

AZ-500 - Comprehensive Azure Study Guide

AZ-500: Microsoft Azure Security Technologies — Comprehensive Study Guide

Audience: Experienced Azure Infrastructure Architect relearning security with deep, hands-on detail. Exam Version: Skills measured as of January 22, 2026 [△](#) This exam retires August 31, 2026

1. Secure Identity and Access (15-20%)

1.1 Manage Security Controls for Identity and Access

Azure Built-in Role Assignments

Key security-related roles: - **Security Reader:** View security features in Defender for Cloud - **Security Admin:** View and configure security features - **Key Vault Contributor:** Manage key vaults (NOT access to secrets) - **Key Vault Secrets User:** Read secret values - **Key Vault Crypto User:** Read key values, perform crypto operations - **Managed Identity Contributor:** Create/read/update/delete managed identities - **Managed Identity Operator:** Assign/remove managed identity from resources

Separation of duties: - Key Vault Contributor can manage the vault but NOT read secrets - Key Vault Secrets Officer can manage secrets but NOT the vault - Key Vault Crypto Officer can manage keys but NOT the vault - This separation prevents a single person from both managing infrastructure and accessing secrets

Custom Roles

When built-in roles don't fit, create custom roles:

```
# Create custom role from JSON definition
az role definition create --role-definition '{
  "Name": "Custom VM Operator",
  "Description": "Can restart and deallocate VMs but not create/delete",
  "Actions": [
    "Microsoft.Compute/virtualMachines/read",
    "Microsoft.Compute/virtualMachines/restart/action",
    "Microsoft.Compute/virtualMachines/deallocate/action",
    "Microsoft.Compute/virtualMachines/start/action"
  ],
  "NotActions": [],
  "AssignableScopes": ["/subscriptions/{sub-id}"]
}'
```

Entra ID custom roles: - Manage in Entra ID → Roles & administrators → New custom role - Actions: *microsoft.directory/users/*, *microsoft.directory/groups/*, etc. - More granular than Azure RBAC (e.g., “reset password” vs “full user management”)

Privileged Identity Management (PIM)

PIM provides: - **Just-in-time (JIT) access:** Activate roles only when needed, for a limited time - **Time-bound access:** Set start and end dates for role assignments - **Approval workflow:** Require approval for role activation - **MFA requirement:** Require MFA to activate a role - **Justification:** Require a business justification for activation - **Audit**

trail: Full history of role activations

Key concepts: - **Eligible assignment:** User CAN activate the role (not permanently active) - **Active assignment:** User HAS the role (permanently active — use sparingly) - **Activation:** Eligible user activates → gets role for configured duration (default 8 hours) - **Approval:** Can require one or more approvers for activation

Configuration:

```
# Configure PIM for a role (via Graph API)
# Maximum activation duration: 0.5 - 24 hours
# Require MFA on activation: true
# Require justification on activation: true
# Require ticket information on activation: true
# Approval required: true
```

Best practices: - Make all privileged roles eligible (not permanently active) - Require MFA for all role activations - Set maximum activation duration to minimum needed (1-4 hours) - Require approval for high-privilege roles (Owner, User Access Admin) - Regular access reviews (quarterly) to remove unneeded assignments - Alert on suspicious activations (activation outside business hours)

Multi-Factor Authentication (MFA)

Enforcement methods: 1. **Per-user MFA:** Enable per user in Entra ID. Legacy method. All or nothing. 2. **Conditional Access:** Modern method. Granular control (require MFA for specific apps, locations, risk levels). 3. **Security defaults:** Free tier. Enables MFA for all users. Blocks legacy authentication. All or nothing.

MFA methods (ordered by security): 1. FIDO2 security keys (phishing-resistant, strongest) 2. Microsoft Authenticator app (number matching, passwordless) 3. Windows Hello for Business 4. SMS (weakest, vulnerable to SIM swap) 5. Voice call (weakest)

NIST compliance: Only FIDO2 and Authenticator app meet NIST AAL2/AAL3 requirements.

Conditional Access Policies

Deep dive:

Assignments (the IF): - **Users:** All users, specific users/groups, exclude (always exclude break-glass accounts!) - **Cloud apps:** All cloud apps, specific apps (Office 365, Azure Portal, etc.) - **Conditions:** - User risk (Entra ID Protection): Low/Medium/High - Sign-in risk: Low/Medium/High - Device platforms: Windows, macOS, iOS, Android - Locations: Any location, trusted locations, specific countries - Client apps: Browser, mobile/desktop apps, Exchange ActiveSync, other clients (legacy auth) - Device state: Compliant, hybrid joined, both

Access controls (the THEN): - **Grant:** Block access, or Grant access with: - Require MFA - Require device to be marked as compliant - Require hybrid Azure AD joined device - Require approved client app - Require app protection policy - Require password change (for high user risk) - **Session:** - Use app enforced restrictions - Use Conditional Access App Control (real-time monitoring) - Sign-in frequency (re-auth every X hours) - Persistent browser session - Customize continuous access evaluation

Key policies to implement:

1. Require MFA for all users (exclude break-glass accounts)
2. Block legacy authentication (all clients > other clients)
3. Require compliant device for Azure Portal
4. Require MFA for risky sign-ins (sign-in risk > medium)
5. Require password change for high user risk
6. Block access from unsupported locations
7. Require MFA for privileged role activation

Report-only mode: Test policy impact before enabling. Review the sign-in logs to see what would have been blocked/granted.

1.2 Manage Microsoft Entra Application Access and Managed Identities

Enterprise Application Access

- **Enterprise apps:** Pre-integrated SaaS apps + apps published via App Proxy
- **User assignment required:** By default, all users can access an enterprise app. Enable “User assignment required” to restrict access.
- **OAuth permission grants:** Users can consent to app permissions. Control via:
 - User consent: Allow/Deny user consent to apps
 - Admin consent: Require admin approval for certain permissions
 - Consent policy: Configure which permissions users can consent to

App Registrations

- **App registration:** Create an identity for your application in Entra ID
- **Application ID (Client ID):** Unique identifier for the app
- **Directory (Tenant) ID:** Your Entra ID tenant ID
- **Authentication:** Configure redirect URLs, supported account types
- **Certificates & secrets:** Client secrets (auto-generated, expire) or certificates (you manage)
- **API permissions:** Delegated (on behalf of user) vs Application (app-only, no user)

Permission Scopes and Consent

Delegated permissions: App acts on behalf of a user. User or admin consents. **Application permissions:** App acts as itself. Only admin can consent. More powerful.

