Microsoft

AZ-500

Tag 3

Secure cloud resources with Microsoft security technologies

Guten Morgen!

Agenda

1 Secure identity and access

2 Secure networking

3 Secure compute, storage, and databases

4 Secure Azure using Microsoft Defender for Cloud and Microsoft Sentinel

Learning Path: Secure compute, storage, and databases

Plan and implement advanced security for compute

Plan and implement security for storage

Plan and implement security for Azure SQL Database and Azure SQL Managed Instance

Module labs

Lab 4 Kubernetes
Lab 5 SQL
Lab 6 Storage
Lab 7 Key Vault SQL Always Encrypt ←

Learning Objectives

After completing this learning path, you will be able to:

1. Strengthen compute security through Azure Bastion, AKS configurations, container monitoring, and advanced encryption techniques.
2. Enhance storage security with tailored access controls, protective measures against threats, and multiple encryption strategies.
3. Bolster Azure SQL Database protection through authentication, auditing, data classification, and advanced encryption recommendations.

Plan and implement advanced security for compute

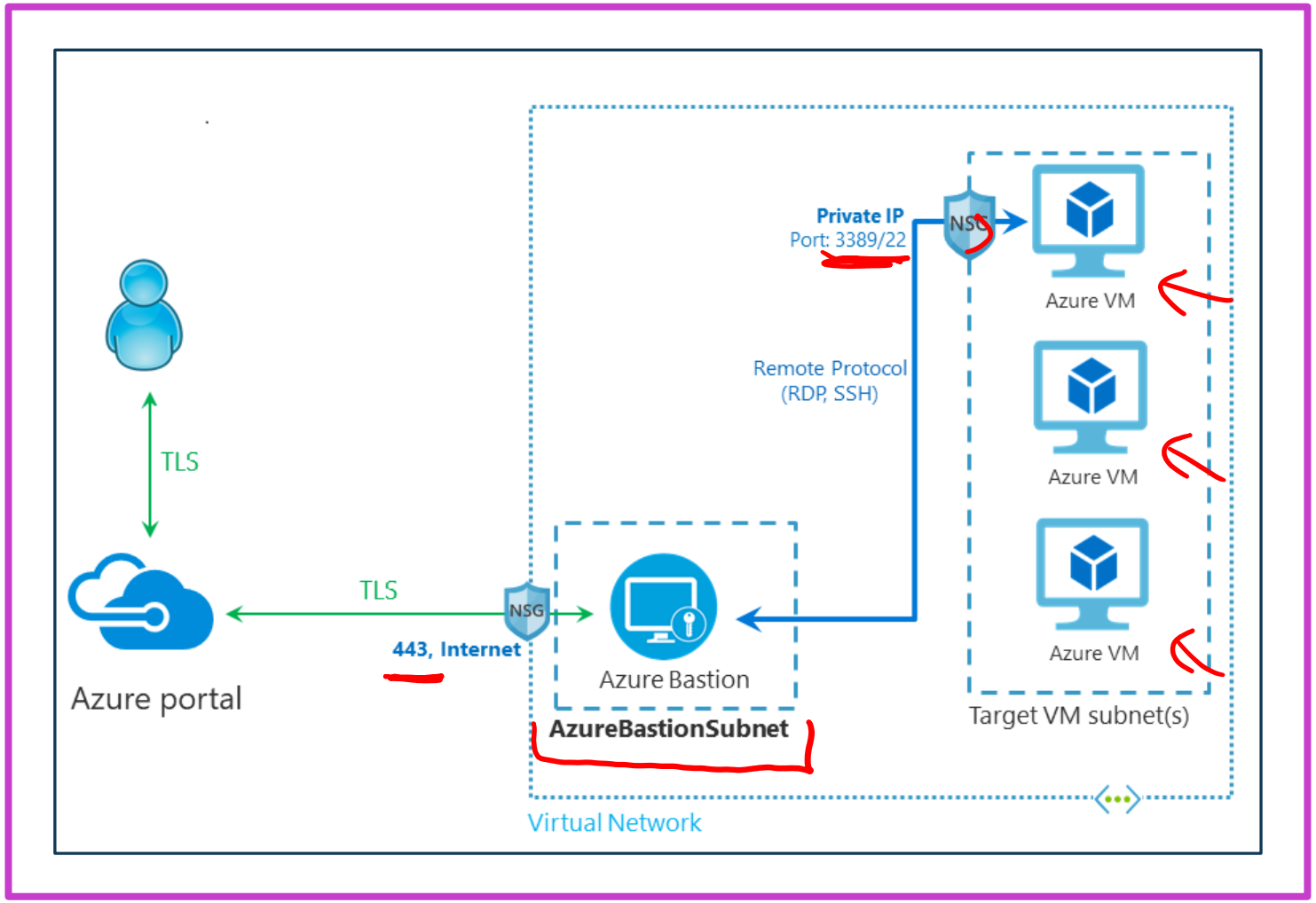
MCSB Security Controls: Data Protection, Identity Management, Network Security, and Privileged Access

- Data Security: Discover, classify, monitor threats, and encrypt sensitive data in transit and at rest.
- Key Management: Use customer-managed keys and secure key management processes for encryption controls.
- Identity Protection: Centralize authentication, restrict credential exposure, and manage identity lifecycles securely.
- Network and Access Control: Secure cloud services, deploy edge firewalls, and enforce least privilege access principles.

Windows Server Core
notepad
regedit
mmc

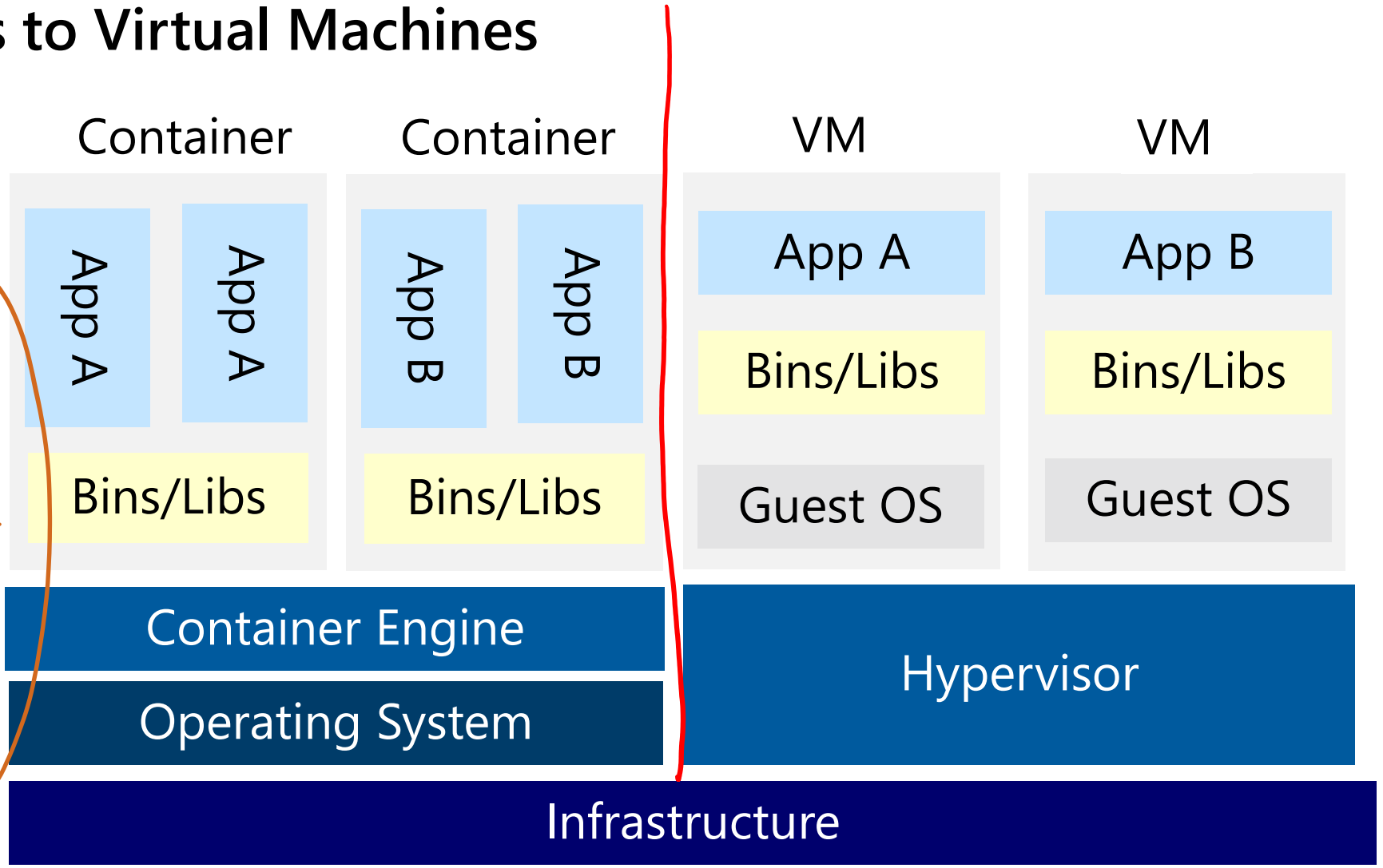
Plan and implement remote access to public endpoints, including Azure Bastion

- Secure VM Access: Azure Bastion provides agentless RDP/SSH over TLS using private IPs, eliminating public exposure.
- SKU-Based Features: Developer is basic, Standard adds scaling, Premium enables session recording and private-only mode.
- Simplified Management: No public IPs, NSGs, or separate bastion hosts needed for secure connectivity.
- Scalability & Redundancy: Standard+ supports host scaling; availability zones are in preview for select regions.



Compare Containers to Virtual Machines

- Isolation
- Operating System
- Deployment
- Persistent storage
- Fault tolerance

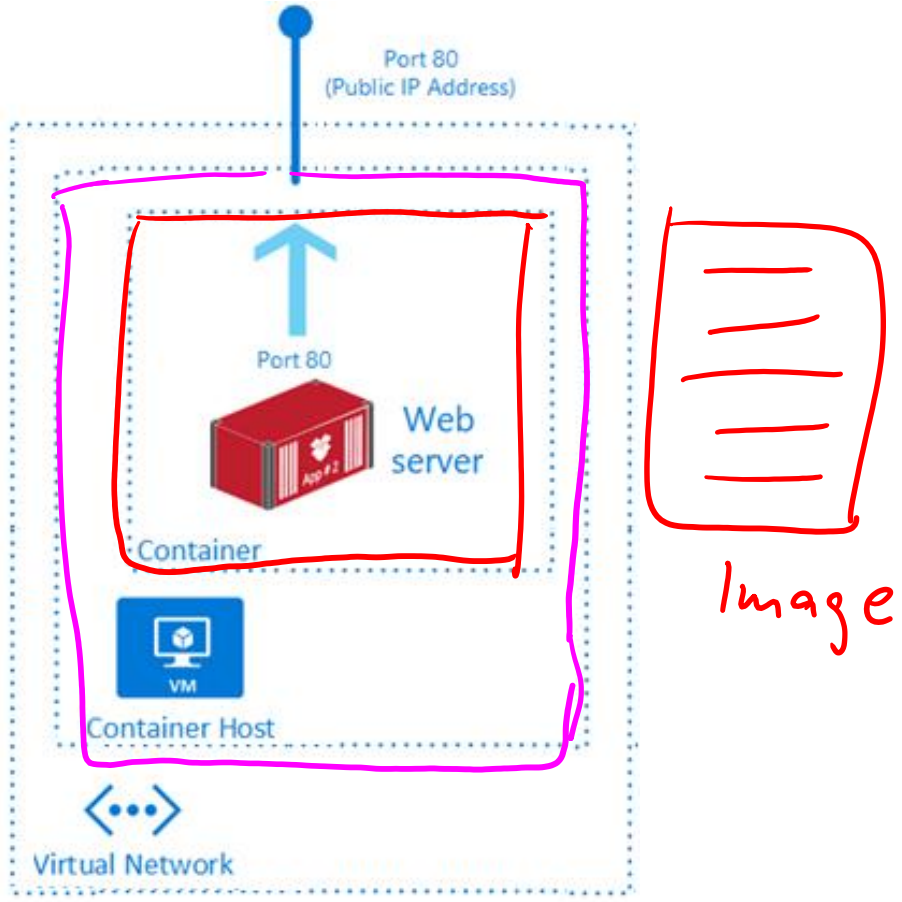


Review Azure Container Instances

ACI

Docker Hub Container Registry ACR

- PaaS Service
- Fast startup times
- Public IP connectivity and DNS name
- Isolation features
- Custom sizes
- Persistent storage
- Linux and Windows Containers
- Co-scheduled Groups
- Virtual network Deployment

