**Problem 1 \*\*: Azure App Service Plan**

You have to provide a website hosting environment that meets the following scalability and security requirements:

* At peak loads, the web application should be able to scale up to 10 host instances.
* The web application storage requirements are minimal and will not exceed 5 GB.
* The web application will perform complex calculations and will require enhanced compute capabilities.
* The virtual machines where the web applications are hosted should be dedicated to your company.

You need to propose an Azure App Service plan to host the application. The solution must ensure minimal costs.

Which Azure App Service plan should you use?

**Answer**

To respond to this question let’s put the company’s requirements together with Azure Service plan features

| Question | Free/Shared | Basic | Standard | Premium V2/V3 | Isolated |
| --- | --- | --- | --- | --- | --- |
| Scale Up to 10 instances | No | No | Yes | Yes | Yes |
| 5 GB Storage | No | No | Yes | Yes | Yes |
| Heavy workloads | No | No | No | Yes | Yes |
| Dedicated and Isolated Environments | No | Yes | Yes | Yes | Yes |

So, the best option which feat with Company’s requirements is the Premium V2/V3 plan. This plan grantee that application sharing the same Azure plan are on dedicated virtual machines and they are not shared with other company’s resources. The premium plan grants a storage up to 250Gb and it is designed a heavy workload due to faster processors, SSD storage, memory-optimized options, and quadruple the memory-to-core ratio.

To summary, here is a recapitulative table for all azure app service plans, that focus on scalability, workload, dedicated resources and estimated cost for each plan:

| Azure App Service Plan | Scalability (max instances) | Compute Capabilities | Dedicated Resources | Approximate Cost | Cores | RAM | Storage |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Basic | 3 | Dedicated, basic capacity | VMs shared among apps in the plan | Low | 1-2 | 1.75-3.5 GiB | 10-50 GiB |
| Standard | 10 | Dedicated, better capacity | VMs shared among apps in the plan | Medium | 1-4 | 1.75-7 GiB | 50-250 GiB |
| Premium | 20 | Dedicated, high performance | VMs shared among apps in the plan | High | 1-4 | 3.5-14 GiB | 250-500 GiB |
| Premium V2 | 30 | Dedicated, very high performance | VMs shared among apps in the plan | Higher | 1-4 | 3.5-14 GiB | 250-500 GiB |
| Premium V3 | 30 | Dedicated, optimized performance for heavy computations | VMs shared among apps in the plan | Higher | 2-8 | 8-32 GiB | 250-1000 GiB |
| Isolated | 100 | Dedicated, maximum isolation and high performance | Completely isolated, non-shared resources | Very High | 2-8 | 8-32 GiB | 250-1000 GiB |

Technical Details:

* Cores and RAM: These specifications depend on the VM sizes used in each plan. Premium plans typically use more powerful VMs with more cores and RAM.
* Storage: The allocated storage also varies by plan, with more expensive plans offering more SSD storage space for improved performance.

**Problem 2 \*: Azure Monitor**

You need to create an alert for a virtual machine (VM) named VM1 that will be fired when the VM's central processing unit (CPU) utilization is greater than 95 percent for at least 10 minutes. You also need to add an action group named AG1 to this alert.  
What should you do? To answer, complete the command by selecting the appropriate parts from the drop-down menus.

**Answer**

az monitor metrics alert create --condition

--name

--resource-group

--scopes

[--action]

[--auto-mitigate {false, true}]

[--description]

[--disabled {false, true}]

[--evaluation-frequency]

[--region]

[--severity]

[--tags]

[--target-resource-type]

[--window-size]

**--evaluation-frequency**

Frequency with which to evaluate the rule in "##h##m##s" format.

default value: 1m

**--window-size**

Time over which to aggregate metrics in "##h##m##s" format.

default value: 5m

**Problem 3 \*: Azure Provider list**

Your organization has an existing Azure subscription that currently hosts VMs and storage.

The Dev team wants to start using Azure Container Apps to allow it to run microservices and containerized applications without deploying serverless VMs.

You decide to set up the Azure Container App environment via Bash. You install the containerapp extension for Command-Line Interface (CLI) and register the Microsoft.App namespace. You try to create the Azure Monitor Log Analytics workspace but you receive an error message that indicates that the command is not recognized.

You need to use the az provider register command to register the Azure Monitor Log Analytics workspace.

Which provider namespace do you need to register?

**Answer:**

In order to view Azure provider list please run this command

az provider list --query "[].namespace" --output tsv

Based on Microsoft Documentation, the correct response is Microsoft.OperationInsights

* We should register the Microsoft.OperationalInsights provider namespace for the Azure Log Analytics workspace. This is the first time that the Log Analytics workspace is being used in the tenant given that the company currently only uses the Azure subscription for virtual machines (VMs) and storage; therefore, we need to register this provider namespace. Once we have registered the Microsoft.OperationalInsights provider namespace, we can create the App container environment, which includes the Log Analytics workspace that is required as part of the environment.
* We would register the Microsoft.PolicyInsights provider namespace if we were using Azure Policy and we wanted to configure this via Command-Line Interface (CLI). Azure Policy is used for governance and guard rails within the Azure landing zone.
* We would register the Microsoft.Automation provider namespace if we were using the Azure Automation namespace. Azure Automation is an Azure native feature that is used to automate different tasks within wer environment. For example, patching servers.
* We would register the Microsoft.NotificationHubs provider namespace if we were configuring Azure Event Hubs. Azure Event Hubs is a big data streaming platform and event ingestion service that can be integrated with Azure native services.

<https://learn.microsoft.com/en-us/azure/container-apps/get-started?tabs=bash>

<https://learn.microsoft.com/en-us/azure/container-apps/overview>

**Problem 4 \*\*\*\*\*: Azure Entra ID for Hybrid Identities, SMB 3.0 over Kerberoes**

Your company is using Microsoft Entra ID and Azure file shares. You are using the standard file shares (GPv2) with Geo-Zone Redundant Storage (GZRS) configured.

You are tasked with configuring identity-based authentication options for Server Message Block (SMB) access.

The following three conditions must be met:

* You should be able to use Microsoft Entra ID for authenticating hybrid user identities.
* Your end users should be able to access Azure file shares over the internet.
* Your end users do not require a line-of-sight to domain controllers.
* You need to implement the solution.

What should you do?

