**WHAT IS CLOUD COMPUTING?**

**Cloud computing** refers to the delivery of computing services—such as servers, storage, databases, networking, software, analytics, and intelligence—over the internet ("the cloud").

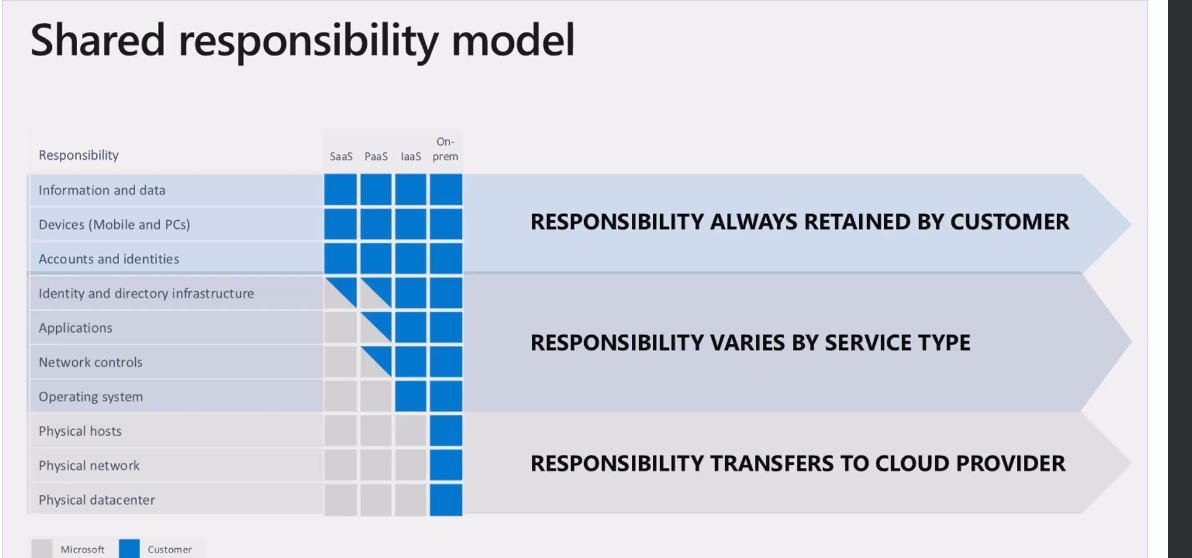
1. **Ownership**: Cloud services are hosted on infrastructure owned by providers such as Microsoft Azure.
2. **On-Demand Availability**: Users can access these services as needed, paying only for what they use.
3. **Popular Services**: Businesses often leverage cloud computing for its vast array of services, which are available on-demand.

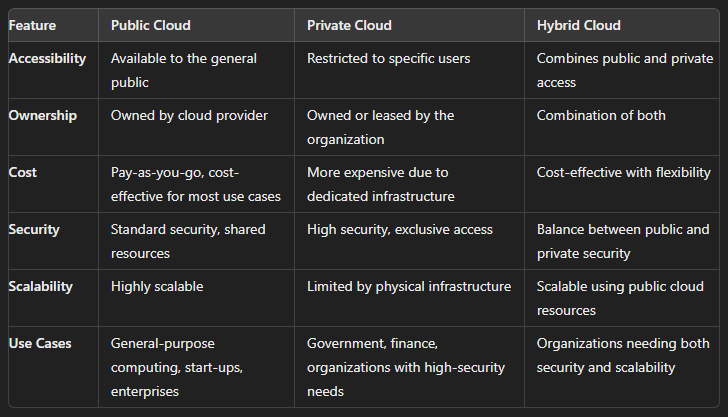
**Benefits of cloud services:**

* **Scalability**: Easily scale resources up or down based on demand.
* **Cost Efficiency**: Pay only for what you use, avoiding large upfront investments in hardware.
* **Flexibility**: Access a wide variety of services and tools to meet specific needs, from AI to database management.
* **Security**: Cloud providers offer robust security measures, including encryption and compliance with various regulations.

**Cloud Service Types:**

* **Infrastructure as a Service (IaaS):** Provides virtualized computing resources over the internet. Users manage the OS, storage, and applications, while the provider manages the hardware.
* **Platform as a Service (PaaS):** Offers a platform allowing customers to develop, run, and manage applications without dealing with the underlying infrastructure.
* **Software as a Service (SaaS):** Delivers software applications over the internet, with the provider managing everything from the infrastructure to the application itself.





**Cloud Pricing:**

**Factors Affecting Cloud Pricing**

* Region
* Operating System
* Hybrid Benefit
* Hardware Specifications
  + CPU and RAM
  + Storage Type
* Bandwidth
* Backups
* Reservations

**BENEFITS OF CLOUD COMPUTING**

**1.High Availability**

The ability of a system to remain operational during planned and unplanned outages.

**Planned Outages:** Necessary system maintenance, such as updates, patches, and hardware upgrades.

**Unplanned Outages:** Unexpected disruptions like hardware failures, network issues, power outages, natural disasters, cyber-attacks, software bugs, or poor design.

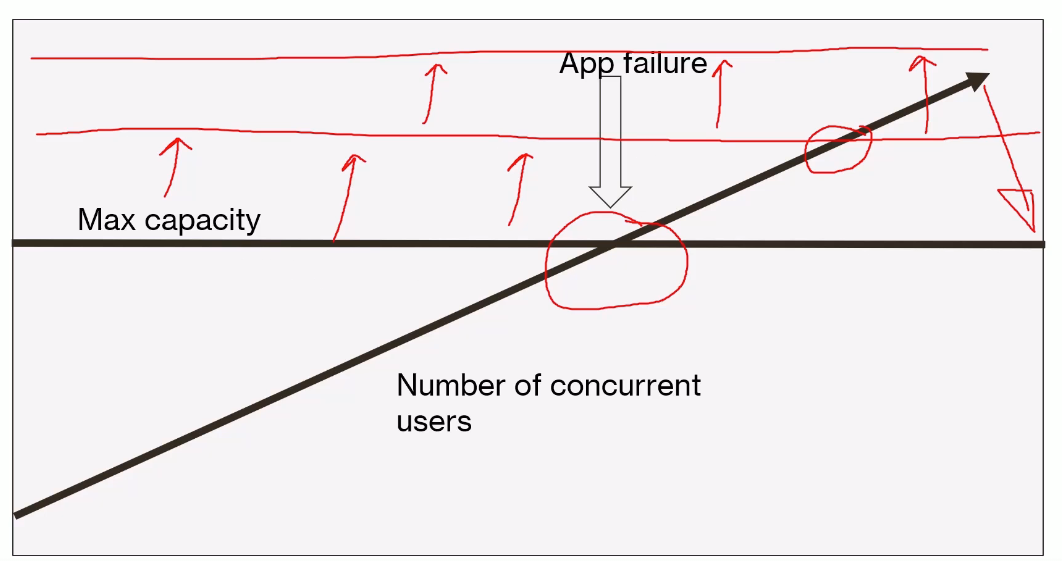
**Mitigation Strategies:**

* **Gradual Deployment:** Deploy updates in stages to minimize risks.
* **Redundancy:** Ensure every component has a backup to avoid single points of failure.
* **Health Monitoring:** Use real-time monitoring to detect and address issues quickly.
* **Disaster Recovery Plan:** Have a tested plan in place to recover from outages quickly.
* **Security Practices:** Implement strong security measures to prevent cyber-attacks.
* **Geographical Distribution:** Deploy applications in multiple regions to withstand location-based disruptions.
* **Load Testing:** Test the system’s capacity to handle high traffic and make necessary optimizations.

