integrated with Azure Application Gateway.
  - **ReplicaSets:** Ensure that a specified number of pod replicas are running at any given time.

    - **DaemonSets:** Ensure that all (or some) nodes run a copy of a pod. Useful for running cluster-level services like logging agents.
    - **CronJobs:** For time-based job scheduling.
    - **Jobs:** For one-off tasks.
    - **StatefulSets:** Used for stateful applications that require stable, unique network identifiers and persistent storage.
- **Management Tools:**
    - `kubectl`**:** The command-line tool for interacting with Kubernetes clusters.
    - **Azure CLI:** Used for managing AKS clusters (e.g., `az aks get-credentials`).
- **Use Cases:** Complex microservices architectures, applications requiring high scalability, portability, and advanced orchestration capabilities.

7.5. Containers inside a VM

- **Description:** While Azure offers managed container services, you can also run Docker containers directly on Azure Virtual Machines.
- **Use Case:** When you need more control over the underlying operating system, specific software dependencies, or custom networking configurations that are not available in managed container services.

8. Azure Database Services

Azure offers a wide range of fully managed database services, catering to various application needs, from relational to NoSQL databases.

8.1. Azure SQL Database

- **Description:** A fully managed, intelligent, and scalable relational database service built on the latest stable version of the Microsoft SQL Server database engine. It provides a platform as a service (PaaS) offering for SQL Server.
- **Deployment Options:**
    - **Single Database:** A fully managed, isolated database suitable for modern cloud applications that need a single reliable data source.
    - **Elastic Pools:** A cost-effective solution for managing multiple databases with varying and unpredictable usage demands. Resources are shared among databases in the pool.
    - **Managed Instance:** Provides near 100% compatibility with the latest SQL Server on-premises database engine, offering a hybrid solution for migrating existing SQL

Server applications to Azure with minimal changes.

- **Service Tiers (Purchasing Models):**
  - **vCore-based purchasing model:** Offers flexibility, control, and transparency over resource consumption. You can choose the number of vCores, memory, and storage.
  - **DTU-based purchasing model:** Provides a blend of compute, memory, and I/O resources in a single measure (Database Transaction Unit). Simpler for estimating performance.
- **Key Features:**
  - **Built-in High Availability:** Automatic backups, point-in-time restore, and geo-replication for disaster recovery.
  - **Intelligent Performance:** Built-in intelligence for performance tuning, threat detection, and vulnerability assessment.
  - **Scalability:** Easily scale up or down compute and storage resources independently.
  - **Security:** Advanced threat protection, data encryption (at rest and in transit), and network security (VNet integration, Private Link).
  - **Serverless compute tier:** Automatically scales compute based on workload demand and bills for the compute used per second.

8.2. Azure Database for MySQL

- **Description:** A fully managed relational database service for MySQL, based on the open-source MySQL Community Edition. It provides high availability, scalability, and security for your MySQL applications.
- **Deployment Options:**
  - **Flexible Server:** Provides more granular control and customization options, including zone-redundant high availability, planned maintenance windows, and custom server parameters. Recommended for most new deployments.
  - **Single Server:** An older deployment option that is being phased out. It offers less control and fewer features compared to Flexible Server.
- **Key Features:**
  - **High Availability:** Automatic failover with zone-redundant high availability in Flexible Server.
  - **Scalability:** Scale compute and storage independently.
  - **Security:** Network isolation with VNet integration, Private Link, and data encryption.
  - **Automated Backups:** Automatic backups and point-in-time restore.

    - **Read Replicas:** Create read-only replicas of your database to offload read-heavy workloads and improve performance.
    - **Monitoring:** Integration with Azure Monitor for performance and health monitoring.

8.3. Azure Database for PostgreSQL

- **Description:** A fully managed relational database service for PostgreSQL, based on the open-source PostgreSQL Community Edition. It provides high availability, scalability, and security for your PostgreSQL applications.
- **Deployment Options:**
    - **Flexible Server:** Provides more granular control and customization options, including zone-redundant high availability, planned maintenance windows, and custom server parameters. Recommended for most new deployments.
    - **Single Server:** An older deployment option that is being phased out. It offers less control and fewer features compared to Flexible Server.
- **Key Features:**
    - **High Availability:** Automatic failover with zone-redundant high availability in Flexible Server.
    - **Scalability:** Scale compute and storage independently.
    - **Security:** Network isolation with VNet integration, Private Link, and data encryption.
    - **Automated Backups:** Automatic backups and point-in-time restore.
    - **Read Replicas:** Create read-only replicas of your database to offload read-heavy workloads and improve performance.
    - **Monitoring:** Integration with Azure Monitor for performance and health monitoring.
    - **Extensions:** Supports a wide range of PostgreSQL extensions.