Consent types: - **User consent:** User agrees to delegated permissions. Can be disabled org-wide. - **Admin consent:** Global admin agrees to permissions. Required for application permissions. - **Incremental consent:** Request permissions as needed, not all upfront.

Service Principals

- A service principal is the local representation of an app registration in a tenant
- **Types:** Application (identity for an app), Managed Identity (auto-managed identity), Legacy (created before app registrations)
- **Enterprise apps = service principals** in the Entra portal

```
# Create service principal for an app
az ad sp create --id <application-id>

# Create service principal with certificate
az ad sp create-for-rbac --name myApp --cert @myCert.pem --skip-assignment
```

Managed Identities

System-assigned: - One identity per Azure resource. Lifecycle tied to the resource. - When resource is deleted, identity is deleted. - Use for: Single resource accessing Azure services

User-assigned: - Standalone identity resource. Can be assigned to multiple resources. - Independent lifecycle. Deleted separately. - Use for: Multiple resources sharing the same identity, pre-created identities for DR

```
# Enable system-assigned managed identity on a VM
az vm identity assign --resource-group myRG --name myVM

# Create user-assigned managed identity
az identity create --resource-group myRG --name myUserIdentity

# Assign user-assigned identity to a VM
az vm identity assign --resource-group myRG --name myVM --identities myUserIdentity
```

Using managed identities: - With Azure services: VM -> IMDS endpoint (169.254.169.254) -> get token for resource - **Token flow:** 1. VM requests token from IMDS: GET http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://storage.azure.com 2. IMDS returns access token 3. VM uses token to authenticate to Azure Storage

Gotcha: Managed identities cannot be used across tenants. If you need cross-tenant access, use service principals instead.

2. Secure Networking (20-25%)

2.1 Plan and Implement Security for Virtual Networks

NSGs and ASGs

NSG rule evaluation: Lowest priority number wins. Rules evaluated 100→4096. First match wins.

ASG usage:

```
# Create ASG
az network asg create --resource-group myRG --name webAsg
az network asg create --resource-group myRG --name dbAsg

# NSG rule using ASG instead of IP addresses
az network nsg rule create --resource-group myRG --nsg-name myNSG \
  --name allow-web-to-db --priority 100 --access allow --direction inbound \
  --source-asgs webAsg --destination-asgs dbAsg --destination-port-ranges 1433 --protocol tcp
```

Azure Virtual Network Manager

- Centralized network management across subscriptions
- **Network groups:** Logical grouping of VNets
- **Configurations:**
 - **Connectivity:** Mesh or hub-spoke topology. Auto-provision peering.
 - **Security admin rules:** Higher priority than NSGs. Enforce/block specific traffic across VNets.

User-Defined Routes

- Force traffic through NVAs (firewalls, IDS/IPS)
- Override default Azure routing
- **Common pattern:** Hub-spoke with Azure Firewall in hub. All spoke traffic routed through firewall via UDR.

VNet Peering and VPN Gateway Security

VNet peering security: - Peering itself has no security controls (it's a connectivity construct) - Security is enforced via NSGs on subnets and NVAs - **Gateway transit:** Allow spoke VNets to use hub's VPN/ExpressRoute gateway -

Forwarded traffic: Control whether traffic from a peered VNet can flow through the local VNet

VPN security: - IPsec/IKE encryption (AES-256, SHA-384, DH Group 24) - P2S: IKEv2, OpenVPN, SSTP. Certificate-based or Entra ID authentication. - S2S: Pre-shared key (PSK) or certificate-based. Use custom IPsec/IKE policies for stronger encryption.

Virtual WAN and Secured Virtual Hub

- **Virtual WAN:** Unified networking service. Connect branches via VPN/ExpressRoute/SD-WAN.
- **Secured virtual hub:** Virtual hub with Azure Firewall integrated. All traffic through hub is inspected.
- **Secured by default:** Firewall policies applied at hub level. All branch and VNet traffic inspected.

ExpressRoute Encryption

- **MACsec (IEEE 802.1AE):** Layer 2 encryption on ExpressRoute Direct. End-to-end between your routers and Microsoft.
- **IPsec over ExpressRoute:** Layer 3 encryption. Configure VPN gateway on ExpressRoute circuit. Double encryption possible.

Network Watcher Security Features

- **NSG Flow Logs:** Capture allowed/denied flows. Store in Storage Account. Analyze with Traffic Analytics.
- **Traffic Analytics:** Visualize network activity. Identify security issues (unauthorized access attempts, data exfiltration).
- **Packet Capture:** Deep packet inspection for security analysis.

2.2 Plan and Implement Security for Private Access to Azure Resources

Service Endpoints vs Private Endpoints

Feature	Service Endpoint	Private Endpoint
Access method	VNet identity over public endpoint	Private IP in VNet
Data exfiltration risk	Possible (DNS still resolves to public IP)	Eliminated (private IP)
NSG support	On subnet level	On NIC/private endpoint
On-prem access	No (VNet only)	Yes (via ExpressRoute/VPN)
DNS	No change (public FQDN)	Private DNS zone required
Cost	Free	Per-endpoint + hourly charge

Best practice: Use Private Endpoints for all PaaS services in production. Service Endpoints for non-critical or when cost is a constraint.

Private Link Service

- Create your own private link service (your service behind a standard load balancer, consumed by others via private endpoints)
- **Use case:** SaaS providers offering private connectivity to their service

App Service and Functions Network Security

- **VNet integration:** Outbound traffic from app goes through VNet (access on-prem resources)
- **Private endpoints:** Inbound traffic to app via private IP
- **Access restrictions:** IP-based and VNet-based inbound restrictions
- **Service endpoints:** Restrict inbound access to specific VNets
- **App Service Environment (ASE):** Fully isolated app hosting in your VNet. All traffic stays in VNet.

SQL Managed Instance Network Security

- Must be deployed in its own subnet (dedicated, delegated)
- Requires NSG and route table on the subnet
- Managed by Azure (you don't fully control the NSG rules — Azure adds required rules)

2.3 Plan and Implement Security for Public Access to Azure Resources

TLS for Applications

- **App Service:** Minimum TLS version configurable (1.2 recommended). Client certificates for mutual TLS.
- **API Management:** Backend and frontend TLS. Custom CA certificates. Mutual TLS.