**Answer:**

To configure identity-based authentication for SMB access to your Azure file shares while satisfying the specified requirements, employing Microsoft Entra Kerberos for hybrid identities is a robust choice. Here’s a structured explanation and implementation guide:

* **Microsoft Entra ID Integration**: Microsoft Entra Kerberos leverages your existing Microsoft Entra ID (formerly known as Azure AD) for user authentication. This setup centralizes user identity management, simplifying administration.
* **Internet Access for Users**: With Microsoft Entra Kerberos, users can access Azure file shares over the internet, ensuring that access isn't limited to the internal corporate network.
* **No Line-of-Sight to Domain Controllers Required**: This solution utilizes Kerberos authentication managed through Azure AD, which means users don't need a direct connection to on-premises domain controllers to authenticate.

**Implementation Steps**

1. **Enable Microsoft Entra Kerberos**: Configure Kerberos for hybrid identities on your Azure storage account. This setup involves adjustments within Microsoft Entra ID and specific settings in the storage account.
2. **Grant Permissions**: Ensure that the relevant Microsoft Entra identities (users or groups) are granted appropriate share-level permissions on the Azure file share.
3. **Client Configuration**: Set up client machines to obtain Kerberos tickets from Microsoft Entra ID, typically requiring the installation and configuration of the Azure AD Connect tool.

**How It Works in a Hybrid Environment**

1. **User Authentication**: Users sign in using their regular credentials through Microsoft Entra ID, which validates these credentials and issues a Kerberos token.
2. **Kerberos Validation with Azure AD DS**: The Kerberos token is then used to authenticate against the Azure Files server. This server contacts Azure AD DS to validate the Kerberos token, which in turn verifies it with Microsoft Entra ID to confirm the user's identity.
3. **Access Granted**: Once the Kerberos token is validated and the user is confirmed to have the necessary permissions, access to the Azure Files shares is granted.

**Benefits of This Solution**

* **Centralized Authentication**: Users authenticate through a single system (Microsoft Entra ID), reducing the complexity of managing multiple authentication systems.
* **Enhanced Security**: Kerberos protocol enhances security by using cryptographic encryption to protect credentials and authentication tickets.
* **Accessibility from the Internet**: Users can access Azure Files shares from any location with internet connectivity, without the need for a direct link to domain controllers.
* **Seamless Integration of On-Premises Identities**: Azure AD DS enables users to utilize their familiar on-premises identities to access Azure Files shares, easing the transition for existing users.

In summary, Microsoft Entra Kerberos for hybrid identities, combined with Azure AD DS and Azure Files Share, provides a secure and streamlined method for users to access Azure file shares in a hybrid environment. This approach leverages the centralized authentication capabilities of Microsoft Entra ID, the secure Kerberos validation, and the support for on-premises identities through Azure AD DS.

* <https://learn.microsoft.com/en-us/azure/storage/files/storage-files-active-directory-overview>.
* <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview>
* <https://learn.microsoft.com/en-us/entra/identity/domain-services/overview>
* <https://learn.microsoft.com/en-us/windows-server/security/kerberos/kerberos-authentication-overview>
* <https://learn.microsoft.com/en-us/windows/win32/fileio/microsoft-smb-protocol-and-cifs-protocol-overview>

**Problem 5 \*\*: Network Troubleshooting**

You work for an organization that uses Azure Virtual Desktop (AVD) to facilitate remote working. At present, your users only have local profiles, but you are now implementing FSLogix to allow them to have network profiles. As part of this deployment, you create an Azure file share that will store the user profiles, and you configure the relevant file path onto each Windows 10 session host in the AVD host pool.

During initial login testing you find that users are logging in with an FSLogix temporary profile instead of a network profile. However, admins can successfully mount the file share on the same session hosts.

What is causing this to happen?

Choose the correct answer:

* Local profiles already exist for the users.
* The users do not have the relevant permissions to the Azure file share.
* The session host firewall is blocking the user profiles from loading.
* Azure file share network settings are too restrictive.

**Answer:**

The correct answer is: The users do not have the relevant permissions to the Azure file share.

**Explanation:**

The provided information indicates that users are unable to access their network profiles stored on the Azure file share, while admins can successfully mount the file share. This suggests that the issue lies with user permissions rather than with the file share configuration, firewall settings, or the presence of local profiles.

To resolve this issue:

* Verify user permissions: Ensure that the affected users have the necessary permissions to access the Azure file share and the specific folders containing their network profiles.
* Grant appropriate permissions: Grant the required permissions to the affected users or user groups. This may involve granting read, write, and execute permissions to the relevant folders.
* Test user access: After granting permissions, retry logging in with the affected users to verify that they can access their network profiles stored on the Azure file share.

Additional Considerations:

* Inheritance: Check if permissions are inherited from parent folders or if explicit permissions are required for the specific folders containing the network profiles.
* Permission propagation: Ensure that permission changes are propagated to all relevant subfolders and files within the user profile folders.
* Troubleshooting: If the issue persists, consider using tools like Active Directory Users and Computers or PowerShell to manage permissions and verify access rights.

By addressing user permissions, you should be able to resolve the issue of users logging in with FSLogix temporary profiles instead of their network profiles in the AVD environment.

**Problem 6 \*\*: Storage Account Choice**

You are creating a new Azure Storage account named mystorageaccount in the resource group named MyResourceGroup. The storage account should meet the following requirements:

* Support Hot, Cool, and Archive storage tiers
* Support automated blob lifecycle
* Provide fault tolerance in case of an Azure region-wide failure
* Minimize costs

You need to complete the command string to create the storage account.

How should you complete the command? To answer, select the command options from the drop-down menus.

**Answer:**

To set up an Azure Storage account that supports Hot, Cool, and Archive storage tiers, facilitates automated blob lifecycle management, provides fault tolerance in case of an Azure region-wide failure, and minimizes costs, follow these recommendations and use the specified command:

az storage account create -n mystorageaccount -g MyResourceGroup --kind StorageV2 --sku Standard\_GRS

Explanation and Justification:

* **--kind StorageV2**:
  + The **StorageV2** (general-purpose v2) account type is optimal as it supports all required storage tiers: Hot, Cool, and Archive. It also enables the setting up of automated blob lifecycle management, which allows you to automate the movement and deletion of blobs according to defined rules. This helps in efficiently managing data according to its usage patterns and reducing costs by moving less frequently accessed data to cheaper storage tiers.
* **--sku Standard\_GRS**:
  + Choosing **Standard\_GRS** (Geo-Redundant Storage Standard) ensures that your data is replicated in a secondary Azure region, offering high availability and regional fault tolerance. This setup is crucial for protecting against regional failures, fulfilling one of your core requirements.