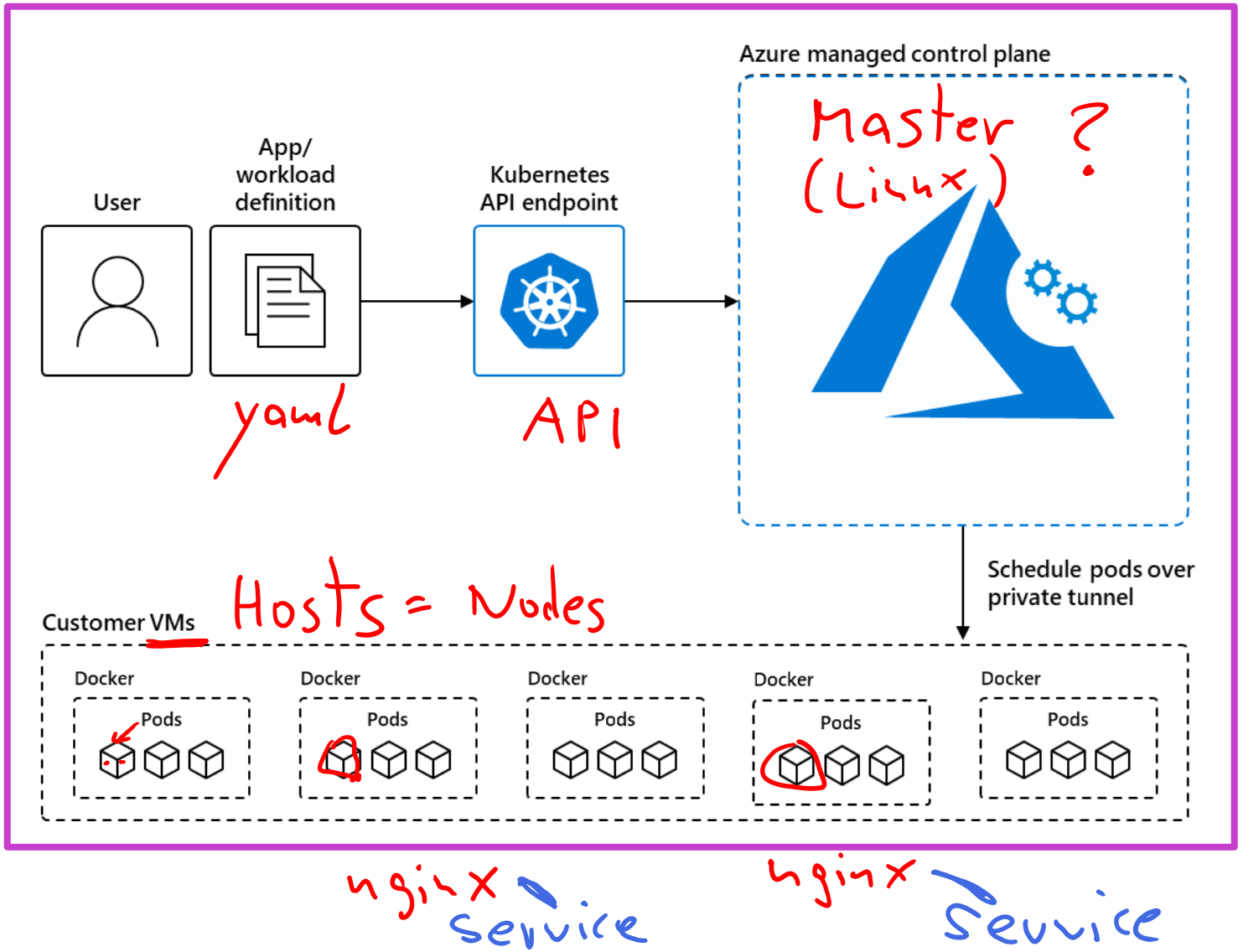


Fastest way to run a container in Azure without provisioning a VM

Azure Kubernetes Service (AKS)

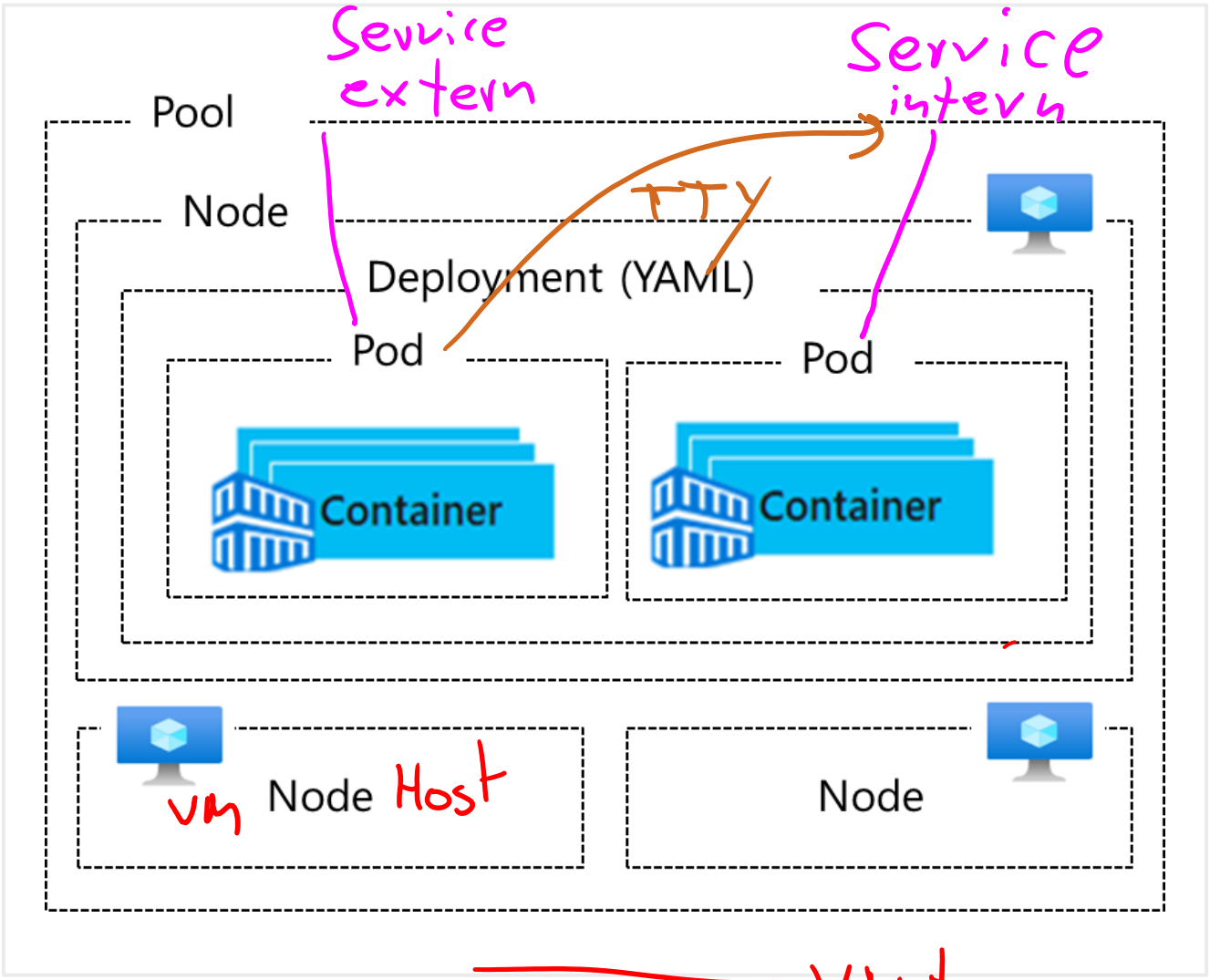
Cluster

- AKS simplifies Kubernetes management, providing high availability, scalability, and integration with DevOps tools.
- Azure manages AKS control plane for free, focusing on health monitoring and maintenance; users pay for nodes.
- AKS use cases include microservices, secure DevOps, data streaming, and running Windows containers.

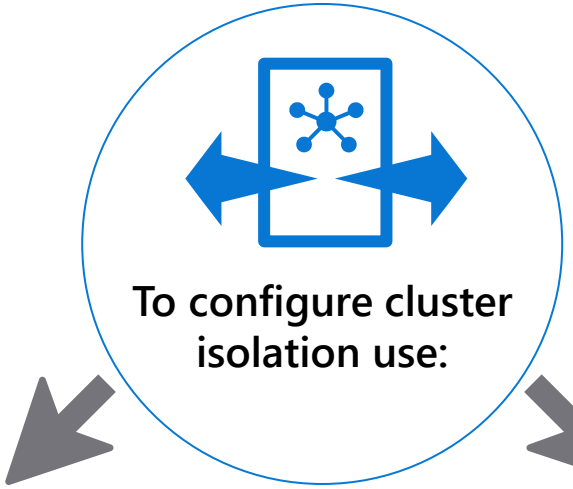


Understand AKS Terminology

Term	Description
Pools	Groups of nodes with identical configurations
Nodes	Individual VMs running containerized applications
Pods	Single instance of an application. A pod can contain multiple containers
Deployment	One or more identical pods managed by Kubernetes
Manifest	YAML file describing a deployment



Configure network isolation for Azure Kubernetes Service



Logical isolation

- Has high pod density
- Additional security features, like Kubernetes RBAC for nodes, efficiently block exploit
- For true security when running hostile multi-tenant workloads, you should only trust a hypervisor.

Physical isolation

- Has low pod density
- it adds management and financial overhead.
- Use only for hostile multi-tenant workloads
- For other scenarios, it is recommended to use Logical Isolation.

Secure and monitor AKS

Use Microsoft Defender for Containers to protect AKS by:

Environment hardening: Defender for Containers continuously assesses clusters to provide visibility into misconfigurations and guidelines to help mitigate identified threats.

Vulnerability assessment: Vulnerability assessment and management tools for images are stored in ACR registries and runs in Azure Kubernetes Service.

Run-time threat protection for nodes and clusters: Threat protection for clusters and Linux nodes generates security alerts for suspicious activities.

Configure authentication for AKS

To configure authentication for AKS:

- Configure Microsoft Entra ID authentication for AKS clusters with OpenID Connect.
- Enable AKS-managed Microsoft Entra ID Integration on your existing Kubernetes RBAC-enabled cluster.
- Upgrade to AKS-managed Microsoft Entra ID Integration if you have legacy Azure AD Integration.
- Use kubelogin to access the cluster with non-interactive service principal sign-in.
- Use Conditional Access to control access while integrating Microsoft Entra ID with your AKS cluster.
- Use Privileged Identity Management (PIM) for just-in-time requests for cluster access control.

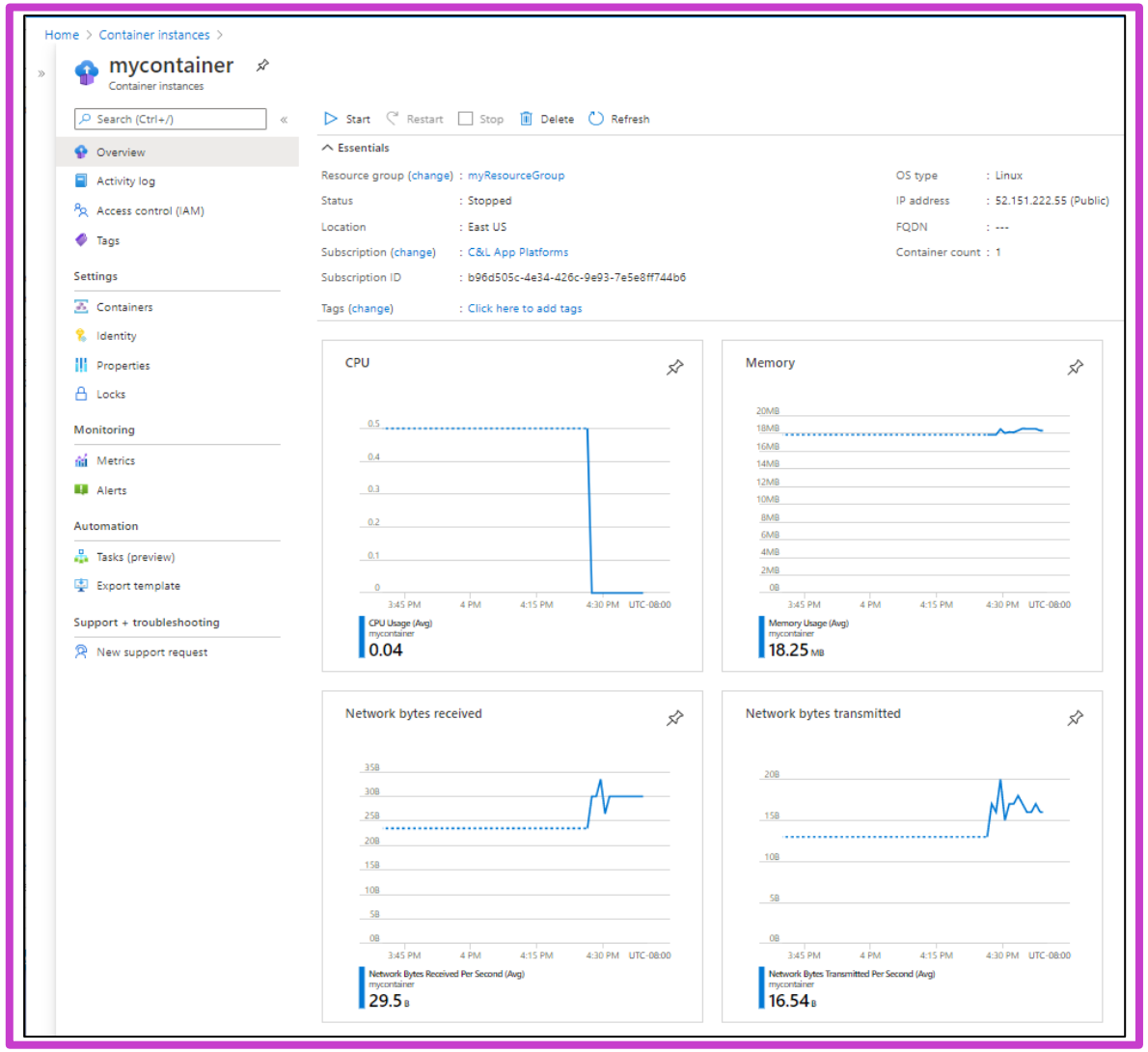
Remember these limitations:

- You can't disable AKS-managed Microsoft Entra ID integration.
- You can't change an AKS-managed Microsoft Entra ID integrated cluster to legacy AAD.
- AKS-managed Microsoft Entra ID integration doesn't support clusters that are not Kubernetes RBAC-enabled.

Configure security monitoring for Azure Container Instances

Get metrics - Azure portal

- When a container group is made, Azure Monitor data is in the Azure portal.
- Metrics are on the **Overview** page, with pre-created charts for each metric.