Azure Firewall

- **SKUs:** Standard (L3-L7 filtering), Premium (TLS inspection, IDPS, URL filtering, web categories)
- **Deployment:** In a dedicated subnet (AzureFirewallSubnet, minimum /26)
- **Features:**
 - **DNAT:** Translate inbound traffic to backend (port forwarding)
 - **Network rules:** L3/L4 filtering (IP, port, protocol)
 - **Application rules:** FQDN-based filtering (allow *.microsoft.com*, deny *.facebook.com*)
 - **NAT rules:** Source NAT for outbound, Destination NAT for inbound
 - **Threat intelligence:** Alert/deny traffic from known malicious IPs
 - **TLS inspection (Premium):** Decrypt, inspect, re-encrypt traffic. Requires CA certificate deployment to clients.
 - **IDPS (Premium):** Intrusion detection and prevention. Signature-based and anomaly-based.
 - **DNS proxy:** Forward DNS queries through firewall. Log and filter DNS requests.

```
# Create Azure Firewall
az network firewall create --resource-group myRG --name myFirewall --location eastus --sku Premium

# Add application rule
az network firewall application-rule create --resource-group myRG --firewall-name myFirewall \
  --collection-name allow-msft --action Allow --priority 100 --name allow-microsoft \
  --source-addresses 10.0.0.0/16 --protocols http=80 https=443 --target-fqdns *.microsoft.com
```

Azure Firewall Manager

- Centralized policy management for Azure Firewalls
- **Firewall policies:** Hierarchical (parent/child). Child inherits from parent. Override at child level.
- **Secured virtual hubs:** Deploy firewall in Virtual WAN hub
- **DNS proxy:** Centralized DNS resolution through firewall

Azure Application Gateway with WAF

- **WAF modes:** Detection (log only), Prevention (log and block)
- **Rule sets:** OWASP 3.2 (default), OWASP 2.2.9, Microsoft Default Rule Set 2.0
- **Custom rules:** Priority-based, match conditions (IP, string, size, geo), action (allow/deny/log)
- **Exclusions:** Exclude specific request attributes from WAF evaluation (reduce false positives)
- **Rate limiting:** Limit requests per client IP. Protect against DDoS and brute force.

Azure Front Door and CDN

- **Front Door:** Global load balancer + WAF + CDN. Layer 7 routing.
- **CDN:** Content caching at edge. Azure CDN from Microsoft, Akamai, Verizon.
- **Security features:** WAF integration, custom WAF rules, bot protection, private link to origins
- **TLS:** End-to-end TLS. Managed certificates (free) or bring your own.

DDoS Protection

Tier	Features	Cost
Basic	Free, automatic, always-on for all Azure resources	Free
Standard	Adaptive tuning, attack analytics, cost protection, rapid response support	\$199/month + usage

Standard features: - **Adaptive tuning:** Learns your traffic patterns, auto-adjusts thresholds - **Attack analytics:** Detailed metrics and reports during attacks - **Cost protection:** Service credits for scale-out costs during DDoS attacks - **Rapid response:** DDoS Rapid Response (DRR) team engagement during active attacks

3. Secure Compute, Storage, and Databases (20-25%)

3.1 Plan and Implement Advanced Security for Compute

Remote Access to VMs

Azure Bastion: - RDP/SSH over TLS (port 443) in browser - No public IP on VMs. No agent on VMs. - SKU: Basic, Standard (custom ports, shareable links), Premium (private-only) - **Hardening:** Restrict NSG on AzureBastionSubnet to only allow HTTPS from trusted IPs

Just-in-Time (JIT) VM Access: - Part of Microsoft Defender for Cloud - NSG rules block management ports (RDP 3389, SSH 22) by default - When needed, user requests access via Portal/API → NSG rule opened for specified time → auto-closed after timeout - **Integration:** Works with PIM for approval workflows

AKS Security

Network isolation: - **CNI:** Each pod gets IP from VNet subnet. Network policies (Calico, Azure) for pod-level isolation. - **kubenet:** Pods get IPs from address range, NAT through node IP. Simpler but less isolation. - **Private cluster:** API server has no public IP. Access via private endpoint.

Security best practices: - Use Azure AD (Entra ID) integration for cluster authentication - Use Azure RBAC for Kubernetes authorization - Use pod-managed identities (Workload Identity) instead of storing secrets - Enable Azure Policy for AKS (enforce pod security, restrict privileged containers) - Use Azure Defender for Kubernetes (threat detection, vulnerability scanning) - Restrict egress traffic with Azure Firewall or UDRs - Use secrets-store-csi-driver with Azure Key Vault provider for secrets management

AKS authentication: - **Entra ID integration:** Kubernetes RBAC backed by Entra ID groups - **Managed identity:** Cluster uses managed identity for Azure resource access - **Workload Identity:** Pods use managed identities via federated credentials (replaces pod identity)

Container Security

ACR security: - Content trust (signed images) - Vulnerability scanning (Defender for Container Registries) - Private endpoints for network isolation - Admin account disabled by default (use Entra ID or service principal) - **Token access:** Fine-grained permissions for pull/push per repository

ACI/ACA security monitoring: - Enable Azure Defender for Containers - Use managed identities for Azure resource access - Network isolation via VNet injection (ACI) or internal environment (ACA)

Disk Encryption

Method	Scope	Key Management	When to Use
Azure Disk Encryption (ADE)	OS + data disks	Key Vault (customer-managed)	When you need OS-level encryption
Encryption at host	Temp disk + caches	Platform-managed or CMK	When you need end-to-end encryption
Confidential disk encryption	VM disk	Managed by VM	When you need encryption within the VM
Server-side encryption (SSE)	All managed disks	Platform-managed or CMK	Default, always enabled

ADE requirements: - VM must be in same region as Key Vault - Key Vault must have soft delete and purge protection enabled - Key Vault access policy must grant the VM access - ADE encrypts using BitLocker (Windows) or DM-Crypt (Linux)

```
# Enable ADE on a VM
az vm encryption enable --resource-group myRG --name myVM \
  --disk-encryption-keyvault myKeyVault --key-encryption-key myKEK
```

API Management Security

- **Authentication:** OAuth 2.0, OpenID Connect, client certificates, managed identity
- **Authorization:** Validate JWT tokens, check scopes/roles
- **Rate limiting:** Limit calls per subscription/key
- **IP filtering:** Allow/deny specific IPs
- **Policies:** Validate, transform, and cache requests. Inbound, backend, outbound, on-error policies.

