

Azure Storage & Advanced Networking - 3 Session Plan

Course Overview

- Total Duration: 6 Hours (3 Sessions of 2 hours each)
 - Target Audience: Beginners with basic Azure infrastructure knowledge
 - Approach: Deep dive into storage services and virtual networking with practical scenarios
-

Session 1: Virtual Networking Deep Dive (2 Hours)

Learning Objectives

By the end of this session, learners will be able to: - Design and implement virtual networks with proper subnetting - Configure Network Security Groups (NSGs) effectively - Understand and implement network routing - Connect Azure networks to on-premises environments

3.1 Compute and Networking Continued

Virtual Network Fundamentals (45 mins) What is Azure Virtual Network (VNet)? - Definition: The fundamental building block for private networks in Azure - Purpose: Isolate and secure Azure resources - Key Capabilities: - Azure resource communication - Internet connectivity - On-premises connectivity - Traffic filtering and routing

VNet Components: - Address Space: The IP range for the entire VNet (e.g., 10.0.0.0/16) - Subnets: Segments within the VNet address space - Network Interfaces: Connection points for VMs - IP Addresses: Private and public IP configurations

Subnet Design Best Practices:

VNet: 10.0.0.0/16

web-subnet: 10.0.1.0/24	(Web tier - App Services, VMs)
app-subnet: 10.0.2.0/24	(Application tier - Logic, Functions)
data-subnet: 10.0.3.0/24	(Data tier - SQL, Storage)
mgmt-subnet: 10.0.4.0/24	(Management - Bastion, Monitoring)
GatewaySubnet: 10.0.255.0/27	(VPN/ExpressRoute gateways)

Network Security Groups (NSGs) (45 mins) What are Network Security Groups? - Definition: Firewall for Azure resources - Purpose: Filter network traffic to and from Azure resources - Scope: Can be applied to subnets or individual network interfaces

NSG Rule Structure: - Priority: Lower numbers evaluated first (100-4096)
 - Source/Destination: IP ranges, service tags, application security groups -
 Protocol: TCP, UDP, or Any - Action: Allow or Deny

Default NSG Rules: | Priority | Source | Destination | Port | Action | Purpose |
 |-----|-----|-----|-----|-----|-----|
 | 65000 | 0.0.0.0/0 | 0.0.0.0/0 | * | Allow | Inbound VNet traffic |
 | 65001 | 0.0.0.0/0 | 0.0.0.0/0 | * | Allow | Outbound internet traffic |
 | 65500 | 0.0.0.0/0 | 0.0.0.0/0 | * | Deny | Catch-all deny rule |

Example Web Server NSG Rules:

Inbound Rules:

- Priority 100: Allow HTTP (Port 80) from Internet
- Priority 110: Allow HTTPS (Port 443) from Internet
- Priority 120: Allow SSH (Port 22) from Corporate IP
- Priority 130: Allow RDP (Port 3389) from Corporate IP

Outbound Rules:

- Priority 100: Allow all to Internet
- Priority 110: Allow all to VNet

Network Routing and Connectivity (30 mins) Route Tables: - Purpose: Control how traffic is routed between subnets - System Routes: Automatically created by Azure - Custom Routes: User-defined for specific routing needs

Common Routing Scenarios: - Force Tunneling: Route all internet-bound traffic through on-premises network - Network Virtual Appliances: Route traffic through firewalls or other security devices - Service Endpoints: Optimize routing to Azure services

Hybrid Connectivity Options:

VPN Gateway: - Site-to-Site: Connect entire networks (office to Azure) - Point-to-Site: Connect individual devices to Azure - Features: IPsec/IKE VPN tunnels, up to 10 Gbps

ExpressRoute: - What: Private, dedicated connection to Azure - Benefits: - Higher reliability (99.95% SLA) - Faster speeds (up to 100 Gbps) - Lower and consistent latency - Enhanced security (private connection)

Connection Comparison: | Feature | VPN Gateway | ExpressRoute | |-----|-----|
 |-----|-----| | Speed | Up to 10 Gbps | Up to 100 Gbps | | Latency | Variable | Consistent | | Cost | Lower | Higher | | Security | Encrypted over internet | Private connection | | Use Case | Development, small offices | Enterprise, production |