**Additional Considerations:**

* **Avoiding "Storage" and "BlockBlobStorage" Types**:
  + The **Storage** (GPv1) account type does not support the necessary storage tiers (Hot, Cool, Archive), which fails to meet your requirements.
  + The **BlockBlobStorage** type only supports premium locally redundant and premium zone-redundant storage options, which could be costlier and do not align with the requirement to minimize costs.
* **Data Replication**:
  + Standard\_LRS (Locally Redundant Storage) and Standard\_ZRS (Zone-Redundant Storage) do not provide the same level of regional fault tolerance as Standard\_GRS, which is essential for your setup.

**Summary on Automated Blob Lifecycle:**

The automated blob lifecycle feature in Azure Blob Storage is critical for managing large-scale data economically and efficiently. By setting up policies within the Azure Management Portal, you can automate tasks such as transitioning older blobs to cooler storage or deleting outdated data that is no longer necessary. This not only optimizes storage costs but also reduces manual overhead in managing data retention and compliance.

By configuring your Azure Storage account with these settings, you ensure it is optimized for high availability, fault tolerance, efficient data management, and cost-effectiveness, according to your specific needs.

**Problem 7 \*\*\*: Network Troubleshooting / Network Monitor**

You are an Azure administrator for an organization that is running an on-premises site and Azure for hosting its mission-critical line-of-business (LOB) applications. One of your on-premises applications and an application hosted on Azure Virtual Machine Scale Sets connect to the same Azure storage account endpoint.

You need to do a comparative analysis of the latencies of the on-premises site and that of the application hosted in Azure.

What should you use?

**Answer:**

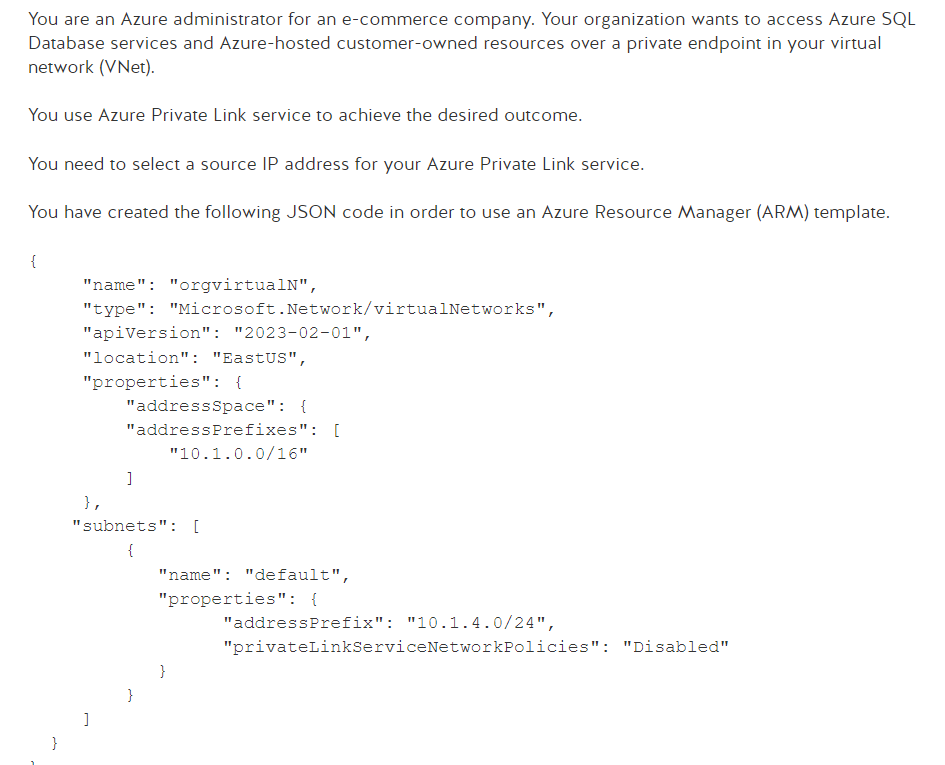
In this scenario, since you want to compare connectivity issues between the on-premises site and that of the application hosted in Azure Virtual Machine Scale Sets, which are connected to the same Azure storage account endpoint, you need to monitor, diagnose, and view connectivity-related metrics for a hybrid deployment. You should use the Azure Network Watcher Connection Monitor tool in this case. Connection Monitor provides a host of features, which are relevant to this scenario, including providing a unified, intuitive experience for Azure and hybrid monitoring needs, a cross-region, cross-workspace connectivity monitoring and higher probing frequencies and better visibility into network performance.

You should not use the Azure Network Watcher NSG diagnostics tool. You should use the network security group (NSG) diagnostic tool in Azure Network Watcher when you aim to diagnose and subsequently troubleshoot data flow between NSGs. This tool helps you to understand which network traffic is allowed or denied in your Azure virtual network (VNet) along with detailed information for debugging. NSG diagnostics can help you to verify that your NSG rules are set up properly.

You should not use the Azure Network Watcher VPN Troubleshoot feature from the Azure portal. You should use this tool when you aim to diagnose and subsequently troubleshoot VNet gateways and their connections. You can access VPN Troubleshoot feature either via Azure portal, Azure PowerShell, Azure CLI or REST API. This feature diagnoses the health of the gateway, or connection, and returns the appropriate results.

You should not use the Azure Network Watcher IP flow verify tool when you aim to diagnose and subsequently troubleshoot network connectivity at a packet level, i.e., which packets are allowed or denied to or from a virtual machine. The details consist of direction, protocol, local IP, remote IP, local port, and a remote port. Although any source or destination IP can be chosen, IP flow verify helps to quickly identify connectivity issues from or to the internet and from or to the on-premises environment.

**Problem 8 \*\*\*: Private Link**

****

**Answer:**

The privateLinkServiceNetworkPolicies setting in an Azure subnet configuration determines whether the subnet can be used for Private Link services. Azure Private Link is a service that enables secure access to Azure services over a private network rather than the public internet. Here are the main roles of this setting:

* Enhanced Security: By disabling privateLinkServiceNetworkPolicies, it allows resources within the subnet to connect to Azure services such as Azure SQL Database via Private Link, ensuring that traffic to and from these services never traverses the public internet.
* Network Isolation: This configuration helps isolate the virtual network and protect your data in transit by using the Azure private network for all communications, reducing the risk of exposure and interception by malicious actors.
* Connectivity Flexibility: Allowing or forbidding Private Link policies can help control data flow between your network and Azure services, thus enabling better network traffic management and compliance with corporate security policies.

