AZ-104 - Comprehensive Azure Study Guide

AZ-104: Microsoft Azure Administrator — Comprehensive Study Guide

> Audience: Experienced Azure Infrastructure Architect relearning with deep, hands-on detail. Exam Version: Skills measured as of April 17, 2026

---

1. Manage Azure Identities and Governance (20-25%)

1.1 Manage Microsoft Entra Users and Groups

Users

- **Member users:** Created in your Entra ID tenant. Full directory rights.
- **Guest users:** Invited from external orgs (B2B). Limited rights by default.
- **Creating users:**

```
        # CLI
        az ad user create --display-name "John Doe" --password "P@ssw0rd123!" --user-principal-name
john@contoso.onmicrosoft.com --mail-nickname john

        # PowerShell
        New-AzADUser -DisplayName "John Doe" -Password (ConvertTo-SecureString "P@ssw0rd123!" -AsPlainText -Force)
UserPrincipalName john@contoso.onmicrosoft.com -MailNickname john
```

Groups

- **Security groups:** Manage user and computer access to resources. Max 100,000 members.
- **Microsoft 365 groups:** Collaboration group with shared mailbox, calendar, SharePoint site, Teams.
- **Dynamic groups:** Membership based on user attributes (e.g., department == “IT”). Requires Entra ID P1/P2 license. Rules use attribute-based expressions.
  - Example: `user.department -eq "Engineering" -and user.country -eq "US"`
  - **Gotcha:** Dynamic groups can only contain users OR devices, not both. You cannot convert a dynamic group to assigned or vice versa.

Licenses

- **Entra ID Free:** 50K objects, SSO, MFA for cloud apps
- **Entra ID P1:** Dynamic groups, Conditional Access, self-service password reset, PIM
- **Entra ID P2:** Identity Protection, PIM with just-in-time, access reviews, Entitlement Management
- **License assignment:** Per-user or group-based (recommended). Group-based auto-assigns to all members.

External Users (B2B)

- Invite via email, direct link, or self-service sign-up
- Guest accounts in your directory, authenticated by their home tenant
- Configure: guest access restrictions, collaboration restrictions (allowlist/blocklist domains)
- **Redemption:** Guest receives invite email → clicks link → consents → gains access

Self-Service Password Reset (SSPR)

- Requires Entra ID P1/P2 for writeback to on-prem AD
- **Authentication methods:** Mobile app notification, mobile app code, email, mobile phone, office phone, security questions
- **Configuration:** Number of methods required to reset (1 or 2), registration required at sign-in
- **Writeback:** Password changes in cloud sync back to on-prem AD (requires Entra Connect)

```
        # Enable SSPR for all users
        az rest --method PATCH --uri "https://graph.microsoft.com/v1.0/policies/authenticationMethodsPolicy" \
          --body '{"properties":{"allowSelfServicePasswordReset":true}}'
```

1.2 Manage Access to Azure Resources

Built-in Azure Roles

Key roles: - **Owner:** Full access + can delegate (DO NOT assign broadly) - **Contributor:** Full access, cannot delegate access - **Reader:** View all resources - **User Access Administrator:** Manage user access to Azure resources - **Storage-specific:** Storage Account Contributor, Storage Blob Data Reader, Storage Blob Data Contributor, Storage Queue Data Contributor - **Network-specific:** Network Contributor, DNS Zone Contributor, Private DNS Zone Contributor - **Compute-specific:** Virtual Machine Contributor, Virtual Machine Administrator Login, Virtual Machine User Login

Gotcha: Storage Account Contributor can manage the storage account but NOT read/write data. For data access, use Storage Blob Data * roles. Same pattern for other data plane roles.

Role Assignments

Scope hierarchy (inheritance): Management Group → Subscription → Resource Group → Resource

```
        # Assign Reader role at subscription scope
        az role assignment create --assignee john@contoso.onmicrosoft.com --role "Reader" --scope "/subscriptions/{
id}"

        # Assign Contributor at resource group scope
        az role assignment create --assignee john@contoso.onmicrosoft.com --role "Contributor" --scope
"/subscriptions/{sub-id}/resourceGroups/myRG"

        # List role assignments
        az role assignment list --resource-group myRG --include-inherited
```

Interpreting Access Assignments

- **Deny assignments:** Take precedence over allow assignments. Created by Azure (e.g., Blueprint locks, Managed Apps) and some deployments. You cannot create custom deny assignments.
- **Priority:** Explicit deny > inherited deny > explicit allow > inherited allow
- **Access evaluation:** Check deny assignments first, then allow assignments at all scopes
- **Tools:** Access Review (IAM → Check Access), `az role assignment list`, `Get-AzRoleAssignment`

1.3 Manage Azure Subscriptions and Governance

Azure Policy

Deep dive on policy effects:

Effect	Behavior	Use Case
Deny	Blocks non-compliant resource creation	Hard enforcement (allowed locations, SKUs)
Audit	Logs warning, allows creation	Soft enforcement, compliance tracking

AuditIfNotExists	Audits if a related resource doesn't exist	"VM should have extension X"
DeployIfNotExists	Auto-deploys if related resource missing	Auto-install Monitoring Agent, encryption
Modify	Adds/updates tags/properties during creation/update	Tag inheritance, enforce settings
Disabled	No effect	Temporarily disable without deleting

Policy initiatives: Group related policies. Example: "ISO 27001:2013" = 50+ policies. Assign at management group for broad coverage.

```
        # Create custom policy definition
        az policy definition create --name "require-tag-environment" \
          --display-name "Require environment tag" \
          --description "Requires an environment tag on all resources" \
          --rules '{
            "if": { "field": "tags[environment]", "exists": "false" },
            "then": { "effect": "deny" }
          }' \
          --mode Indexed

        # Assign policy
        az policy assignment create --name "require-tag-env" \
          --policy "/subscriptions/{sub-id}/providers/Microsoft.Authorization/policyDefinitions/require-tag-environ
\
          --scope "/subscriptions/{sub-id}"
```

Resource Locks

Lock Type	Can Read?	Can Modify?	Can Delete?
CanNotDelete	Yes	Yes	No
ReadOnly	Yes	No	No

- Locks are inherited. Lock on RG = lock on all resources in it.
- Locks apply to ALL users, including Owners. Must remove lock first.
- ReadOnly lock prevents any write operation, including auto-scaling and backup jobs.
- **Gotcha:** ReadOnly lock on a storage account prevents listing keys (listKeys is a POST operation).

```
        # Apply Delete lock to resource group
        az lock create --name "prod-lock" --lock-type CanNotDelete --resource-group myRG

        # Apply ReadOnly lock to a specific resource
        az lock create --name "critical-readonly" --lock-type ReadOnly --resource-group myRG --resource-name myVM -
resource-type Microsoft.Compute/virtualMachines
```