**2.Scalability**

Scalability refers to the ability of a system to handle increasing or decreasing demand by adding or removing resources.

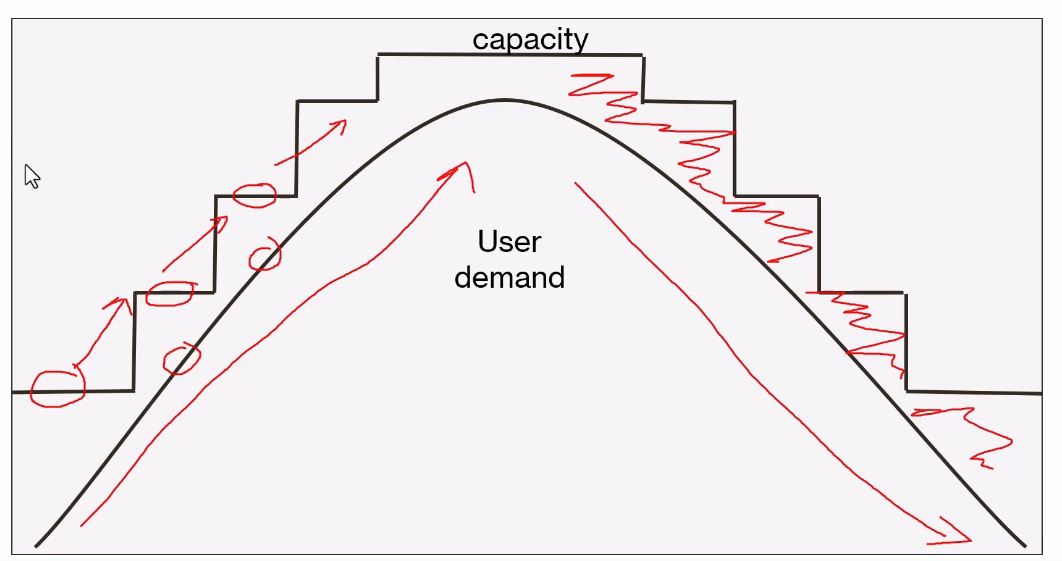
1. **Adaptation to Fluctuating Traffic:** Scalable systems can easily handle variations in traffic, such as high demand during peak times and lower demand during off-peak times. This is particularly useful for businesses with seasonal traffic patterns, like e-commerce sites during holidays or tax systems during tax season.
2. **Types of Scaling:**
   1. **Vertical Scaling (Scaling Up/Down):** This involves increasing or decreasing the capacity of a single server by adding more memory, CPUs, or other resources. While vertical scaling is straightforward and can be done easily in cloud environments, it has an upper limit. Once the maximum capacity of the server is reached, no further scaling can be done.
   2. **Horizontal Scaling (Scaling Out/In):** This involves adding more servers to distribute the load. Horizontal scaling has no real limits, allowing for the addition of as many servers as needed. It also improves availability, as the load is spread across multiple servers, reducing the risk of downtime.
3. **Impact on Costs:** Scaling, whether vertical or horizontal, affects costs. Vertical scaling typically doubles the cost as resources are doubled, while horizontal scaling increases costs linearly with the number of servers added. However, cloud computing allows for cost optimization by enabling the reduction of resources during periods of lower demand.
4. **Zero Waste Concept:** In traditional self-hosted environments, resources are often over-provisioned to accommodate future growth, leading to inefficiencies. In cloud environments, scalability allows for just-in-time resource allocation, minimizing waste and optimizing costs.

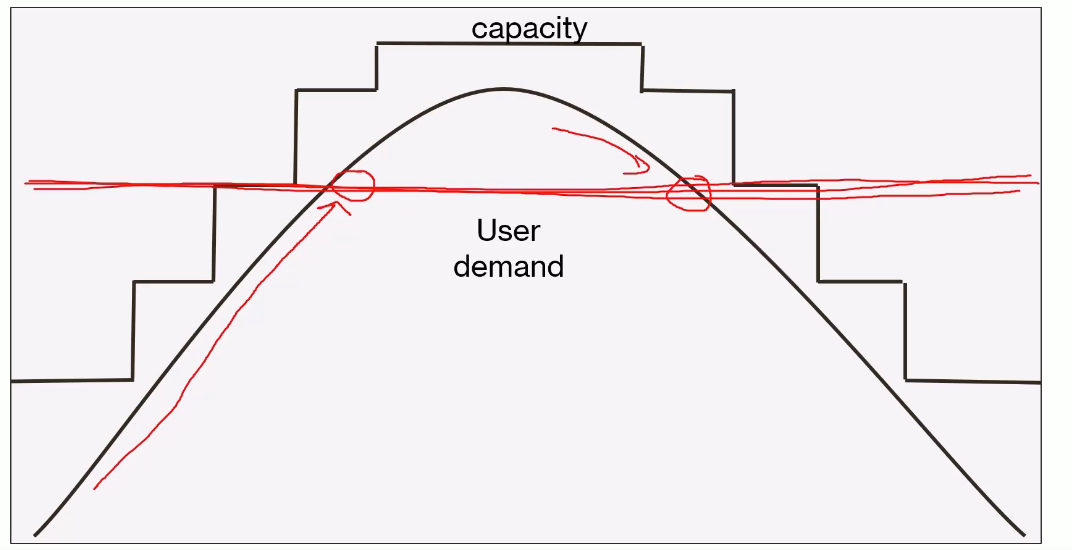


**3.Elasticity**

Elasticity is a crucial feature of cloud computing that refers to the ability of a system to automatically adjust its resources to match the current demand. This means that a system can quickly and easily scale up or down, depending on the workload, ensuring that it has just the right number of resources at any given time.

1. **Automated Resource Adjustment:** Elasticity involves automation, where the system itself detects changes in demand and adjusts resources accordingly. This is often referred to as **auto-scaling**. Auto-scaling allows the system to monitor specific metrics, like CPU utilization or memory usage, and automatically add or remove resources when certain thresholds are reached.
2. **Efficiency and Cost-Effectiveness:** Elasticity is more efficient and cost-effective compared to static resource allocation. In traditional self-hosted environments, resources are often over-provisioned to handle potential future growth, leading to waste. Elasticity minimizes waste by allocating resources dynamically, ensuring that you're only paying for what you actually use.
3. **Reducing Computing Waste:** In cloud environments, elasticity reduces computing waste by scaling resources up during peak demand and scaling them down during low demand. This dynamic adjustment means you avoid paying for unused resources, optimizing your costs and making your system more efficient.
4. **Preventing System Overload:** Elastic systems help prevent application failures by scaling up resources to meet increasing user demand. In contrast, systems with fixed capacity may fail when demand exceeds their maximum capability, leading to slow performance or downtime. Elasticity ensures that your system can handle fluctuations in demand without degrading performance.
5. **Dynamic Capacity vs. Fixed Capacity:** A system with elasticity dynamically adjusts its capacity to meet user demand, reducing waste and avoiding the risk of overload. In contrast, a system with fixed capacity may either over-provision (leading to waste) or under-provision (leading to potential failures during peak demand).





**4.Reliability**

**Reliability** in cloud computing refers to the ability of a system to consistently perform its intended function without interruption and with a high degree of accuracy.