In your JSON configuration, "privateLinkServiceNetworkPolicies": "Disabled" means that the private network policies for Private Link services are disabled, thus allowing resources in this specific subnet to connect to Azure service instances configured with Azure Private Link. This is often used in scenarios where data security and isolation are a priority.

----

In this scenario, you need to choose a source IP address for your Azure Private Link service. To disable network policies for the Private Link service source IP, you need to use the privateLinkServiceNetworkPolicies: Disabled setting. This specific setting is only applicable for the specific private IP address you choose as the source IP of the Private Link service.

The privateLinkServiceNetworkPolicies: Disabled setting is configured automatically if you are using Azure portal to create a Private Link service. If you are using PowerShell, or Azure CLI, or Azure Resource Manager (ARM) templates as in this scenario, you need to explicitly use the privateLinkServiceNetworkPolicies: Disabled configuration to be able to select a source IP address for your Azure Private Link service and disable network policies for the Private Link source IP.

For other resources in the subnet, network traffic is not filtered by Access Control Lists (ACL). ACL is applicable and offers low-level control of access to the folders in Azure Data Lake storage. For other resources in the subnet, network traffic is filtered by defined security rules of the network security groups (NSG).

**Probleme 9 \*\*\*: Entra ID and Entitlement management**

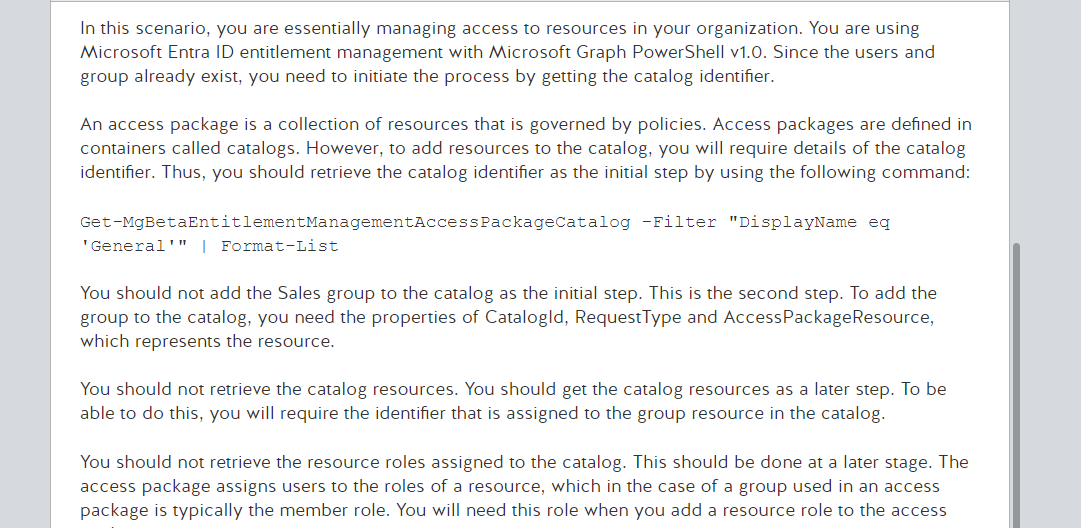
You are an Azure administrator in your organization and use Microsoft Azure AD Governance to manage identities and access across Azure. To assign rights to the "Sales" group, you need to use entitlement management, using Microsoft Graph PowerShell to manage these accesses.

**Answer:**

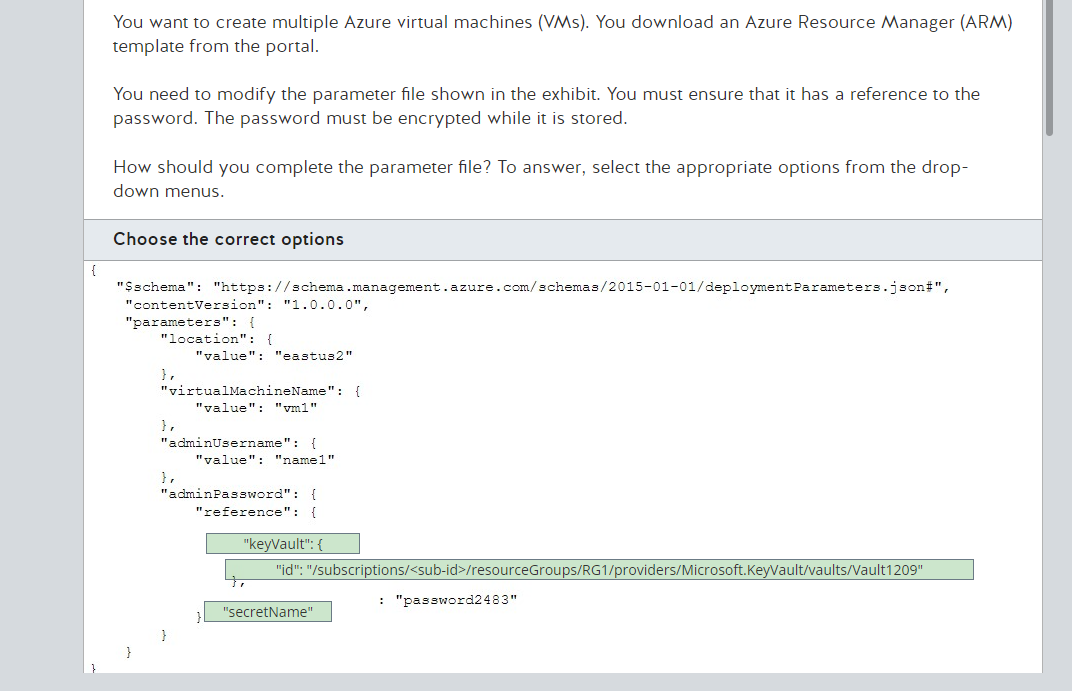
* First, you need to consult the general catalog which lists all available resources and access rights. This catalog is crucial as it clearly defines the resources that organization members can request or be assigned to, including applications, groups, licenses, and roles.
* Second, identify the specific rights that the "Sales" group requires to operate effectively. This may include rights to specific applications, groups, or other resources listed in the catalog.
* Third, configure access entitlements using Microsoft Graph PowerShell, applying policies and ensuring that each member of the "Sales" group receives the appropriate access according to the rules defined by your identity and access governance policy.