Tags

- Key-value pairs. Max 50 per resource.
- **NOT inherited** from RG to resources (use Azure Policy `Modify` effect for inheritance)
- **Tag policies:** Require tag, Add default tag, Inherit tag from RG
- Use cases: Cost tracking (cost center), environment classification, ownership, automation triggers
- **Bulk tagging:**

```
        # Tag all resources in an RG
        az resource list --resource-group myRG --query "[].id" -o tsv | xargs -I {} az resource tag --ids {} --tags
Environment=Production Owner=PlatformTeam
```

```
# Tag the RG itself
az group update --name myRG --tags Environment=Production Owner=PlatformTeam
```

Resource Groups

- **Moving resources:** Can move between RGs and subscriptions. Some resources can't be moved (Managed Identity, Key Vault in some cases, Azure AD B2C).
- **Moving across subscriptions:** Source and destination subs must be in same Entra tenant. Quota checks apply.
- **Steps:** Validate move → Execute move → Update dependencies/scripts

```
# Validate a move
az resource invoke-action --action validateMoveResources --ids "/subscriptions/{sub-id}" \
  --request-body '{"resources": ["/subscriptions/{sub-id}/resourceGroups/srcRG/providers/Microsoft.Compute/virtualMachines/myVM"], "targetResourceGroup": "/subscriptions/{sub-id}/resourceGroups/destRG"}'

# Move resources
az resource move --destination-group destRG --ids "/subscriptions/{sub-id}/resourceGroups/srcRG/providers/Microsoft.Compute/virtualMachines/myVM"
```

Subscriptions

- **Transfer subscription:** Between Entra tenants possible but complex (managed identities lost, RBAC reset)
- **Cost management per subscription:** Budgets, alerts, cost analysis
- **Limits:** Default: 25K VMs, 980 RGs. Can be increased via support request.

Management Groups

- Up to 10,000 MGs, 6 levels deep
- Root MG contains all subscriptions
- **Important:** Any MG with subscriptions as direct children cannot be deleted
- **Policy inheritance:** Policies at MG level flow down to all subscriptions and their resources

```
# Create management group
az account management-group create --name "production-mg" --display-name "Production"

# Add subscription to management group
az account management-group subscription add --name "production-mg" --subscription "{sub-id}"
```

---

2. Implement and Manage Storage (15-20%)

2.1 Configure Access to Storage

Storage Firewalls and Virtual Networks

By default, storage accounts accept traffic from all networks. Restrict to: - **Selected networks:** Specific VNets/subnets (service endpoints must be enabled), specific public IPs - **Private endpoints:** Assign private IP from VNet to storage account

```
# Enable service endpoint for Storage on a subnet
az network vnet subnet update --name mySubnet --resource-group myRG --vnet-name myVNet \
  --service-endpoints Microsoft.Storage
```

```
# Restrict storage account to specific VNets
az storage account network-rule add --account-name mystorage --vnet-name myVNet --subnet mySubnet

# Add IP rule
az storage account network-rule add --account-name mystorage --ip-address 203.0.113.0/24
```

Shared Access Signatures (SAS)

Types: - **Account SAS:** Delegates access to resources across the storage account (service-level operations) - **Service SAS:** Delegates access to a specific service (Blob, File, Queue, Table) - **User delegation SAS:** Signed with Entra ID credentials (most secure, no storage key exposure)

SAS parameters: - **Signed version (sv):** API version - **Signed services (ss):** b (blob), f (file), q (queue), t (table) - **Signed resource types (srt):** s (service), c (container), o (object) - **Signed permissions (sp):** r (read), w (write), d (delete), l (list), a (add), c (create), u (update), p (process) - **Signed expiry (se):** Expiration time - **Signed IP (sip):** Allowed IP ranges - **Signed protocol (spr):** HTTPS, HTTPSHTTP - **Signed start (st):** Start time

```
# Generate a user delegation SAS for a blob (most secure)
az storage blob generate-sas --account-name mystorage --container-name mycontainer --name myfile.txt \
  --permissions r --expiry 2026-04-24T00:00:00Z --auth-mode login --as-user
```

Best practices: - Always use HTTPS (spr=https) - Set shortest possible expiry - Use user delegation SAS when possible (no key exposure) - Use stored access policies for service SAS (allows revocation) - Never store SAS tokens in code — use Key Vault

Stored Access Policies

- Define permissions and expiry for service SAS tokens
- Can be modified or revoked at any time (invalidate all SAS tokens using that policy)
- Only supported for service SAS, not account SAS or user delegation SAS

Access Keys

- Two keys (key1, key2) per storage account. Allows rotation without downtime.
- **Rotation process:** Update all apps to use key2 → Regenerate key1 → Update apps to use key1 → Regenerate key2
- **Best practice:** Avoid using access keys directly in code. Use Entra ID authentication or SAS instead.
- **Gotcha:** Access keys provide FULL access to the storage account — read, write, delete everything. Extremely dangerous if leaked.

```
# List access keys
az storage account keys list --account-name mystorage --resource-group myRG

# Regenerate a key
az storage account keys renew --account-name mystorage --key key1
```

Identity-Based Access for Azure Files

- **Entra ID authentication over SMB:** Users authenticate with Entra ID to access file shares
- Requires: Storage account joined to domain (Entra Domain Services or AD DS)
- **NTFS-like ACLs:** Configure directory/file level permissions
- **Share-level permissions:** Azure RBAC roles (Storage File Data SMB Share Reader/Contributor/Elevated Contributor)

2.2 Configure and Manage Storage Accounts

Creating Storage Accounts

```
az storage account create --name mystorageacct123 --resource-group myRG \
  --location eastus --sku Standard_GZRS --kind StorageV2 \
  --min-tls-version TLS1_2 --allow-blob-public-access false \
  --default-action Deny
```

Key parameters: - **Kind:** StorageV2 (always), BlockBlobStorage (premium blobs), FileStorage (premium files) - **SKU:** Standard_LRS/ZRS/GRS/GZRS/RA-GRS/RA-GZRS, Premium_LRS/ZRS - **Access tier:** Hot (default), Cool, Cold - **min-tls-version:** TLS1_2 (recommended) - **allow-blob-public-access:** False (recommended — prevents anonymous blob access) - **default-action:** Deny (for network rules — block all by default)

Storage Redundancy (Deep Dive)

LRS (Locally Redundant Storage): - 3 synchronous copies in a single datacenter - Protects against: server/rack failure - Does NOT protect against: datacenter outage - Cost: Lowest - Use: Non-critical, easily reproducible data, temp/scratch data

ZRS (Zone-Redundant Storage): - 3 synchronous copies across 3 availability zones - Protects against: zone outage - Does NOT protect against: regional outage - Cost: Moderate - Use: Production workloads needing HA within a region

