Secure and Scalable Infrastructure Documentation using IaC tools

Overview

This documentation outlines the secure and scalable design of an Azure-based web infrastructure stack, implemented using Infrastructure as Code (IaC) tools and adhering to best practices. The infrastructure utilizes appropriate Azure services for web hosting and load balancing to ensure reliability, security, and scalability.

Infrastructure Overview

The infrastructure comprises the following components:

1. **Virtual Network (VNet)**:
    - Address Space: 10.0.0.0/16
    - Subnets:
        - Web Tier Subnet: 10.0.1.0/24
        - Database Tier Subnet: 10.0.2.0/24
2. **Network Security Groups (NSGs)**:
    - Web Tier NSG:
        - Inbound Rules: Allow HTTP (port 80) and HTTPS (port 443) traffic.
    - Database Tier NSG:
        - Inbound Rule: Allow SQL traffic (port 1433) from the web tier subnet.
3. **Virtual Machines (VMs)**:
    - Web Tier VMs (2 instances):
        - OS: Windows Server 2019
        - Size: Standard_D2s_v3 (2 vCPUs, 8 GB RAM)
        - Managed Disks: Premium SSD, 128 GB
        - Availability Set: Ensures high availability for web tier VMs.
    - Database Tier VM (1 instance):
        - OS: Windows Server 2019
        - Size: Standard_D4s_v3 (4 vCPUs, 16 GB RAM)

- Managed Disks: Premium SSD, 256 GB

4. **Azure Load Balancer**:
    - SKU: Standard
    - Frontend IP Configuration: Public IP address for external access.
    - Backend Pool: Includes web tier VM instances.
    - Health Probe: Monitors the health of web tier VMs.
    - Load Balancing Rule: Distributes incoming HTTP traffic (port 80) to the backend pool.
5. **Azure Application Gateway**:
    - SKU: Standard_v2
    - Tier: Standard
    - Frontend IP Configuration: Public IP address for external access.
    - Backend Pool: Includes web tier VM instances.
    - HTTP Settings: Configured for port 80 (HTTP) traffic.
    - Listener: Listens for incoming HTTP traffic on port 80.
    - Routing Rule: Routes incoming traffic to the backend pool.
6. **Azure SQL Database**:
    - Database Name: [YourDatabaseName]
    - Server: New server created in the same region as VMs.
    - Pricing Tier: Standard S0
    - Connectivity: Azure services allowed to access the server.

Best Practices and Security Measures

1. **Infrastructure as Code (IaC)**:
    - Utilized Azure Resource Manager (ARM) templates for automated and consistent deployment.
    - Stored sensitive information like database connection strings in Azure Key Vault for secure management.
2. **Scalability**:

- VM sizes chosen for optimal performance and scalability based on workload requirements.
- Azure Load Balancer and Application Gateway facilitate load distribution and scalability.

3. **Security**:
    - Network Security Groups (NSGs) restrict traffic to necessary ports and sources, enhancing security posture.
    - Azure Security Center configured for continuous monitoring and management of security policies.
    - Implemented Azure Backup for regular backups of VMs and databases, ensuring data integrity and recovery options.

Deployment and Configuration

1. **Deployment Instructions**:
    - Refer to the README file in the GitHub repository for step-by-step deployment instructions using ARM templates.
    - Ensure Azure CLI or Azure PowerShell is installed for executing deployment scripts.
2. **Example Configuration Files**:
    - ARM templates provided in the repository with detailed comments explaining each component and parameter.
    - Follow best practices for parameterization, modularization, and version control of ARM templates.

Conclusion

This documentation illustrates the secure and scalable design of an Azure-based web infrastructure stack, showcasing the proper use of IaC tools, adherence to best practices, and appropriate utilization of Azure services for web hosting and load balancing. By following these guidelines, you can deploy a robust and reliable infrastructure that meets modern application hosting requirements.