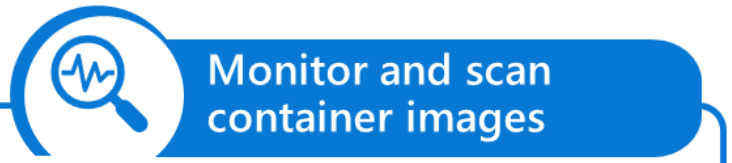


Get metrics - Azure CLI

- Metrics for container instances can also be gathered using the Azure CLI.
- First, get the ID of the container group using the following command:

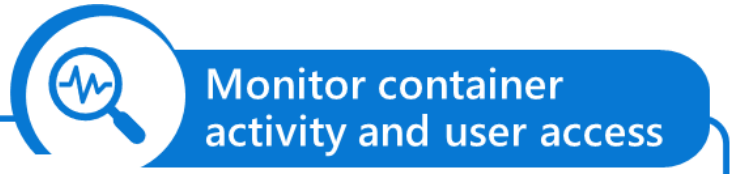
```
CONTAINER_GROUP=$(az container show --
resource-group <resource-group> --name
<container-group> --query id --output tsv)
```

Configure security monitoring for Azure Container Apps



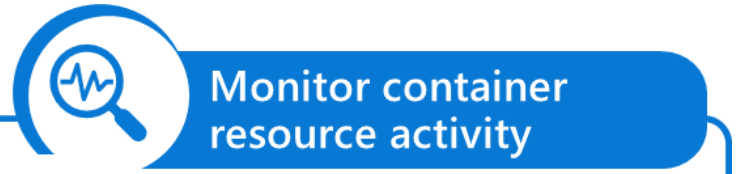
Monitor and scan container images

- Use solutions to scan container images in a private registry and identify potential vulnerabilities.
- Solutions include Microsoft Defender for Cloud's integrated Qualys scanner, Twistlock, and Aqua Security.



Monitor container activity and user access

- Monitor activity and user access to your container ecosystem consistently to identify suspicious or malicious activities.
- Use container monitoring solutions provided by Azure, such as Container Insights.

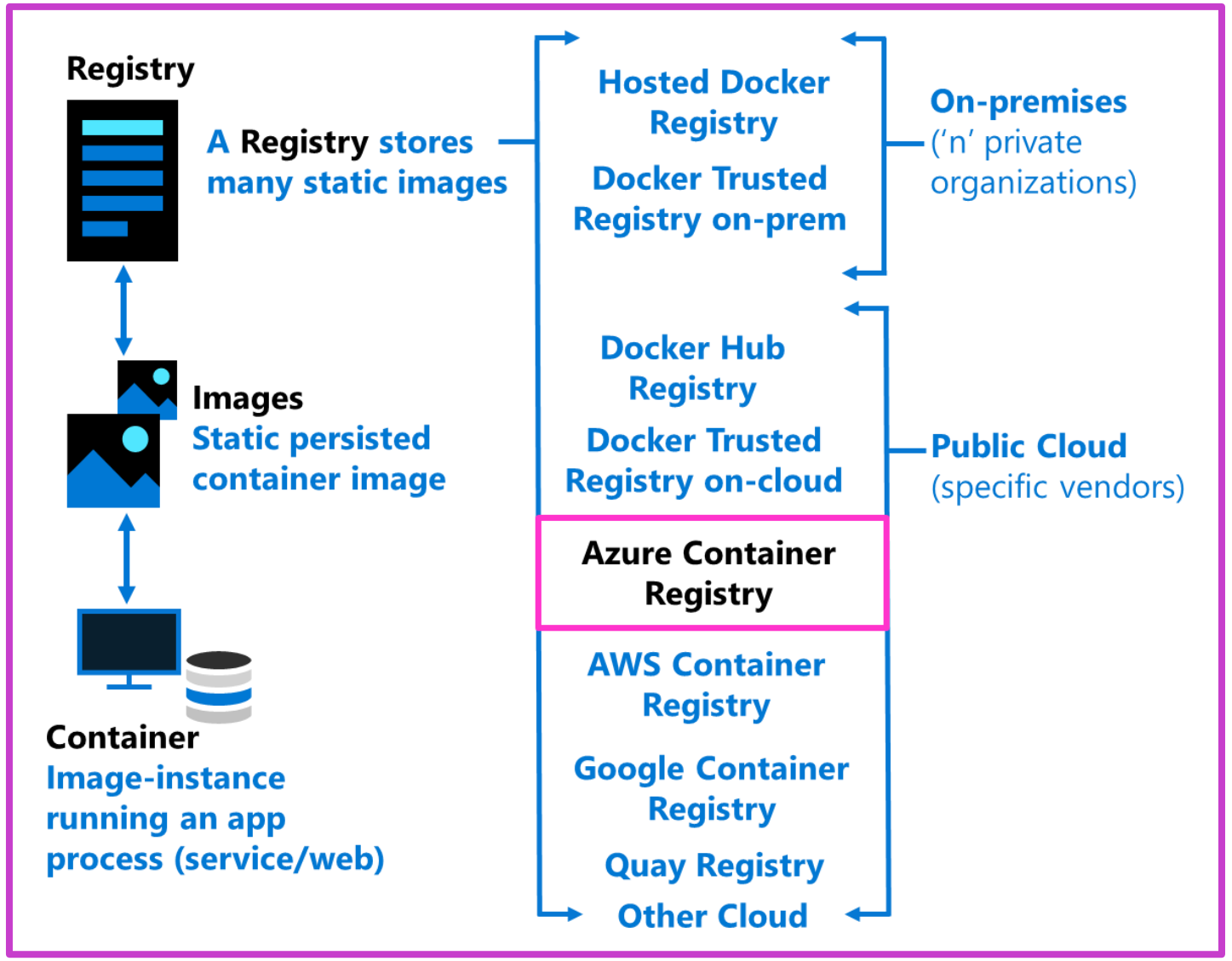


Monitor container resource activity

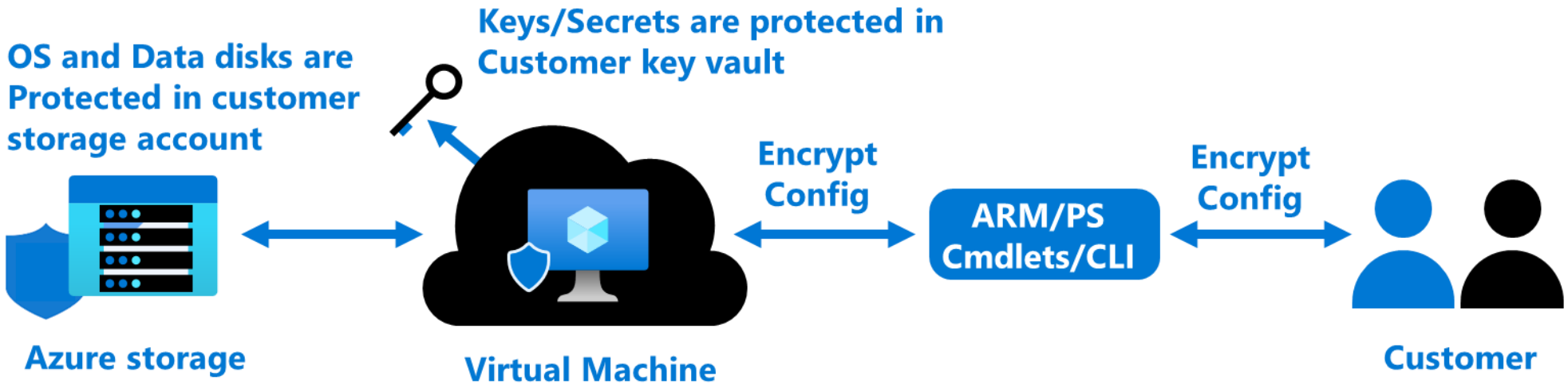
- Monitor your resource activity such as files and other resources that your containers access.
- Use Azure Monitor for the collection of metrics, activity logs, and diagnostic logs.
- Review metrics for performance statistics for different resources and the operating system inside a VM.

Manage access to an Azure Container Registry

- Docker registry service
- Private and hosted in Azure
- Build, store, and manage images
- Push and pull with the Docker CLI or the Azure CLI
- Access with Microsoft Entra ID
- RBAC to assign permissions
- Automate using DevOps



Configure disk encryption, including Azure Disk Encryption (ADE), encryption at host, and confidential disk encryption



Provides encryption of the operating system and data disks

Windows VMs (BitLocker) and Linux VMs (DM-Crypt)

Stores encryption keys in a customer-managed key vault

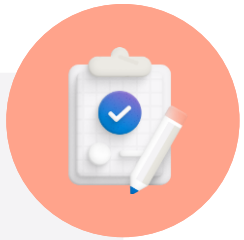
Security configurations for Azure API Management

Use the following Azure security baseline for API Management:

Azure security baseline for API Management

- Azure security baseline for API Management incorporates Microsoft cloud security benchmark v1.0 guidance.
- The benchmark offers Azure security recommendations, categorized by controls, tailored for API Management.
- Microsoft Defender for Cloud enables monitoring, listing Azure Policy definitions for compliance, with some recommendations dependent on paid Defender plans for specific security scenarios.

Additional Study – Planning and Implementing Advanced Security for Compute



Microsoft Learn Modules (docs.microsoft.com/Learn)

Module Review Questions

- Secure Remote Access to Virtual Machines: Implement Azure Bastion and Just-in-Time (JIT) access to protect VMs from unauthorized access.
- Secure Azure Kubernetes Service (AKS):Configure network isolation, authentication, and continuous security monitoring for AKS clusters.
- Monitor Azure Containers: Enable security monitoring for Azure Container Instances (ACIs) and Azure Container Apps (ACAs).
- Manage Azure Container Registry (ACR) Access: Securely manage access to ACR using role-based access control (RBAC) and network rules.
- Implement Disk and API Security:
  Configure Azure Disk Encryption (ADE), host encryption, and secure Azure API Management.

In Rest Encr.

BYOK

Storage account

Container – Blob

Shares (SMB)

Queues

Tables

In Transit Enc

HTTPS

SMB HTTPS

Storage API (REST)

Plan and implement security for storage

→ Data Lake v2

Log Analytics Workspace

LA

KQL

Table | Perf | Event | ...

Configure access control for storage accounts

Every storage request must be authorized. There are various authorization methods, including anonymous.

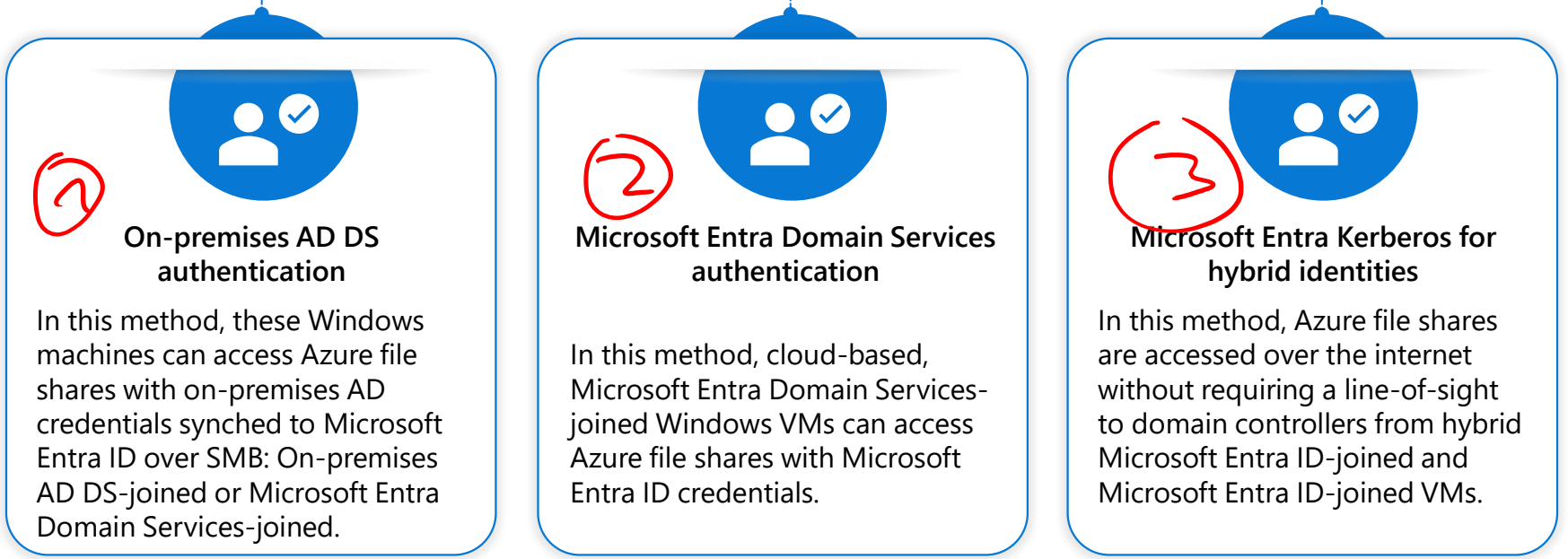
Storage	Storage Account Shared Key	Shared access signature	Microsoft Entra ID	Active Directory Domain Services (on-prem ADDS)	Anonymous public read access
Azure Blobs	Supported	Supported	Supported	Not supported	Supported
Azure Files (SMB)	Supported	Not supported	Supported, only with Microsoft Entra Domain Services	Supported, credentials must be synced to Microsoft Entra ID	Not supported
Azure Files (REST)	Supported	Supported	Supported	Not supported	Not supported
Azure Queues	Supported	Supported	Supported	Not supported	Not supported
Azure Tables	Supported	Supported	Supported	Not supported	Not supported

Manage storage account access keys

- Key Management: Use Azure Key Vault to securely manage, rotate, and protect storage access keys.
- Enhanced Authorization: Leverage Microsoft Entra ID and managed identities for superior security over shared keys.
- Key Rotation and Monitoring: Regularly rotate keys, set expiration policies, and monitor compliance using Azure Policy.