GRS (Geo-Redundant Storage): - LRS in primary + async replication to paired region - Protects against: regional outage (failover required) - RPO: ~15 minutes (async replication lag) - Read access requires: RA-GRS (read from secondary) - **Failover:** Customer-initiated failover converts secondary to primary

GZRS (Geo-Zone-Redundant Storage): - ZRS in primary + async replication to paired region - Best of both: zone HA + geo DR - Use: Mission-critical production workloads

Object Replication

- Asynchronous replication of block blobs between storage accounts
- **Source and destination can be in any region** (not limited to paired regions)
- Requires: versioning enabled on both accounts, point-in-time restore on destination
- **Use case:** Content distribution (replicate to regions close to users), data aggregation (centralize logs)

Storage Account Encryption

- **Microsoft-managed keys (default):** Microsoft generates and manages encryption keys. No action required.
- **Customer-managed keys (CMK):** You provide keys stored in Azure Key Vault. Full control over key lifecycle, rotation, and revocation.
- **Customer-provided keys:** You provide the encryption key per operation. Most control, most complexity.
- **Double encryption:** Infrastructure-level encryption (additional layer using Microsoft-managed keys). Available on new storage accounts.

Azure Storage Explorer and AzCopy

AzCopy:

```
# Upload file
azcopy copy "C:\data\file.txt" "https://mystorage.blob.core.windows.net/mycontainer/?sv=..."

# Download file
azcopy copy "https://mystorage.blob.core.windows.net/mycontainer/file.txt?sv=..." "C:\data\"

# Sync directory (one-way, like rsync)
azcopy sync "C:\data" "https://mystorage.blob.core.windows.net/mycontainer/?sv=..." --delete-destination tr

# Copy between storage accounts
azcopy copy "https://source.blob.core.windows.net/container/?sv=..."
"https://dest.blob.core.windows.net/container/?sv=..."
```

2.3 Configure Azure Files and Azure Blob Storage

Azure Files

- **Standard vs Premium:** Standard (HDD-backed, transaction-optimized), Premium (SSD-backed, IOPS/throughput provisioned)
- **SMB vs NFS:** SMB (Windows, authentication), NFS (Linux, no authentication — use network security)
- **Azure File Sync:** Sync on-prem files to cloud. Cloud tiering: cache hot files locally, cold files in Azure only.
- **Snapshots:** Read-only point-in-time copy. Max 200 per share. Incremental (only changes stored).
- **Soft delete:** Recover deleted files/shares. Retention: 1-365 days.

```
        # Create file share
        az storage share-rm create --name myshare --storage-account mystorage --quota 1024

        # Create snapshot
        az storage share snapshot --name myshare --account-name mystorage
```

Azure Blob Storage

- **Access tiers:** Hot, Cool, Cold, Archive (at blob level, not just account level)
- **Blob types:** Block blobs (text/binary, 190 TB), Append blobs (append-only, logs), Page blobs (VHDs, 8 TB)
- **Containers:** Organize blobs. Public access: Private (default), Blob (anonymous read blobs only), Container (anonymous read list + blobs)
- **Soft delete:** Retention 1-365 days. Recover deleted blobs and versions.
- **Versioning:** Auto-save previous version on write. Each version is a separate blob (cost implications).
- **Blob lifecycle management:** Rules to auto-tier, auto-delete, auto-archive based on age/conditions.
- **Immutable storage:** WORM (Write Once, Read Many). Legal hold or time-based retention. Cannot be modified/deleted for the duration. Required for compliance (SEC, FINRA).

```
        # Create container
        az storage container create --name mycontainer --account-name mystorage --public-access off

        # Upload blob
        az storage blob upload --account-name mystorage --container-name mycontainer --name file.txt --file
"C:\data\file.txt"

        # Set blob tier
        az storage blob set-tier --account-name mystorage --container-name mycontainer --name file.txt --tier Cool
```

Lifecycle policy example (JSON):

```
        {
          "rules": [
            {
              "name": "move-to-cool",
              "type": "Lifecycle",
              "definition": {
                "filters": { "blobTypes": ["blockBlob"] },
                "actions": {
                  "baseBlob": {
                    "tierToCool": { "daysAfterModificationGreaterThan": 30 },
                    "tierToArchive": { "daysAfterModificationGreaterThan": 90 },
                    "delete": { "daysAfterModificationGreaterThan": 2555 }
                  }
                }
              }
            }
          ]
```

```
        }
```

---

3. Deploy and Manage Azure Compute Resources (20-25%)

3.1 ARM Templates and Bicep

ARM Template Structure

```
        {
          "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
          "contentVersion": "1.0.0.0",
          "parameters": {},
          "variables": {},
          "functions": [],
          "resources": [],
          "outputs": {}
        }
```

Bicep (Recommended)

- Compiles to ARM JSON. Simpler syntax, type safety, better IDE support.
- **Key advantages over ARM JSON:**
  - No JSON boilerplate
  - Automatic dependency detection (no dependsOn needed)
  - String interpolation vs. concat()
  - Type checking at design time
  - Modules for reusability

```
// Parameters
param location string = resourceGroup().location
param vmName string
param adminUsername string
@secure()
param adminPassword string

// Resource
resource vm 'Microsoft.Compute/virtualMachines@2024-03-01' = {
  name: vmName
  location: location
  tags: {
    environment: 'production'
  }
  properties: {
    hardwareProfile: {
      vmSize: 'Standard_D2s_v5'
    }
    osProfile: {
      computerName: vmName
      adminUsername: adminUsername
      adminPassword: adminPassword
    }
    storageProfile: {
```

```
      imageReference: {
        publisher: 'Canonical'
        offer: 'UbuntuServer'
        sku: '22_04-lts'
        version: 'latest'
      }
      osDisk: {
        createOption: 'FromImage'
        managedDisk: {
          storageAccountType: 'Premium_LRS'
        }
      }
      dataDisks: [
        {
          lun: 0
          createOption: 'Empty'
          diskSizeGB: 128
          managedDisk: {
            storageAccountType: 'Premium_LRS'
          }
        }
      ]
    }
    networkProfile: {
      networkInterfaces: [
        {
          id: nic.id
        }
      ]
    }
  }
}

// Output
output vmId string = vm.id
```

Deploying

```
        # Deploy Bicep file
        az deployment group create --resource-group myRG --template-file main.bicep --parameters vmName=myVM
adminUsername=azureuser

        # Deploy ARM template
        az deployment group create --resource-group myRG --template-file azuredeploy.json --parameters @parameters.

        # Export a deployment as ARM template
        az deployment group export --resource-group myRG --name myDeployment

        # Convert ARM to Bicep (decompile)
        az bicep decompile --file azuredeploy.json
```

