**1. What is a Virtual Network (VNet) in Azure?**

**Definition**

An **Azure Virtual Network (VNet)** is a logically isolated network in the cloud, similar to a traditional on-premises network, but with the scalability and flexibility of Azure.

**Real-World & Realtime Examples**

* **Example 1 (Enterprise Scenario):** A company migrates its on-premises servers to Azure. They create a **VNet** to host their **web servers (frontend)**, **application servers (middleware)**, and **database servers (backend)**—just like their physical data center but in the cloud.
* **Example 2 (Hybrid Cloud):** A retail business keeps its **Point-of-Sale (POS) systems** on-premises but connects them securely to Azure VNet for **inventory management** hosted in the cloud.
* **Example 3 (Multi-tier Applications):** A SaaS provider uses **multiple VNets** for different customers (isolated environments) while maintaining security and compliance.

**2. Why is a Virtual Network Needed?**

**Key Reasons:**

1. **Isolation & Segmentation:** Prevents unauthorized access between resources.
2. **Secure Communication:** Enables private connectivity (without public internet).
3. **Hybrid Connectivity:** Connects to on-premises via **VPN or ExpressRoute**.
4. **Traffic Control:** Controls inbound/outbound traffic using **NSGs, Firewalls, and UDRs**.
5. **Integration with Azure Services:** Connects to **Azure VMs, Kubernetes (AKS), App Services (with VNet Integration), and PaaS services (Private Endpoints).**

**Example:**

A bank cannot expose its **core banking database** to the public internet. A **VNet** ensures only authorized internal applications can access it.

**3. How to Use Virtual Network Effectively?**

**Best Practices:**

✔ **Plan IP Addressing Carefully** (Avoid overlapping with on-premises networks).  
✔ **Use Subnets for Segmentation** (e.g., Web, App, DB tiers).  
✔ **Enable Network Security Groups (NSGs)** for traffic filtering.  
✔ **Use Private Endpoints for PaaS Services** (Avoid public exposure).  
✔ **Implement Hub-Spoke Model** for large enterprises (Centralized security in Hub VNet).  
✔ **Monitor with Azure Network Watcher** (Diagnose connectivity issues).

**Example:**

A healthcare provider:

* **Web Tier (Public Subnet)** – Hosts patient portal (accessible via internet).
* **App Tier (Private Subnet)** – Processes medical data (no public access).
* **DB Tier (Private Subnet + NSG)** – Stores sensitive patient records (only allows App Tier).

**4. What is CIDR? (Classless Inter-Domain Routing)**

**Definition:**

CIDR is a method for allocating IP addresses and routing IP packets more efficiently than traditional class-based (A, B, C) networks.

**Format:**

<Base IP>/<Prefix Length>

* **Example:** 10.0.0.0/16 means:
  + **Network IP:** 10.0.0.0
  + **Subnet Mask:** 255.255.0.0
  + **Usable IPs:** 10.0.0.1 to 10.0.255.254 (65,534 hosts)

**Examples:**

| **CIDR Block** | **Subnet Mask** | **Usable IPs** | **Use Case** |
| --- | --- | --- | --- |
| 192.168.1.0/24 | 255.255.255.0 | 254 | Small office |
| 10.0.0.0/16 | 255.255.0.0 | 65,534 | Enterprise VNet |
| 172.16.0.0/12 | 255.240.0.0 | 1M+ | Large orgs |

**5. Why is CIDR Needed?**

* **Efficient IP Allocation:** Prevents wastage (unlike Class A/B/C).
* **Flexible Subnetting:** Allows variable-sized networks.
* **Route Aggregation:** Reduces routing table size in large networks.

**Example:**

An Azure VNet with 10.0.0.0/16 can be split into:

* 10.0.1.0/24 (Web Tier)
* 10.0.2.0/24 (App Tier)
* 10.0.3.0/24 (DB Tier)

Without CIDR, we’d waste IPs using fixed Class B (10.0.0.0/255.255.0.0).

**6. What is a Subnet?**

**Definition:**

A **subnet** is a segmented part of a VNet, used to organize and secure resources.