1. **Distinction from Availability:** While availability focuses on whether a system is accessible and responsive to user requests, reliability ensures that the system is actually doing what it's supposed to do. For example, a calculator app that is always accessible (high availability) but consistently provides wrong answers would be considered unreliable.
2. **Dependability and Trust:** A reliable system builds trust with users. When you interact with a cloud service, you expect it to perform correctly every time. Reliability involves the system operating as expected, handling data correctly, and maintaining consistency across operations. Users must be confident that their data is secure, their computations are correct, and their transactions are processed without error.
3. **Reliability Mechanisms in Cloud Computing:**

* **Auto-Scaling**
* **Global Reach and Redundancy**
* **Data Backups and Redundancy**
* **Health Monitoring and Probes**

**5.Predictability**

**Predictability** in cloud computing refers to the ability to forecast and control the performance, behaviour, and costs of a system over time. This concept is vital because it gives businesses the confidence that their applications will perform consistently, and they won't face unexpected financial surprises.

How Cloud Platforms Enable Predictability:

1. Auto-Scaling
2. Load Balancing
3. Instance and Tier Selection
4. Cost Management Tools
   1. Cost Analysis
   2. Budgeting Features
   3. APIs for Budgeting
   4. Pricing Calculators

**6.Security**

**Shared Responsibility:**

* **Cloud Provider's Role:** The cloud provider secures the infrastructure, including physical data centres and network layers.
* **Customer's Role:** As a customer, you're responsible for securing your data and applications. This includes managing who can access your resources, encrypting data, and keeping your systems up to date.

**Ensure Security:**

* **Compliance:** Cloud providers meet international security standards, ensuring they follow best practices.
* **Security Teams:** Providers have dedicated teams that monitor their networks for threats and respond quickly to any issues. (MSRC)
* **Denial of Service (DoS) Protection:** Cloud platforms automatically protect against attacks that try to overwhelm your network with traffic.
* **Policies and Governance:** Tools like Azure Policy help you enforce security rules across your environment, such as requiring encryption or strong passwords.
* **Access Management:** Azure Active Directory and role-based access control (RBAC) help you control who can access your resources, reducing the risk of unauthorized access.
* **Automatic Updates:** For some services, the cloud provider automatically applies security updates, so you're always protected with the latest security measures.
* **Data Protection:** Cloud providers encrypt your data both when it's stored (at rest) and when it's being transmitted (in transit), ensuring it's secure.
* **Firewall:** Azure Firewall is an optional service that adds an extra layer of protection by controlling the flow of network traffic.

**7.Governance**

**Governance** is an essential aspect of using cloud services, closely tied to security. It involves setting rules and policies that guide how your organization operates in the cloud, ensuring that everything aligns with your business goals and compliance requirements.

Governance is Important:

* **Policy Enforcement:** Governance ensures that your organization's policies, such as security standards or cost limits, are consistently applied across all cloud operations.
* **Compliance:** Many industries have specific regulations, like HIPAA for healthcare or GDPR for data protection in the EU. Proper governance helps ensure your cloud environment meets these legal requirements.
* **Azure Policy and Blueprint:** These tools allow you to enforce company policies across all your cloud resources. For example, you can require certain metadata tags on resources or enforce specific security settings.
* **Role-Based Access Control (RBAC):** RBAC lets you define who can access what within your cloud environment. This ensures that only authorized personnel can interact with critical resources.
* **Management Groups:** Azure allows you to organize multiple subscriptions into hierarchical groups, making it easier to apply policies across your entire organization.
* **Soft Deletes:** Features like soft deletes in storage accounts prevent data from being permanently deleted immediately, protecting against accidental or malicious deletions.

**8.Manageability**

**Manageability** in cloud computing refers to how easily you can manage both your applications and the cloud environment itself.

* **Management of the Cloud:**
  + Automation & Monitoring: Tools like auto-scaling, health checks, and alerts help manage applications efficiently without constant manual effort.
* **Management in the Cloud:**
  + Resource Management: Azure offers web portals, CLI, scripting (PowerShell/Bash), and REST APIs to control and configure resources.
  + Azure Tools: Azure Portal, Cloud Shell, and Azure Monitor simplify resource management, monitoring, and scaling.

**CLOUD SERVICE TYPES**

**TYPES:**

**1. Infrastructure as a Service (IaaS):**

* Provides essential IT services like computing, storage, and networking.
* **Features:**
  + You rent infrastructure instead of owning it.
  + Services include virtual machines (VMs), storage (e.g., Azure Storage), and networking (e.g., virtual networks).
  + **Example:** Azure Virtual Machines (VMs) where you can choose specific CPU types, RAM sizes, and performance optimizations.
  + Pay only for what you use, down to the second for VMs.
  + **Storage:** Azure Storage offers up to five petabytes for various file types (blobs, files, queues, tables) at a low cost.
  + **Networking:** Virtual networks are free, but bandwidth (ingress and egress) incurs costs.

**2. Platform as a Service (PaaS):**

* Builds on IaaS by adding a managed software layer, simplifying development and deployment.
* **Features:**
  + Provides infrastructure plus middleware, development tools, and deployment tools.
  + **Example:** Azure App Services allows you to upload your code and configuration, and Azure handles the infrastructure.
  + Includes features like auto-scaling, continuous integration/deployment, and container support.
  + **Storage:** Managed storage and database services like Azure SQL Database are included.
  + **Networking:** Services like Azure Front Door, Load Balancer, and Firewall

**3. Software as a Service (SaaS):**

* Fully developed applications ready to use without managing underlying infrastructure.
* **Features:**
  + The application is ready to go after basic setup, with no need for additional configuration.
  + **Examples:** Office 365, OneDrive, Skype, Dropbox, Google Drive, and Google Docs.
  + SaaS applications are typically accessible via a web interface with an account and password.

**Serverless**

**What is Serverless?**

Serverless means you don't manage the underlying servers. The infrastructure is abstracted, and you pay based on service usage rather than specific hardware.

**Pricing Models:**

* **Platform as a Service (PaaS):**
  + **Example:** SQL Database using DTU pricing, where you select a performance level. Alternatively, V-Core pricing lets you specify the exact number of CPU cores, memory, and storage.
* **Serverless:**
  + **Example:** Serverless SQL Database, where Azure handles scaling based on demand, and you pay per CPU second. Costs are dynamic and can be lower for infrequent usage, as it can scale down to zero when idle.