It is important to note that entitlement management in Microsoft Azure AD Governance is not limited solely to traditional Azure resources but extends to various digital resources, thus facilitating dynamic and flexible access governance. This method allows users to request access to resources via a self-service portal, which can be automatically or manually approved according to established policies.

By using this approach, you ensure secure and efficient management of identities and access in your organization, specifically addressing the needs of the "Sales" group while adhering to your company's security and compliance guidelines.



**Problem 10 \*\*\*: Key Vault with ARM Template**



Example

{

"$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentParameters.json#",

"contentVersion": "1.0.0.0",

"parameters": {

"adminLogin": {

"value": "exampleadmin"

},

"adminPassword": {

"reference": {

"keyVault": {

"id": "/subscriptions/<subscription-id>/resourceGroups/<rg-name>/providers/Microsoft.KeyVault/vaults/<vault-name>"

},

"secretName": "ExamplePassword"

}

},

"sqlServerName": {

"value": "<your-server-name>"

}

}

}

**Problem 11 \*\*: Azure Container Roles**

Your organization is using Azure Container Registry for storing and managing container images and related artifacts for various DevOps projects. You are an Azure administrator in the organization. You have been tasked with setting up appropriate authentication methods with an Azure container registry.

You assign a Microsoft Entra service principal to your registry.

You need to identify which Azure container registry roles would allow pull of a non-quarantined image from the registry.

Which three roles should you identify? Each correct answer presents part of the solution.

**Answer:**

In the scenario described, you need to identify the Azure Container Registry roles that allow the pull of a non-quarantined image from the registry. Based on the information provided, the relevant roles are:

* AcrPull: This role grants the ability to pull a non-quarantined image or other supported artifacts such as a Helm chart from a registry. It requires authentication with the registry using an authorized identity. This role is specifically designed for pulling images and is suitable for service principals used in automated workflows where only image retrieval is necessary.
* AcrPush: While this role primarily allows both pulling and pushing of images and artifacts like Helm charts, it's relevant here because it includes the capability to pull images. This role would be useful if the service principal also needs to push updates or new images to the registry.
* Owner: This is a more privileged role that includes the ability to pull and push images and assign roles to other users. It also allows for managing resources within the registry, such as creating or deleting the registry itself, pushing images, pulling images, deleting image data, and changing policies. This role is suitable if comprehensive control over the registry and its images is required.

Roles like AcrDelete and AcrImageSigner should not be used in this context as they are for deleting images and signing images, respectively, and do not pertain to the task of pulling non-quarantined images.

**Problem 12 \*\*\*: Roles and Subscriptions**

You manage a number of Azure subscriptions for a global organization and have ownership of all the subscriptions. You have been asked to use PowerShell to migrate the resources on an existing subscription called sub010 to a new subscription called sub020. After the migration, you find that all the Azure role assignments for individual resources have been orphaned on the virtual machines (VMs) but are still in place for the Resource Groups.

You need to find what has caused the missing role assignments to ensure that it is mitigated in future migrations.

**Answer:**

The migration was between subscriptions, and therefore, **any roles assigned directly to the resources were not moved**. All role assignments that are directly assigned to a resource or a child resource are not fully migrated, but instead orphaned in the destination subscription. Once the move has been completed, all Azure role assignments need to be re-created and the orphaned role assignments will be removed automatically.

It is not necessary to have used the Azure Portal for the migration. In this scenario, you were using PowerShell to complete the migration of resources from sub010 to sub020, but it is irrelevant if the task is done via PowerShell or via the Azure portal, the outcome would still be orphaned role assignments directly assigned to resources.

The user account moving the resources to sub020 did have the relevant permissions. The scenario states that you are the owner of all the subscriptions and you therefore have the highest level of permissions.

<https://learn.microsoft.com/en-us/azure/azure-resource-manager/management/move-resource-group-and-subscription>

**Problem 13 \*\*\*: Azure Recovery Services**

your company wants to start using Azure Recovery Services vaults to protect all virtual machines (VMs) and store backup data in their Azure subscription.

The question is:

What is the minimum number of Azure Recovery Services vaults required?

**Answer:**

To determine the minimum number of Azure Recovery Services vaults required to protect all virtual machines (VMs) and store backup data within an Azure subscription, we should consider several factors including scope, scale, and redundancy requirements.

Based on the Azure documentation, here's a relevant overview:

* Single Vault: Azure Recovery Services Vault is a management entity that stores backups and recovery points created over time. It manages backup policies and offers an integrated solution to protect your data. Technically, you can use a single Recovery Services vault to manage and store backups for all your VMs within the same region, assuming you don't exceed the scale limits of the vault.
* Scale Limits: Each Recovery Services vault supports a specific number of protected instances and data volume. For extensive deployments, *where the number of VMs or the volume of data exceeds these limits, you might need additional vaults.*
* Redundancy and Availability: For higher redundancy and availability, particularly for critical workloads, it's advisable to replicate backups across multiple regions. However, Azure Recovery Services Vaults do not support direct cross-region replication within the service. Instead, you can configure geo-redundant storage (GRS) which replicates your backup data to a secondary region automatically. This setup doesn't require a second vault but ensures data is available even if the primary region is down.
* Regulatory and Compliance Requirements: Sometimes, regulatory requirements dictate that data must be stored in multiple locations or specific geographical areas. In such cases, more than one vault might be necessary, especially if these locations span across different Azure regions which are not covered by the GRS.

Considering these points, the minimum number of Azure Recovery Services vaults required depends primarily on your specific requirements:

For basic backup needs with regional coverage and under the scale limits, one vault is sufficient.

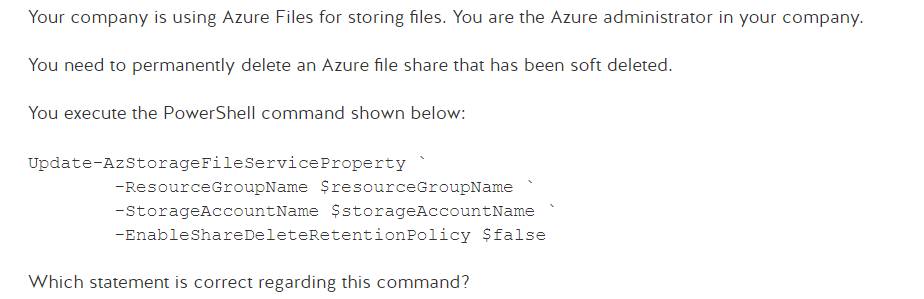
For compliance, cross-regional redundancy, or scale beyond the limits of a single vault, additional vaults or different configurations might be necessary.