8.4. Azure Database for MariaDB (Retiring)

- **Description:** A fully managed relational database service for MariaDB. **Important Note: Azure Database for MariaDB is scheduled for retirement on September 19, 2025.**
- **Migration Recommendation:** Microsoft recommends migrating to Azure Database for MySQL Flexible Server by the retirement date.

8.5. Azure Cosmos DB

- **Description:** A fully managed, globally distributed, multi-model database service that provides turn-key global distribution, elastic scaling of throughput and storage, and guaranteed low-latency access.

- **Key Features:**
  - **Global Distribution:** Distribute your data across any number of Azure regions with a single click, enabling low-latency access for users worldwide.
  - **Multi-model API Support:** Supports various APIs, including:
    - **NoSQL (Core) API:** For document data, compatible with MongoDB.
    - **MongoDB API:** For MongoDB applications.
    - **Cassandra API:** For Apache Cassandra applications.
    - **Gremlin API:** For graph databases.
    - **Table API:** For key-value data, compatible with Azure Table Storage.
  - **Elastic Scalability:** Independently scale throughput (Request Units per second - RUs) and storage globally and elastically.
  - **Guaranteed Low Latency:** Offers single-digit millisecond latency at the 99th percentile, backed by SLAs.
  - **Five Consistency Models:** Provides a spectrum of consistency choices: Strong, Bounded Staleness, Session, Consistent Prefix, and Eventual.
  - **Automatic Indexing:** Automatically indexes all data without requiring schema or index management.
  - **Change Feed:** Provides a persistent, ordered record of changes that happen in your database, enabling real-time data processing.
  - **Serverless:** A consumption-based model where you only pay for the Request Units and storage consumed by your database operations.
  - **Integrated Cache:** Improves read latency and reduces costs for read-heavy workloads.
- **Use Cases:** Web, mobile, gaming, and IoT applications that need to handle massive amounts of data with low latency and high availability.

8.6. Azure Cache for Redis

- **Description:** A secure, dedicated Redis cache that provides an in-memory data store based on the open-source Redis (Remote Dictionary Server) software. It significantly improves the performance and scalability of applications that rely on backend data stores.
- **Key Features:**
  - **In-Memory Data Store:** Stores frequently accessed data in memory, enabling very fast read and write operations.

  - **High Performance:** Achieves sub-millisecond latency and high throughput.
  - **Scalability:** Easily scale up or down to meet changing application demands.
  - **Data Structures:** Supports various Redis data structures like strings, hashes, lists, sets, sorted sets, and more.
  - **Persistence:** Can be configured for data persistence to disk.
  - **Security:** Integrated with Azure Virtual Network for network isolation, and supports SSL/TLS encryption for data in transit.
  - **Patching and Updates:** Azure manages the patching and updates of the Redis software, ensuring your cache is always up-to-date and secure.
  - **Use Cases:** Caching frequently accessed data, session management, leaderboards, real-time analytics, message brokering, and distributed caching.
- **Tiers:**
  - **Basic:** Single node, suitable for development/test and non-critical workloads.
  - **Standard:** Two-node primary/replica configuration for high availability, suitable for production workloads.
  - **Premium:** Includes all Standard features plus advanced capabilities like clustering, VNet integration, persistence, and geo-replication.
  - **Enterprise/Enterprise Flash:** Newer tiers offering enhanced performance, scalability, and enterprise-grade features, including support for Redis 7.2 and Redis modules.

9. Microsoft Entra Connect (formerly Azure AD Connect)

Microsoft Entra Connect is a tool designed to meet and accomplish your hybrid identity goals. It provides a way to synchronize identities between your on-premises Active Directory and Microsoft Entra ID.

9.1. Key Features and Components

- **Synchronization Services:**
  - **Password Hash Synchronization (PHS):** A simple method to synchronize a hash of a user's on-premises AD password to Microsoft Entra ID. This allows users to use the same password for both on-premises and cloud resources.
  - **Pass-through Authentication (PTA):** Provides simple password validation against on-premises Active Directory for cloud authentication. When users try to sign in to Microsoft Entra ID, PTA agents validate their passwords directly against on-premises AD.