Select and configure an appropriate method for access to Azure Files

Azure Files supports identity-based authentication for Windows file shares over Server Message Block (SMB) through the following methods. You can only use one method per storage account.



On-premises AD DS authentication

In this method, these Windows machines can access Azure file shares with on-premises AD credentials synched to Microsoft Entra ID over SMB: On-premises AD DS-joined or Microsoft Entra Domain Services-joined.

Microsoft Entra Domain Services authentication

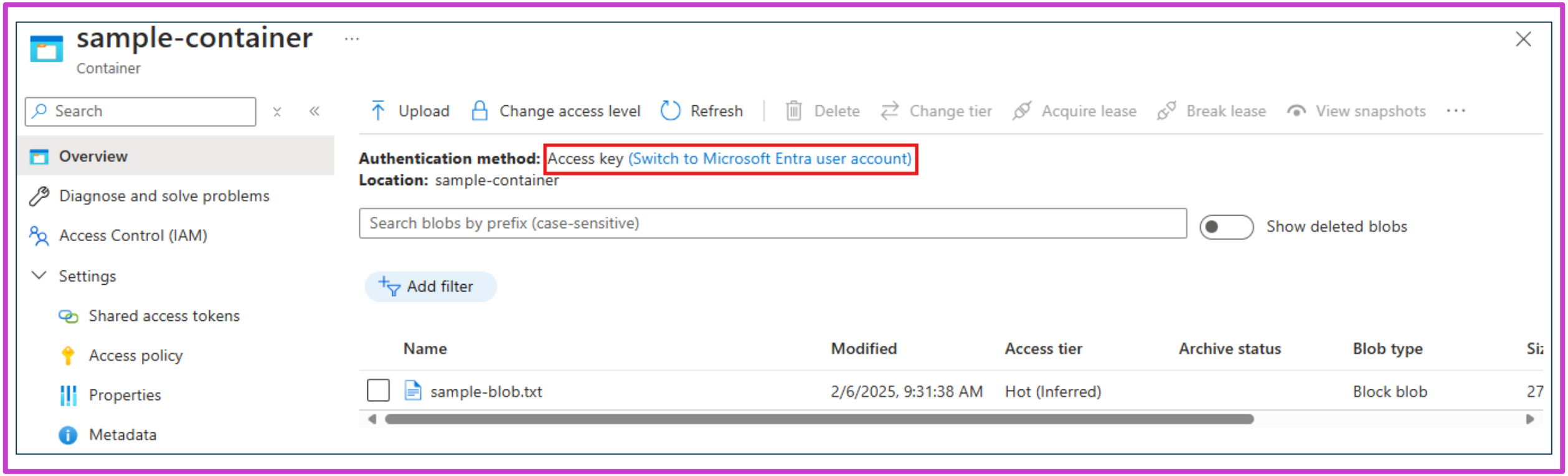
In this method, cloud-based, Microsoft Entra Domain Services-joined Windows VMs can access Azure file shares with Microsoft Entra ID credentials.

Microsoft Entra Kerberos for hybrid identities

In this method, Azure file shares are accessed over the internet without requiring a line-of-sight to domain controllers from hybrid Microsoft Entra ID-joined and Microsoft Entra ID-joined VMs.

Note: Azure Files supports the NFS protocol supporting Linux.

Select and configure an appropriate methods for access to Azure Blob Storage



- Access blob data via Azure portal using Microsoft Entra account or storage account key.
- Permissions managed through Azure RBAC roles for accessing blob data.
- Switch authentication methods or specify authorization for blob uploads.

Select and configure an appropriate method for access to Azure Tables

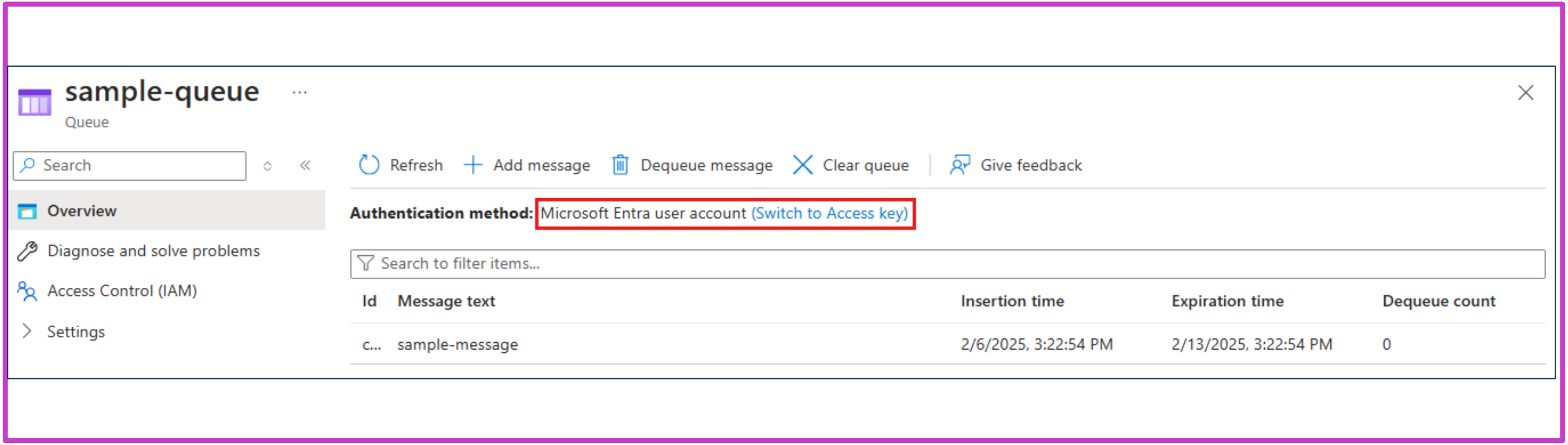
Remember the following considerations while configuring access to Azure Tables:

Accessing a table resource involves a two-step process in Microsoft Entra ID:
1. Authentication of the security principal's identity to get an OAuth 2.0 token
2. Using the token for authorizing access through the Table service

For authentication, applications running within Azure entities (e.g., Azure VM, Azure Functions) can utilize a managed identity to request an OAuth 2.0 access token.

The authorization phase necessitates assigning specific Azure roles to the security principal; these roles, provided by Azure Storage, dictate the permissions the principal possesses for table data access.

Authorize access to queue data in the Azure portal



- Queue Data Access: The Azure portal accesses queue data using either a Microsoft Entra account or a storage account access key.
- Permissions & Authentication: Azure RBAC roles define access; users can switch authentication methods based on their permissions.
- Default Authentication: Microsoft Entra authorization can be set as the default but can be overridden if needed.

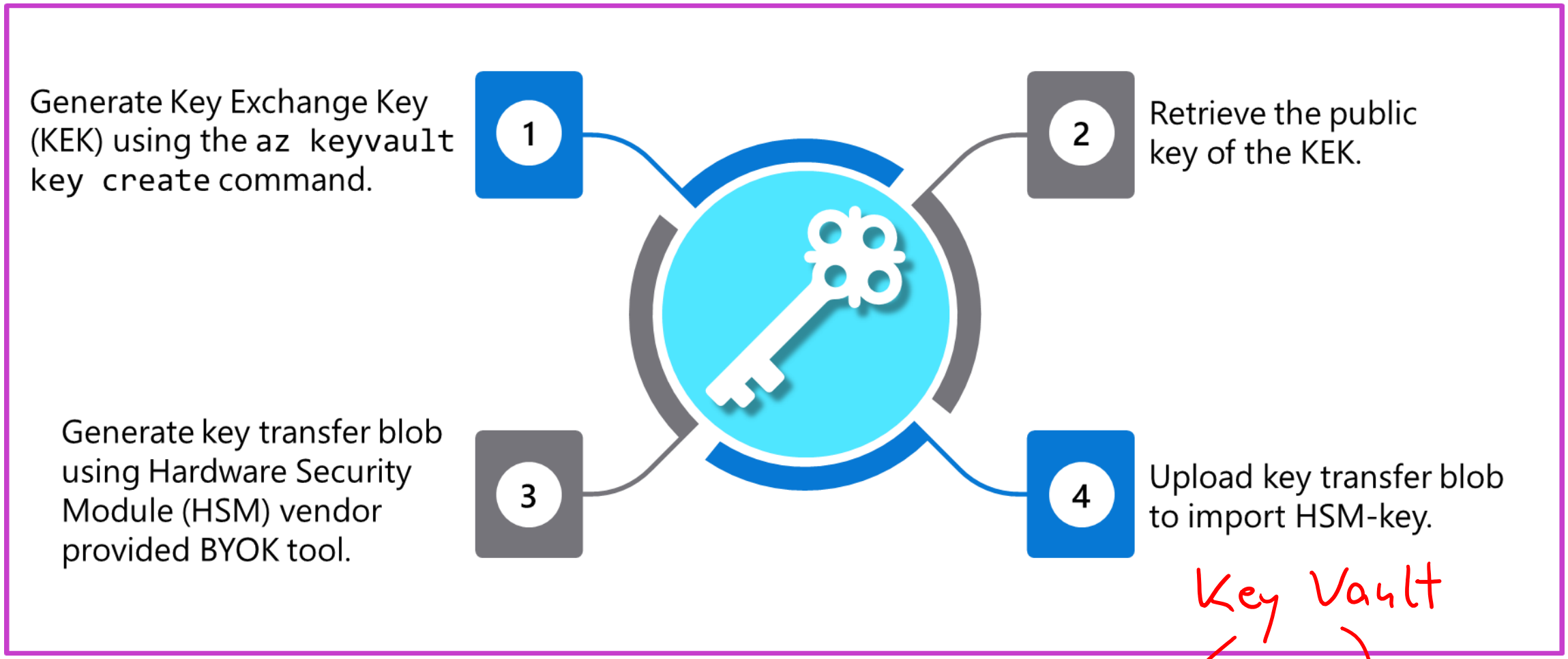
Data protection overview

Recommendations for basic data protection

If you're looking for basic data protection coverage for your storage account and the data that it contains, then Microsoft recommends taking the following steps to begin with:

- Setting up Azure Resource Manager lock to avoid deletions or changes.
- Activating container soft delete for recovery of deleted content.
- Preserving blob state periodically:
    - In Blob Storage: use blob versioning for overwrite events.
    - In Azure Data Lake: utilize manual snapshots for data milestones.

Bring your own key specification (BYOK)

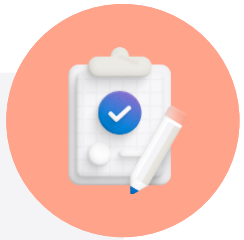


Enable infrastructure encryption for double encryption of data

- Azure Storage Encryption: Uses 256-bit AES and is FIPS 140-2 compliant; optional infrastructure-level double encryption adds an additional security layer.
- Infrastructure Encryption: Encrypts data twice with distinct algorithms and keys, applicable to entire storage accounts or specific scopes.
- Key Management: Service-level supports both Microsoft and customer-managed keys; infrastructure-level strictly uses Microsoft-managed keys.

Important: Infrastructure encryption is advised for compliance-driven double encryption needs. However, for most cases, Azure Storage encryption alone is typically sufficient and beneficial.

Additional Study – Planning and Implementing Security for Azure Storage



Microsoft Learn Modules (docs.microsoft.com/Learn)