You can manage and protect all VMs in a single vault if they are located in the same region and the amount of data and number of VMs do not exceed the vault’s capacity. If these conditions are met and no specific compliance or cross-regional requirements exist, then the minimum number of Recovery Services vaults required is one.

**What I should Know about Azure Recovery Services**

* Understanding Backup Policies:
  + Backup policies in Azure dictate the schedule (when backups are taken) and the retention period (how long backups are kept). Policies should be tailored based on the backup needs, compliance requirements, and the type of data being backed up. For instance, mission-critical resources may require more frequent backups to ensure minimal data loss in disaster scenarios​ (Microsoft Learn)​.
* VM Agent and Backup Extension:
  + Azure VMs must have the Azure VM agent installed to facilitate backups. For VMs created from Azure Marketplace images, the agent is typically pre-installed. However, for custom or migrated VMs, manual installation might be necessary. Once enabled, Azure Backup automatically manages the backup extension on these agents​ (Microsoft Learn)​.
* Storage and Replication Types:
  + Recovery Services vaults offer different storage replication options such as Locally Redundant Storage (LRS), Zone-Redundant Storage (ZRS), and Geo-Redundant Storage (GRS). The choice depends on your backup durability requirements and budget. Modifying the replication type after setup requires vault recreation​ (Microsoft Learn)​.
* Backup from VM Settings:
  + Backing up directly from the VM settings in the Azure portal is straightforward. You can select an existing vault or create a new one, choose or create a backup policy, and enable backup. This process involves choosing backup policies that specify the frequency and retention of backups​ (Microsoft Learn)​.
* Snapshot and Restore Point Considerations:
  + Azure Backup provides different snapshot consistency levels depending on the VM's state and configuration. These include application-consistent, file-system consistent, and crash-consistent snapshots, each suitable for different recovery scenarios​ (Microsoft Learn)​.
* Incremental Backup and Restore:
  + Azure Backup is generally incremental, meaning only the changes since the last backup are saved, which helps in optimizing storage usage and reducing backup time. The backup process includes preparing backups, data transfer, and might have a delay between snapshot creation and copying to the vault​ (Microsoft Learn)​.
* Enhanced Backup Policies:
  + For scenarios requiring frequent backups, enhanced backup policies can be configured to take backups hourly, with options to set specific start times, intervals, and retention durations within the policy settings. This is crucial for high-transaction environments where data changes frequently​ (Microsoft Learn)​.

**Problem 14 \*\*: Soft Delete**



**Answer:**

To permanently delete a file share that has been soft deleted, you must undelete it, disable soft delete, and then delete it again. The PowerShell command does not disable soft delete on your storage account and permanently delete an azure file share that has been soft deleted. The Powershell command only disables soft delete on your storage account and permanently delete an azure file share that has been soft deleted. The PowerShell command does not undelete the file share..

**Problem 15 \*\*: VNET Peering**

You have two Azure virtual machines (VMs) running Windows as shown in the first exhibit.

You create a virtual network (VNet) peering by executing the following PowerShell commands:

$vnet1 = Get-AzVirtualNetwork -Name 'VNet1' -ResourceGroupName 'RG1'

$vnet2 = Get-AzVirtualNetwork -Name 'VNet2' -ResourceGroupName 'RG2' Add-AzVirtualNetworkPeering -Name 'VNet1-VNet2' -VirtualNetwork $vnet1 -RemoteVirtualNetworkId $vnet2.Id

The peering overview of VNet1 is shown in the second exhibit.

You open the local Windows firewalls by running the following PowerShell command on both VMs:

New-NetFirewallRule DisplayName 'Enable ping' -Protocol 'ICMPv4'

A ping from VM1 to the private IP address of VM2 fails.

You need to make sure that VM1 can connect to VM2.

What should you do?

**Answer:**

You should add a virtual network (VNet) peering from VNet2 to VNet1. When peering two VNets**, peering connections are required in both directions.**

The peering from VNet1 to VNet2 has already been created, so now you need to create the reverse peering. As shown in the second exhibit, the VNet1 to VNet2 peering status is Initiated. The peering status should be Connected. This will be the case when two-way peering is created.

You should not create an network security group (NSG). An NSG is used to filter network traffic. This cannot solve the problem here.

You should not modify the VNet address space of VNet2. VNets can only be peered if their address spaces do not overlap. VNet1 has a classless inter-domain routing (CIDR) range of 10.10.0.0/16, which converts to an IP range of 10.10.0.0 to 10.10.255.255. VNet2 has a CIDR range of 10.20.0.0/16. This translates to an IP range of 10.20.0.0 to 10.20.255.255. So there is no overlap.

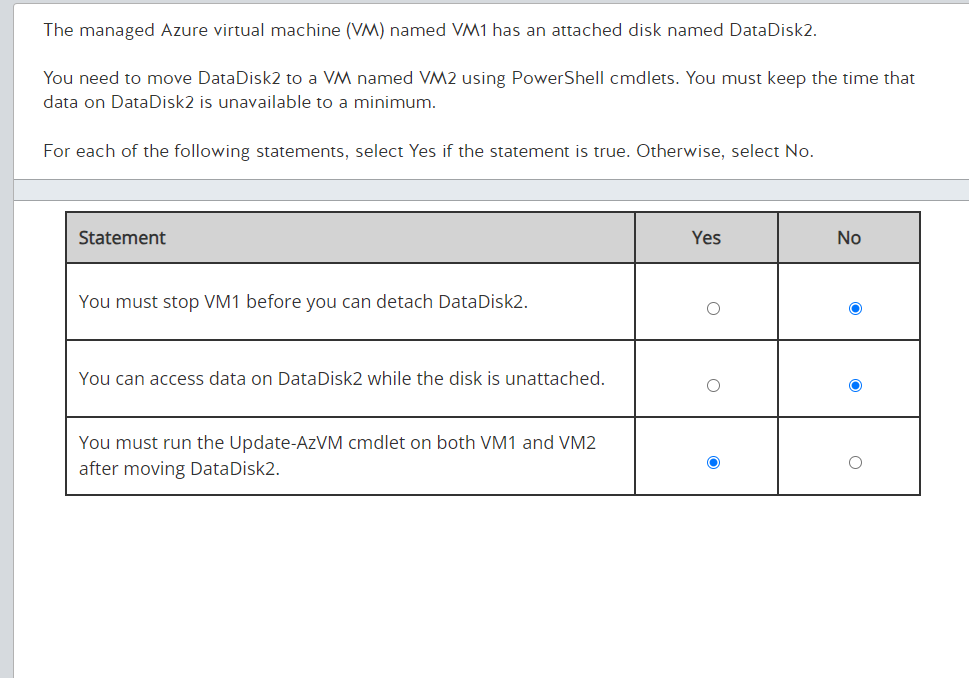
You should not modify the subnet Default2 address range. The IP ranges in the VNets do not overlap. This means that the address ranges of the subnets also do not overlap because the subnet range is inside the VNet range.