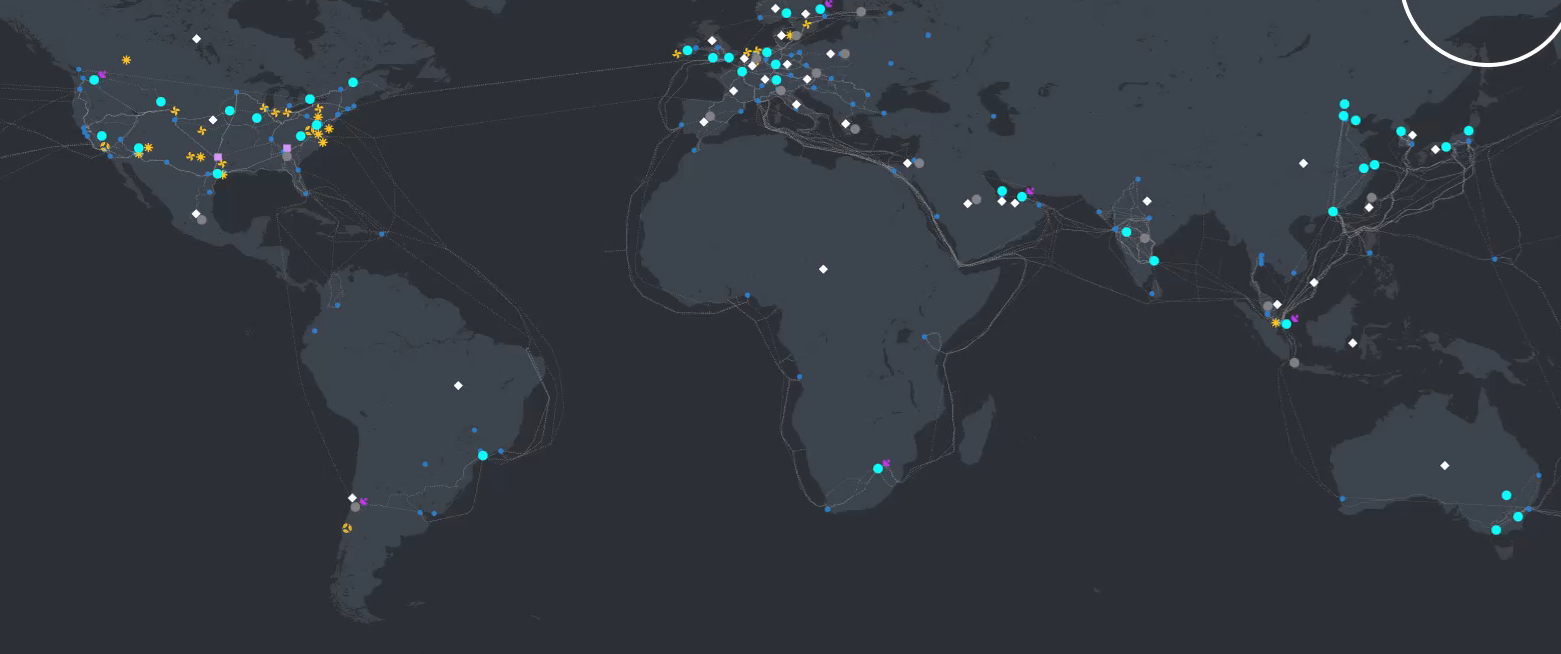
**Examples of Serverless Services:**

* **Azure Functions:** Offers a generous free tier and automatically scales based on execution demand, charging only for what you use.
* **Cost Efficiency:** Serverless can save money, especially for variable workloads. However, the cost can be less predictable due to its dynamic nature.

**CORE ARCHITECTURAL COMPENENTS OF AZURE**

**Core Azure Architectural Components:**

* + **Regions & Region Pairs:** Azure has over 60 regions worldwide. Regions are clusters of data centres, often paired with another for faster connectivity and safe updates.
  + **Availability Zones & Data Centres:** Fundamental to building resilient solutions in Azure.



**Region Pairing:**

* Regions are paired for efficient data backup and rapid updates. Choosing a region pair is advantageous for high availability and disaster recovery.

**Global Presence:**

* Azure's regions are spread across the globe, with dense concentrations in North America and Europe. Some regions are coming soon, reflecting Azure’s growth.

**Sovereign Regions:**

* **U.S. Government, Secret, and Top-Secret Clouds:** These are separate from Azure's public cloud, meeting stringent security and compliance standards. Access is restricted to eligible entities.
* **China:** Operates under a separate agreement, requiring a Chinese presence to use its services.

**Availability Zones in Azure**

**What Are Availability Zones?**

**Availability Zones** are physically separate data centres within a region, ensuring high availability by being independent in power, cooling, and networking.

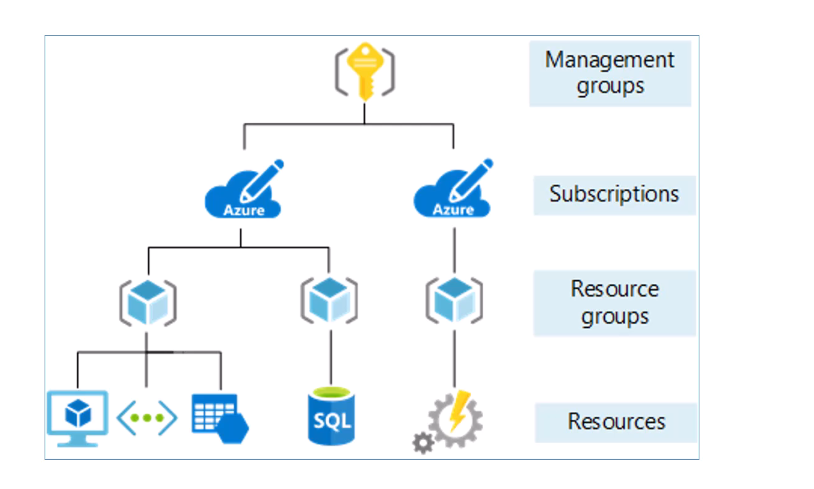
* **Separation:** Zones can be multiple buildings on the same site or miles apart, connected by high-speed links (around 5 milliseconds).

**Availability Zones in Regions:**

* **Selective Support:** Not all regions have Availability Zones. For instance, Canada Central does, while Canada East does not.
* **Regions with Zones:** Many regions in the Americas, Europe, the Middle East, Africa, and Asia Pacific now support these zones, with more being added over time.

**Service Types and Zone Support:**

* **Zonal Services:** Let you choose specific zones within a region for resource deployment, providing redundancy against localized failures.
* **Zone-Redundant Services:** Automatically distribute resources across zones, like Azure Cosmos DB and Azure SQL Database.
* **Always Available Services:** Global services not tied to any specific region or zone, like Azure Portal and Azure Active Directory, designed to remain available even if an entire zone or region goes down.

****

**Resources and Resource Groups**

**1. Azure Resources:**

**Resources** are individual Azure services, such as virtual machines, storage accounts, or databases. They are instantiated and accessible services within Azure.

* **Creation:** Resources can be created through the Azure portal, command line, PowerShell, ARM templates, or Bicep. Naming conventions are recommended for easy identification.
* **Location:** Resources need to be deployed in a specific region. This is a geographic aspect of where the resource runs.

**2. Resource Groups:**

A **resource group** is a container for organizing and managing related resources. It groups resources that are related to a project or application.

* **Flexibility:** Resources in a group can be from different regions. The key is to group resources that are logically related, often managed or deleted together.
* **Permissions:** You can set permissions (read, write, delete) at the resource group level. However, resource groups do not provide additional security boundaries; resources within different groups can interact unless restricted by other security measures.

**Azure Subscriptions**

A **subscription** in Azure is essentially a billing unit that manages costs associated with resources and services. Every subscription has a payment method linked to it, like a credit card or a corporate account.

* **In Hierarchy:** Resources belong to resource groups, and resource groups are organized under subscriptions.

**Types:**

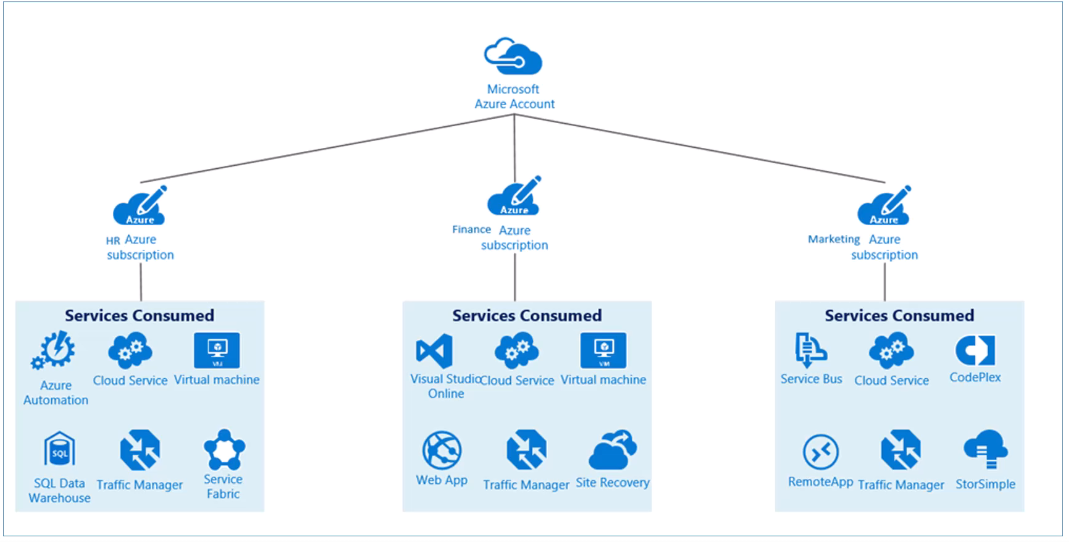
* **Free Plan:** $200 credits for 30 days, free services for 12 months.
* **Pay-As-You-Go:** Monthly billing based on usage.
* **Enterprise & Special Plans:** Includes business agreements, MSDN, student plans.