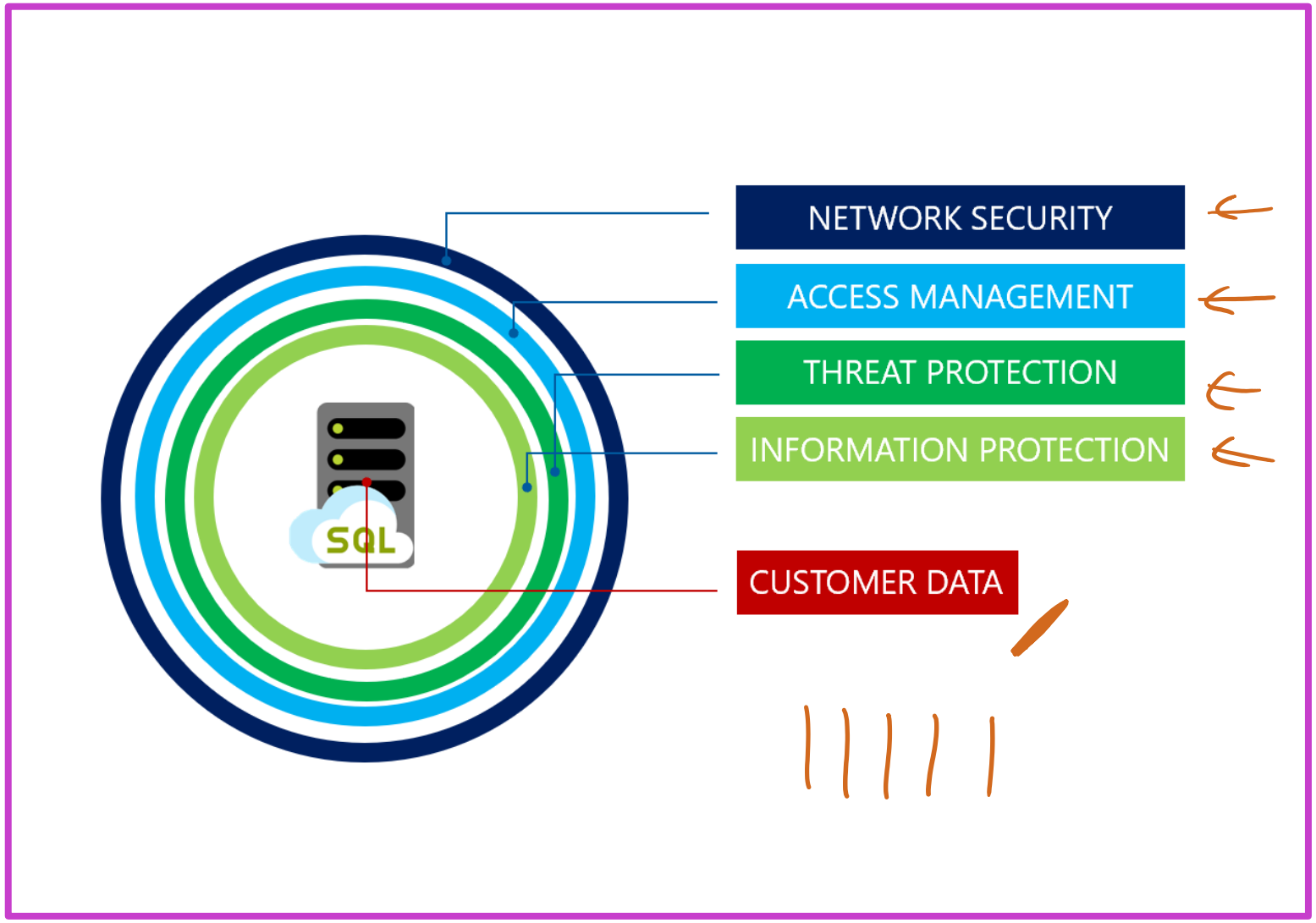
Module Review Questions

- Configure Access Control for Storage Accounts: Implement role-based access control (RBAC) and shared access signatures (SAS) for secure storage access.
- Manage Storage Account Access Keys: Rotate and manage storage account keys to prevent unauthorized access.
- Secure Access to Azure Files and Blob Storage: Configure appropriate access methods for Azure Files and Blob Storage using identity-based authentication.
- Protect Data Against Security Threats: Use soft delete, backups, versioning, and immutable storage for data protection.
- Implement Encryption and Key Management: Configure Bring Your Own Key (BYOK) and enable double encryption for enhanced data security.

Plan and implement security for Azure SQL Database and Azure SQL Managed Instance

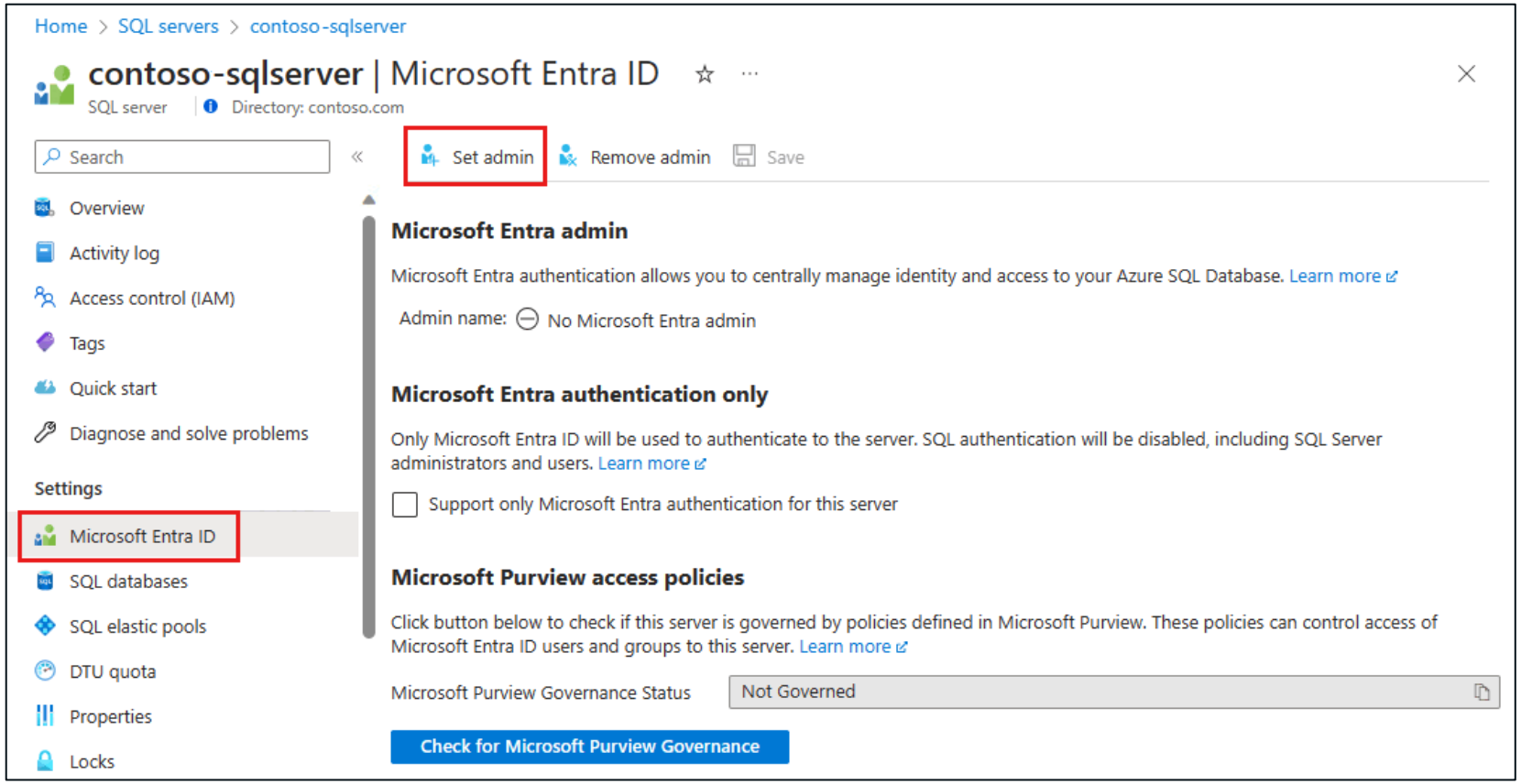
Azure SQL Database and SQL Managed Instance security

- Implements firewalls, IP and virtual network rules for robust network security.
- Supports SQL, Microsoft Entra authentication, and Windows authentication for secure access management.
- Uses encryption for data in transit and at rest and offers advanced threat protection.



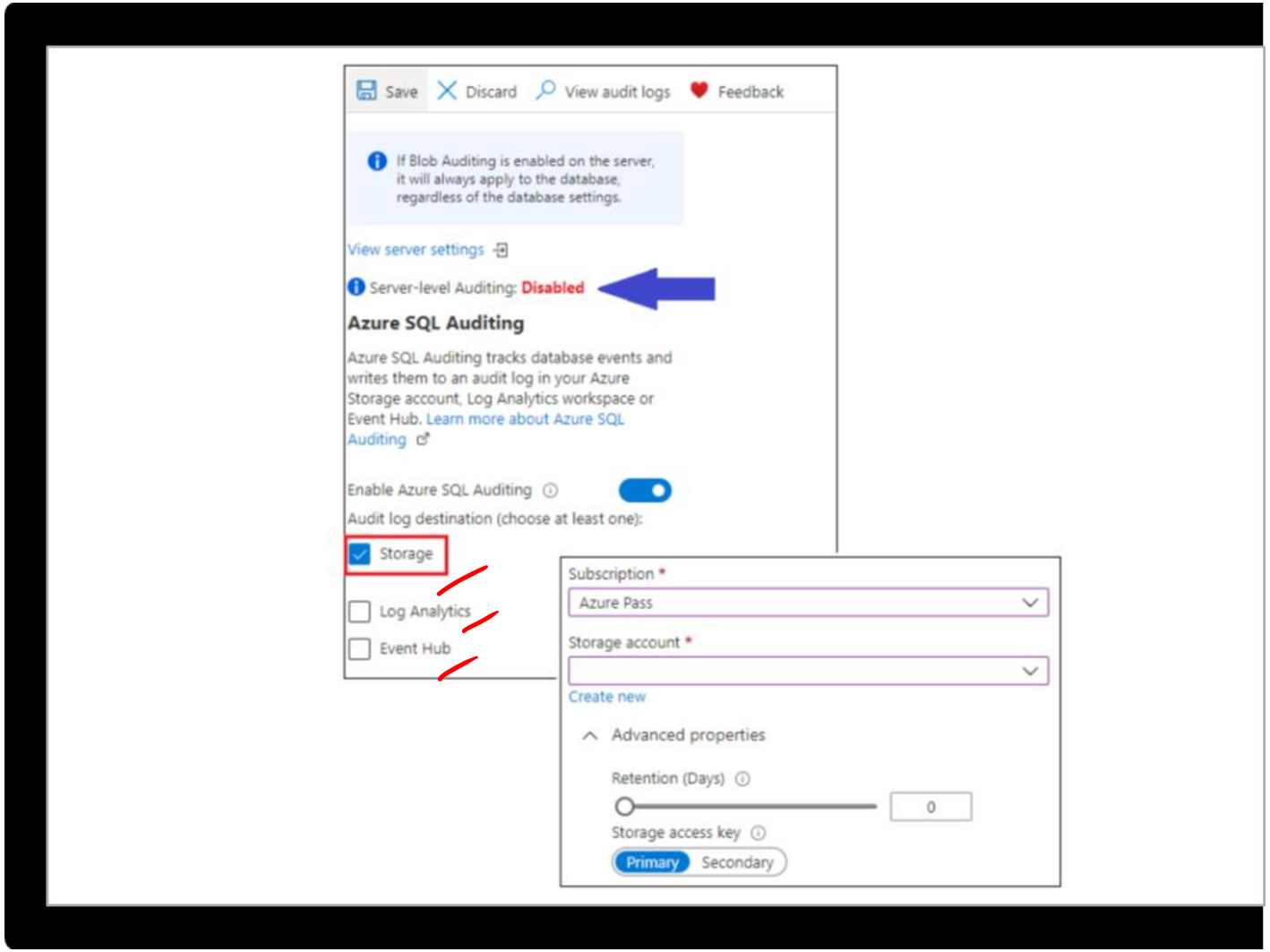
Enable Microsoft Entra database authentication

- Use Microsoft Entra ID for authentication with Azure SQL Database, Managed Instance, and Synapse Analytics.
- Ensure proper setup with a Microsoft Entra tenant, admin configuration, and permissions.
- Enable secure connections using Microsoft Entra authentication, MFA, and client integration.



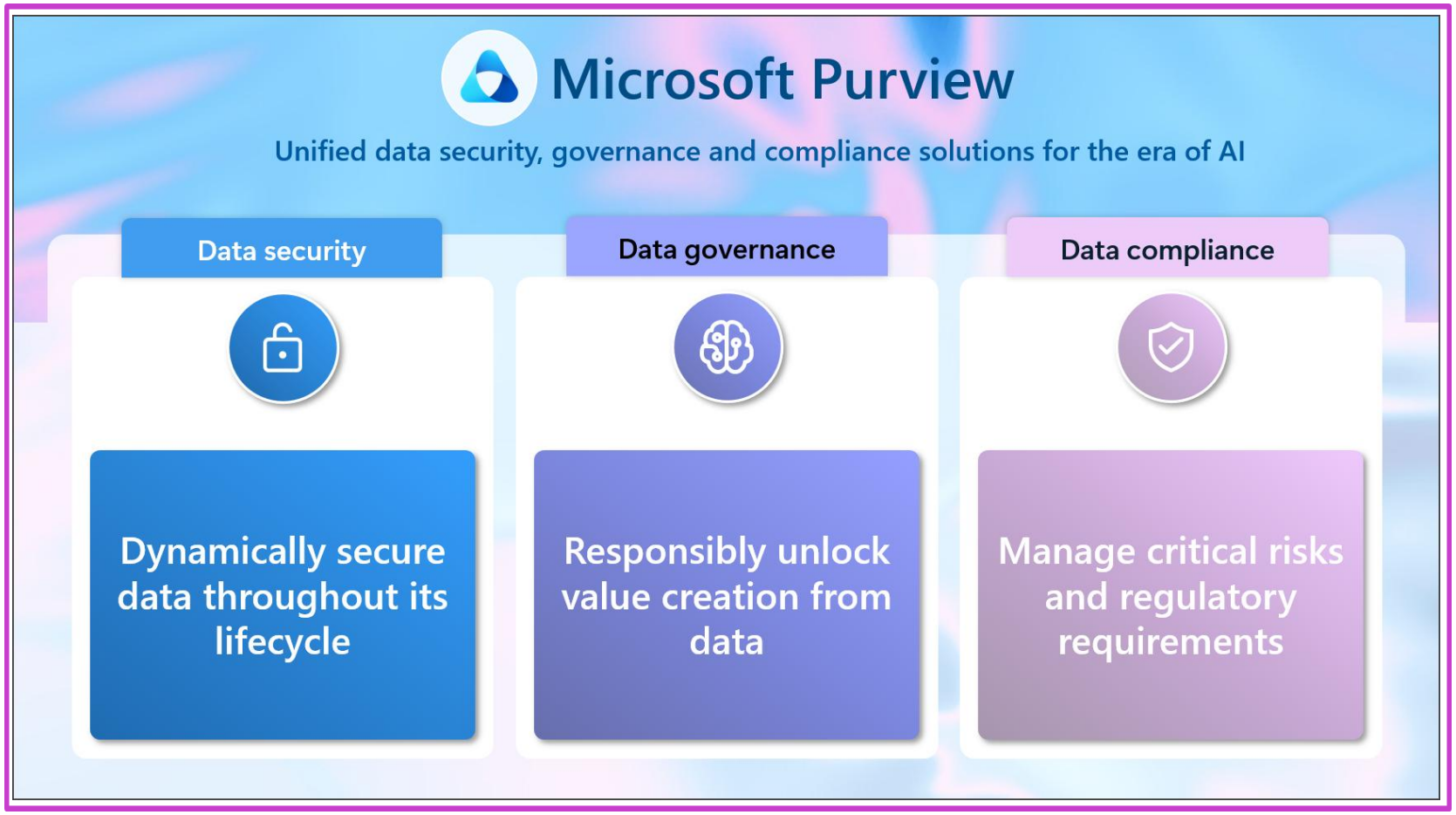
Enable database auditing

- Tracks database events to audit logs in Azure Storage, Log Analytics, or Event Hubs.
- Supports compliance, activity monitoring, and identifying security threats or suspicious database activity.
- Has limitations with Synapse pools, managed identities, and network-restricted storage accounts.