You should not create a virtual network gateway. In this case, VNet peering is used. This configuration does not use a virtual network gateway but, generally, it is possible to connect two VNets using two VNet gateways.

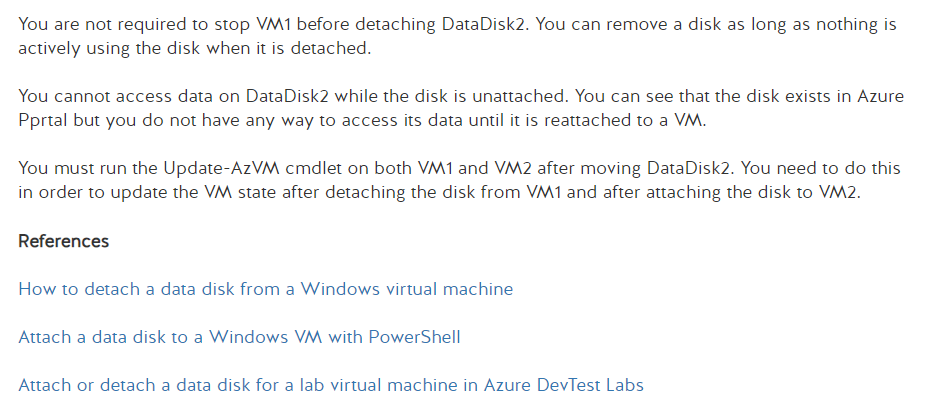
You should not move VM2 to the East US region. Peering across Azure regions is supported (global virtual network peering). So peering between VM1 (East US) and VM2 (East US 2) is possible.

You should not use the public IP address to connect. VNet peering is set up to create connectivity using the private IP addresses of the virtual machines.

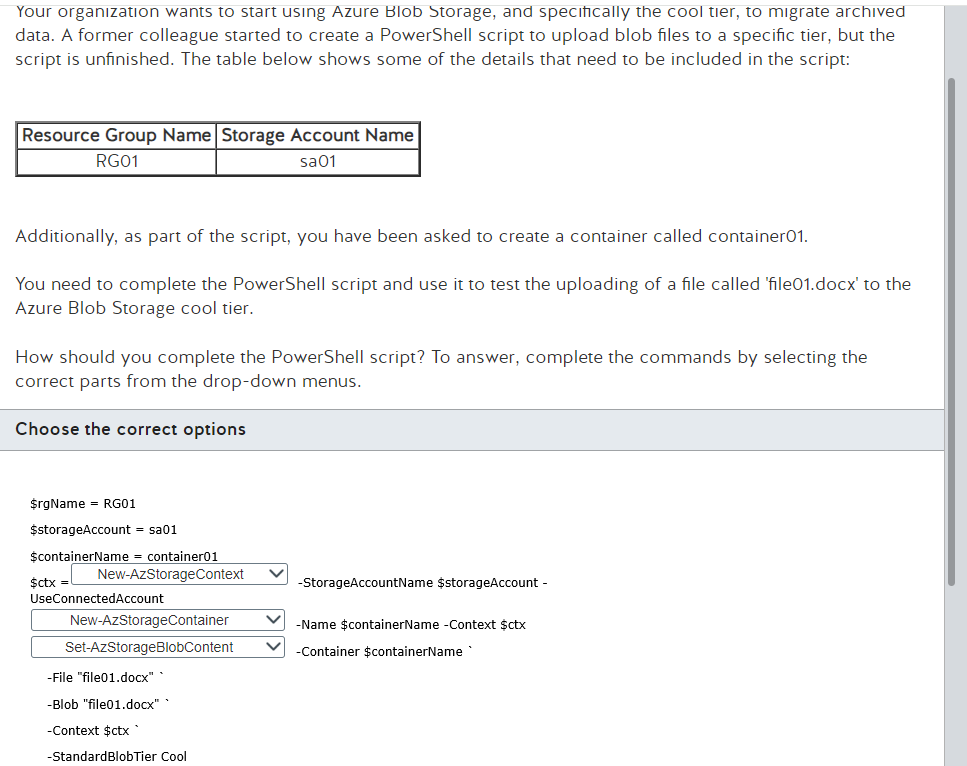
**Problem 16 \*\*: Disk Attachments**



**Answer:**



**Problem 17 \*\*: Storage Context**

****

**Answer:**

The New-AZStorageContext and Get-AzStorageAccount are both Azure PowerShell cmdlets used in managing Azure Storage, but they serve different purposes and are used in different scenarios. Here’s a detailed explanation of each, including when to use one over the other:

1. **New-AZStorageContext**

Purpose: The New-AZStorageContext cmdlet creates a storage context object that Azure PowerShell can use to authenticate requests against an Azure Storage account. This context encapsulates credentials and storage account details (like storage account name, access keys, or a connection string) required to access Azure Storage services (Blob, Queue, Table, and File services).

Usage:

* Creating Context: It is used when you need to perform operations on the storage services and you need a way to pass authentication details efficiently between cmdlets.
* Authentication: Useful for scenarios where you need to specify different credentials or use specific authentication methods, such as access keys, SAS tokens, or Azure AD credentials.
* Multiple Operations: Particularly useful in scripts where multiple operations need to be performed on the storage account, and you want to avoid passing credentials each time.
* Example: Creating a storage context to use in subsequent blob operations.

