Pavan Kumar D

ACE12555

Testing

**AZURE TASK**

**Scenario** **1:**

Your team needs to deploy a virtual machine in azure to test a software application. The team has requested both windows and linex machine

**Question:**

how could you set up these machines and what considerations are need for pricing and OS licensing

**Answer:**

**Step 1:** Log in to the Azure Portal

**Step 2:** Create a Resource Group

**Step 3:** Create a Windows VM

Go to the "Virtual Machines" service and click on "Create"> "Azure Virtual Machine."

**Configure the VM details:**

**Subscription:** Choose the subscription.

**Resource Group:** Select the resource group created in Step 2.

**Instance Details:**

**Name:** Provide a name for the VM (e.g., Win-VM-Test).

**Region:** Choose the region closest to your users.

**Image:** Select a Windows OS.

**Size:** Select the appropriate VM size based on the testing requirements.

**Administrator Account:** Set a username and password.

Configure other settings like networking, disks, and monitoring (enable or disable as needed).

Review and create the VM.

**Step 4:** Create a Linux VM

Follow the same steps as above but select a Linux-based OS in the "Image" section

Provide SSH public key or username/password credentials for admin access.

Configure other settings similarly and create the VM.

**Considerations for Pricing an OS License:**

Azure offers two pricing options for operating systems:

1. Bring Your Own License (BYOL): If your organization already owns OS licenses

2. Azure Pay-As-You-Go Licensing: Azure automatically includes the license cost for the OS in the VM's pricing when using default OS images.

**Scenario 2**:

The IT security team has requested the sensitive data stored in azure storage account be encrypted to meet completed required

**Question:**

How could ensure the data stored azure is encrypted, and what encryption types are available

**Answer:**

**Default Encryption:** Azure Storage accounts are automatically encrypted by default using Microsoft-managed keys (MMK). No manual configuration is needed unless you require more control.

**Customer-Managed Keys:** Use Azure Key Vault to manage your own encryption keys for additional control over key rotation and compliance.

**Data in Transit:** Enable Secure transfer required to enforce HTTPS for secure data transfer. For Azure Files, enable SMB 3.0 encryption.

**Azure Disk Encryption:** For VMs, use BitLocker (Windows) or DM-Crypt (Linux) to encrypt OS and data disks.

**Client-Side Encryption:** Encrypt data on the client side before uploading it to Azure for added security.

**Encryption Types:**

**Server-Side Encryption:** Microsoft-managed keys or customer-managed keys.

**Client-Side Encryption:** Encrypt data before uploading to Azure.

**Encryption in Transit:** HTTPS/TLS for secure communication.

To ensure compliance, use Azure Key Vault, Azure Policy, and Azure Security Center to manage and monitor encryption settings....

**Scenario 3:**

You're responsible for setting up Devops pipeline in azure Devops for your application. The pipeline must deploy code to an

azure app services and notifies the team if the deployment failed

**Question:**

How could you configure this pipeline to meet this requirement

**Answer:**

If the deployment fails, notifying the team and providing actionable insights is critical. Here's how you can configure the pipeline to notify the team effectively while providing relevant failure

1. Enable Built-In Azure DevOps Notifications

Go to your Azure DevOps Project → Project Settings → Notifications.

Create a New Notification Subscription:

**Trigger:** Pipeline Fails

**Scope:** Select the specific pipeline(s) where this applies.

**Recipients:** Add team members, group emails, or service accounts for integrations with Slack, Microsoft Teams, etc.

**Details:** Include failure information in the notification.

3.Integrate Application Insights Alerts:

Configure Azure Monitor with Application Insights to detect runtime failures or performance issues after deployment.

Set up alert rules for error codes, latency, or availability issues.

Send notifications to the team via email, SMS, or integration with tools like PagerDuty, Slack, or Teams.

**Scenario** **4:**

Your organization is moving On-premises SQL database to azure. the data base must remain assessable during migration with minimal downtime

**Question:**

Which azure service would you use and how could you perform the migration

**Answer:**

Post-Migration Tasks

**a. Validate the Migration**

Compare the source and target databases to ensure data consistency and accuracy.

Use query testing tools or run application-specific tests.

**b. Optimize the Target Database**

Scale up or down based on workload requirements.

Optimize database performance using indexing, partitioning, and query tuning.

**c. Implement Monitoring and Alerts**

Enable Azure Monitor and Application Insights to track database performance and detect issues.

Configure alerts for key metrics such as CPU usage, query response time, and storage utilization.

**d. Enable Backups**

Configure automated backups for the Azure database:

Point-in-Time Restore for Azure SQL Database and Managed Instance.

Full and incremental backups for SQL Virtual Machines.

**e. Decommission On-Premises Database**

After confirming the success of the migration, decommission the on-premises SQL Server to free up resources.

**Key Considerations for a Successful Migration**

**1. Minimal Downtime**

Online migration ensures the database remains accessible during the process, reducing downtime to a few minutes during the cutover.

**2. Secure Data Transfer**

Use encrypted connections (e.g., SSL/TLS) to secure data in transit.

Configure Azure Private Link or ExpressRoute for enhanced security.

**3. Scalability**

Ensure the Azure database tier and compute/storage configurations can handle future workload growth.

**4. High Availability**

Enable high availability features such as Auto-Failover Groups or Zone-Redundant Configurations to minimize disruptions in case of failures.

**5. Cost Management**

Use Azure Cost Management to monitor and control expenses during and after migration.

Scale resources as needed to optimize costs.