**Multiple Subscriptions:**

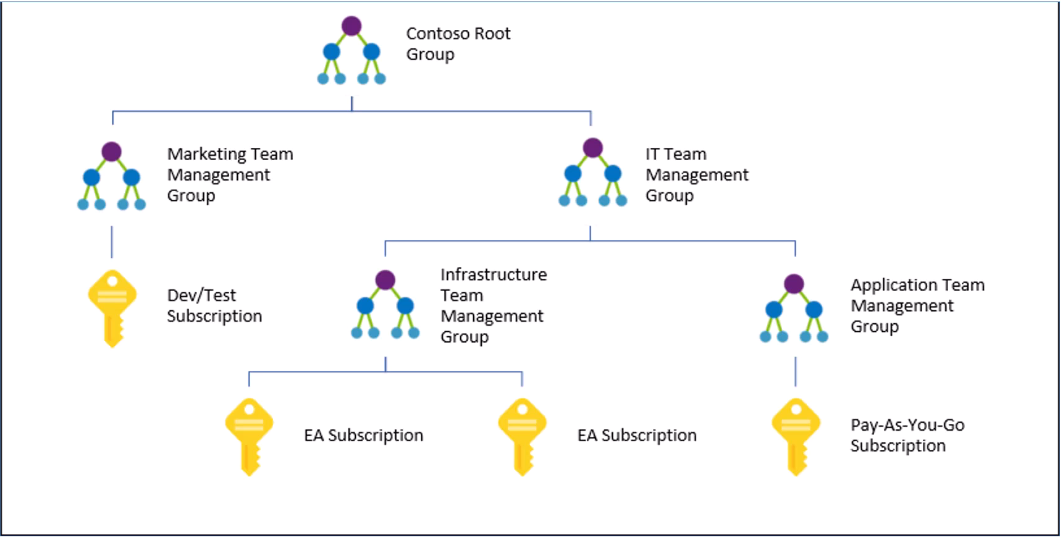
* **User Roles:** Users can have different roles across subscriptions.
* **Organizational Use:** Separates departments or regions for billing and security.

**Security:** Can serve as a boundary, isolating resources between departments (e.g., finance vs. marketing).

**Flexibility:** Single or multiple subscriptions can be used depending on organizational needs.



**Azure Management Groups**

****

**Centralized Management:** Manage subscriptions with a unified approach, applying policies and security controls across multiple subscriptions.

1. **Azure Policies/Blueprints:** Assign resource groups, policies, and ARM templates to subscriptions via management groups to enforce governance and security across all associated subscriptions.
2. **Security Enhancement:** Management groups provide an additional layer of security and policy enforcement across multiple subscriptions within an organization.

**AZURE COMPUTE AND NETWORKING SERVICES**

Azure offers various compute services like Virtual Machines, App Services, and more. Networking services include Virtual Networks, VPNs, and DNS.

**Core Pillars:** Compute, Networking, Storage, and Database services are foundational in Azure.

**Compute Services:**

* Execute code in the cloud for apps, websites, and other services.

**Types:**

* **Virtual Machines (VMs):** Analogous to physical servers, providing full control over the OS. VMs are Infrastructure as a Service (IaaS).
* **Scale Sets:** Abstractions on top of VMs for scaling.
* **Web Apps:** Cloud-native services for running applications.
* **Container Services:** Includes Azure Container Instances, Container Apps, and Kubernetes.
* **Azure Virtual Desktop:** Allows remote access to desktops as a cloud service.

**Virtual Machines (VMs):**

* **Analogy:** Like owning a detached house, offering complete control.
* **Virtualization:** VMs are related to renting an apartment in a building, sharing some services with others but maintaining private control.
* **Infrastructure as a Service (IaaS):** VMs are slices of physical machines, supporting both Windows and Linux OS.
* **AWS Comparison:** Azure VMs are similar to AWS EC2 instances.
* **Predefined Images:** Over 700 VM types available to meet various needs.

**Scaling Virtual Machines**

**1. Scaling Basics:**

* **Scaling:** Allows applications to handle increased traffic as demand grows.
* **Two Types:**
  + **Scaling Up:** Increase the size of the VM instance (e.g., from 4 CPUs to 8 CPUs).
  + **Scaling Out**: Increase the number of VMs working together, often preferred for handling larger loads.

**2. Scaling Up:**

* Upgrading a VM to a higher CPU version within the same family.
* **Limitations:** Maximum typically around 120 CPUs per instance.

**3. Scaling Out:**

* Adding more VMs to distribute the load.
* **Benefits**: No practical limit on the number of VMs; allows for flexibility and better resource management.

**4. Virtual Machine Scale Sets (VMSS):**

* A predefined group of VMs that can scale automatically based on demand.
* **Scaling Triggers:**
  + **CPU Utilization:** Automatically adds VMs when CPU load increases.
  + **Scheduled Scaling:** Adjusts the number of VMs based on predefined time schedules.
  + **Other Triggers:** Queue length, message queues, etc.
* **Elasticity:** Supports cloud computing's need to dynamically adjust resources.

**5. Load Balancer:**

* **Purpose:** Distributes traffic evenly across VMs in the scale set, ensuring efficient use of resources.

**6. Scale Set Limits:**

* **VM Count:** Typically supports up to 100 VMs per scale set, configurable to handle up to 1,000 VMs.
* **Expansion:** Additional scale sets can be added if needed, with a load balancer in front to manage traffic.

**7. Cost Efficiency:**

* **Dynamic Scaling:** Saves money by scaling down during low demand periods and scaling up when necessary.

**App Services (Web Apps):**

* **Platform as a Service (PaaS):** Unlike VMs, you don’t control the server; you just upload your code and configuration.
* **Performance Promise:** Azure promises a specific performance level without revealing the underlying server details.
* **Developer Benefits:** Includes integrations with GitHub, A/B testing, deployment slots, etc.
* **Limited Access:** You can access files and logs but can't modify server settings or install arbitrary software.

**Containers:**

* **Container Concept:** Containers package all necessary files and libraries with the code, creating a portable image.
* **Deployment Flexibility:** The same container image can be deployed across various environments (staging, production, etc.).
* **Azure Container Options:**
  + **Azure Container Instances:** Quick deployment for single containers; minimal scaling.
  + **Azure Container Apps:** More features than basic web services but less complex than Kubernetes; ideal for mid-tier applications.
  + **Azure Kubernetes Service (AKS):** Enterprise-grade, cluster-based container orchestration with advanced features.

**Azure Virtual Desktop:**

* **Virtualized Windows Desktop:** Your Windows desktop runs in the cloud, accessible from any device.
* **Consistency:** Same files and settings regardless of where you log in (office, home, mobile).
* **Cross-Device Access:** Accessible via desktop, iOS, or Android.