3.2 Plan and Implement Security for Storage

Access Control for Storage Accounts

Authorization hierarchy: 1. **Entra ID (recommended):** Use RBAC roles (Storage Blob Data Owner/Reader/Contributor). Most secure. 2. **Shared Key (access key):** Full access. Least secure. Avoid in production. 3. **SAS tokens:** Delegated access. Time-limited. Use user delegation SAS (Entra ID-backed). 4. **Anonymous public access:** Disable at account level (allow-blob-public-access false).

Azure Files Access

- **SMB with Entra ID:** Authenticate with Entra ID over SMB. ACLs on files/folders.
- **SMB with AD DS:** On-prem AD authentication. Requires Entra Connect sync.
- **NFS:** No authentication. Use network security (private endpoints, NSGs).
- **Identity-based:** Use Storage File Data SMB Share roles for share-level permissions.

Blob Storage Access

- **Access tiers:** Hot, Cool, Cold, Archive (at account or blob level)
- **Anonymous access:** Can be enabled per container (disabled at account level recommended)
- **SAS for blobs:** Service SAS, Account SAS, User delegation SAS (most secure)

Data Protection

- **Soft delete:** Recovery of deleted blobs (1-365 days), containers (1-365 days), file shares (1-365 days)
- **Versioning:** Auto-versions on write. Each version stored separately (cost!).
- **Immutable storage:** WORM policy. Legal hold or time-based retention. Cannot be overridden.
- **Point-in-time restore:** Restore blob data to a previous state (requires versioning + soft delete)
- **Backup:** Azure Backup for blobs (operational backup, no agent, uses point-in-time restore)

Customer-Managed Keys (BYOK)

```
# Configure CMK for storage account
az storage account update --name mystorage --resource-group myRG \
  --encryption-key-vault myKeyVault --encryption-key-name myKey --encryption-key-version <version>
```

- Key must be in Key Vault with soft delete and purge protection
- Auto-rotation: Enable auto-rotation to use latest key version
- If key is revoked or deleted, storage account becomes inaccessible

Infrastructure Double Encryption

- Additional layer of encryption at the infrastructure level
- Uses Microsoft-managed keys (separate from SSE CMK)
- Enabled at storage account creation
- Use when: regulated workloads requiring multiple encryption layers

3.3 Plan and Implement Security for Azure SQL

Entra ID Authentication

- **Entra ID admin:** Must be set first. Can be a user or group.
- **Contained database users:** Create users in each database mapped to Entra ID identities
- **Benefits:** Centralized identity, MFA, Conditional Access, audit trail

```
-- Create contained database user from Entra ID
CREATE USER [john@contoso.com] FROM EXTERNAL PROVIDER;
-- Grant access
ALTER ROLE db_datareader ADD MEMBER [john@contoso.com];
```

Database Auditing

- Tracks database events and writes to audit log
- **Destination:** Log Analytics workspace, Storage Account, Event Hub
- **Server-level vs database-level:** Server-level applies to all databases. Database-level overrides.
- **Events tracked:** Login failures, data modifications, schema changes, permission changes

Dynamic Data Masking

- Masks sensitive data in query results (does NOT change stored data)
- **Masking rules:** Defined per column per table
- **Masking functions:**
 - Default: Full masking based on data type (XXX for strings, 0 for numbers)
 - Email: aXXX@XXXX.com
 - Random: Random number for numeric types
 - Custom string: Custom prefix/padding/suffix
- **Unmasked access:** Granted via SQL permissions (UNMASK). Entra ID admins always unmasked.

```
-- Add dynamic data mask to a column
ALTER TABLE Customers ALTER COLUMN Email ADD MASKED WITH (FUNCTION = 'email()');
-- Grant unmask permission
GRANT UNMASK TO [john@contoso.com];
```

Transparent Data Encryption (TDE)

- Encrypts data at rest: database files, log files, backups
- Enabled by default on all new Azure SQL databases
- **Service-managed TDE:** Microsoft manages the TDE protector (DEK encrypted by service-managed key)
- **Customer-managed TDE:** DEK encrypted by your key in Key Vault. Full control.
- **Always Encrypted:** Client-side encryption. Data encrypted in the client driver BEFORE being sent to the database. Database never sees plaintext. Column-level. Use for: PII, PHI, financial data.

4. Secure Azure Using Microsoft Defender for Cloud and Microsoft Sentinel (30-35%)

4.1 Implement and Manage Enforcement of Cloud Governance Policies

Azure Policy for Security

Security-specific policies: - Require encryption on storage accounts - Require SQL TDE - Require NSGs on subnets - Audit diagnostic settings - Require approved VM extensions - Deny resource types (e.g., deny public load balancers)

Initiatives for security: - Microsoft Cloud Security Benchmark initiative (150+ policies) - NIST SP 800-53 R5 - CIS Microsoft Azure Foundations Benchmark - ISO 27001

Azure Key Vault

Network settings: - **Firewall:** Allow public access from specific networks, or deny all public access - **Private endpoints:** Access Key Vault via private IP (recommended for production) - **Trusted services:** Allow Azure services (Backup, Compute, etc.) to bypass firewall

Access models: - **Vault access policy (legacy):** Granular per-operation permissions (get, list, set, delete, backup, recover, purge for each of keys/secrets/certificates) - **Azure RBAC (recommended):** Built-in roles for Key Vault. Easier management, consistent with Azure RBAC.

RBAC roles: - Key Vault Administrator: Full management - Key Vault Secrets User: Read secrets - Key Vault Crypto User: Perform crypto operations with keys - Key Vault Certificates Officer: Manage certificates

Key, Secret, and Certificate Management

Keys: RSA and EC keys for encryption, signing, wrapping. **Secrets:** Any string value (passwords, connection strings, API keys). Max 25KB. **Certificates:** X.509 certificates. Auto-renewal from integrated CAs (DigiCert, GlobalSign).

Key rotation: - **Auto-rotation:** Configure automatic key rotation on a schedule (e.g., every 90 days) - **Manual rotation:** Create new key version. Update application to use new version. - **Application impact:** If using key identifier without version, auto-rotation is transparent. If using specific version, must update reference.

```
# Create key with auto-rotation policy
az keyvault key create --vault-name myKeyVault --name myKey --key RSA --size 2048
az keyvault key rotation-policy update --vault-name myKeyVault --name myKey \
  --value '{"lifetimeActions":[{"trigger":{"timeAfterCreate":"P90D"},"action":"Rotate"}],"attributes":
{"expiryTime":"P365D"}}'
```

Backup and Recovery of Key Vault Objects

- **Soft delete:** Enabled by default (90-day retention). Recover deleted vaults/objects.
- **Purge protection:** Permanent delete after retention period. Must explicitly purge.
- **Backup:** Can backup individual keys, secrets, certificates. Stored as encrypted blob.