**Example:**

A **VNet (**10.0.0.0/16**)** can have:

* **Subnet 1:** 10.0.1.0/24 (Web Servers)
* **Subnet 2:** 10.0.2.0/24 (Application Servers)
* **Subnet 3:** 10.0.3.0/24 (Database Servers)

**7. Why is a Subnet Needed?**

**Key Reasons:**

1. **Security Isolation:** Apply different **NSGs** per subnet.
2. **Traffic Control:** Route tables can differ per subnet.
3. **Resource Grouping:** Logical separation (e.g., Dev/Test/Prod).

**Example:**

* **Public Subnet:** Hosts **load balancers** (allows HTTP/HTTPS from the internet).
* **Private Subnet:** Hosts **SQL DB** (blocks all public traffic, only allows App Tier).

**8. What is a Security Rule in Azure?**

**Definition:**

A **security rule** defines **allow/deny** traffic based on:

* **Source/Destination IP**
* **Port (e.g., 80, 443, 3389)**
* **Protocol (TCP/UDP/ICMP)**

**Example Rule:**

| **Name** | **Priority** | **Source** | **Port** | **Destination** | **Protocol** | **Action** |
| --- | --- | --- | --- | --- | --- | --- |
| Allow-HTTP | 100 | Internet | 80 | Any | TCP | Allow |
| Deny-RDP | 200 | Any | 3389 | Any | TCP | Deny |

**9. What is a Network Security Group (NSG)?**

**Definition:**

An **NSG** is a firewall that filters traffic at **subnet or NIC level**.

**Example:**

* **Web Subnet NSG:** Allows **HTTP (80), HTTPS (443)** from the internet.
* **DB Subnet NSG:** Allows **SQL (1433)** only from App Subnet.

**10. Why is NSG Important?**

**Key Reasons:**

✔ **Micro-Segmentation:** Restrict lateral movement (e.g., DB only allows App Tier).  
✔ **Compliance:** Enforces security policies (e.g., PCI-DSS).  
✔ **DDoS Protection:** Blocks malicious traffic.

**Example:**

A **bank’s database** only allows:

* **App Servers (10.0.2.0/24) → Port 1433 (SQL)**
* Blocks **all other traffic** (including internal Azure services).

**11. What is an Application Security Group (ASG)?**

**Definition:**

An **ASG** groups VMs by **application role** (e.g., "Web Servers", "DB Servers") and applies NSG rules to them **dynamically**.

**Example:**

* **ASG Name:** WebServers
* **Members:** VM1, VM2 (running IIS)
* **NSG Rule:** Allow **HTTP (80)** from Internet to WebServers ASG.

**12. Why is ASG Important?**

**Key Benefits:**

✔ **Simplified Management:** No need to update NSGs per VM.  
✔ **Dynamic Scaling:** New VMs auto-inherit rules if tagged.

**Example:**

An **auto-scaling web farm**:

* New VMs added to WebServers ASG **automatically** get HTTP/HTTPS rules.
* No manual NSG updates required.

**Final Thoughts**

* **VNet = Foundation of Azure Networking** (Isolation, Security, Routing).
* **Subnets = Logical Segmentation** (Tier-based security).
* **NSG = Firewall** (Traffic filtering at subnet/NIC level).
* **ASG = Dynamic Grouping** (Simplifies security for scaling apps).

**1. CIDR (Classless Inter-Domain Routing) – Explained Simply**

**What is CIDR?**

CIDR is a way to **allocate IP addresses efficiently** without wasting them. It replaces the old **Class A/B/C** system, which was rigid and led to IP wastage.

**How Does CIDR Work?**

* **Format:** <Base IP>/<Prefix Length>
  + Example: 10.0.0.0/16
* **Prefix Length:** The number of **fixed bits** in the subnet mask.
  + /16 means the first **16 bits** are fixed (network part).
  + The remaining bits are for hosts.

