Azure Virtual Networks

Introduction

Azure Virtual Networks (VNets) provide the foundational building blocks for private networking in Azure. They enable resources like virtual machines (VMs) to securely communicate with each other, the internet, and on-premises networks. This paper delves into the essential components, configurations, and best practices for leveraging Azure VNets.

Overview of Virtual Networks in Azure

Importance of Virtual Networks

Virtual Networks in Azure are critical for creating isolated and secure environments for deploying applications and services. They facilitate secure communication between different Azure resources, allowing for segmented network architectures that can be tailored to specific needs and compliance requirements.

Common Use Cases

1. **Isolated Environments**: Creating separate environments for development, testing, and production.

2. **Hybrid Connectivity**: Extending on-premises networks to the cloud.

3. **Security and Compliance**: Implementing stringent security controls and meeting regulatory requirements.

Creating Virtual Machines in a Private Subnet

Creating VMs in a private subnet involves the following steps:

1. 1. **Navigate to Virtual Machines**: In the Azure portal, search for and select 'Virtual machines'.
2. 2. **Initiate VM Creation**: Select '+ Create' and choose 'Azure virtual machine'.
3. 3. **Fill in Basic Information**: Provide details such as the VM name, region, and image (e.g., Ubuntu Server 22.04 LTS).
4. 4. **Configure Network Settings**:

- **Virtual Network**: Select the desired VNet.

- **Subnet**: Choose a private subnet.

- **Public IP**: Ensure that no public IP is assigned.

Connecting Bastion Host VM in Public Subnet to Private Subnet VM

Azure Bastion provides secure and seamless RDP/SSH connectivity to VMs without exposing them to the public internet:

5. 1. **Create a Public Subnet for Bastion**: Ensure that the subnet has a /27 or larger IP range.
6. 2. **Deploy Azure Bastion**: Enable Bastion in the VNet settings and assign it a public IP.
7. 3. **Connect to VMs**: Use Bastion to connect to VMs in the private subnet by navigating to the VM in the portal and selecting 'Connect' via Bastion.

NAT Gateway

NAT Gateway provides outbound internet connectivity for VMs in a private subnet:

8. 1. **Create a NAT Gateway**: In the Azure portal, navigate to 'NAT gateways' and create a new NAT gateway.
9. 2. **Associate with Subnet**: Link the NAT gateway to the private subnet to manage outbound traffic.

Network Access Control Lists (NACLs) and Security Groups

Network Security Groups (NSGs)

NSGs contain security rules that allow or deny inbound and outbound traffic:

10. 1. **Create NSG**: Navigate to 'Network Security Groups' and create a new NSG.
11. 2. **Associate with Subnet/VM**: Apply the NSG to a specific subnet or network interface of a VM.
12. 3. **Define Rules**: Set up security rules to control traffic flow.

Network Access Control Lists (NACLs)

NACLs provide an additional layer of security by controlling traffic at the subnet level. They are stateless and can be used to complement NSGs.

Internet Gateway

Azure does not have a direct equivalent to AWS Internet Gateway. However, the combination of public IPs and the Azure load balancer provides similar functionality:

13. 1. **Public IPs**: Assign public IPs to resources requiring internet access.
14. 2. **Load Balancer**: Use an Azure load balancer to route internet traffic to VMs.

Configuring Two Availability Zones with Private Subnets

Deploying resources across multiple availability zones enhances reliability and fault tolerance:

15. 1. **Create Virtual Network**: Ensure the VNet spans multiple availability zones.
16. 2. **Create Subnets**: Define private subnets in each availability zone.

17. 3. **Deploy Resources**: Distribute VMs and other resources across the subnets in different zones.

Using a Load Balancer to Route Traffic

Azure Load Balancer distributes incoming traffic across multiple VMs, ensuring high availability:

18. 1. **Create Load Balancer**: In the Azure portal, navigate to 'Load Balancers' and create a new load balancer.
19. 2. **Configure Frontend IP**: Set up the frontend IP configuration.
20. 3. **Backend Pool**: Add the VMs from different availability zones to the backend pool.
21. 4. **Health Probes**: Define health probes to monitor the status of the VMs.
22. 5. **Load Balancing Rules**: Set up rules to distribute traffic based on defined criteria.

3rd Week
By Lekhraj jadon
MBM University Jodhpur