Session 2: Azure Storage Services (2 Hours)

Learning Objectives

By the end of this session, learners will be able to: - Differentiate between Azure storage services - Select appropriate storage solutions for different scenarios - Understand storage redundancy options - Implement basic storage security

3.2 Storage Services

Azure Storage Overview (30 mins) What is Azure Storage? - Definition: Microsoft's cloud storage solution for modern data storage scenarios - Characteristics: Durable, highly available, massively scalable, secure - Access Methods: REST API, SDKs, Azure Portal, PowerShell/CLI

Core Storage Services:

1. Blob Storage: - Purpose: Optimized for storing massive amounts of unstructured data - Use Cases: - Serving images or documents directly to browsers - Storing files for distributed access - Streaming video and audio - Storing data for backup, archiving, and disaster recovery
2. File Storage: - Purpose: Fully managed file shares in the cloud - Use Cases: - Lift and shift applications that need file shares - Shared application settings - Development and testing environments
3. Disk Storage: - Purpose: Block-level storage volumes for Azure VMs - Types: - Ultra Disks: High-performance for IO-intensive workloads - Premium SSDs: High-performance for production workloads - Standard SSDs: Balanced performance for web servers - Standard HDDs: Low-cost for backup and infrequent access
4. Table Storage: - Purpose: NoSQL key-value store for semi-structured data - Use Cases: Storing TBs of structured, non-relational data

Storage Account Types (30 mins) Storage Account: - Definition: The fundamental management unit for Azure Storage - Purpose: Provides a unique namespace for your Azure Storage data

Storage Account Type	Type	Best For	Performance	Supported Services
Standard General Purpose v2	Standard	Most scenarios	Blobs, Files, Queues, Tables	Most scenarios
Premium Block Blobs	Premium	High transaction rates	Block blobs only	Premium Block Blobs
Premium File Shares	Premium	Enterprise file shares	Azure Files only	Premium File Shares
Premium Page Blobs	Premium	Virtual machine disks	Page blobs only	Premium Page Blobs

Storage Tiers for Blob Storage: - Hot Tier: Optimized for frequently accessed data - Cool Tier: Optimized for infrequently accessed data (lower storage costs, higher access costs) - Archive Tier: Optimized for rarely accessed data (lowest storage costs, highest access costs)

Example Use Cases by Tier: - Hot: Active application data, frequently accessed files - Cool: Short-term backup, disaster recovery datasets - Archive: Long-term backups, compliance data, raw sensor data

Storage Redundancy Options (30 mins) What is Storage Redundancy? - Definition: Strategy for storing multiple copies of your data - Purpose: Protect your data from planned and unplanned events

Redundancy Types:

1. Locally Redundant Storage (LRS): - Replication: Synchronously copies data three times within a single datacenter - Use Case: Data that can be easily reconstructed, non-critical data - Durability: 99.999999999% (11 nines)
2. Zone-Redundant Storage (ZRS): - Replication: Synchronously copies data across three availability zones in the region - Use Case: Production applications requiring high availability - Durability: 99.999999999% (12 nines)
3. Geo-Redundant Storage (GRS): - Replication: LRS in primary region + asynchronous copy to secondary region - Use Case: Maximum durability across regions - Durability: 99.9999999999999% (16 nines)
4. Read-Access Geo-Redundant Storage (RA-GRS): - Replication: GRS + read access to data in secondary region - Use Case: Read access during regional outage

Redundancy Comparison: | Type | Copies | Survives Datacenter Failure | Survives Regional Failure | Cost |

LRS	3 in one DC	No	No	Lowest
ZRS	3 across AZs	Yes	No	Medium
GRS	6 (3+3)	Yes	Yes	Higher
RA-GRS	6 (3+3)	Yes	Yes	Highest

Session 3: File Management & Migration (2 Hours)

Learning Objectives

By the end of this session, learners will be able to: - Implement effective file management strategies - Plan and execute storage migrations - Configure storage security and access controls - Monitor and optimize storage performance

