Azure Virtual Network (VNet) - Complete Documentation

Status | Not Started

Document Version: 1.0

Date: January 2026

Audience: Technical

1. Introduction to Azure Virtual Network

1.1 What is a Virtual Network?

Imagine you're building a private neighborhood in a big city. The city is Azure (Microsoft's cloud), and your neighborhood is a Virtual Network (VNet). Just like your neighborhood has its own streets, houses, and security gates, a VNet has its own private space in the cloud where your applications and data can live securely.

1.2 Simple Analogy

- **Azure Cloud** = A massive city with public spaces
- **Virtual Network** = Your gated community within that city
- **Resources** (VMs, databases) = Houses and buildings in your community
- **Subnets** = Different streets within your community
- **Security Rules** = Gates, guards, and house locks

2. Key Concepts Explained Simply

2.1 Logical Segmentation of Workloads

What it means: Separating different parts of your applications like rooms in a house.

Real-world example:

Think of a shopping mall:

- **Ground floor** = Customer-facing shops (public)
- **First floor** = Restaurants and food court
- **Second floor** = Offices and management (private)
- **Basement** = Storage and delivery area (very private)

In Azure VNet:

- **Web tier subnet** = Your website (like ground floor shops)
- **Application tier subnet** = Business logic (like first floor restaurants)
- **Database tier subnet** = Data storage (like basement storage)

Why it's important:

- Security: If someone breaks into the "ground floor," they can't access the "basement"
- Organization: Easier to manage different types of resources
- Performance: Different needs get different settings

2.2 CIDR Block Planning and IP Addressing

What it means: Giving addresses to everything in your network.

Simple explanation:

Your VNet is like a new housing development. You need to:

1. **Name the development** (VNet name)
2. **Give it a postal code range** (CIDR block)
3. **Number all the houses** (IP addresses)

CIDR Example:

text

```
VNet Address Space: 10.0.0.0/16
(This means addresses from 10.0.0.1 to 10.0.255.254)
```

- **10.0.0.0/24** = Web tier (254 addresses)
- **10.0.1.0/24** = App tier (254 addresses)
- **10.0.2.0/24** = Database tier (254 addresses)

IP Address Types:

- **Private IPs** = Phone extensions within an office
- **Public IPs** = Full phone numbers people can call from outside
- **Dynamic IP** = Hotel room number (changes each visit)
- **Static IP** = Your home address (always the same)

2.3 Integration with Different Services

Think of VNet as the foundation where you can build different types of buildings:

Azure Service	Analogy	How it connects to VNet
Virtual Machines	Individual houses	Directly placed inside VNet
App Services	Apartment buildings	Special tunnel connection
AKS (Kubernetes)	Condominium complex	Entire cluster inside VNet
Databases	Banks or storage facilities	Private endpoints (back doors)

Simple Rule: If resources are in the same VNet, they can talk to each other privately. If they're in different VNets, they need special connections (like bridges between islands).