    - **Federation Integration (with AD FS):** Allows you to use your on-premises Active Directory Federation Services (AD FS) infrastructure to enable single sign-on (SSO) for Microsoft Entra ID. This is suitable for organizations with complex authentication requirements.
    - **Cloud Sync (Microsoft Entra Connect Cloud Sync):** A lightweight agent-based synchronization service that is an alternative to Microsoft Entra Connect Sync. It is ideal for organizations with multiple disconnected AD forests or those looking for a simpler synchronization solution.
- **Health Monitoring (Microsoft Entra Connect Health):** Provides robust monitoring capabilities for your hybrid identity infrastructure, including synchronization services, AD FS, and AD DS.
- **Automatic Upgrades:** Microsoft Entra Connect supports automatic upgrades to ensure you are always running the latest version with the most recent features and security updates.

9.2. Synchronization Scenarios

- **User and Group Synchronization:** Synchronizes user accounts, groups, and their attributes from on-premises AD to Microsoft Entra ID.
- **Device Writeback:** Writes back device objects from Microsoft Entra ID to on-premises AD for Hybrid Microsoft Entra joined devices.
- **Password Writeback:** Enables users to reset their passwords in Microsoft Entra ID and have those changes written back to their on-premises AD accounts.

9.3. Considerations

- **Minimum Version Requirements:** Ensure you are running a supported version of Microsoft Entra Connect to receive updates and support.
- **Staging Mode:** Use staging mode to test configuration changes or new versions of Microsoft Entra Connect before promoting them to production.
- **High Availability:** Implement high availability for Microsoft Entra Connect by deploying multiple servers in staging mode or using a highly available database for the synchronization service.

10. Azure Monitoring

Azure Monitor is a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud and on-premises environments. It helps you understand how your applications and infrastructure are performing and proactively identifies issues.

10.1. Azure Monitor Components

- **Metrics:**
  - **Description:** Numerical values that describe some aspect of a system at a particular point in time. They are lightweight and support near real-time scenarios.
  - **Use Case:** Performance monitoring, real-time alerting, and quick diagnostics.
- **Logs:**
  - **Description:** Event data that is collected from various sources (e.g., Azure resources, applications, operating systems) and stored in a Log Analytics workspace.
  - **Log Analytics Workspace:** A unique environment for storing, querying, and analyzing log data. It uses Kusto Query Language (KQL) for powerful data analysis.
  - **Use Case:** Deep diagnostics, root cause analysis, security auditing, and compliance.
- **Activity Log:**
  - **Description:** Provides a history of subscription-level events that occurred in Azure. It tells you what, who, and when for any write operations (PUT, POST, DELETE) taken on resources in your subscription.
  - **Use Case:** Auditing, troubleshooting, and tracking changes to your Azure environment.
- **Diagnostic Settings:**
  - **Description:** Configures the export of platform logs and metrics from Azure resources to various destinations, such as Log Analytics, Storage Accounts, or Event Hubs.
  - **Use Case:** Centralized logging, long-term archiving, and integration with third-party monitoring tools.

10.2. Azure Monitor Features

- **Alerts:**
  - **Purpose:** Proactively notify you when conditions in your monitoring data are met. They can trigger actions like sending emails, SMS, or calling webhooks.
  - **Types:**
    - **Metric Alerts:** Triggered when a metric crosses a threshold.
    - **Log Alerts:** Triggered when a log query returns results that meet specified criteria.
    - **Activity Log Alerts:** Triggered by specific events in the Activity Log.

    - **Action Groups:** A collection of notification preferences and actions that can be triggered by an alert.
- **Dashboards:**
    - **Purpose:** Provide a customizable, consolidated view of your Azure resources and their performance.
    - **Features:** Display metrics, logs, and other data visualizations in a single pane of glass.
    - **Customization:** Drag-and-drop interface to create personalized dashboards.
- **Workbooks:**
    - **Purpose:** Flexible canvases for data analysis and the creation of rich visual reports within the Azure portal.
    - **Features:** Combine text, analytics queries, Azure Metrics, and parameters into rich interactive reports.
    - **Use Case:** Troubleshooting guides, operational playbooks, and post-mortem analysis.
- **Application Insights:**
    - **Description:** An Application Performance Management (APM) service that monitors live web applications. It automatically detects performance anomalies and includes powerful analytics tools to help you diagnose issues.
    - **Features:** Performance monitoring, dependency mapping, live metrics, usage tracking, and custom events.
- **Network Watcher:**
    - **Description:** Provides tools to monitor, diagnose, and gain insights into your Azure network performance.
    - **Features:** IP flow verify, NSG flow logs, connection troubleshoot, packet capture, and VPN diagnostics.