3.2 Create and Configure Virtual Machines

VM Creation Deep Dive

```
        # Full VM creation with all options
        az vm create \
```

```
    --resource-group myRG \
    --name myVM \
    --image Ubuntu2204 \
    --size Standard_D4s_v5 \
    --admin-username azureuser \
    --generate-ssh-keys \
    --os-disk-size-gb 128 \
    --os-disk-premium-storage \
    --data-disk-sizes-gb 256 512 \
    --vnet-name myVNet \
    --subnet mySubnet \
    --nsg myNSG \
    --public-ip-address myPublicIP \
    --availability-set myAvailSet \
    --tags Environment=Production Owner=PlatformTeam
```

Encryption at Host

- Data encrypted at rest on the VM host BEFORE written to storage
- End-to-end encryption: from VM → host → storage
- **When to use:** When you need encryption from the application all the way to storage (e.g., regulated workloads)
- **Requirements:** Specific VM series (Dv3, Ev3, etc.), must enable on the resource provider
- **Does NOT replace:** Azure Disk Encryption (ADE) or Storage Service Encryption (SSE)

Moving VMs

- **Between resource groups:** Straightforward. Update dependencies (VNets may need to stay).
- **Between subscriptions:** Same tenant. Check quotas, resource provider registration, managed identities (lost in cross-tenant moves).
- **Between regions:** Use Azure Site Recovery to replicate and failover. Not a simple move operation.

VM Sizing

Workload	Recommended Series	Example
Dev/test	B (burstable)	B2s
General purpose	Ds v5	D4s_v5
Memory-optimized	Es v5	E8s_v5
Compute-optimized	Fs v2	F8s_v2
GPU/AI	NC/ND	NC6s_v3
Storage-optimized	Ls v3	L8s_v3
HPC	HB/HC	HB120rs_v3

VM Disks

- **OS disk:** Created from image. Premium SSD recommended.
- **Data disks:** Up to 64 per VM (depends on VM size). Can be different SKU than OS disk.
- **Ultra Disk:** Sub-millisecond latency, configurable IOPS and throughput. For mission-critical databases. Requires availability zone.
- **Ephemeral OS disk:** Created on the local VM storage. Faster re-image. No storage cost. Lost on VM deallocation.

Availability Sets and Zones

```
# Create availability set
az vm availability-set create --resource-group myRG --name myAvailSet --platform-fault-domain-count 3 --
```

```
platform-update-domain-count 5

        # Create VM in specific availability zone
        az vm create --resource-group myRG --name myVM --image Ubuntu2204 --zone 1
```

VM Scale Sets

```
        # Create scale set
        az vmss create --resource-group myRG --name myScaleSet --image Ubuntu2204 \
          --upgrade-policy-mode Automatic --instance-count 3 --admin-username azureuser --generate-ssh-keys

        # Auto-scale rules
        az monitor autoscale create --resource-group myRG --resource myScaleSet \
          --min-count 2 --max-count 10 --count 3
        az monitor autoscale rule create --resource-group myRG --autoscale-name myScaleSet \
          --scale out 1 --condition "Percentage CPU > 80 avg 5m"
        az monitor autoscale rule create --resource-group myRG --autoscale-name myScaleSet \
          --scale in 1 --condition "Percentage CPU < 30 avg 5m"
```

3.3 Containers

Azure Container Registry (ACR)

- **SKUs:** Basic (dev), Standard (prod), Premium (geo-replication, private endpoints, zone-redundancy)
- **Geo-replication:** Premium only. Replicate images to multiple regions for fast pulls.
- **Tasks:** Build container images in Azure (ACR Tasks). Trigger on source code commit or base image update.
- **Security:** Content trust (signed images), vulnerability scanning (Microsoft Defender), private endpoints.

```
        # Create ACR
        az acr create --resource-group myRG --name myacr123 --sku Premium --admin-enabled true

        # Build and push image
        az acr build --registry myacr123 --image myapp:v1 .

        # Import image from Docker Hub
        az acr import --name myacr123 --source docker.io/library/nginx:latest --image nginx:latest
```

Azure Container Instances (ACI)

```
        # Create container instance
        az container create --resource-group myRG --name mycontainer \
          --image myacr123.azurecr.io/myapp:v1 --cpu 2 --memory 4 \
          --ports 80 --dns-name-label myapp-unique \
          --registry-login-server myacr123.azurecr.io \
          --registry-username myacr123 --registry-password <password>
```

Azure Container Apps

```
        # Create Container Apps environment
        az containerapp env create --name myEnv --resource-group myRG --location eastus

        # Create container app
        az containerapp create --name myapp --resource-group myRG --environment myEnv \
          --image myacr123.azurecr.io/myapp:v1 --target-port 80 --ingress external \
          --cpu 0.5 --memory 1.0Gi --min-replicas 1 --max-replicas 10
```

3.4 Azure App Service

App Service Plan

- **Tiers:** Free (F1), Shared (D1), Basic (B1-B3), Standard (S1-S3), Premium (P1v3-P3v3), Isolated (I1v2-I3v2)
- **Scaling:** Manual (Basic+), Auto-scale (Standard+), Deployment slots (Standard+)
- **Key differences:** Free/Shared = shared infrastructure, no SLA. Standard+ = dedicated instances, SLA.

Deployment Slots

- **Available:** Standard tier and above
- **Purpose:** Zero-downtime deployments, A/B testing, staging environments
- **Swap:** Swap staging slot → production slot. Traffic instantly routes to new code.
- **Swap with preview:** Validate before swapping. Warm up the staging slot.

```
        # Create deployment slot
        az webapp deployment slot create --name mywebapp --resource-group myRG --slot staging

        # Swap slots
        az webapp deployment slot swap --name mywebapp --resource-group myRG --slot staging --target-slot productio
```

App Service Configuration

```
        # Configure custom domain
        az webapp config hostname add --webapp-name mywebapp --resource-group myRG --hostname www.contoso.com

        # Configure TLS/SSL binding
        az webapp config ssl bind --certificate-thumbprint <thumbprint> --name mywebapp --resource-group myRG --ssl
SNI

        # Configure backup
        az webapp config backup update --webapp-name mywebapp --resource-group myRG \
          --storage-account mystorage --container-name backups --retention 30

        # Configure VNet integration
        az webapp vnet-integration add --name mywebapp --resource-group myRG --vnet myVNet --subnet mySubnet

        # Configure deployment source (GitHub)
        az webapp deployment source config --name mywebapp --resource-group myRG \
          --repo-url https://github.com/contoso/myapp --branch main --manual-integration
```

---

4. Implement and Manage Virtual Networking (15-20%)

4.1 Configure and Manage Virtual Networks

VNet and Subnets

```
        # Create VNet with subnets
        az network vnet create --resource-group myRG --name myVNet --address-prefix 10.0.0.0/16 \
          --subnet-name frontend --subnet-prefix 10.0.1.0/24

        az network vnet subnet create --resource-group myRG --vnet-name myVNet \
```

```
  --name backend --address-prefix 10.0.2.0/24 \
  --service-endpoints Microsoft.Storage Microsoft.Sql
```

VNet Peering

```
# Create VNet peering
az network vnet peering create --resource-group myRG --name peer1to2 \
  --vnet-name vnet1 --remote-vnet vnet2 --allow-vnet-access

# Both directions needed for full connectivity
az network vnet peering create --resource-group myRG --name peer2to1 \
  --vnet-name vnet2 --remote-vnet vnet1 --allow-vnet-access

# Gateway transit (allow vnet2 to use vnet1's VPN gateway)
az network vnet peering create --resource-group myRG --name peer2to1 \
  --vnet-name vnet2 --remote-vnet vnet1 --allow-vnet-access --allow-gateway-transit --use-remote-gateways
```

Gotcha: VNet peering is NOT transitive. If A peers with B and B peers with C, A cannot reach C. Solutions: hub-spoke with NVA, Azure Virtual WAN, or BGP with VPN gateways.

