Exam 70-535: Architecting Microsoft Azure Solutions (replaces 70-534)

Audience Profile

Candidates for this exam define the appropriate cloud native, cloud migration, and hybrid cloud solutions to meet the required functional, operational, and deployment requirements through the solution lifecycle. Candidates should know the features and capabilities of Azure services to be able to identify tradeoffs and make decisions for designing public and hybrid cloud solutions.

The candidate should understand DevOps technologies, provisioning Azure resources using ARM templates, and designing highly resilient workloads running on Azure.

Objective Domain
Note: This document shows tracked changes that are effective as of March 22, 2018.

Design Compute Infrastructure (20-25%)
Design solutions using virtual machines (VMs)

> Design VM deployments by leveraging availability sets, fault domains, and update domains in Azure; use web app for containers; design VM Scale Sets; design for compute-intensive tasks using Azure Batch and Azure Batch AI; define a migration strategy from cloud services; determine when to use reserved instances; design for VMs in a DevTest Lab environment (including formulas, images, artifacts, claiming and un-claiming VMs); determine when to use Accelerated Networking; recommend use of Azure Backup and Azure Site Recovery including support for Linux in Azure Backup and integrating Azure Backup in the VM creation process; recommend when to use availability zones

Design solutions for serverless computing
> Use Azure Functions to implement event-driven actions; design data storage solutions for serverless computing; design for serverless computing using Azure Container Instances; design application solutions by using Azure Logic Apps, Azure Functions, or both; determine when to use API management service; design event routing solutions using Azure Event Grid; design solutions that integrate stream processing and bot messaging

Design microservices-based solutions

> Determine when a container-based solution is appropriate; determine when container-orchestration is appropriate; determine when Azure Service Fabric (ASF) is appropriate; determine when Azure Functions is appropriate; determine when to use API management service; determine when Web API is appropriate; determine which platform is appropriate for container orchestration; consider migrating existing assets versus cloud native deployment; design lifecycle management strategies

Design web applications

Design Azure App Service Web Apps; design custom web API; secure Web API; design Web Apps for scalability and performance; design for high availability using Azure Web Apps in multiple regions; determine which App service plan to use; design Web Apps for business continuity; determine when to use Azure App Service ~~Environment (ASE)~~Isolated; design for API apps; determine when to use API management service; determine when to use Web Apps on Linux; determine when to use a CDN; determine when to use a cache, including Azure Redis cache

Create compute-intensive applications

Design high-performance computing (HPC) and other compute-intensive applications using Azure Services; determine when to use Azure Batch; design stateless components to accommodate scale; design lifecycle strategy for Azure Batch; design solution that implement low priority batching and job task counting

Design Data Implementation (15-20%)

Design for Azure Storage solutions

Determine when to use Azure Blob Storage, blob tiers (hot, cool, archive), Azure Files, disks, Azure Data Box, Azure Storage Service Encryption, and Azure StorSimple

Design for Azure Data Services

Determine when to use Azure Data Catalog, Azure Data Factory, Azure SQL Data Warehouse, Azure Data Lake Analytics, Azure Analysis Services, and Azure HDInsight

Design for relational database storage

Determine when to use Azure SQL Database and SQL Server Stretch Database; design for scalability and features; determine when to use Azure Database for MySQL and Azure Database for PostgreSQL; design for HA/DR, geo-replication; design a backup and recovery strategy; design optimization strategies for Azure SQL Data Warehouse columnar storage

Design for NoSQL storage

Determine when to use Azure Redis Cache, Azure Table Storage, Azure Data Lake, Azure Search, Time Series Insights; design pipelines for managing recurring jobs

Design for CosmosDB storage

Determine when to use MongoDB API, Azure Cosmos DB SQL~~DocumentDB~~ API, Graph API, Azure Tables API; design for cost, performance, data consistency, availability, and business continuity

Design Networking Implementation (15-20%)

Design Azure virtual networks

Design solutions that use Azure networking services: design for load balancing using Azure Load Balancer and Azure Traffic Manager; define DNS, DHCP, and IP strategies; determine when to use Azure Application Gateway; determine when to use virtual network (VNet) service endpoints; determine when to use multi-node application gateways, Traffic Manager and load balancers