4.2 Manage Security Posture with Microsoft Defender for Cloud

Secure Score

- Measures security posture as a percentage (0-100%)
- Based on completed security recommendations
- **Controls:** Grouped recommendations (e.g., "Enable MFA" has 2 recommendations = max 10 points)
- **Maximum score:** Varies based on your resources and recommendations applicable

Compliance Assessment

- Compare your configuration against security frameworks
- **Built-in standards:** Azure CIS, NIST SP 800-53, ISO 27001, SOC 2, PCI DSS, SWIFT CSP
- **Custom standards:** Create your own compliance standards
- **Compliance dashboard:** Visual overview of pass/fail per control per standard
- **Compliance reports:** Download PDF/CSV reports for auditors

Multi-Cloud Security

- **AWS integration:** Connect AWS account via CloudFormation stack. Defender assesses EC2, S3, EKS, etc.
- **GCP integration:** Connect GCP project via service account. Defender assesses GCE, GKE, Cloud Storage, etc.
- **Defender for Containers:** Protect Kubernetes across Azure (AKS), AWS (EKS), GCP (GKE), and Arc-enabled clusters

Microsoft Defender External Attack Surface Management (EASM)

- Discovers your internet-facing assets (domains, IPs, ports, services)
- Continuous monitoring for new exposures
- Identifies vulnerabilities and misconfigurations on external surfaces
- **Use case:** Know your external attack surface. Find shadow IT and forgotten assets.

4.3 Configure and Manage Threat Protection

Cloud Workload Protection Plans

Plan	Protects	Key Features
Defender for Servers	Windows/Linux VMs, Arc servers	Vulnerability scanning (Qualys), file integrity monitoring, adaptive application controls, JIT VM access
Defender for Databases	SQL (Azure, VMs, MI), open-source	Vulnerability assessment, anomaly detection
Defender for Storage	Blob, Files, Data Lake	Malware scanning, sensitivity discovery, anomaly detection
Defender for Containers	AKS, Arc K8s, ACR, ACI	Vulnerability scanning on images and runtime, K8s-level threat detection
Defender for App Service	Web apps, API apps, function apps	Runtime threat detection, WAF integration
Defender for Key Vault	Key Vault	Suspicious access patterns, unusual operations
Defender for DNS	Azure DNS	DNS-based threat detection (malware C2, data exfil)
Defender for Resource Manager	ARM operations	Suspicious ARM operations, privilege escalation

Agentless Scanning for VMs

- No agent required on VMs
- Scans VM disks using cloud-native API (snapshot-based)
- Complements agent-based scanning (not a replacement)
- **Benefits:** Zero deployment effort, no performance impact on VM, covers unmanaged VMs

Defender Vulnerability Management

- Powered by Microsoft Defender Vulnerability Management (MDVM, formerly Qualys)
- Scans VMs for missing security updates, configuration issues, CVEs
- **Continuous scanning:** New scans triggered on changes. On-demand scanning available.
- **Integration:** Secure Score, recommendations, alerts

DevOps Security

- Connect GitHub, Azure DevOps, and GitLab repositories
- **Features:**
 - Secret scanning: Detect leaked secrets (keys, passwords, tokens)
 - Code scanning: SAST analysis for vulnerabilities
 - Dependency scanning: SCA for vulnerable dependencies
 - Infrastructure as Code scanning: Analyze ARM/Bicep/Terraform for misconfigurations
 - Pull request annotations: Inline security findings in PRs

4.4 Configure and Manage Security Monitoring and Automation

Security Alert Management

- **Alert severity:** Informational, Low, Medium, High
- **Alert lifecycle:** Active → Dismissed → Resolved
- **Response actions:** Investigate, dismiss (false positive), remediate, escalate
- **Automation:** Workflow automation triggered by alert severity/type

Workflow Automation

- Automatically trigger Logic Apps on specific Defender for Cloud events
- **Triggers:** Security alert, recommendation, regulatory compliance change
- **Use cases:**
 - Auto-create ticket in ITSM when High severity alert fires
 - Auto-notify security team via Teams/Slack
 - Auto-remediate specific recommendations

Data Collection Rules (DCRs)

- Define what data to collect from VMs and where to send it
- **Data sources:** Performance counters, Windows event logs, Linux syslog, custom logs, IIS logs
- **Destinations:** Log Analytics workspace, Azure Monitor metrics, Azure Storage, Event Hub
- **Association:** Link DCR to specific VMs (via Azure Monitor Agent)

Microsoft Sentinel

SIEM + SOAR — Security Information and Event Management + Security Orchestration, Automation, and Response.

Data connectors: - Microsoft 365 (Exchange, SharePoint, Teams) - Azure Activity logs - Microsoft Defender for Cloud alerts - Entra ID (sign-in logs, audit logs, provisioning logs) - Third-party: Palo Alto, Cisco, Fortinet, Check Point, AWS CloudTrail, etc. - Custom: CEF, Syslog, REST API

Analytics rules: - **Scheduled:** KQL query on a schedule (e.g., “failed logins > 10 in 5 minutes”) - **NRT (Near Real-Time):** KQL query running continuously (within minutes of event) - **Microsoft incident creation:** Group related alerts into incidents - **Rule logic:** - Query: KQL that identifies suspicious activity - Alert threshold: Number of results to trigger - Grouping: Group alerts by entity, time window - Mapping: Map entities (account, host, IP) for investigation graph

Automation (Playbooks): - Logic Apps triggered by Sentinel events - **Trigger types:** Alert, Incident, Entity - **Use cases:** - Enrich alerts with threat intelligence - Block IP in firewall on detection - Send notification to SOC - Auto-close benign alerts - Create investigation package

```
# Enable Microsoft Sentinel on a Log Analytics workspace
az sentinel workspace create --resource-group myRG --workspace-name myWorkspace
```