Public IP Addresses

- **Basic:** Static or Dynamic. No availability zones. Open by default. Being retired.
- **Standard:** Static only. Zone-redundant or zonal. Secure by default (must opt in to inbound flows). Recommended.

```
az network public-ip create --resource-group myRG --name myPublicIP --sku Standard --zone 1 2 3
```

User-Defined Routes (UDRs)

- Override default Azure routing. Force traffic through NVAs (firewalls, load balancers).
- **Next hop types:** Virtual Appliance, Virtual Network Gateway, Internet, None (drop), Virtual Network

```
# Create route table
az network route-table create --resource-group myRG --name myRouteTable

# Add route: send all internet traffic through NVA
az network route-table route create --resource-group myRG --route-table-name myRouteTable \
  --name to-internet --address-prefix 0.0.0.0/0 --next-hop-type VirtualAppliance --next-hop-ip-address 10.0

# Associate with subnet
az network vnet subnet update --resource-group myRG --vnet-name myVNet --name mySubnet \
  --route-table myRouteTable
```

Troubleshooting Network Connectivity

- **Network Watcher:** Regional service. Must be enabled in each region. Free.
- **IP Flow Verify:** Check if a packet is allowed/denied by NSG rules between source and destination
- **Next Hop:** Determine the next hop for a packet (find routing issues)
- **NSG Flow Logs:** Log allowed/denied traffic to Storage Account. Analyze with Traffic Analytics.
- **Connection Troubleshoot:** Check connectivity between VM and endpoint (TCP ping with path analysis)
- **Packet Capture:** Capture packets on VM NIC for deep analysis

4.2 Configure Secure Access to Virtual Networks

Network Security Groups (NSGs)

- **Inbound and outbound rules:** Priority 100-4096 (lower = evaluated first). Default: allow VNet inbound/outbound, deny internet inbound, allow internet outbound.
- **Applied to:** Subnet or NIC. Subnet NSG evaluated first, then NIC NSG for inbound.
- **ASGs (Application Security Groups):** Group VMs by workload. Use ASG in NSG rules instead of individual IPs.

```
# Create NSG
az network nsg create --resource-group myRG --name myNSG

# Allow HTTP from internet
az network nsg rule create --resource-group myRG --nsg-name myNSG \
  --name allow-http --priority 100 --direction inbound --access allow \
  --protocol tcp --destination-port-range 80 --source-address-prefixes Internet

# Allow SSH from specific IP
az network nsg rule create --resource-group myRG --nsg-name myNSG \
  --name allow-ssh --priority 110 --direction inbound --access allow \
  --protocol tcp --destination-port-range 22 --source-address-prefixes 203.0.113.0/24

# Deny all other inbound
az network nsg rule create --resource-group myRG --nsg-name myNSG \
  --name deny-all-inbound --priority 4096 --direction inbound --access deny
```

Effective Security Rules

```
# View effective rules for a NIC
az network nic list-effective-nsg --resource-group myRG --name myNic

# View effective route table
az network nic show-effective-route-table --resource-group myRG --name myNic
```

Azure Bastion

- PaaS service for secure RDP/SSH access to VMs over TLS (port 443)
- No public IPs on VMs needed. No agent on VMs.
- Deploys in a dedicated subnet (AzureBastionSubnet, minimum /26)
- **SKU:** Basic, Standard, Premium. Standard+: custom ports, shareable links.

Service Endpoints

- Extend VNet identity to Azure PaaS services
- Traffic stays on Azure backbone (not internet)
- **Supported services:** Storage, SQL, Key Vault, Service Bus, Event Hubs, Cosmos DB, and more
- **Limitations:** Only works from VNet to Azure services in the SAME region (mostly)

Private Endpoints

- Assign private IP from your VNet to an Azure PaaS resource
- **Supported:** Storage, SQL, Cosmos DB, Key Vault, App Service, and many more
- **DNS:** Private DNS zone (privatelink.blob.core.windows.net) maps to private IP
- **NSG:** Can apply NSG rules to private endpoint (subnet NSG)
- **Vs Service Endpoints:** Private endpoints provide private IP (data exfiltration protection), service endpoints provide VNet identity but still use public IPs

4.3 Name Resolution and Load Balancing

Azure DNS

- **Public DNS zones:** Host internet-facing domains. SOA, A, AAAA, CNAME, MX, NS, PTR, SRV, TXT records
- **Private DNS zones:** Resolve within VNets. Auto-registration (VMs auto-register their A records)
- **DNS resolution flow:**
  1. VM queries Azure-provided DNS (168.63.129.16)
  2. If private zone linked to VNet, returns private IP
  3. Otherwise, resolves via public DNS

Load Balancers

Azure Load Balancer (Layer 4):

Feature	Public LB	Internal LB
Frontend	Public IP	Private IP
Inbound traffic	From internet	From VNet
Use case	Web tier, public APIs	Internal microservices, database

- **SKU:** Basic (being retired), Standard (recommended)
- **Rules:** Load balancing rules (frontend → backend pool), NAT rules (frontend:port → specific backend), outbound rules
- **Health probes:** HTTP/HTTPS/TCP. Unhealthy instance removed from rotation after failed probes.
- **Session affinity:** SourceIP (2-tuple) or SourceIPProtocol (3-tuple)
- **Standard LB:** Zone-redundant by default, HTTPS probes, outbound rules, 99.99% SLA

```
# Create public load balancer
az network lb create --resource-group myRG --name myLB --sku Standard \
  --public-ip-address myPublicIP --frontend-ip-name myFrontEnd --backend-pool-name myBackEndPool

# Create health probe
az network lb probe create --resource-group myRG --lb-name myLB \
  --name myProbe --protocol http --port 80 --path /health

# Create load balancing rule
az network lb rule create --resource-group myRG --lb-name myLB \
  --name myHTTPRule --protocol tcp --frontend-port 80 --backend-port 80 \
  --frontend-ip-name myFrontEnd --backend-pool-name myBackEndPool --probe-name myProbe
```

Application Gateway (Layer 7)

- **Features:** URL-based routing, SSL termination, cookie affinity, WAF, URL rewrite, redirect
- **SKU:** Standard_v2, WAF_v2. v2 supports autoscaling and zone-redundancy.
- **WAF:** OWASP 3.2 rule set. Custom rules. Exclusions.
- **Listeners:** Frontend IP + port + protocol + host name. Basic vs multi-site.
- **Routing rules:** Listener → backend pool. Path-based routing (/images → pool1, /api → pool2).