---

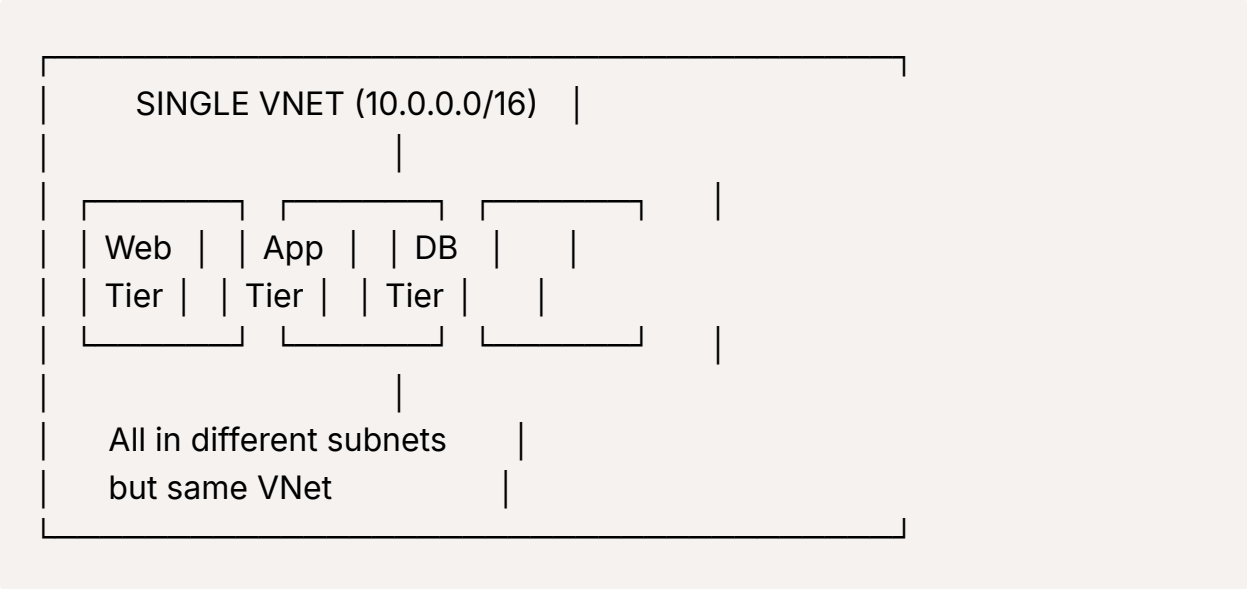
3. Network Design Patterns

3.1 Flat VNet Design

What it is: One big network with everything inside.

Visual:

text



When to use:

- Small to medium projects
- Single department applications
- Testing and development environments
- Simple architectures

Pros:

- ✅ Easy to set up
- ✅ Simple to manage
- ✅ All services can communicate easily
- ✅ Cost-effective

Cons:

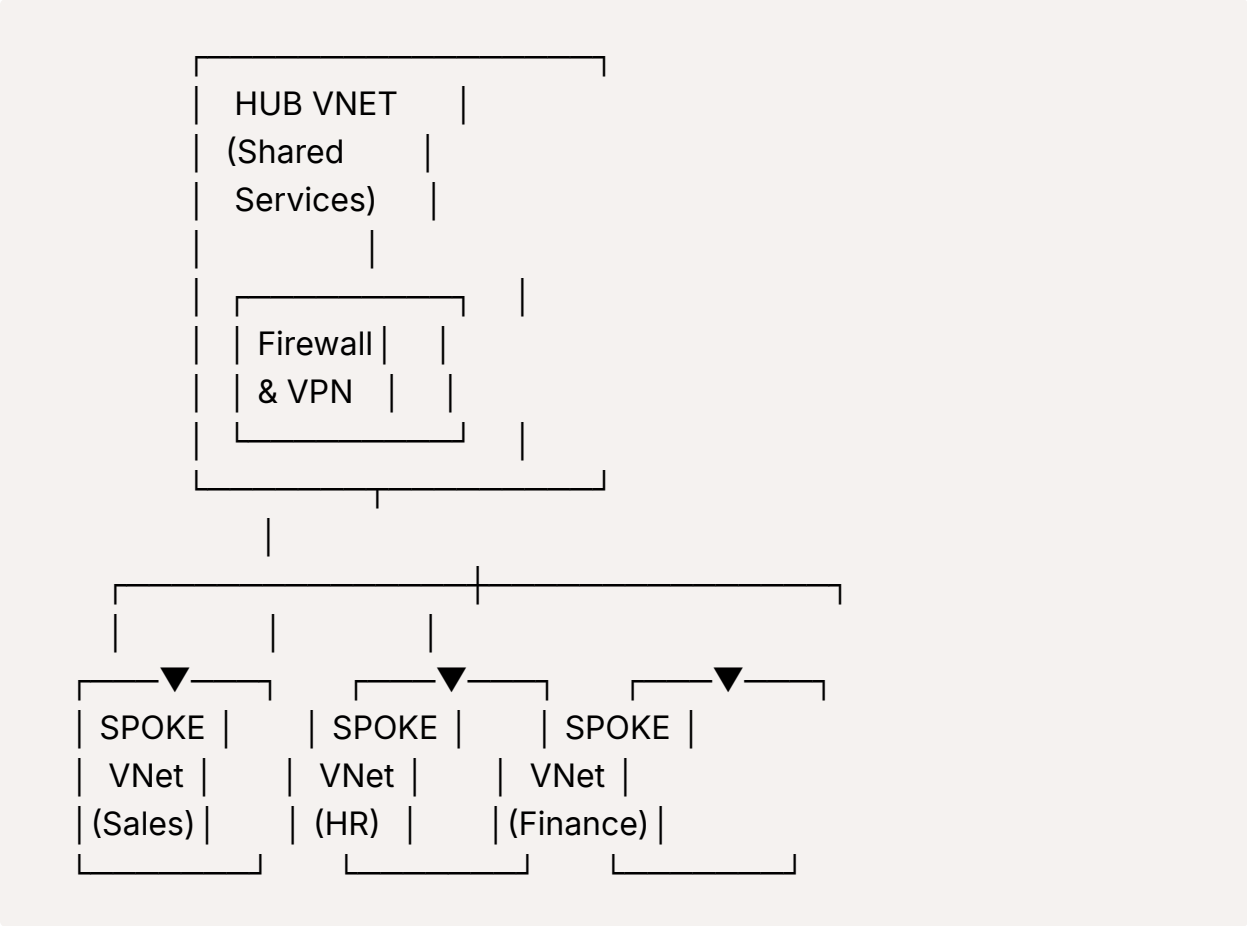
- ❌ Limited growth potential
- ❌ Hard to separate different departments
- ❌ Security boundaries are weaker

3.2 Hub-Spoke Design

What it is: Central hub connects to multiple spoke networks.