11. Monitoring and Backup

Effective monitoring and robust backup strategies are crucial for maintaining the health, performance, and availability of your Azure resources.

11.1. Azure Backup

- **Description:** A highly scalable and reliable cloud-based backup solution that protects your data in Azure, on-premises, and from other clouds.

- **Key Features:**
    - **Centralized Management:** Manage backups for various workloads from a single console.
    - **Application-Consistent Backups:** Ensures that application data is consistent at the time of backup, preventing data corruption during recovery.
    - **Long-Term Retention:** Retain backups for years, meeting compliance requirements.
    - **Cost-Effective:** Pay-as-you-go pricing, with no upfront costs.
    - **Encryption:** Data is encrypted in transit and at rest.
    - **Supported Workloads:**
        - **Azure VMs:** Backup Windows and Linux VMs.
        - **SQL Server in Azure VMs:** Application-consistent backups for SQL databases.
        - **SAP HANA in Azure VMs:** Application-consistent backups for SAP HANA databases.
        - **Azure Files:** Backup Azure file shares.
        - **Azure Blobs:** Operational backup for blob data.
        - **On-premises workloads:** Using the Microsoft Azure Recovery Services (MARS) agent or Azure Backup Server (MABS).
- **Recovery Services Vault:**
    - **Purpose:** A management entity in Azure that stores backup data and recovery points for various Azure services.
    - **Features:** Centralized monitoring, reporting, and security features for backups.
    - **Replication Options:** Supports LRS, GRS, and ZRS for backup data.

11.2. Azure Site Recovery (ASR)

- **Description:** A disaster recovery as a service (DRaaS) solution that helps ensure business continuity by keeping business apps and workloads running during outages.
- **Key Features:**
    - **Replication:** Replicates workloads running on physical and virtual machines from a primary site to a secondary location (Azure or a secondary on-premises datacenter).
    - **Orchestrated Failover:** Automates the failover process to the secondary site with minimal downtime.
    - **Failback:** Restores workloads to the primary site after an outage is resolved.

    - **Non-Disruptive Testing:** Perform disaster recovery drills without impacting production workloads.
    - **Supported Workloads:** Azure VMs, on-premises VMware VMs, Hyper-V VMs, and physical servers.
    - **RPO (Recovery Point Objective):** Measures data loss tolerance. ASR can achieve low RPOs.
    - **RTO (Recovery Time Objective):** Measures downtime tolerance. ASR helps achieve low RTOs.

12. Azure Cloud Shell

Azure Cloud Shell is an interactive, browser-accessible shell for managing Azure resources. It provides a flexible way to work with Azure, offering the choice of Bash or PowerShell experience.

12.1. Key Features

- **Browser-Based:** Accessible directly from the Azure portal, eliminating the need for local installations.
- **Pre-configured Environment:** Comes pre-installed with popular Azure command-line tools and utilities, including:
    - **Azure CLI:** For managing Azure resources.
    - **Azure PowerShell:** For managing Azure resources.
    - **Git:** For version control.
    - **Terraform:** For Infrastructure as Code.
    - **Ansible:** For automation and configuration management.
    - **Kubectl:** For managing Kubernetes clusters.
    - **Docker:** For container management.
    - **Various programming languages:** Python, Node.js, .NET, etc.
- **Persistent Storage:** Provides a 5 GB persistent storage (mounted as `clouddrive`) for your files, scripts, and configurations, ensuring your work is saved across sessions.
- **Authentication:** Automatically authenticates to your Azure subscription, so you don't need to log in separately.
- **Customization:** You can customize your Cloud Shell environment by installing additional tools or configuring your shell profile.

12.2. Use Cases

- **Quick Management:** Perform quick administrative tasks without setting up a local environment.
- **Script Execution:** Run Azure CLI or Azure PowerShell scripts directly.
- **Learning and Experimentation:** Experiment with Azure commands and services in a pre-configured environment.
- **Troubleshooting:** Diagnose and troubleshoot issues with Azure resources.