**Azure Functions**

Azure functions are designed for running small, focused pieces of code in the cloud. They are ideal for tasks that don't require a full server or application setup. Here’s how they work:

* **Triggers:** Each function is triggered by a specific event, like an HTTP request, a new file in Blob storage, or a timer that runs at scheduled intervals.
* **Consumption-Based Model:** You only pay for the compute power used when your function is actually running. There’s even a generous free tier that offers up to 1 million executions per month at no cost.

**Examples of Azure Functions:**

1. **Daily Summary Report:** Imagine a function that runs every day at midnight, analyses data from the previous day, and updates a report. It’s a small task that doesn’t need a dedicated server.
2. **File Processing:** A function could monitor a Blob storage container. Whenever a new file is added, the function could automatically compress the file or create a backup.
3. **Weather API Fetcher:** Say you have a weather widget on your website. You could use a function to call an external weather API every six hours to update the weather data on your site.

**Virtual Networks (VNets)**

* VNets are like your own private network within Azure, allowing you to set up IP address ranges and control traffic between your virtual machines and other resources.
* **Address Space:** When you create a VNet, you define a range of private IP addresses (IPv4 or IPv6) for it. This address space is private, meaning it isn’t accessible from the internet.
* **Private vs. Public IPs:** VNets use private IP addresses. Public IP addresses can be assigned to VMs for internet access, but this is typically not recommended for security reasons.

**Subnets**

* Subnets are subdivisions of a VNet’s address space. They help organize and segment the network.
* **Usage:** Each subnet has its own range of IP addresses within the VNet. You can have multiple subnets within a single VNet, each with unique address ranges.

**Network Interface Cards (NICs)**

* NICs connect VMs to subnets. Each VM must have at least one NIC.
* **Multiple NICs:** VMs can have more than one NIC, allowing them to connect to multiple subnets and have multiple private IP addresses.

**Network Security Groups (NSGs)**

* NSGs act as a firewall, controlling inbound and outbound traffic to and from subnets or individual VMs.
* **Rules:** NSGs use rules to allow or deny traffic based on factors like source IP, destination IP, source port, destination port, and protocol (TCP/UDP).
* **Priorities:** Rules are evaluated based on priority, with the highest priority rule applied first. You can configure rules to allow or deny traffic as needed.

**Peering, Azure DNS and VPN Gateway:**

**Peering**

**What is Peering?** Peering allows communication between virtual machines (VMs) or subnets in different virtual networks (VNets).

**Why Use It?** By default, VNets can't talk to each other. Peering connects them, enabling cross-network communication.

**Example:** If you have a WebNetwork and a DatabaseNetwork, peering allows a VM in WebNetwork to access a VM in DatabaseNetwork.

**Azure DNS**

**What is Azure DNS?** Azure DNS provides domain name services for internal Azure resources.

**Private DNS:** You can assign easy-to-remember names (like webserver.local) to private IP addresses within Azure.

**Public DNS:** You can also use public domains but only within Azure unless you set up VPN or modify local hosts files.

**Example:** Instead of remembering IP addresses, you can refer to internal services using names like database.local.

**VPN (Virtual Private Network)**

**What is a VPN?** A VPN creates a secure, encrypted connection between two networks or between a user and a network.

**Types:**

* + Point-to-Site (P2S): Connects individual devices (like a laptop) to an Azure network.
  + Site-to-Site (S2S): Connects entire networks (like office networks) to Azure.

**Example:** If you work from home, a VPN allows you to securely connect to your company’s network as if you were physically there.

**ExpressRoute**

**What is ExpressRoute?** ExpressRoute provides a private, high-speed connection between your on-premises network and Azure, bypassing the public internet.

**Benefits:** It’s faster and more secure than a VPN because it avoids the public internet altogether.

**Example:** Large enterprises needing high-speed and secure connections to Azure might use ExpressRoute to link their data centres directly with Azure.

**Public and Private Endpoints**

**Public Access:**

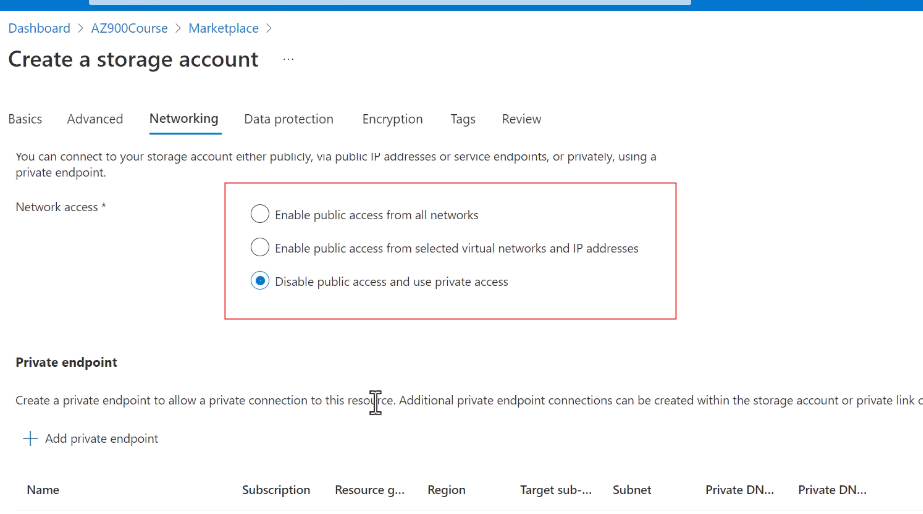
Even though public access is enabled, users still need proper credentials (like access keys) to access the resource. However, the resource has a door open for anyone to try and enter from the public Internet.

**Public Access from Selected Virtual Networks:**

Provides an extra layer of security by ensuring only devices within the selected VNets can access the resource, even though it still has a public endpoint.