File Management and Data Organization (45 mins) Blob Storage Hierarchy:

Storage Account

Container 1 (like a folder)

Blob 1 (actual file)

Blob 2

Container 2

Blob 3

\$web (special container for static websites)

Container Access Levels: - Private: No anonymous read access (default) - Blob: Anonymous read access for blobs only - Container: Anonymous read access to containers and blobs

Best Practices for Organization: 1. Use meaningful container names: "images", "documents", "backups" 2. Implement folder structure: "documents/invoices/2024/" 3. Use descriptive blob names: "customer-contract-v2-signed.pdf" 4. Apply metadata: Add key-value pairs for better organization

Azure Files Features: - SMB Protocol: Compatible with Windows, Linux, macOS - Active Directory Integration: Use existing AD for authentication - Snapshot Support: Point-in-time backups of file shares - Size: Up to 100 TiB per share

Storage Security and Access Control (45 mins) Storage Security Layers:

1. Network Security: - Firewall Rules: Restrict access to specific IP ranges/VNets - Service Endpoints: Secure connection from VNets to storage - Private Endpoints: Private IP addresses for storage accounts

2. Authentication & Authorization: - Access Keys: Full access to storage account (use sparingly) - Shared Access Signatures (SAS): Time-limited, granular access - Azure AD Integration: Role-based access control (RBAC) - Storage Access Policies: Revocable SAS tokens

Shared Access Signature (SAS) Types: - Service SAS: Delegates access to a specific storage service - Account SAS: Delegates access to multiple storage services - User Delegation SAS: Secured with Azure AD credentials

Example SAS URL Structure:

```
https://storageaccount.blob.core.windows.net/container/file.txt
?sv=2020-08-04
&ss=b
&srt=s
&sp=r
&se=2024-12-31T23:59:59Z
&st=2024-01-01T00:00:00Z
&spr=https
&sig=Signature
```

Data Migration Strategies (30 mins) Migration Tools Overview:

1. Azure Storage Explorer: - Purpose: GUI tool for managing storage accounts - Best For: Small to medium migrations, manual operations

2. AzCopy: - Purpose: Command-line tool for high-performance data transfer - Best For: Large-scale automated migrations - Features: Resume capability,

parallel transfers

3. Azure Data Box: - Purpose: Physical appliance for offline data transfer - Best For: Terabytes to petabytes of data, limited bandwidth - Variants: Data Box Disk, Data Box, Data Box Heavy

4. Azure File Sync: - Purpose: Sync on-premises file servers with Azure Files - Best For: Hybrid file server scenarios, disaster recovery

Migration Planning Steps: 1. Assessment: Inventory current data and access patterns 2. Tool Selection: Choose appropriate migration tools 3. Network Planning: Estimate bandwidth and time requirements 4. Testing: Validate migration with sample data 5. Execution: Perform actual migration 6. Validation: Verify data integrity and access

Example Migration Scenario:

```
On-Premises File Server
  ↓ (Azure File Sync)
Azure File Share (primary cloud endpoint)
  ↓ (Sync group)
Branch Office Servers (secondary endpoints)
```

Session Wrap-up & Best Practices

Storage Design Patterns: - Use lifecycle management: Automatically move blobs to cooler tiers - Implement versioning: Protect against accidental deletion/modification - Enable soft delete: Recover deleted blobs and containers - Use immutable storage: Write-once-read-many for compliance

Cost Optimization Tips: 1. Choose right redundancy: Don't over-provision for non-critical data 2. Use appropriate tiers: Move infrequently accessed data to cool/archive 3. Implement lifecycle policies: Automate tier movement 4. Monitor usage: Use Storage Analytics to identify optimization opportunities

Security Checklist: - ☐ Enable storage account firewall - ☐ Use private endpoints for VNet connectivity - ☐ Rotate access keys regularly - ☐ Use SAS tokens instead of access keys when possible - ☐ Enable logging and monitoring

Next Steps: - Practice creating storage accounts with different redundancy options - Experiment with AzCopy for data transfer - Explore Storage Explorer for management tasks - Review AZ-104 certification storage objectives