---

5. Monitor and Maintain Azure Resources (10-15%)

5.1 Monitor Resources in Azure

Azure Monitor Components

- **Metrics:** Numeric telemetry (CPU, memory, network). 93-day retention. Export to Log Analytics or Storage.
- **Logs:** Log Analytics workspace. KQL queries. Custom logs via Data Collection Rules (DCRs). 30-730 day retention.
- **Alerts:** Metric alerts (fast), Log alerts (KQL-based), Activity log alerts (service health, policy, etc.)
- **Action Groups:** Email, SMS, push, voice, webhook, Logic App, Function, ITSM, secure webhook

- **Application Insights:** APM for web apps. Auto-instrumentation or SDK. Request tracking, dependency tracking, live metrics, availability tests, usage analysis.

```
        # Create diagnostic setting to send logs to Log Analytics
        az monitor diagnostic-settings create --resource "/subscriptions/{sub-
id}/resourceGroups/myRG/providers/Microsoft.Compute/virtualMachines/myVM" \
          --name myDiagSetting --workspace "/subscriptions/{sub-
id}/resourcegroups/myRG/providers/microsoft.operationalinsights/workspaces/myWorkspace" \
          --logs '[{"category": "Administrative","enabled": true},{"category": "Security","enabled": true}]' \
          --metrics '[{"category": "AllMetrics","enabled": true}]'
```

Network Watcher

- **IP Flow Verify:** Test NSG allow/deny for a specific flow
- **Next Hop:** Find routing path for a packet
- **NSG Flow Logs:** Log traffic to Storage Account, analyze with Traffic Analytics
- **Connection Troubleshoot:** Check TCP connectivity between VM and endpoint
- **Packet Capture:** Capture network packets for deep troubleshooting
- **Topology:** Visual map of VNet resources and connections

5.2 Implement Backup and Recovery

Recovery Services Vault vs Backup Vault

- **Recovery Services Vault:** For Azure VMs, SQL in VMs, Azure Files, SAP HANA. Supports Site Recovery.
- **Backup Vault:** Newer. For Azure Blobs, Azure Disks, and PostgreSQL. Uses Data Protection API.

Azure Backup

VM Backup: - Application-consistent (Windows) or file-system-consistent (Linux) snapshots - Snapshot stored in Recovery Services vault - Frequency: Daily, weekly. Retention: 7-9999 days, or weekly/monthly/yearly points - **Instant restore:** Restore VM directly from snapshot (not from vault) — much faster - **Soft delete:** Deleted backup data retained 14 days (configurable). Protection against ransomware.

```
        # Create Recovery Services vault
        az backup vault create --resource-group myRG --name myVault --location eastus

        # Enable VM backup
        az backup protection enable-for-vm --resource-group myRG --vault-name myVault \
          --vm myVM --policy-name DefaultPolicy

        # Trigger on-demand backup
        az backup protection backup-now --resource-group myRG --vault-name myVault \
          --container-name myVM --item-name myVM --retain-until 2026-05-01

        # Restore VM
        az backup restore restore-disks --resource-group myRG --vault-name myVault \
          --container-name myVM --item-name myVM --storage-account mystorage
```

Azure Site Recovery

- **Business continuity:** Replicate VMs from primary region to secondary region
- **RTO:** Minutes (depends on VM size and application startup time)
- **RPO:** Near-zero (continuous async replication)
- **Failover:** Test failover (no impact), Planned failover (no data loss), Unplanned failover (minimal data loss)
- **Failback:** Reverse replication after failover

```
# Create Recovery Services vault for Site Recovery
az backup vault create --resource-group myRG --name myASRVault --location eastus2

# Enable replication for a VM
az site-recovery fabric create --resource-group myRG --vault-name myASRVault --name eastus-fabric
# (Full ASR setup requires multiple steps - fabric, container, mapping, policy, protection)
```

Backup Reports and Alerts

- **Backup Reports:** Power BI-based reports on backup jobs, storage usage, compliance
- **Alerts:** Built-in alerts for backup failures, configured via Azure Monitor
- **Monitoring:** Backup center for centralized management across vaults

---

Quick Reference: AZ-104 Key Commands

Task	Command
Create VM	`az vm create`
Create VNet	`az network vnet create`
Create NSG	`az network nsg create`
Create storage account	`az storage account create`
Create App Service	`az webapp create`
Deploy Bicep	`az deployment group create --template-file`
Assign RBAC	`az role assignment create`
Assign Policy	`az policy assignment create`
Create backup	`az backup protection enable-for-vm`
Create alert	`az monitor metrics alert create`