Microsoft Purview

- Gain visibility into data across your organization
- Safeguard and manage sensitive data across its lifecycle
- Govern data seamlessly
- Manage critical data risks and regulatory requirements
- Protect against accidental oversharing and sensitive data leakage



Unified Data Security Platform

Comprehensive Security Integration

Purview integrates DLP, Information Protection, Insider Risk Management, and Data Security Investigations into one platform.

Data Security Posture Management

DSPM continuously monitors and evaluates data security posture across multicloud and hybrid environments.

Zero Trust Enforcement

Purview enables enforcement of Zero Trust principles by dynamically securing sensitive data and managing compliance.

AI-Driven Threat Response

The platform supports rapid response to emerging threats including those from generative AI applications.

Data Map and Unified Catalog for Visibility

Comprehensive Data Discovery

Automatically discover and classify data across on-premises, cloud, and SaaS environments for complete visibility.

Real-Time Data Inventory

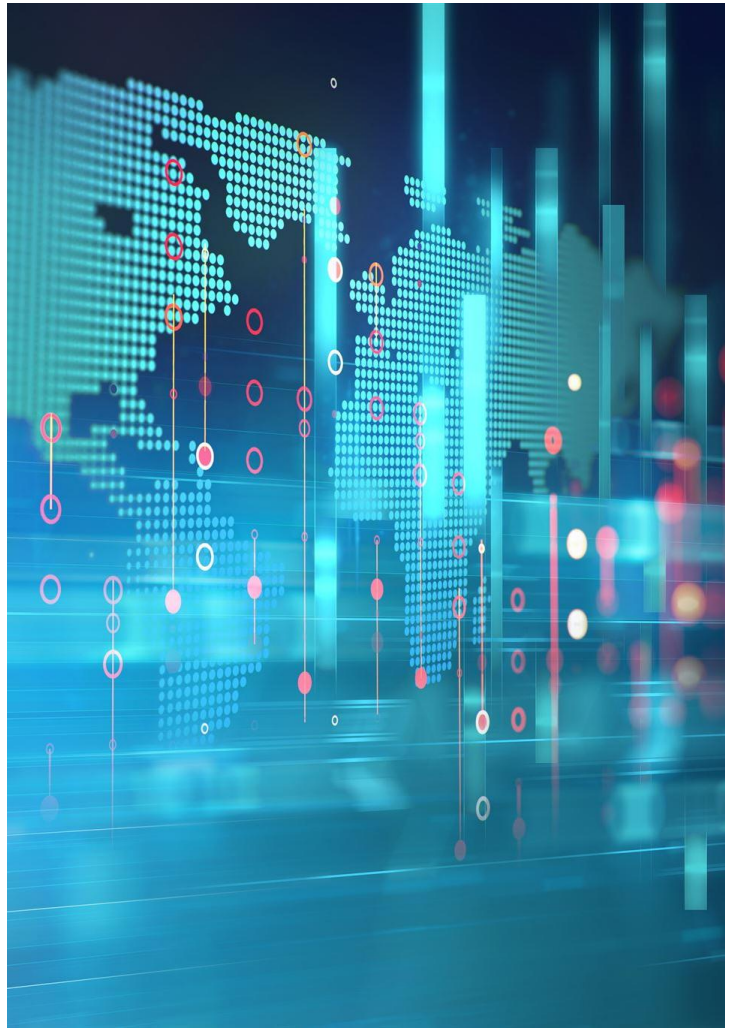
Maintain an up-to-date inventory of sensitive data assets, their classifications, and data lineage for security insights.

Zero Trust Enforcement

Use visibility into data flows and access to enforce precise access controls and monitor potential vulnerabilities.

Self-Service Data Discovery

Enable authorized users to find data easily, reducing shadow IT risks while maintaining governance and compliance.



Data Security Posture Management (DSPM) for AI and Cloud

Continuous Data Risk Assessment

DSPM continuously scans and assesses sensitive data risks across cloud and AI platforms ensuring real-time security posture visibility.

Unified Security Signals

DSPM consolidates signals from DLP, sensitivity labels, insider risks, and data graphs for a comprehensive security overview.

Proactive Risk Mitigation

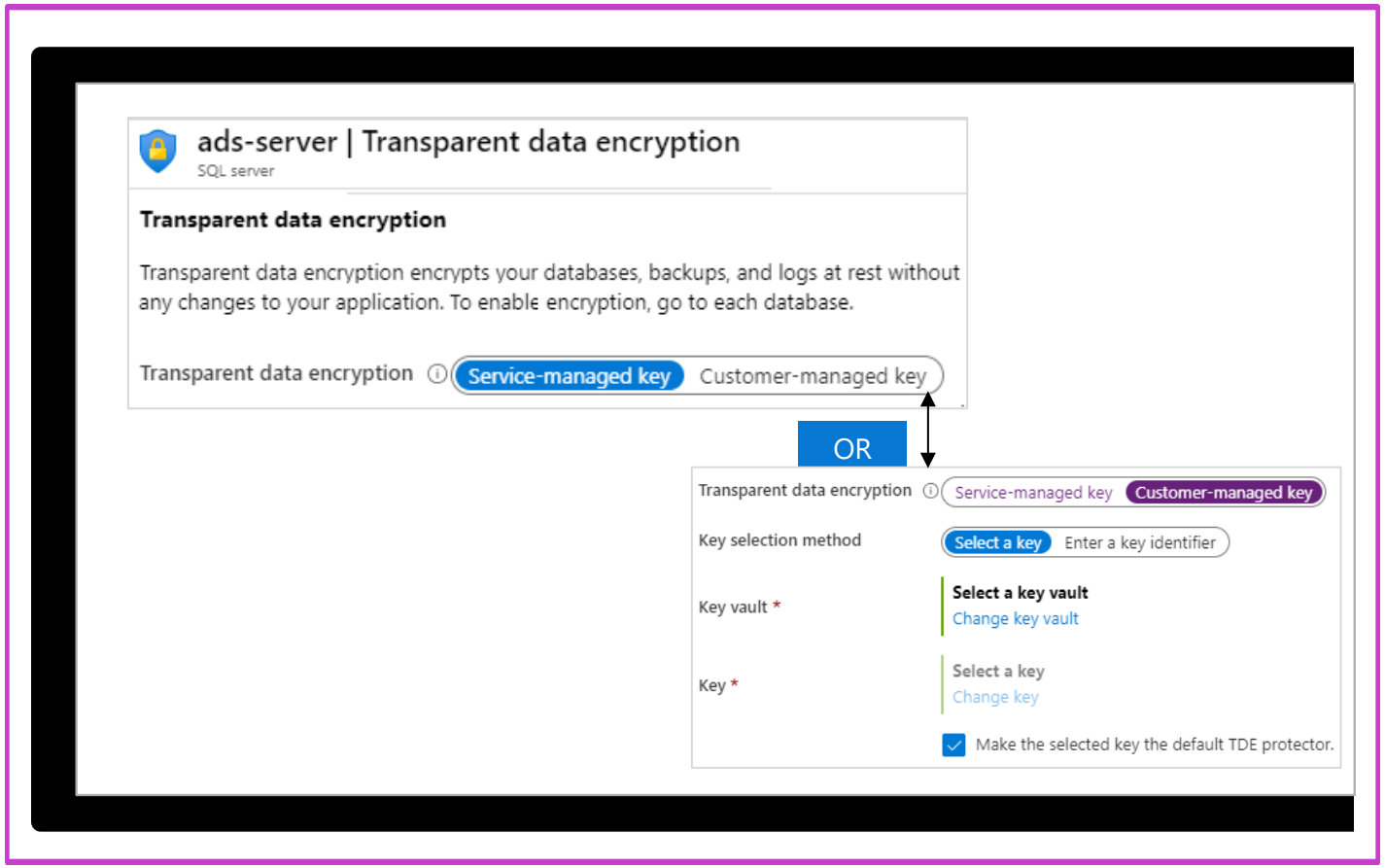
Engineers use DSPM to detect oversharing and misconfigurations, enabling proactive remediation before incidents occur.

AI and Cloud Security Focus

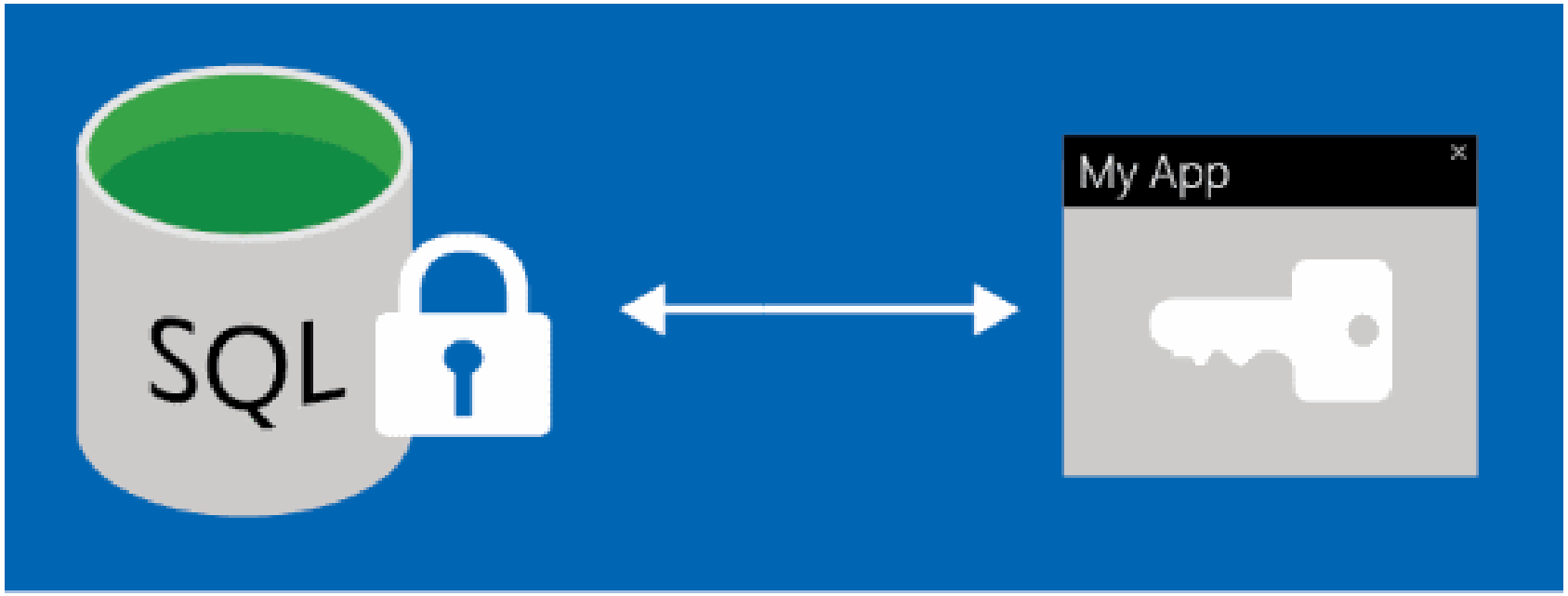
DSPM focuses security posture on data itself, supporting AI agents and cloud environments for improved compliance and risk reduction.