**CIDR Examples & Breakdown**

| **CIDR Notation** | **Subnet Mask** | **Usable IPs** | **Explanation** |
| --- | --- | --- | --- |
| 192.168.1.0/24 | 255.255.255.0 | 254 IPs | Small office network |
| 10.0.0.0/16 | 255.255.0.0 | 65,534 IPs | Large Azure VNet |
| 172.16.0.0/12 | 255.240.0.0 | 1M+ IPs | Enterprise networks |

**Why CIDR?**

* **Avoids IP wastage** (unlike Class A/B/C).
* **Allows flexible subnet sizes** (e.g., /24 for small subnets, /16 for large ones).
* **Used in Azure VNets, AWS, and on-premises networks.**

**Real-World Example**

* **Company VNet:** 10.0.0.0/16 (65k+ IPs)
  + **Web Subnet:** 10.0.1.0/24 (254 IPs)
  + **DB Subnet:** 10.0.2.0/24 (254 IPs)

**2. What is a Route Table in Azure?**

**Definition**

A **route table** defines how traffic flows between subnets, VNets, and the internet.

**Default Routes (Automatically Added)**

| **Destination** | **Next Hop** | **Example** |
| --- | --- | --- |
| Same VNet | Local | 10.0.0.0/16 → Local |
| Internet | Internet | 0.0.0.0/0 → Internet |
| On-Premises (VPN/ExpressRoute) | Virtual Network Gateway | 192.168.1.0/24 → VPN Gateway |

**When Do You Need a Custom Route Table?**

* **Force traffic through a firewall** (e.g., Azure Firewall).
* **Custom routing for hybrid networks** (e.g., specific traffic to on-premises).
* **Override default Azure routing.**

**3. What is a User-Defined Route (UDR)?**

**Definition**

A **UDR** is a custom rule in a route table that **overrides Azure’s default routing**.

**Common Uses of UDR**

1. **Force traffic through a Firewall/NVA**
   * Example: All outbound internet traffic (0.0.0.0/0) goes through **Azure Firewall**.
2. **Hub-Spoke Routing**
   * Example: Traffic from **Spoke VNet** to **On-Premises** goes through **Hub VNet**.
3. **Bypass Azure Default Routing**
   * Example: Specific subnets use a **different gateway**.

**Example UDR Scenario**

* **Problem:** You want **all outbound internet traffic** from a subnet to go through **Azure Firewall** (not directly to the internet).
* **Solution:**
  + Create a **Route Table** → Add a **UDR**:
    - **Destination:** 0.0.0.0/0 (Internet)
    - **Next Hop Type:** **Virtual Appliance**
    - **Next Hop IP:** 10.0.0.4 (Azure Firewall Private IP)
* **Result:** All traffic from that subnet is forced through the firewall.

**4. Key Differences: Route Table vs. UDR**

| **Feature** | **Route Table** | **User-Defined Route (UDR)** |
| --- | --- | --- |
| **Definition** | A collection of routes | A custom route inside a route table |
| **Default Routes** | Yes (Azure adds them) | No (manually added) |
| **Use Case** | Controls traffic for a subnet | Overrides specific routes |
| **Example** | All subnets use this table | "Send all internet traffic to Firewall" |

**5. Real-World Example: Secure Hub-Spoke Architecture**

**Scenario**

* **Hub VNet:** Contains **Azure Firewall & VPN Gateway**.
* **Spoke VNet 1:** Web Servers (10.1.0.0/16).
* **Spoke VNet 2:** Database (10.2.0.0/16).

**Routing Setup**

1. **Hub Route Table:**
   * 10.1.0.0/16 → **Spoke 1 Peering**
   * 10.2.0.0/16 → **Spoke 2 Peering**
   * 0.0.0.0/0 → **Azure Firewall**
2. **Spoke 1 Route Table (Web Servers):**
   * 0.0.0.0/0 → **Azure Firewall (UDR)**
   * 192.168.1.0/24 (On-Premises) → **Hub VPN Gateway**

**Result**

* **All internet traffic** from Web Servers goes through **Firewall**.
* **On-Premises traffic** goes through **Hub VPN Gateway**.
* **DB traffic stays private.**

**6. Summary**

| **Concept** | **Key Takeaway** |
| --- | --- |
| **CIDR** | Efficient IP allocation (e.g., 10.0.0.0/16). |
| **Route Table** | Controls traffic flow for subnets. |
| **UDR** | Overrides default routing (e.g., force traffic through Firewall). |
| **Use Case** | Hub-Spoke, Firewall filtering, Hybrid networks. |