Design external connectivity for Azure Virtual Networks

Determine when to use Azure VPN, Azure ExpressRoute and Virtual Network Peering architecture and design; determine when to use User Defined Routes (UDRs); determine when to use VPN gateway site-to-site failover for ExpressRoute; determine when to use the Container Networking Interface (CNI) plugin; design solutions that use Global VNet Peering

Design security strategies

Determine when to use network virtual appliances; design a perimeter network (DMZ); determine when to use a Web Application Firewall (WAF), Network Security Group (NSG), and virtual network service tunneling; organize resources by designing solutions that use service tags

Design connectivity for hybrid applications

Design connectivity to on-premises data from Azure applications using Azure Relay Service, Azure Data Management Gateway for Data Factory, Azure On-Premises Data Gateway, Hybrid Connections, or Azure Web App's virtual private network (VPN) capability; identify constraints for connectivity with VPN; identify options for joining VMs to domains

Design Security and Identity Solutions (20-25%)

Design an identity solution

Design AD Connect synchronization; design federated identities using Active Directory Federation Services (AD FS); design solutions for Multi-Factor Authentication (MFA); design an architecture using Active Directory on-premises and Azure Active Directory (AAD); determine when to use Azure AD Domain Services; design security for Mobile Apps using AAD

Secure resources by using identity providers

Design solutions that use external or consumer identity providers such as Microsoft account, Facebook, Google, and Yahoo; determine when to use Azure AD B2C and Azure AD B2B; design mobile apps using AAD B2C or AAD B2B

Design a data security solution

Design data security solutions for Azure services; determine when to use Azure Storage encryption, Azure Disk Encryption, Azure SQL Database security capabilities, and Azure Key Vault; design for protecting secrets in ARM templates using Azure Key Vault; design for protecting application secrets using Azure Key Vault; design a solution for managing certificates using Azure Key Vault; design solutions that use Azure AD Managed Service Identity

Design a mechanism of governance and policies for administering Azure resources

Determine when to use Azure RBAC standard roles and custom roles; define an Azure RBAC strategy; determine when to use Azure resource policies; determine when to use Azure AD Privileged Identity Management; design solutions that use Azure AD Managed Service Identity; determine when to use HSM-backed keys

Manage security risks by using an appropriate security solution

Identify, assess, and mitigate security risks by using Azure Security Center, Operations Management Suite Security and Audit solutions, and other services; determine when to use Azure AD Identity Protection; determine when to use Advanced Threat Detection; determine an appropriate endpoint protection strategy

Design Solutions by using Platform Services (10-15%)

Design for Artificial Intelligence Services

Determine when to use the appropriate Cognitive Services, Azure Bot Service, Azure Machine Learning, and other categories that fall under cognitive AI

Design for IoT

Determine when to use Azure Stream Analytics, Azure IoT Hubs, Azure Event Hubs, real-time analytics, Azure Time Series Insights, Azure IoT Edge, Azure Notification Hubs, Event Grid, and other categories services that fall under IoT

Design messaging solution architectures

Design a messaging architecture; determine when to use Azure Storage Queues, Azure Service Bus, Azure Event Hubs, Azure Event Grid, Azure Relay, Azure Functions, and Azure Logic Apps; design a push notification strategy for Mobile Apps; design for performance and scale

Design for media service solutions

Define solutions using Azure Media Services, video indexer, video API, computer vision API, preview, and other media related services; design solutions that use file-based encoding or Azure Media Analytics

Design for Operations (10-15%)

Design an application monitoring and alerting strategy

Determine the appropriate Microsoft products and services for monitoring applications on Azure; define solutions for analyzing logs and enabling alerts using Azure Log Analytics; define solutions for analyzing performance metrics and enabling alerts using Azure Monitor; define a solution for monitoring applications and enabling alerts using Application Insights

Design a platform monitoring and alerting strategy

Determine the appropriate Microsoft products and services for monitoring Azure platform solutions; define a monitoring solution using Azure Health, Azure Advisor, and Activity Log; define a monitoring

solution for Azure Networks using Log Analytics and Network Watcher service; monitor security with Azure Security Center; design TCP connections

Design an operations automation strategy

Determine when to use Azure Automation, Chef, Puppet, PowerShell, ~~Desired State Configuration~~Azure Automation (DSC), Event Grid, and Azure Logic Apps; define a strategy for auto-scaling; define a strategy for enabling periodic processes and tasks; define an update management strategy