Implement Transparent Data Encryption (TDE)

- Protects databases, backups, and logs at rest – server level
- Real-time page level encryption and decryption – service or customer managed keys
- Supports Azure SQL Database (enabled by default), SQL Managed Instance, SQL Server on VM (IaaS SQL Server), and Azure Synapse Analytics



Recommend when to use Azure SQL Database Always Encrypted

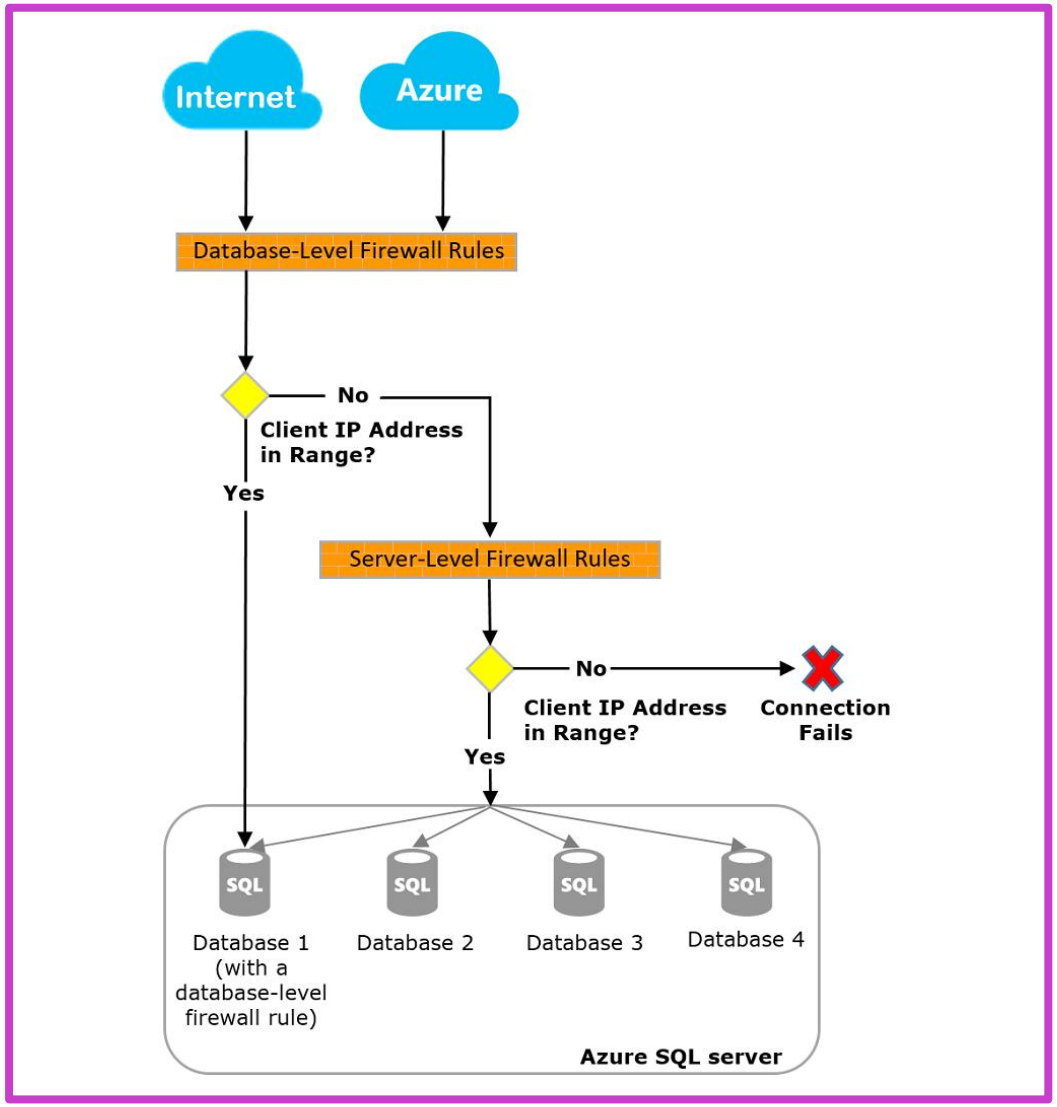


- Always Encrypted protects sensitive data in Azure SQL platforms.
- Clients encrypt data in applications without revealing encryption keys to Database Engine.
- Ensures data owner visibility while preventing unauthorized access, reducing data theft risks.

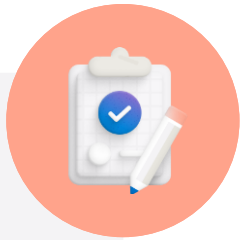
Recommend when to use Always Encrypted

Implement an Azure SQL Database firewall

- Azure SQL Database and Synapse Analytics block public endpoint access by default with firewalls.
- Use server-level or database-level IP firewall rules to manage database access securely.
- Firewall rules can be configured via Azure portal, PowerShell, CLI, or Transact-SQL.



Additional Study – Planning and Implementing Security for Azure SQL Database and Azure SQL Managed Instance



Microsoft Learn Modules (docs.microsoft.com/Learn)

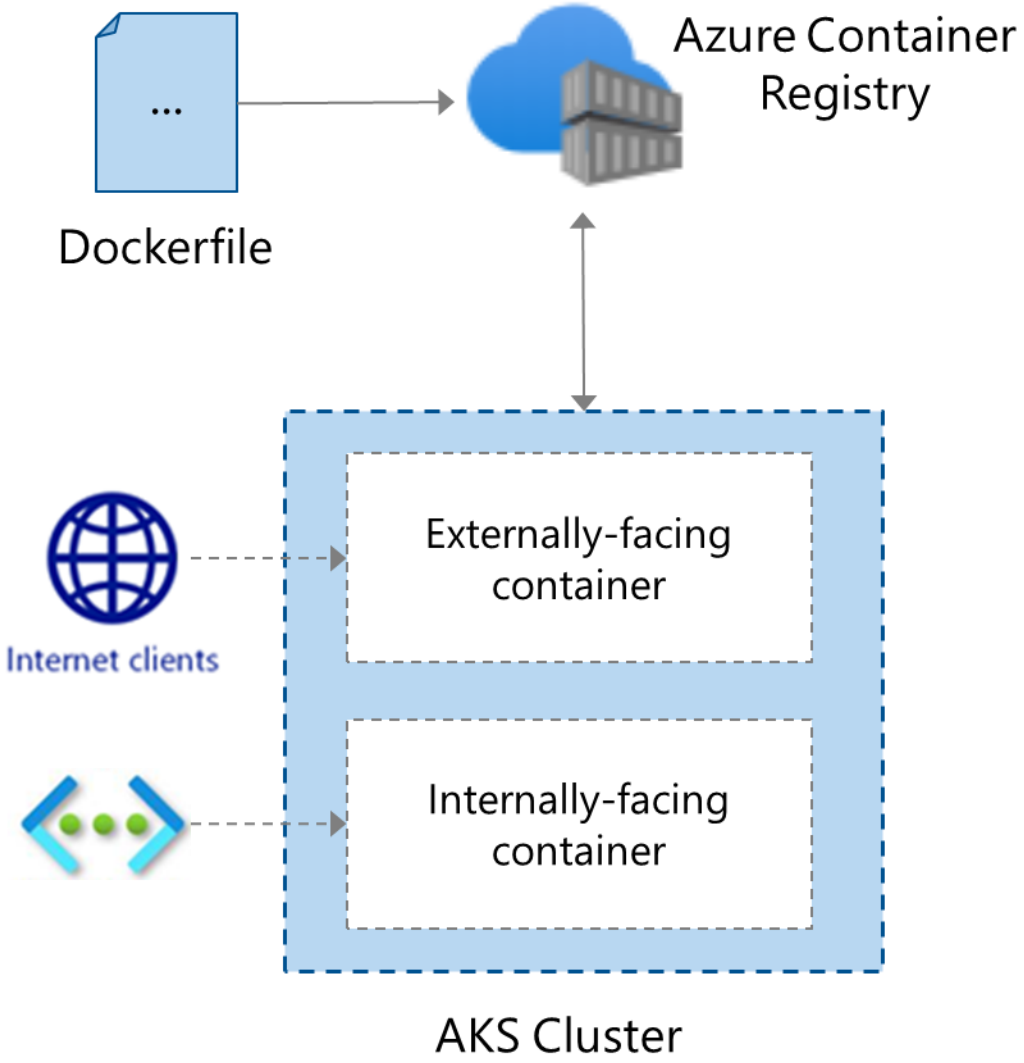
Module Review Questions

- Enable Microsoft Entra Database Authentication: Secure database access using Microsoft Entra ID for identity-based authentication.
- Enable Database Auditing: Track and log database activities to ensure security compliance.
- Implement Dynamic Data Masking: Protect sensitive data by masking it from unauthorized users.
- Apply Transparent Data Encryption (TDE): Encrypt SQL databases at rest to protect data.
- Use Always Encrypted for Sensitive Data: Encrypt sensitive data in use to protect it from unauthorized access.

Module Labs

Lab 04 – Configuring and securing ACR and AKS

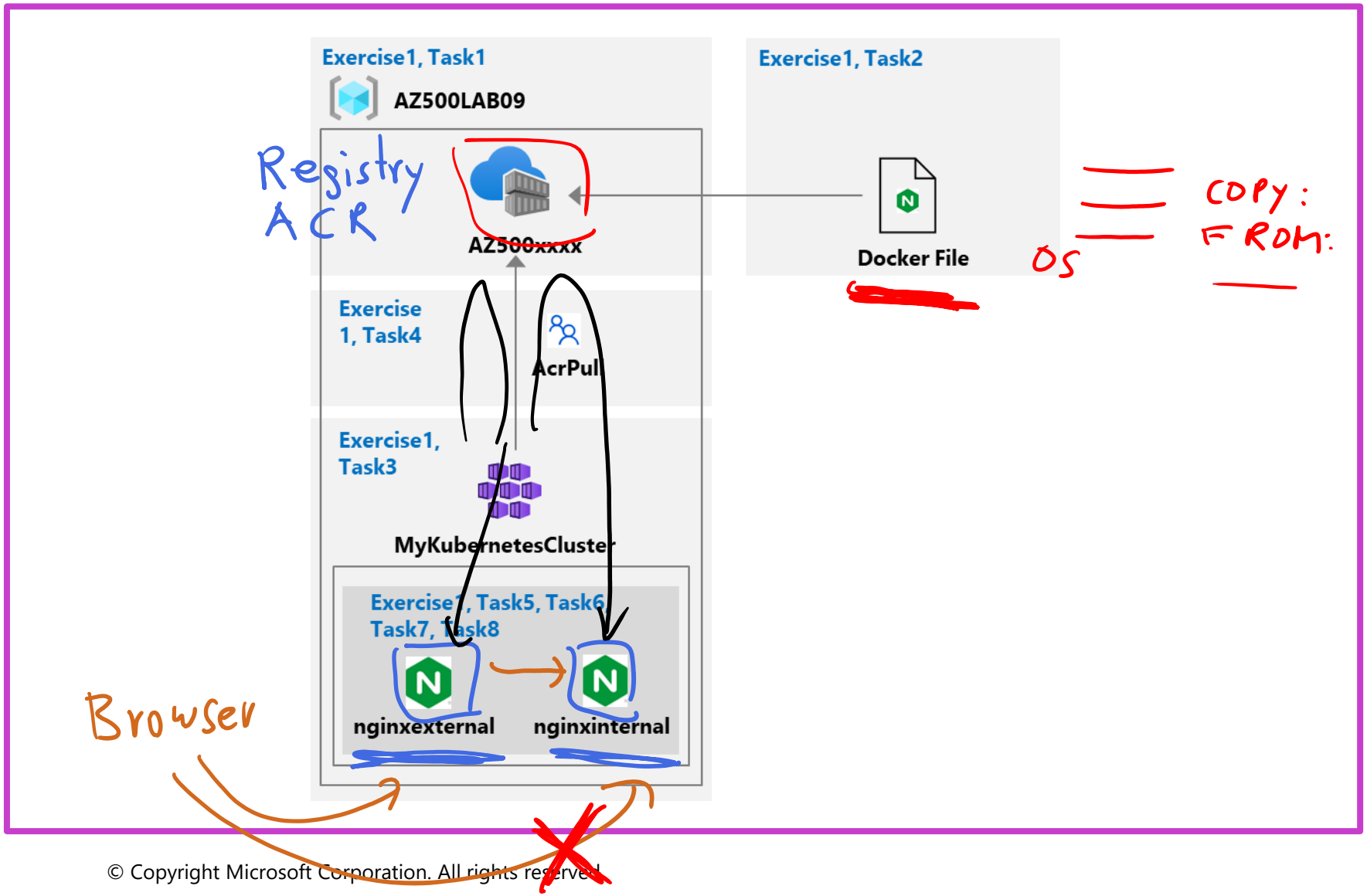
- Create an Azure Container Registry
- Create a Dockerfile, build a container and push it to ACR
- Create an Azure Kubernetes Service
- Give AKS permission to access the ACR
- Deploy an external facing container and test
- Deploy an internal facing container and test



Lab 04 – Configuring and securing ACR and AKS

This exercise teaches students how to deploy a proof of concept with Azure Container Registry and Azure Kubernetes Service by building images with Dockerfile, storing them in ACR, configuring AKS, and securing container app access

[Launch this Exercise in GitHub]



