BAKER COLLEGE
STUDENT LEARNING OUTCOMES

MNP 3120 Advanced Server Administration II
3 Semester Hours

Student Learning Outcomes & Enabling Objectives

1. Design Azure virtual networks
    a. Leverage Azure networking services: implement load balancing using Azure Load Balancer and Azure Traffic Manager; define DNS, DHCP, and IP addressing configuration; define static IP reservations; apply Network Security Groups (NSGs) and User Defined Routes (UDRs); deploy Azure Application Gateway
2. Describe Azure VPN and ExpressRoute architecture and design
    a. Describe Azure P2S and S2S VPN; leverage Azure VPN and ExpressRoute in network architecture
3. Secure resources by using managed identities
    a. Describe the differences between Active Directory on-premises and Azure Active Directory (AAD); programmatically access AAD using Graph API; secure access to resources from AAD applications using OAuth and OpenID Connect
4. Secure resources by using hybrid identities
    a. Use SAML claims to authenticate to on-premises resources; describe AD Connect synchronization; implement federated identities using Active Directory Federation Services (AD FS)
5. Secure resources by using identity providers
    a. Provide access to resources using identity providers such as Microsoft account, Facebook, Google, and Yahoo; manage identity and access by using AAD B2C; implement AAD B2B
6. Identify an appropriate data security solution
    a. Identify security requirements for data in transit and data at rest; implement data security requirements using Azure services, including Azure Storage encryption, Azure Disk Encryption, and Azure SQL Database TDE
7. Design a role-based access control (RBAC) strategy
    a. Secure resource scopes such as the ability to create VMs and Azure Web Apps; implement Azure RBAC standard roles; design Azure RBAC custom roles
8. Manage security risks by using an appropriate security solution
    a. Identify, assess, and mitigate security risks by using Azure Security Center, Operations Management Suite, and other services
9. Design data storage
    a. Design storage options for data including Table Storage, SQL Database, DocumentDB, Blob Storage, MongoDB, and MySQL; design security options for SQL Database or Azure Storage

10. Select the appropriate storage option
    a. Select the appropriate storage for performance; identify storage options for cloud services and hybrid scenarios with compute on-premises and storage on Azure
11. Create compute-intensive applications
    a. Design high-performance computing (HPC) and other compute-intensive applications using Azure Services
12. Create long-running applications
    a. Implement Azure Batch for scalable processing; design stateless components to accommodate scale; use Azure Scheduler
13. Integrate Azure services in a solution
    a. Design Azure architecture using Azure services such as Azure Active Directory (AAD), Azure AppService, API Management, Azure Cache, Azure Search, Service Bus, Event Hubs, Stream Analytics, and IoT Hub; identify the appropriate use of services such as Azure Machine Learning, Big Data, Azure Media Services, and Azure Search services
14. Implement messaging applications
    a. Use a queue-centric pattern for development; select appropriate technology such as Azure Storage Queues, Azure Service Bus queues, topics, subscriptions, and Azure Event Hubs
15. Implement applications for background processing
    a. Implement Azure Batch for compute-intensive tasks; use Azure Web Jobs to implement background tasks; use Azure Functions to implement event-driven actions; leverage Azure Scheduler to run processes at preset/recurring timeslots
16. Design connectivity for hybrid applications
    a. Connect to on-premises data from Azure applications using Service Bus Relay, 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 or cloud services
17. Design web applications
    a. Design Azure App Service Web Apps; design custom web API; offload long-running applications using WebJobs; secure Web API using AAD; design Web Apps for scalability and performance; deploy Azure Web Apps to multiple regions for high availability; deploy Web Apps; create App service plans; design Web Apps for business continuity; configure data replication patterns; update Azure Web Apps with minimal downtime; back up and restore data; design for disaster recovery
18. Design Mobile Applications
    a. Design Azure Mobile Services; consume Mobile Apps from cross-platform clients; integrate offline sync capabilities into an application; extend Mobile Apps using custom code; implement Mobile Apps using Microsoft .NET or Node.js; secure Mobile Apps using AAD; implement push notification services in Mobile Apps; send push notifications to all subscribers, specific subscribers, or a segment of subscribers

19. Design a monitoring strategy
    a. Identify the Microsoft products and services for monitoring Azure solutions; leverage the capabilities of Azure Operations Management Suite and Azure Application Insights for monitoring Azure solutions; leverage built-in Azure capabilities; identify third-party monitoring tools including open source; describe the Azure architecture constructs such as availability sets and update domains and how they impact a patching strategy; analyze logs by using the Azure Operations Management Suite
20. Describe business continuity/disaster recovery (BC/DR) by using Azure
    a. Leverage the architectural capabilities of BC/DR; describe Hyper-V Replica and Azure Site Recovery (ASR); describe use cases for Hyper-V Replica and ASR
21. Design a disaster recovery strategy
    a. Design and deploy Azure Backup and other Microsoft backup solutions for Azure; leverage use cases when StorSimple and System Center Data Protection Manager would be appropriate; design and deploy Azure Site recovery
22. Design Azure Automation and PowerShell workflows
    a. Create a PowerShell script specific to Azure; automate tasks by using the Azure Operations Management Suite
23. Describe the use cases for Azure Automation configuration
    a. Evaluate when to use Azure Automation, Chef, Puppet, PowerShell, or Desired State Configuration (DSC)
24. Design ARM virtual machines (VMs)
    a. Design VM deployments leveraging availability sets, fault domains, and update domains in Azure; select appropriate VM SKUs
25. Design ARM template deployment
    a. Author ARM templates; deploy ARM templates via the portal, PowerShell, and CL
26. Design for availability
    a. Implement regional availability and high availability for Azure deployments

These SLOs are approved for experiential credit.

Effective: Fall 2017