Visual:

text



When to use:

- Large organizations
- Multiple departments/projects
- Need for centralized security
- Compliance requirements

Pros:

- ✅ Excellent security isolation
- ✅ Centralized management

- ✅ Scalable - add more spokes easily
- ✅ Department/team independence

Cons:

- ❌ More complex to set up
- ❌ Higher initial cost
- ❌ Requires more planning

---

4. Step-by-Step Visual Guide

4.1 How Data Travels Through a VNet

text

```
[INTERNET USER]
      |
      ▼
[AZURE PUBLIC IP]
      |
      ▼
[AZURE FIREWALL] ← Security Checkpoint
      |
      ▼
[LOAD BALANCER] ← Traffic Director
      |
      ▼
   [VNET]
      |
   ┌──┴──────────────────┐
   ▼                     ▼
[WEB SUBNET]   [APP SUBNET]
   |                  |
   ▼                  ▼
[Web Server]  [App Server]
                  |
```

```
          ▼
[DB SUBNET]
    │
    ▼
[Database]
```

4.2 Security Layers in VNet

text

```
┌─────────────────────────────────────────┐
│  Layer 1: Network Security Group        │
│  (Like a building security guard)       │
│  • Checks IDs at the door               │
│  • Allows/denies based on rules         │
└─────────────────────────────────────────┘
                  │
┌─────────────────────────────────────────┐
│  Layer 2: Application Security Group    │
│  (Like department access cards)         │
│  • Web servers group                    │
│  • Database servers group               │
└─────────────────────────────────────────┘
                  │
┌─────────────────────────────────────────┐
│  Layer 3: Private Endpoints             │
│  (Like private back entrances)          │
│  • No public exposure                   │
│  • Direct private connections           │
└─────────────────────────────────────────┘
```

5. Practical Examples

Example 1: Small Business Website

text

```
Company: "Bella's Bakery" wants an online store

Architecture:
• One VNet: 10.1.0.0/16
• Two Subnets:
  - Web: 10.1.1.0/24 (for website)
  - Database: 10.1.2.0/24 (for orders)
• One Public IP for the website
• Database has only private IP
```

Example 2: Enterprise Corporation

text

```
Company: "Global Tech Corp" with multiple teams

Architecture:
• Hub VNet: 10.0.0.0/22 (shared services)
• Spoke 1: 10.1.0.0/22 (Development team)
• Spoke 2: 10.2.0.0/22 (Sales team)
• Spoke 3: 10.3.0.0/22 (HR team)
• VPN Gateway in hub for office connection
• Azure Firewall for security
```

---

6. Best Practices for Beginners

6.1 Planning Your Network

1. **Start with requirements:**
   - How many applications?
   - How many teams/departments?
   - What are security needs?

2. **Choose design pattern:**
    - **Flat VNet:** < 50 resources, single team
    - **Hub-Spoke:** > 50 resources, multiple teams
3. **IP Address Planning:**
    - Leave room for growth
    - Use meaningful numbering
    - Document everything

6.2 Security Basics

- **Always** use Network Security Groups
- **Never** expose databases to internet
- **Use** private endpoints when possible
- **Regularly** review security rules

6.3 Cost Optimization

- Combine similar resources in same subnet
- Use smaller subnets for limited resources
- Monitor data transfer between regions
- Delete unused resources

---

7. Common Mistakes to Avoid

Mistake	Why It's Bad	Better Approach
Using too small CIDR	Can't add more resources	Plan for 2x growth
Everything in one subnet	Security risk	Separate by function
Too many public IPs	Security & cost issue	Use load balancers
No documentation	Hard to manage later	Document from day 1
Ignoring DNS	Hard to remember IPs	Use descriptive names

---

8. Quick Reference Guide

VNet Components Cheat Sheet

Component	Purpose	Analogy
VNet	Main network container	Gated community
Subnet	Division within VNet	Street in community
NSG	Traffic filter	Security guard
Route Table	Traffic director	Road signs
Peering	Connect two VNets	Bridge between communities
Gateway	Connect to on-premises	Highway entrance

When to Use What Design

text

```
Is this a simple application?
    ├── Yes → Use FLAT VNET
    └── No → Answer: Multiple teams/departments?
            ├── Yes → Use HUB-SPOKE
            └── No → Need strict compliance?
                    ├── Yes → Use HUB-SPOKE
                    └── No → Use FLAT VNET
```