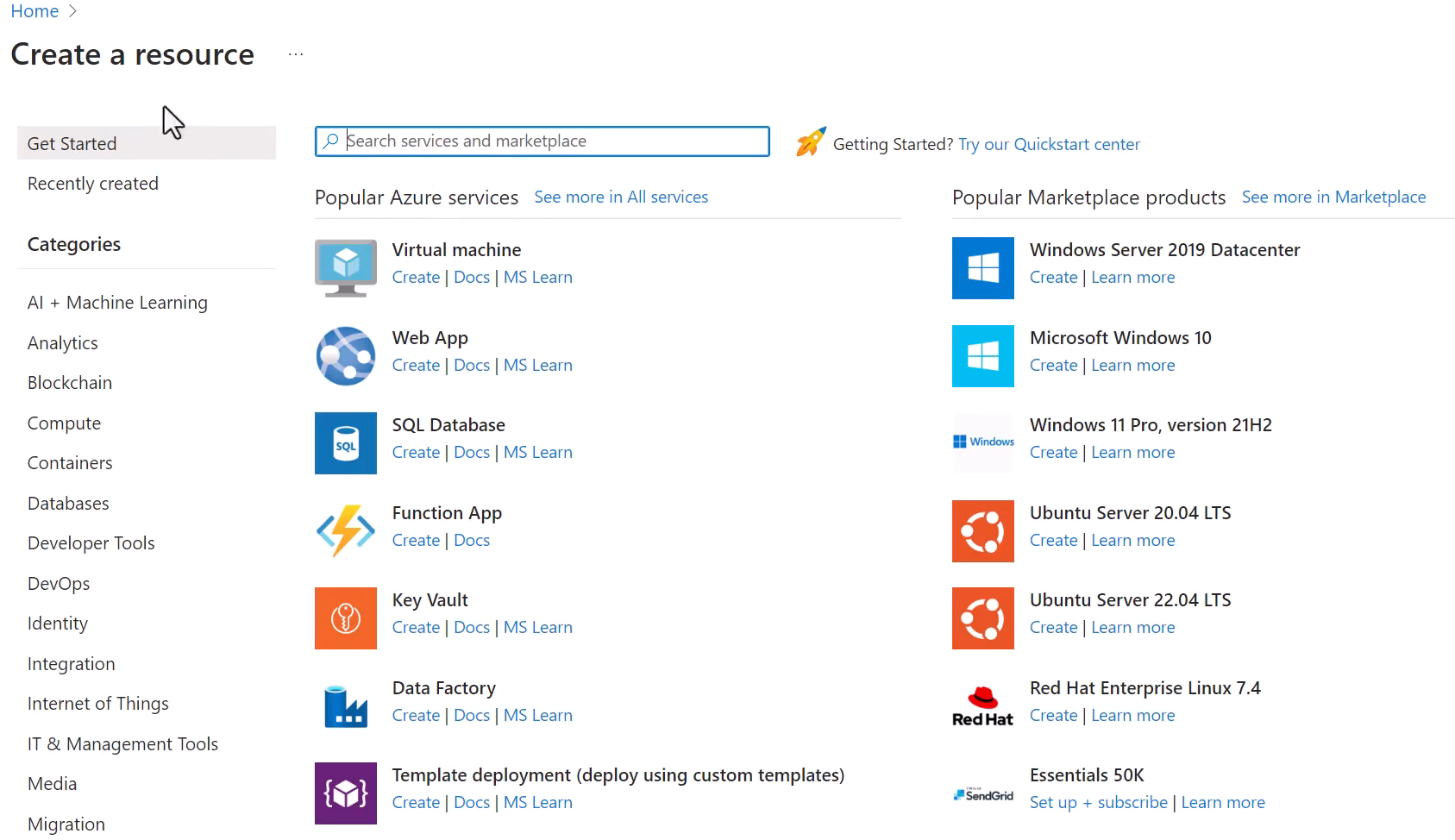
**Private Endpoints:**

Ensures that the resource can only be accessed privately within a specific VNet, making it inaccessible from the public Internet.