Quick Reference: AZ-500 Key Security Controls

Area	Key Controls
Identity	PIM, Conditional Access, MFA, Managed Identities
Network	Private Endpoints, Azure Firewall, WAF, DDoS Standard, NSGs
Compute	JIT VM Access, ADE, AKS security, Defender for Servers
Storage	SAS tokens, CMK, immutable storage, soft delete, private endpoints
Database	Entra ID auth, TDE, Always Encrypted, dynamic masking, auditing
Monitoring	Defender for Cloud, Sentinel, NSG Flow Logs, DCRs
Governance	Azure Policy, Key Vault, RBAC, PIM

Appendix A: Hands-On Security Labs

Lab 1: Implement Conditional Access + PIM

Objective: Build a complete identity security posture with Conditional Access policies and PIM.

```
# Step 1: Create a Conditional Access policy (via Graph API)
# Policy: Require MFA for all users accessing Azure Portal

# First, get the service principal ID for Azure Portal
AZURE_PORTAL_SP_ID=$(az ad sp list --display-name "Windows Azure Service Management API" --query "[0].id" -
tsv)

# Create the policy via Graph API
az rest --method POST --uri "https://graph.microsoft.com/v1.0/identity/conditionalAccess/policies" \
--body '{
  "displayName": "Require MFA for Azure Portal",
  "state": "enabled",
  "conditions": {
    "applications": {
      "includeApplications": ["797f4846-ba00-4fd7-b8b2-5ffc5e4cfe8c"]
    },
    "users": {
      "includeUsers": ["All"],
      "excludeUsers": ["break-glass@contoso.com"]
    }
  },
  "grantControls": {
    "builtInControls": ["mfa"]
  }
}'

# Step 2: Create a policy to block legacy authentication
az rest --method POST --uri "https://graph.microsoft.com/v1.0/identity/conditionalAccess/policies" \
--body '{
  "displayName": "Block Legacy Authentication",
  "state": "enabled",
  "conditions": {
    "applications": {
      "includeApplications": ["All"]
    },
    "users": {
      "includeUsers": ["All"],
      "excludeUsers": ["break-glass@contoso.com"]
    },
    "clientAppTypes": ["exchangeActiveSync", "other"]
  },
  "grantControls": {
    "builtInControls": ["block"]
  }
}'

# Step 3: Configure PIM for Owner role
# Via Portal: Entra ID → Identity Governance → Privileged Identity Management → Azure resources
# Settings:
#   - Maximum activation duration: 4 hours
#   - Require MFA on activation: Yes
```

```
# - Require justification: Yes
# - Require approval: Yes (security-team as approver)
# - Require ticket information: Yes
```

Lab 2: Secure Storage Account End-to-End

Objective: Lock down a storage account with private endpoints, CMK encryption, and audit logging.

```
# Step 1: Create Key Vault (for CMK)
az keyvault create --resource-group rg-securestorage --name kv-securestorage \
  --location eastus --sku Premium --enable-purge-protection true \
  --enable-soft-delete true --default-action Deny

# Step 2: Create key for storage encryption
az keyvault key create --vault-name kv-securestorage --name storage-encryption-key --key-type RSA --size 2048

# Step 3: Create storage account with CMK
az storage account create --resource-group rg-securestorage --name stsecure$(openssl rand -hex 4) \
  --location eastus --sku Standard_GZRS --kind StorageV2 \
  --min-tls-version TLS1_2 --allow-blob-public-access false \
  --default-action Deny --encryption-key-vault kv-securestorage \
  --encryption-key-name storage-encryption-key

# Step 4: Create private endpoint for blob
az network vnet create --resource-group rg-securestorage --name vnet-secure \
  --address-prefix 10.10.0.0/16 --subnet-name snet-privatelink --subnet-prefix 10.10.1.0/24

# Disable private endpoint network policies on the subnet
az network vnet subnet update --resource-group rg-securestorage --vnet-name vnet-secure \
  --name snet-privatelink --private-endpoint-network-policies Disabled

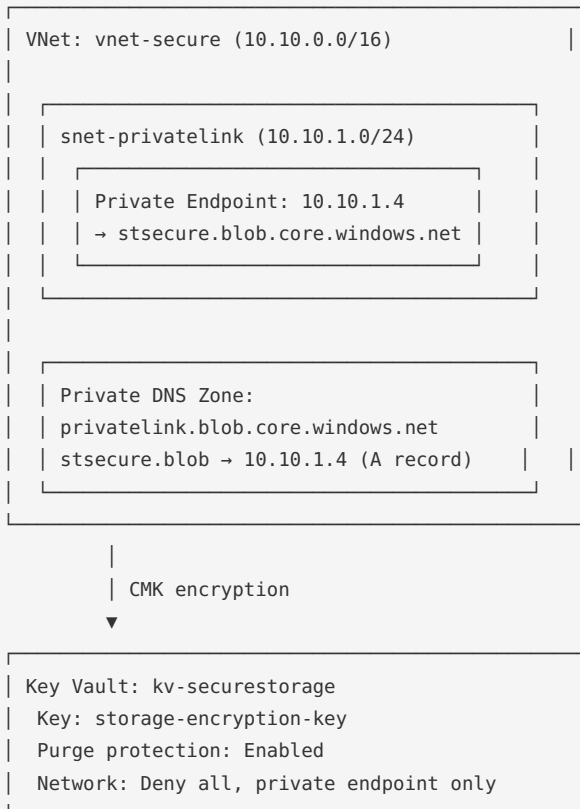
az network private-endpoint create --resource-group rg-securestorage \
  --name pe-securestorage-blob --vnet-name vnet-secure --subnet snet-privatelink \
  --private-connection-resource-id "/subscriptions/{sub-id}/resourceGroups/rg-securestorage/providers/Microsoft.Storage/storageAccounts/stsecurexxxxxxx" \
  --group-id blob --connection-name pe-connection

# Step 5: Create private DNS zone
az network private-dns zone create --resource-group rg-securestorage \
  --name privatelink.blob.core.windows.net

az network private-dns link vnet create --resource-group rg-securestorage \
  --zone-name privatelink.blob.core.windows.net --name dnslink-secure \
  --virtual-network vnet-secure --registration-enabled false

# Step 6: Configure diagnostic settings for audit logging
az monitor diagnostic-settings create \
  --resource "/subscriptions/{sub-id}/resourceGroups/rg-securestorage/providers/Microsoft.Storage/storageAccounts/stsecurexxxxxxx" \
  --name storage-audit --workspace "/subscriptions/{sub-id}/resourceGroups/rg-securestorage/providers/microsoft.operationalinsights/workspaces/law-secure" \
  --logs '[{"category": "StorageRead", "enabled": true}, {"category": "StorageWrite", "enabled": true}, {"category": "StorageDelete", "enabled": true}]' \
  --metrics '[{"category": "Transaction", "enabled": true}]'
```