Appendix A: Hands-On Labs for AZ-104

Lab 1: Complete Enterprise Governance Setup

Objective: Set up management groups, policies, RBAC, and locks for a production environment.

```
# Step 1: Create management group hierarchy
az account management-group create --name mg-production --display-name "Production"
az account management-group create --name mg-nonprod --display-name "Non-Production"
az account management-group create --name mg-sandbox --display-name "Sandbox"

# Step 2: Move subscriptions into management groups
az account management-group subscription add --name mg-production --subscription "{prod-sub-id}"
az account management-group subscription add --name mg-nonprod --subscription "{dev-sub-id}"

# Step 3: Apply policies at management group level

# 3a: Allowed locations (production = East US + West US 2 only)
az policy assignment create --name "prod-allowed-locations" \
  --display-name "Production: Allowed Locations" \
```

```
  --policy "/providers/Microsoft.Authorization/policyDefinitions/e56962a6-4747-49cd-b67b-bf8b01975c4c" \
  --params '{"listOfAllowedLocations": {"value": ["eastus", "westus2"]}}' \
  --scope "/providers/Microsoft.Management/managementGroups/mg-production"

# 3b: Require tag on all resources
az policy assignment create --name "require-costcenter" \
  --display-name "Require CostCenter tag" \
  --policy "/providers/Microsoft.Authorization/policyDefinitions/1e30110a-5ceb-460c-a204-c1c394928265" \
  --params '{"tagName": {"value": "CostCenter"}}' \
  --scope "/providers/Microsoft.Management/managementGroups/mg-production"

# 3c: Enforce HTTPS on storage accounts
az policy assignment create --name "enforce-https-storage" \
  --display-name "Enforce HTTPS on Storage Accounts" \
  --policy "/providers/Microsoft.Authorization/policyDefinitions/404c3081-a854-4457-ae30-26a93ef643f9" \
  --scope "/providers/Microsoft.Management/managementGroups/mg-production"

# 3d: Allowed VM SKUs (prevent expensive VMs in sandbox)
az policy assignment create --name "sandbox-allowed-skus" \
  --display-name "Sandbox: Allowed VM SKUs" \
  --policy "/providers/Microsoft.Authorization/policyDefinitions/cccc23c7-8427-4f53-ad10-5f2bc4e97cfe" \
  --params '{"listOfAllowedSKUs": {"value": ["Standard_B2s", "Standard_B2ms", "Standard_D2s_v5"]}}' \
  --scope "/providers/Microsoft.Management/managementGroups/mg-sandbox"

# Step 4: Create RBAC structure
# 4a: Platform team = Contributor at production MG
az role assignment create --assignee platform-team@contoso.com \
  --role Contributor \
  --scope "/providers/Microsoft.Management/managementGroups/mg-production"

# 4b: Dev team = Contributor at non-prod MG
az role assignment create --assignee dev-team@contoso.com \
  --role Contributor \
  --scope "/providers/Microsoft.Management/managementGroups/mg-nonprod"

# 4c: Security team = Reader at root + Security Reader at root
az role assignment create --assignee security-team@contoso.com \
  --role Reader \
  --scope "/providers/Microsoft.Management/managementGroups/{root-mg-id}"
az role assignment create --assignee security-team@contoso.com \
  --role "Security Reader" \
  --scope "/providers/Microsoft.Management/managementGroups/{root-mg-id}"

# Step 5: Apply resource locks on production
az lock create --name "prod-delete-lock" --lock-type CanNotDelete \
  --resource-group rg-prod-shared --notes "Production lock - requires CTO approval to remove"
```

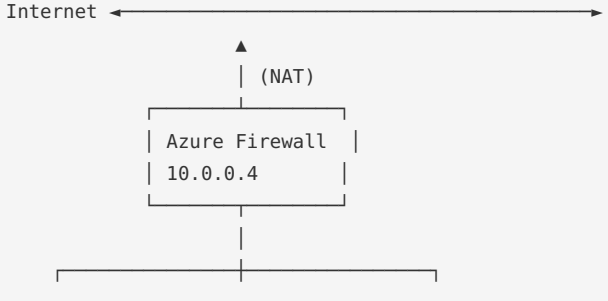
Lab 2: Implement VNet with Azure Firewall (Hub-Spoke)

Objective: Build a hub-spoke topology with Azure Firewall for traffic inspection.

```
# Hub VNet
az network vnet create --resource-group rg-hubspoke --name vnet-hub \
  --address-prefix 10.0.0.0/16 --location eastus

az network vnet subnet create --resource-group rg-hubspoke --vnet-name vnet-hub \
  --name AzureFirewallSubnet --address-prefix 10.0.0.0/26
```

```
az network vnet subnet create --resource-group rg-hubspoke --vnet-name vnet-hub \
  --name GatewaySubnet --address-prefix 10.0.1.0/27

# Spoke 1 VNet
az network vnet create --resource-group rg-hubspoke --name vnet-spoke1 \
  --address-prefix 10.1.0.0/16 --location eastus

az network vnet subnet create --resource-group rg-hubspoke --vnet-name vnet-spoke1 \
  --name snet-workload --address-prefix 10.1.1.0/24

# Peering: Hub <-> Spoke1
az network vnet peering create --resource-group rg-hubspoke --name hub-to-spoke1 \
  --vnet-name vnet-hub --remote-vnet vnet-spoke1 \
  --allow-vnet-access --allow-forwarded-traffic

az network vnet peering create --resource-group rg-hubspoke --name spoke1-to-hub \
  --vnet-name vnet-spoke1 --remote-vnet vnet-hub \
  --allow-vnet-access --allow-forwarded-traffic --allow-gateway-transit

# Deploy Azure Firewall
az network firewall create --resource-group rg-hubspoke --name azfw-hub \
  --location eastus --sku Standard

az network public-ip create --resource-group rg-hubspoke --name fw-pip --sku Standard

az network firewall ip-config create --resource-group rg-hubspoke --firewall-name azfw-hub \
  --name fw-config --public-ip-address fw-pip --vnet-name vnet-hub

# Firewall rule: Allow HTTP/HTTPS to internet from spokes
az network firewall application-rule create --resource-group rg-hubspoke \
  --firewall-name azfw-hub --collection-name allow-web \
  --name allow-http-https --action Allow --priority 100 \
  --source-addresses 10.1.0.0/16 10.2.0.0/16 \
  --protocols Http=80 Https=443 --target-fqdns "*"

# UDR: Force spoke traffic through firewall
FIREWALL_PRIVATE_IP=$(az network firewall show --resource-group rg-hubspoke \
  --name azfw-hub --query "ipConfigurations[0].properties.privateIPAddress" -o tsv)

az network route-table create --resource-group rg-hubspoke --name rt-spoke1
az network route-table route create --resource-group rg-hubspoke --route-table-name rt-spoke1 \
  --name to-internet --address-prefix 0.0.0.0/0 --next-hop-type VirtualAppliance \
  --next-hop-ip-address $FIREWALL_PRIVATE_IP

# Associate route table to spoke subnet
az network vnet subnet update --resource-group rg-hubspoke --vnet-name vnet-spoke1 \
  --name snet-workload --route-table rt-spoke1
```

Architecture diagram:



```
    |    Hub VNet (10.0.0.0/16)     |
    |   AzureFirewallSubnet         |
    |   GatewaySubnet               |
    └──────┬──────────────────┬─────┘
           | peering          | peering
           ▼                  ▼
┌──────────────────┐    ┌──────────────────┐
| Spoke1 VNet      |    | Spoke2 VNet      |
| 10.1.0.0/16      |    | 10.2.0.0/16      |
| UDR → FW         |    | UDR → FW         |
| Workload VMs     |    | Data Services    |
└──────────────────┘    └──────────────────┘
```