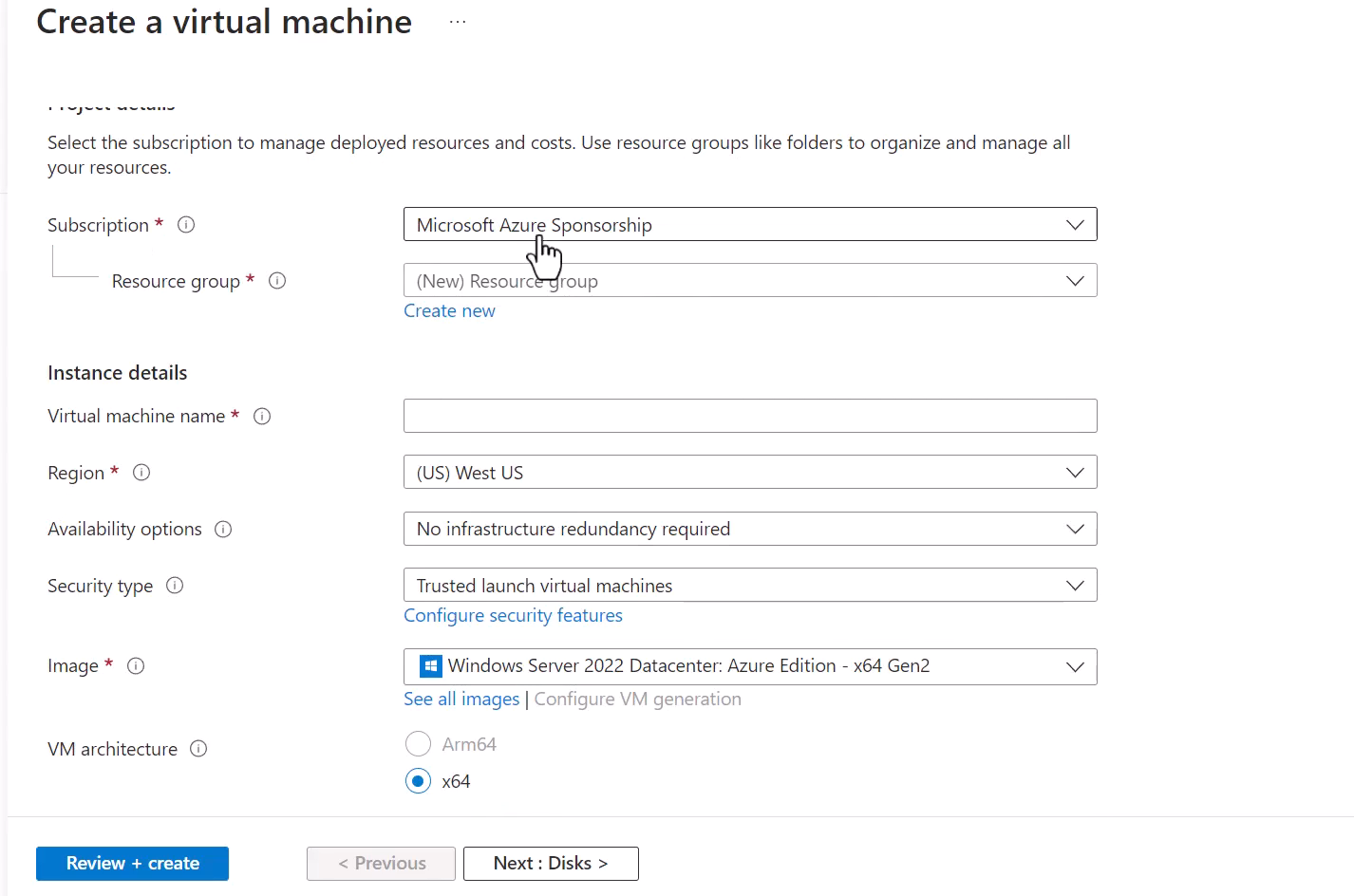


**CREATING A VIRTUAL MACHINE**

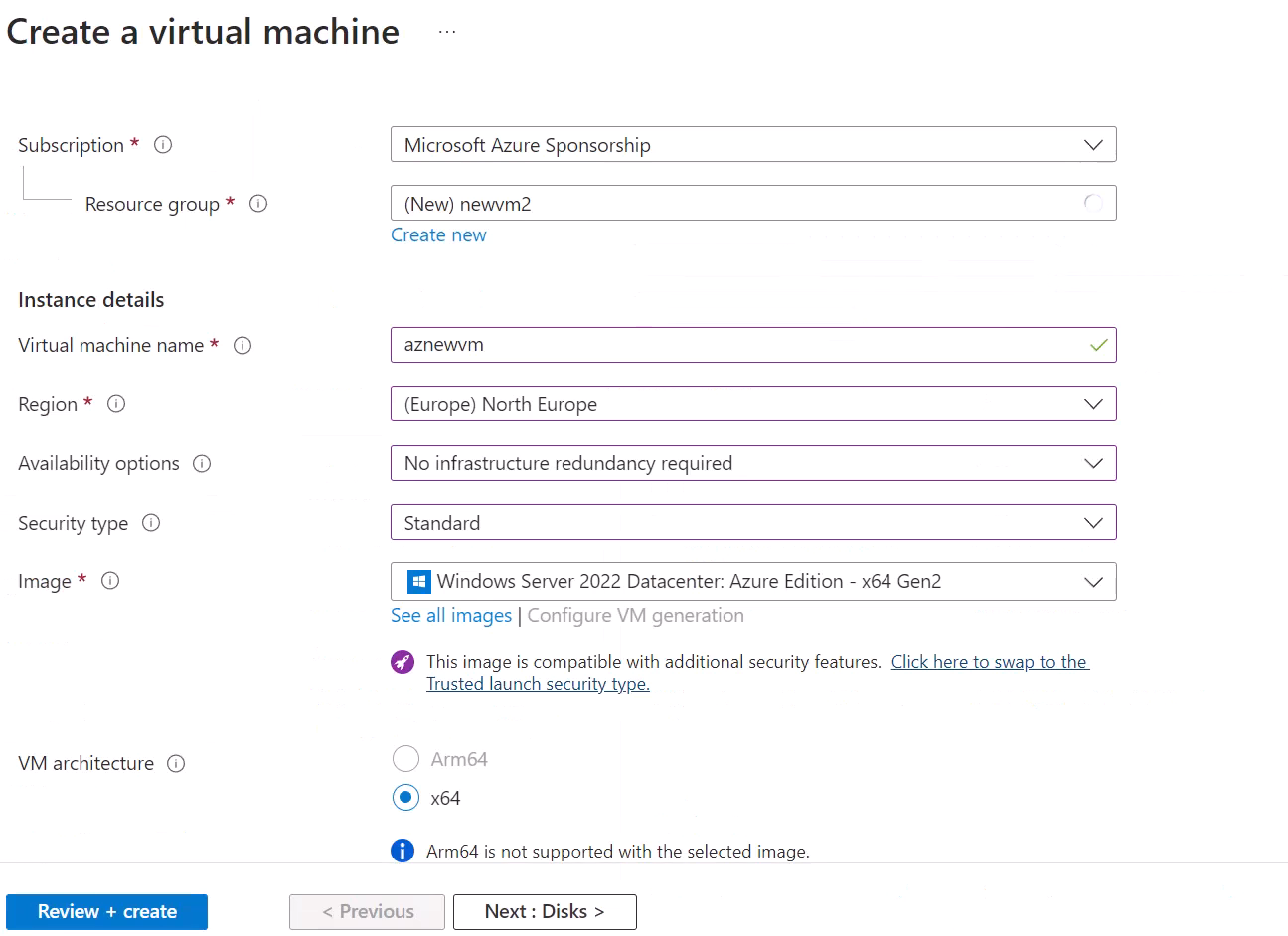
Step1: Click on the create resource + icon:

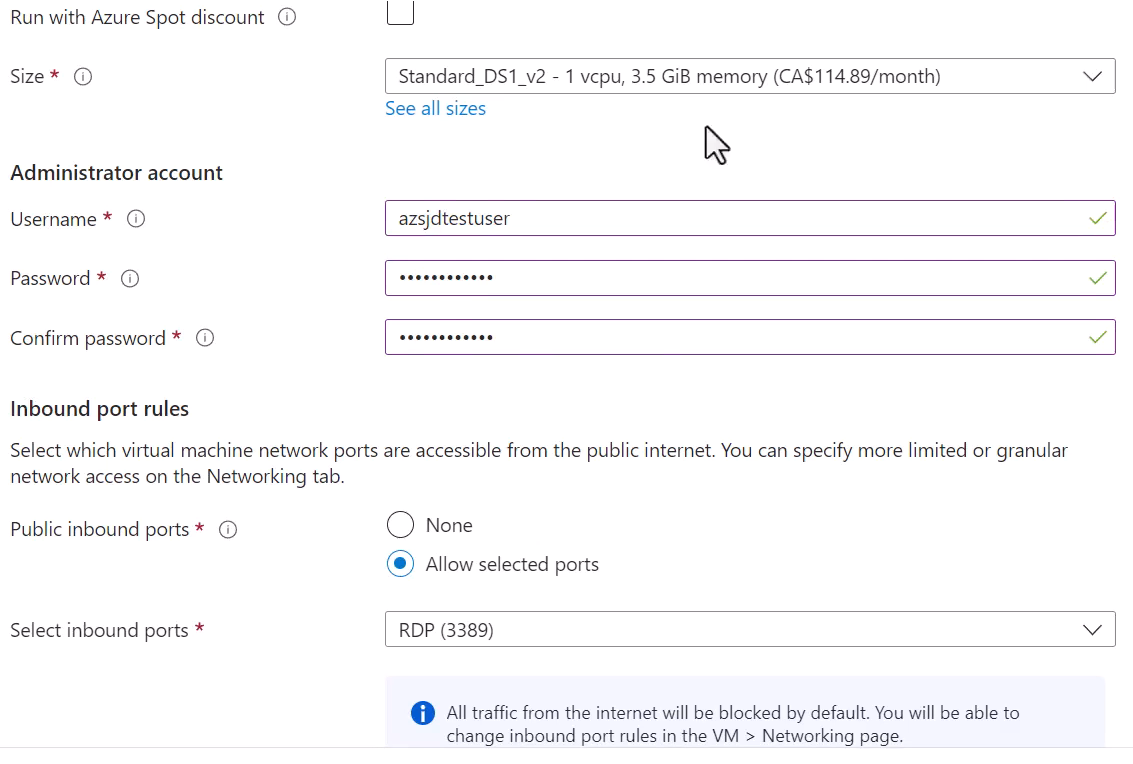


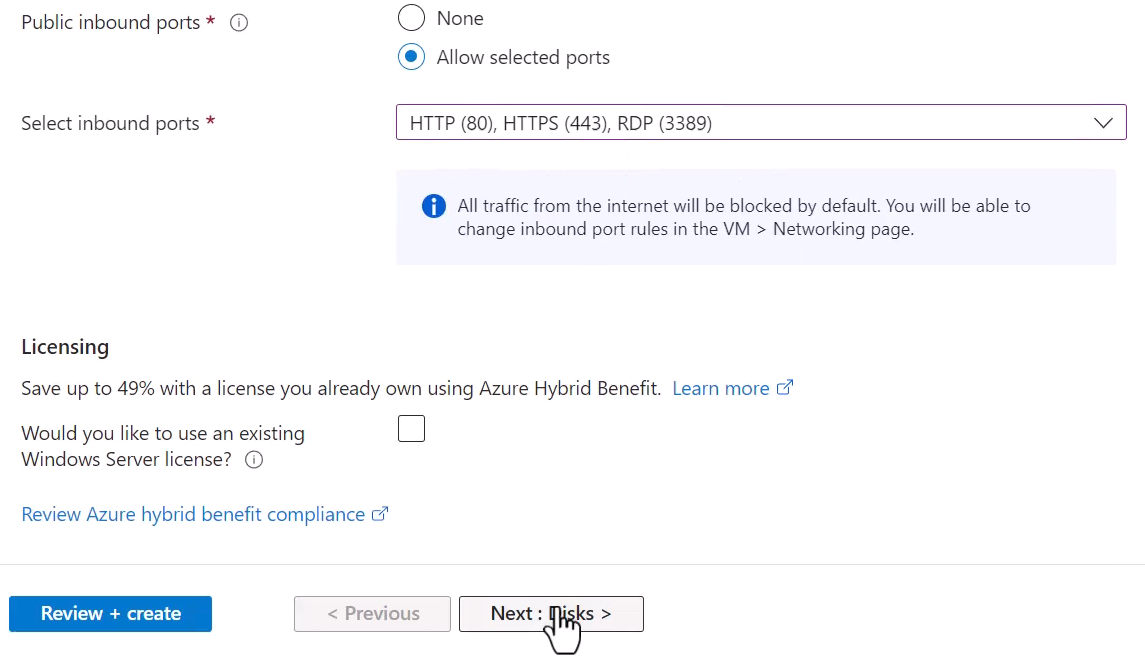
Step2: click the create under the virtual machine



Step2: choose the requirements of VM







Step3: Skip all the options and create the virtual machine