Architecture diagram:



Lab 3: Deploy Azure Firewall Premium with TLS Inspection

```
# Step 1: Create VNet with firewall subnet
az network vnet create --resource-group rg-fwlab --name vnet-fwlab \
  --address-prefix 10.0.0.0/16 --location eastus

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

az network vnet subnet create --resource-group rg-fwlab --vnet-name vnet-fwlab \
  --name snet-workload --address-prefix 10.0.1.0/24

# Step 2: Create Premium Firewall
az network firewall create --resource-group rg-fwlab --name azfw-premium \
  --location eastus --sku Premium --tier Premium

# Step 3: Configure IDPS (Intrusion Detection and Prevention)
az network firewall policy create --resource-group rg-fwlab --name fwpolicy-premium \
  --location eastus --sku Premium --threat-intel-mode Alert

# Enable IDPS in alert mode initially (switch to Deny after tuning)
az network firewall policy intrusion-detection add --resource-group rg-fwlab \
  --policy-name fwpolicy-premium --mode Alert

# Step 4: Add application rules
az network firewall policy rule-collection-group create --resource-group rg-fwlab \
  --policy-name fwpolicy-premium --name app-rcg --priority 100

az network firewall policy rule-collection-group collection add --resource-group rg-fwlab \
  --policy-name fwpolicy-premium --rule-collection-group-name app-rcg \
```

```

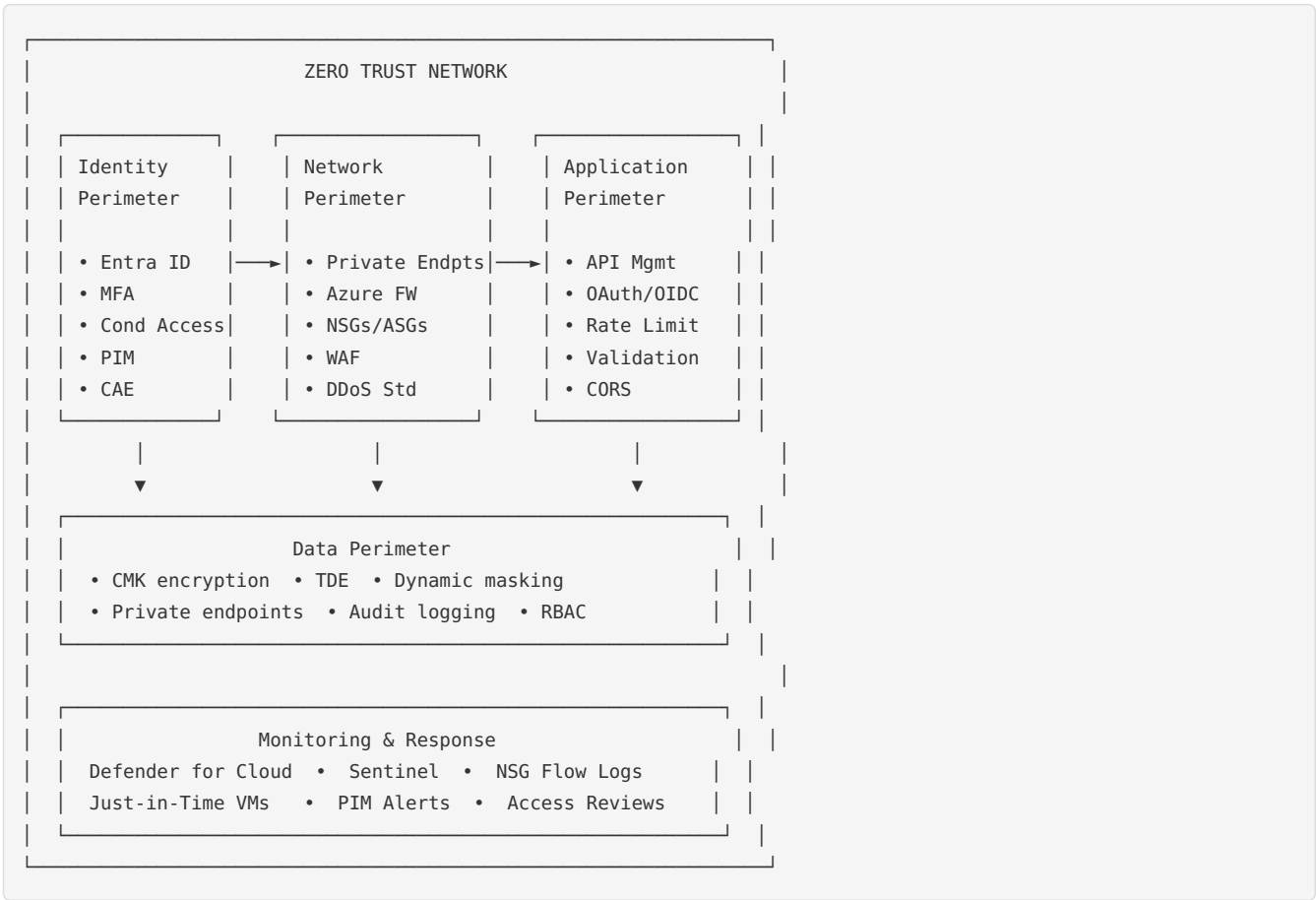
--name allow-web --collection-type ApplicationRuleCollection --priority 100 \
--action Allow --rule-name allow-msft --rule-type ApplicationRule \
--protocols Http=80 Https=443 --target-fqdns "*.microsoft.com" --source-addresses 10.0.1.0/24

# Step 5: Associate policy with firewall
az network firewall update --resource-group rg-fwlab --name azfw-premium \
--firewall-policy fwpolicy-premium

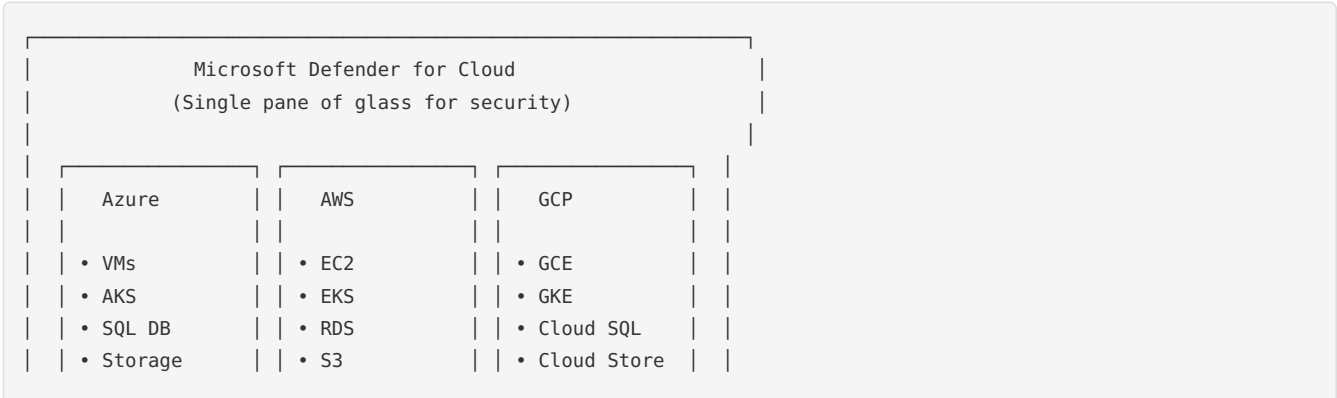
```