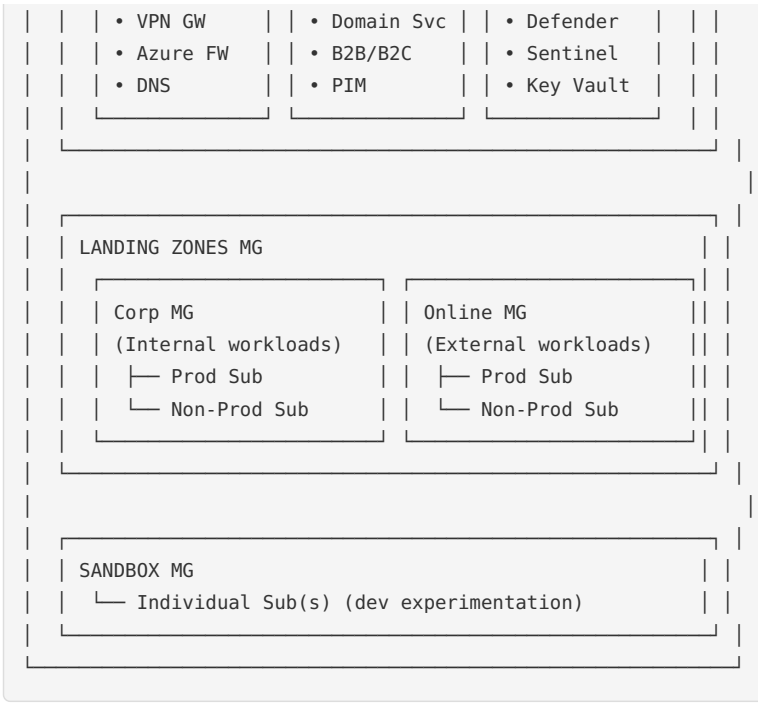
Lab 3: Azure Backup and Site Recovery Setup

```bash
# Step 1: Create Recovery Services vault
az backup vault create --resource-group rg-dr --name rsv-dr --location eastus

# Step 2: Enable backup for a VM
az backup protection enable-for-vm \
  --resource-group rg-dr --vault-name rsv-dr \
  --vm vm-prod-app1 --policy-name DefaultPolicy

# Step 3: Run on-demand backup
az backup protection backup-now \
  --resource-group rg-dr --vault-name rsv-dr \
  --container-name vm-prod-app1 --item-name vm-prod-app1 \
  --retain-until 2026-05-23

# Step 4: Configure Site Recovery for cross-region DR
# (Requires Azure CLI recovery services extension)
# Create vault in secondary region
az backup vault create --resource-group rg-dr-westus --name rsv-dr-westus --location westus

# Step 5: Create and run a test failover
# Via Portal: Recovery Services vault → Site Recovery → Test Failover
# Or via CLI with az site-recovery commands
```

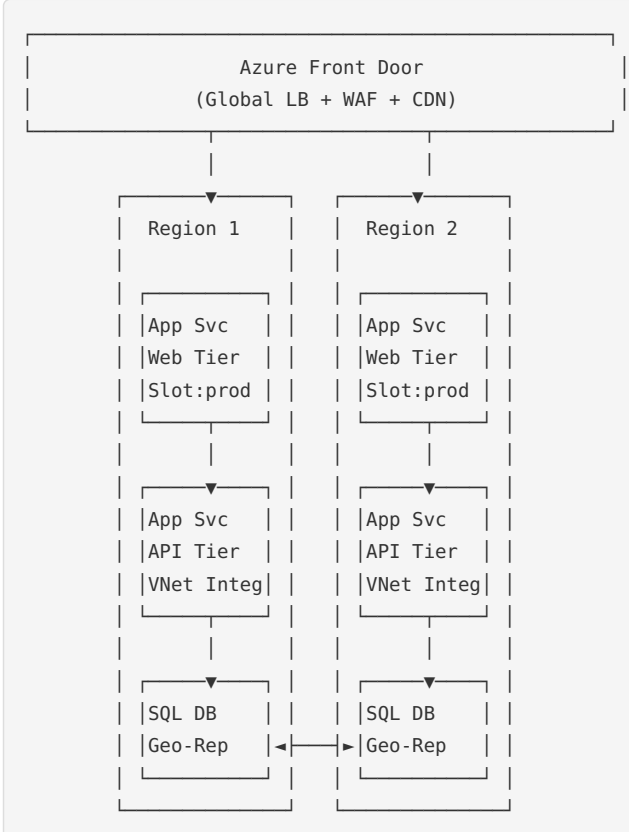
Appendix B: Architecture Patterns

Enterprise Landing Zone (Cloud Adoption Framework)

```
┌──────────────────────────────────────────────────────────┐
|                  ROOT MANAGEMENT GROUP                   |
|                                                          |
|  ┌──────────────────────────────────────────────────┐    |
|  | PLATFORM MG                                      |    |
|  |  ┌──────────────┐ ┌──────────────┐ ┌──────────┐  |    |
|  |  | Connectivity | | Identity     | | Management |  |    |
|  |  | Subscription | | Subscription | | Subscription |  |    |
|  |  |              | |              | |          |  |    |
|  |  | • Hub VNets  | | • Entra      | | • Monitor  |  |    |
|  |  | • ExpressRt  | |   Connect    | | • Automation |  |    |
```



App Service Multi-Tier Architecture



Appendix C: AZ-104 Troubleshooting Guide

VM Issues

Problem	Likely Cause	Resolution
Can’t RDP/SSH	NSG blocking port	Check NSG rules, use Network Watcher IP Flow Verify
VM won’t start	Quota exceeded	Check vCPU quota for the VM series in the region
Slow performance	Undersized VM or disk	Check CPU/memory metrics, consider Premium SSD or upgrade VM
Deployment failed	Policy denial	Check Azure Policy compliance, look for deny assignments
Can’t move VM	Resource type not supported	Check move support: https://learn.microsoft.com/azure/azure-resource-manager/management/move-support-resources

Storage Issues

Problem	Likely Cause	Resolution
Authentication failed	Wrong auth method	Use Entra ID RBAC, check role assignments
SAS token expired	Time limit reached	Generate new SAS or use user delegation SAS
Can’t access from VNet	Firewall blocking	Add VNet/service endpoint to storage network rules
Blob access denied	Public access disabled + no auth	Enable Entra ID auth or use SAS token
High storage cost	Hot tier for cold data	Implement lifecycle management policies

Networking Issues

Problem	Likely Cause	Resolution
Can’t reach on-prem	VPN tunnel down	Check VPN gateway status, BGP status, shared key
VNet peering not working	Asymmetric routing	Check peering settings on both sides, gateway transit
Private endpoint not resolving	Missing DNS zone	Create private DNS zone and link to VNet
Load balancer unhealthy	Probe failing	Check probe path, backend port, health endpoint
NSG not blocking traffic	Rule priority issue	Check effective rules (lower priority = evaluated first)

Appendix D: AZ-104 Key Limits

Resource	Limit
VMs per availability set	200
VMs per VMSS	1,000 (600 with custom image)
NICs per VM	Depends on VM size (1-8)

Data disks per VM	Depends on VM size (2-64)
VNet peering per VNet	500
NSG rules per NSG	3,000 (inbound + outbound)
Route tables per subscription	500
Routes per route table	400
Storage accounts per subscription	250
File shares per storage account	30,000
Backup retention days	7-9,999
Recovery Services vaults per subscription	500