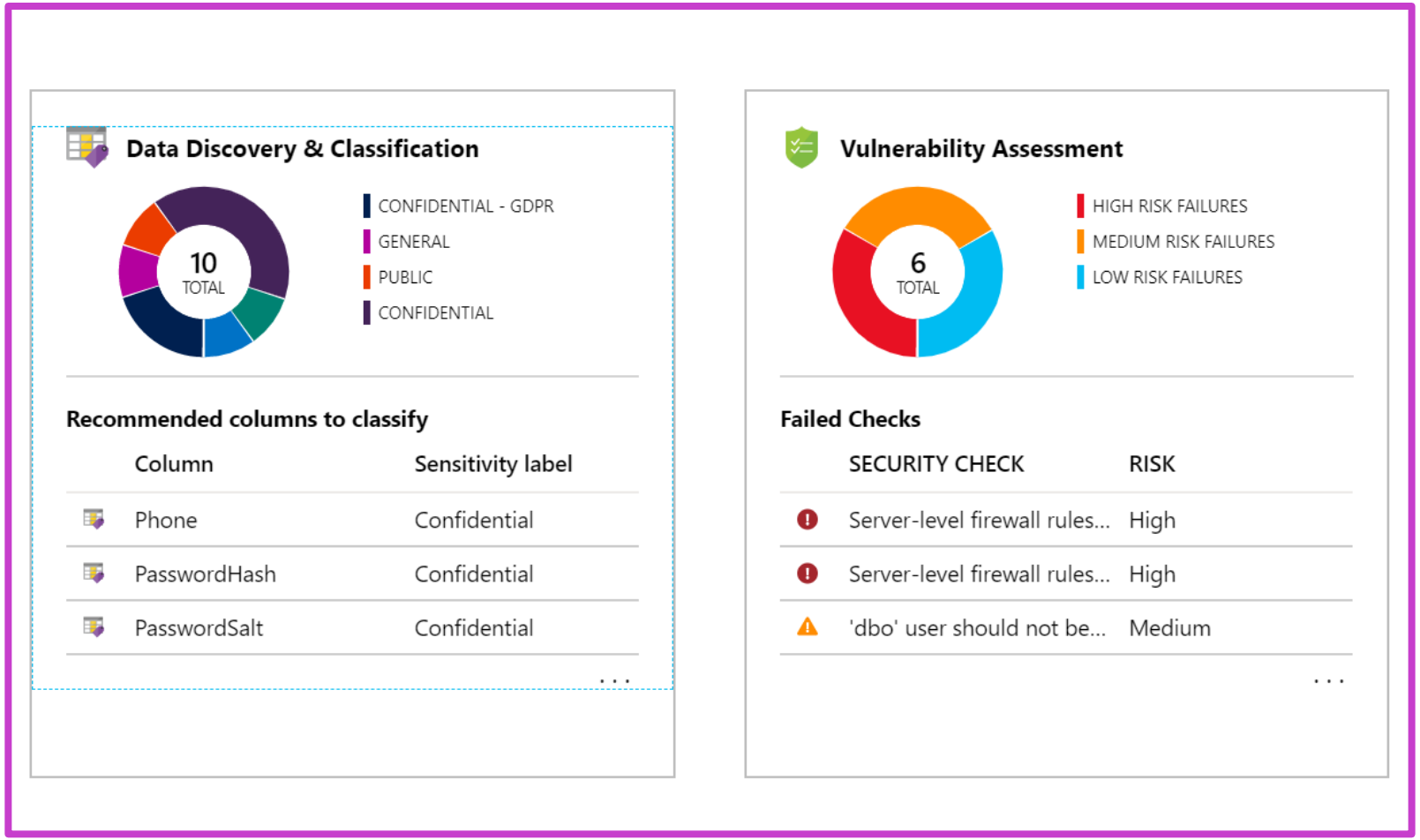
Lab 05 – Securing Azure SQL Database

Deploy an Azure SQL Database

Configure Advanced Data Protection

Configure Data Classification

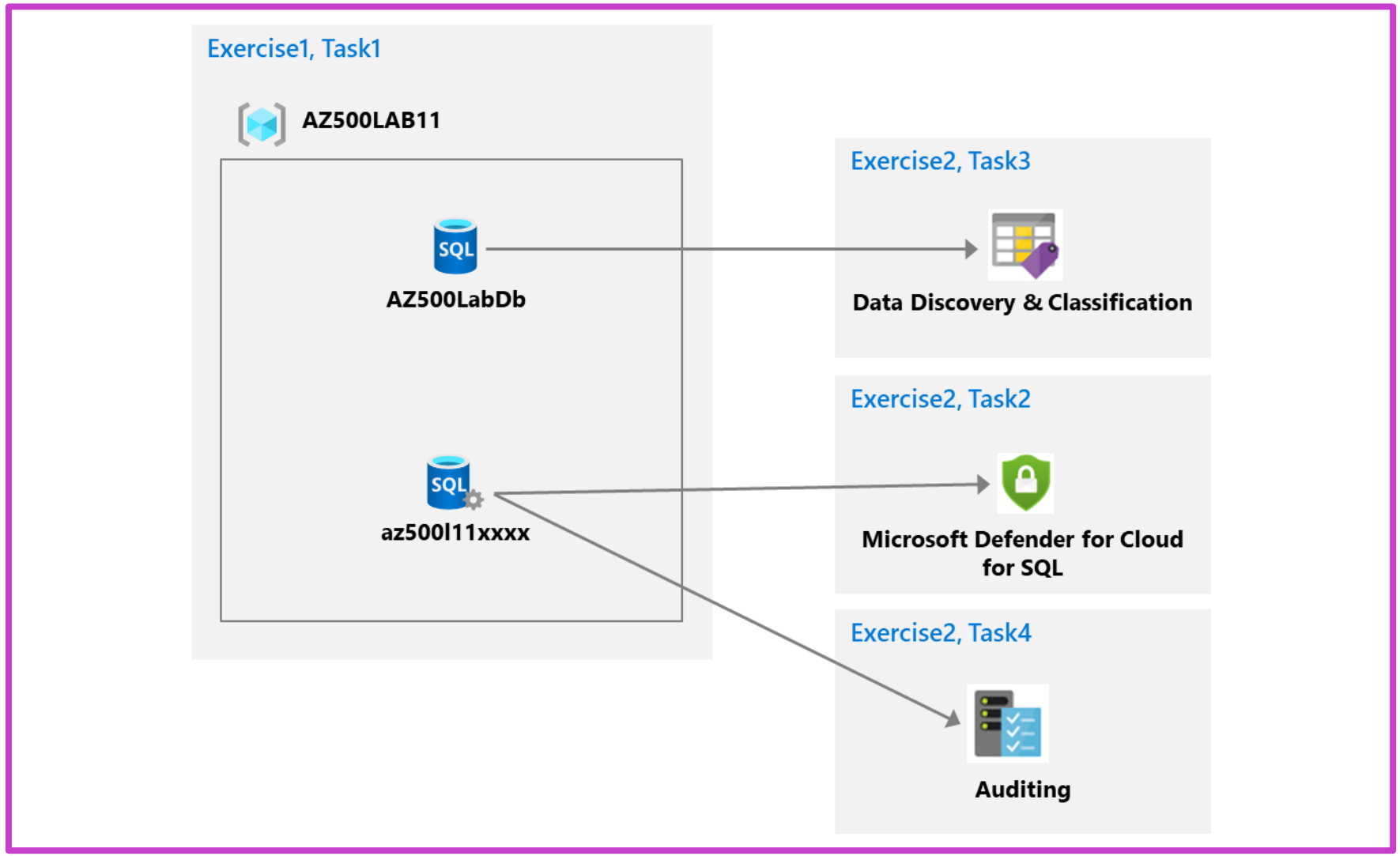
Configure Auditing



Lab 05 – Securing Azure SQL Database

This exercise teaches students to review Azure SQL Database security features, including attack protection, data classification, and auditing of servers, queries, and events.

[Launch this Exercise in GitHub]



Lab 06 – Service Endpoints and Securing Storage

Create a virtual network with a Public and Private subnet

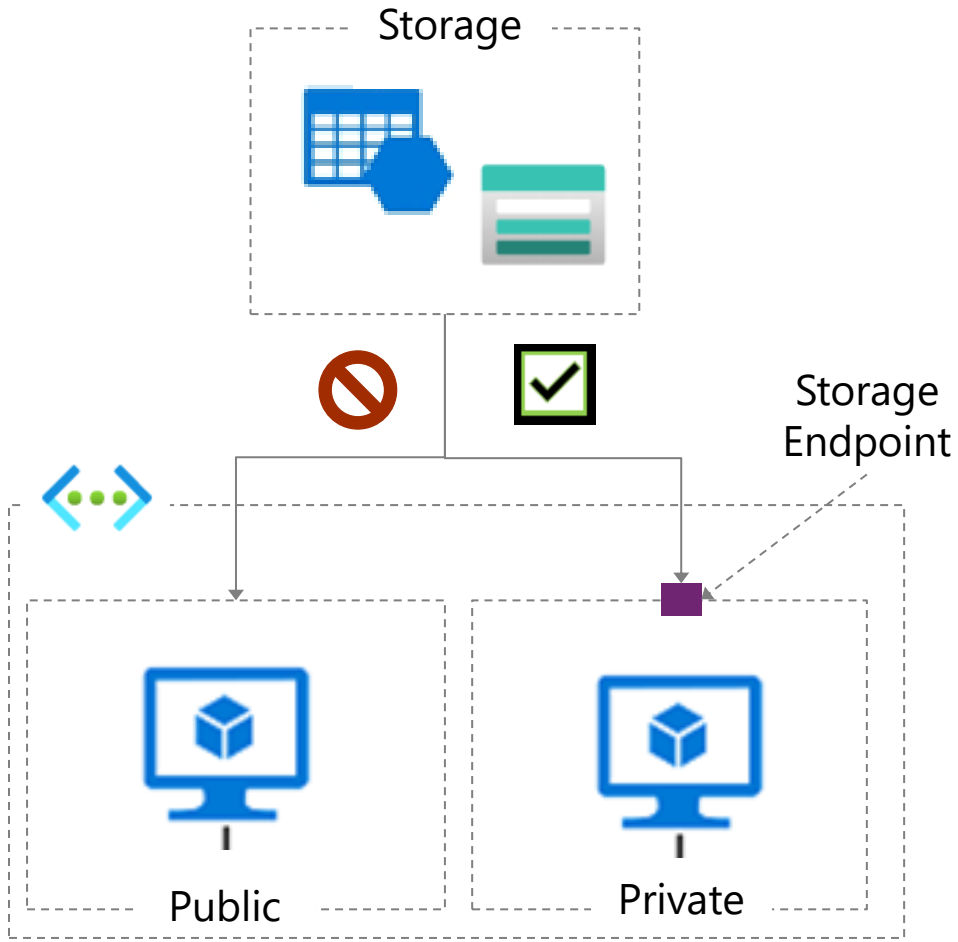
Create a storage endpoint for the Private subnet

Create a storage account with a file share

Configure a NSG with rules to allow access to storage and internet

Confirm storage access from the private subnet

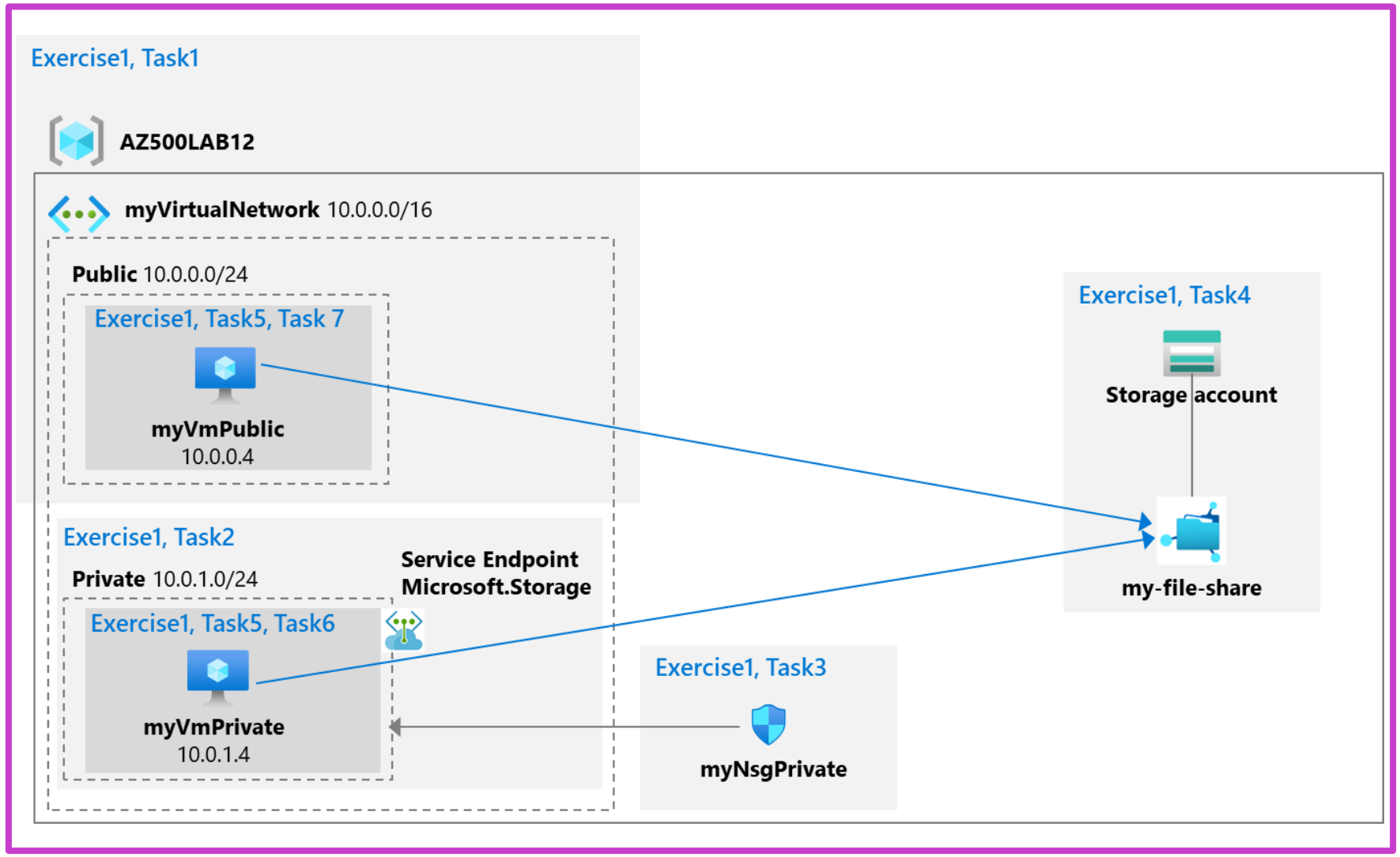
Confirm storage access is denied from the public subnet



Lab 06 – Service Endpoints and Securing Storage

This exercise teaches students how to create a proof of concept to secure Azure file shares by configuring a storage endpoint for Azure backbone traffic, restricting subnet access, and blocking external resources.

[Launch this Exercise in GitHub]



Knowledge check

1 **How does scanning in Microsoft Purview handle classifications and sensitivity labels?**

- ☐ It ignores them during the scanning process
- ☑ It applies them to the gathered technical metadata and schema
- ☐ It deletes them from the system

2 **Which Azure service provides serverless, automatic, and scalable data encryption for data at rest?**

- ☐ Azure Key Vault
- ☑ Azure Storage Service Encryption
- ☐ Azure Sentinel

3 **In Azure SQL Database, what is Transparent Data Encryption (TDE) used for?**

- ☐ Managing access control for Azure SQL Database
- ☑ Encrypting data at rest. Use TLS (transport layer security) for data in motion.
- ☐ Automatically scaling the database resources

Learning Path Recap

In this learning path, we:

Implemented advanced compute security with Azure Bastion, JIT, AKS isolation, authentication, and encryption techniques.

Established robust storage security through access controls, data protection methods, and advanced encryption techniques.

Enhanced Azure SQL Database security via Microsoft Entra ID authentication, auditing, data classification, and encryption methods.

End of presentation