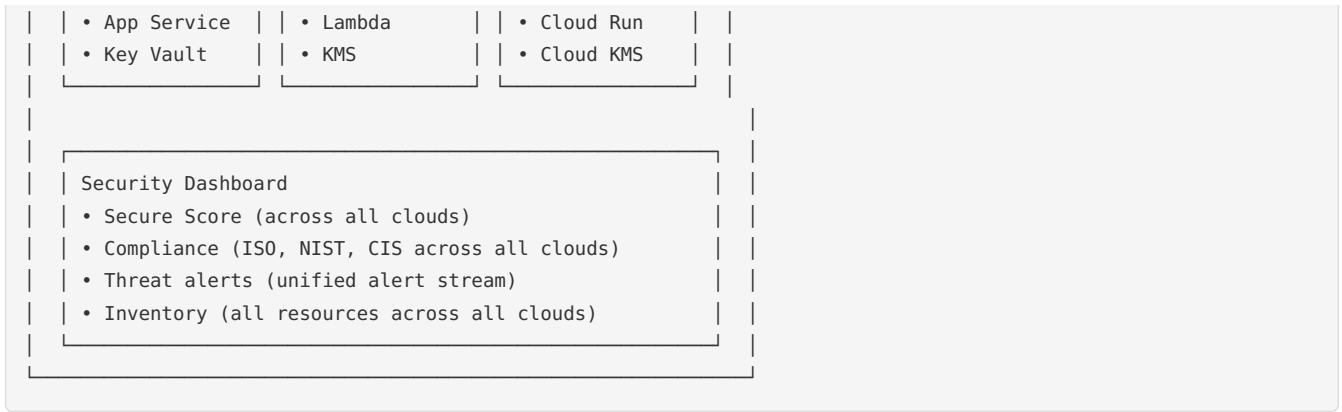
Appendix B: Security Architecture Patterns

Zero Trust Network Architecture



Multi-Cloud Security with Defender for Cloud





Appendix C: Security Hardening Checklist

Identity & Access

- ☐ Enable MFA for all users (Conditional Access, not per-user)
- ☐ Block legacy authentication protocols
- ☐ Configure break-glass accounts (excluded from all CA policies)
- ☐ Enable PIM for all privileged roles
- ☐ Configure access reviews (quarterly at minimum)
- ☐ Disable user consent for applications (require admin consent)
- ☐ Enable password protection (ban common passwords)
- ☐ Configure named locations (trusted IPs)
- ☐ Enable Continuous Access Evaluation (CAE)
- ☐ Review and remove stale guest accounts

Network

- ☐ Private endpoints for all PaaS services
- ☐ Azure Firewall for outbound traffic inspection
- ☐ WAF on Application Gateway / Front Door
- ☐ DDoS Standard on public-facing VIPs
- ☐ NSGs on all subnets (deny all inbound by default)
- ☐ Force tunneling through firewall (UDRs)
- ☐ Disable RDP/SSH from internet (use Bastion + JIT)
- ☐ Enable NSG Flow Logs + Traffic Analytics
- ☐ Use ExpressRoute encryption (MACsec or IPsec)
- ☐ Restrict VNet peering to approved VNets only

Compute

- ☐ Enable Azure Disk Encryption or encryption at host
- ☐ Enable JIT VM access
- ☐ Configure endpoint protection (Defender for Servers)
- ☐ Vulnerability scanning (Defender Vulnerability Management)
- ☐ AKS: Use Workload Identity, Azure RBAC, network policies
- ☐ App Service: VNet integration, private endpoints, disable remote debugging
- ☐ Container security: Scan images in ACR, use private endpoints

Storage & Data

- ☐ Disable public access on storage accounts
- ☐ Enforce minimum TLS 1.2
- ☐ Use CMK encryption for critical workloads
- ☐ Implement immutable storage for compliance
- ☐ Enable soft delete and versioning
- ☐ Use user delegation SAS instead of access keys
- ☐ Azure SQL: Enable auditing, TDE, dynamic masking
- ☐ Use Always Encrypted for PII/PHI
- ☐ Entra ID authentication for SQL (not SQL auth)

Monitoring & Response

- ☐ Enable Microsoft Defender for Cloud (all plans)
- ☐ Deploy Microsoft Sentinel
- ☐ Configure analytics rules for key scenarios
- ☐ Set up automation playbooks for common responses
- ☐ Enable diagnostic settings on all resources
- ☐ Configure alert rules for critical events
- ☐ Regularly review Secure Score and remediate
- ☐ Conduct quarterly access reviews

Appendix D: Key Security Benchmarks

Microsoft Cloud Security Benchmark (MCSB) - Key Controls

Control Domain	Key Controls
Identity Management	MFA for all users, conditional access, PIM, passwordless
Privileged Access	JIT access, least privilege, break-glass accounts, access reviews
Network Security	Private endpoints, microsegmentation, WAF, DDoS protection
Data Protection	Encryption at rest and in transit, CMK, data classification
Asset Management	Inventory, vulnerability scanning, endpoint protection
Logging & Monitoring	Centralized logging, SIEM (Sentinel), alerting, audit trails
Incident Response	IR plan, Sentinel playbooks, communication plan, post-incident review
Posture Management	Secure Score, compliance dashboards, continuous assessment