1. **Get-AzStorageAccount**

Purpose: The Get-AzStorageAccount cmdlet retrieves information about one or more storage accounts in your Azure subscription. It provides details like the storage account type, location, status, and other properties related to the storage account.

Usage:

* Retrieving Information: Used when you need details about the storage accounts in your subscription or a specific storage account.
* Configuration Audits and Monitoring: Useful for monitoring and managing the configurations of storage accounts.
* Direct Operations: While it does not return a storage context, you can extract keys and other data needed to create a storage context or for other configurations.
* Example: Retrieving a storage account and extracting its key to use in creating a storage context.

1. **When to Use Which?**

Use New-AZStorageContext when you need a context object for authenticating requests to Azure Storage services. It’s especially useful in scripts or repetitive operations where you pass the context to various cmdlets that operate on storage objects.

Use Get-AzStorageAccount when you need to retrieve or inspect properties of storage accounts in your Azure subscription. It’s more about data retrieval and less about performing operations on the storage itself.

Both cmdlets are often used together in scripts to manage Azure Storage, where you first retrieve storage account details with Get-AzStorageAccount and then create a context with New-AZStorageContext to perform operations on that account.

**Problem 18 \*\*\*: Azure Policy and Service Provider**

A company has an existing on-premises environment and a newly created Azure subscription. You need to start testing cloud features and services with a view to eventually migrating the company environment to the cloud. You have been given their global administrator rights and the resource policy confirmation role on the subscription level and you need to test Azure Policy first.

You have downloaded version 2.38 of the Azure Command-Line-Interface (CLI) to configure new policies, but you find that the Azure Policies you are creating are not working with your subscription.

You need to find the cause of this problem.

What is causing the Azure Policy to not function with your subscription when using the Azure CLI?

* You do not have the relevant access to the subscription.
* Your version of the Azure CLI needs updating.
* You have not registered the Azure Policy Insights resource provider.
* You do not have the relevant role assignment to manage Azure Policy.

**Answer:**

The issue you're experiencing with Azure Policy not functioning correctly could be due to several factors. Based on the options you've provided and the context of the situation, let's analyze each potential cause:

1. You do not have the relevant access to the subscription.
   * Since you mentioned having global administrator rights and the resource policy confirmation role at the subscription level, this should provide sufficient permissions to manage and apply Azure Policies. Therefore, this is unlikely to be the cause.
2. Your version of the Azure CLI needs updating.
   * Azure CLI is frequently updated to support new features and services, including Azure Policy. Although Azure CLI version 2.38 might still be compatible with Azure Policy commands, it’s possible that certain functionalities or newer policy definitions are not fully supported. However, this is less likely to be the issue unless you're using features introduced after 2.38. It's good practice to use the latest version of Azure CLI, but not typically a blocker for basic policy functionality.
3. You have not registered the Azure Policy Insights resource provider.
   * Azure Policy and its compliance features require the Azure Policy Insights resource provider to be registered with your subscription. If this provider is not registered, Azure Policy may not function as expected because it relies on this provider for compliance data and other insights. This could very well be the cause if the error messages or behaviors you're seeing relate to data retrieval or compliance evaluation.
4. You do not have the relevant role assignment to manage Azure Policy.
   * As you've stated you have global administrator rights and the resource policy confirmation role, this typically includes permissions necessary for managing Azure Policies. Thus, this is unlikely to be the problem unless there's an error in how permissions are assigned or inherited.

Given these possibilities, the most likely cause of your issue is that you have not registered the Azure Policy Insights resource provider. This is a common oversight and can easily be addressed by registering the provider in your Azure subscription. Here’s how you can check and register the Azure Policy Insights resource provider using Azure CLI:

**Problem 19 \*\*\*: SAS over Stored Access Policy**

You are an Azure administrator for a manufacturing organization. You are using shared access signature (SAS) to configure control over storage accounts. You create a stored access policy as an additional level of control over SAS on the server side for file shares.

You need to modify a stored access policy.

What should you do?

Choose the correct answer

* Execute a Set Container ACL operation with public read access for blobs only.
* Execute a Set Table ACL operation.
* Execute a Set Share ACL operation with the SMB protocol.

**Answer:**

To address your question on why the correct action to modify a stored access policy for file shares in Azure involves using a "Set Share ACL" operation, let's first understand the context and components involved:

**Context: Stored Access Policies**

Stored access policies provide an additional layer of control over Shared Access Signatures (SAS) at the service level. These policies are set on the resources like blob containers, file shares, queues, or tables and allow you to manage, define, and constrain policies centrally on the resource itself. This includes aspects like the start time, expiry time, and permissions for the SAS tokens that leverage the policy.

**Key Points in Options:**

1. Execute a Set Container ACL operation with public read access for blobs only.
   * This option refers specifically to blob containers within a Blob storage. The operation modifies the Access Control List (ACL) of a blob container, not a file share. This is used primarily in scenarios dealing with blobs and is irrelevant for file shares.
2. Execute a Set Table ACL operation.
   * This option is related to Azure Table storage, where you set the ACL on a table to control access. Again, this doesn’t apply to file shares but to Table storage, which is used to store structured data in a key-value format.
3. Execute a Set Share ACL operation with the SMB protocol.
   * This is the correct option when dealing with file shares in Azure Storage. Azure file shares support using the SMB protocol, and modifying the ACLs of a file share involves setting policies that dictate how SAS tokens can operate within those shares. This operation directly affects the stored access policies associated with a file share.

**Why the Third Option is Correct for File Shares:**

* Specific to File Shares: The "Set Share ACL" operation is designed to manage access control lists specifically for Azure file shares. Since your requirement is to modify a stored access policy for a file share, this operation is directly relevant.
* Protocol Relevance: The SMB (Server Message Block) protocol is used by Azure File Shares for providing standard file system semantics. When you configure ACLs using the SMB protocol, you are essentially setting up permissions and rules that apply to client interactions via SMB, which is critical for environments that integrate with or emulate traditional file server configurations.
* Control Over SAS via Stored Access Policies: By executing a Set Share ACL operation, you're able to modify the policies that govern how SAS tokens are created and used for the file share, aligning with security and access requirements.

**Summary:**

The reason you need to use the "Set Share ACL" operation for modifying a stored access policy on a file share is that it's the only option that specifically targets the security and access controls for file shares. The other options are targeting different storage types (blobs and tables) and would not impact the access rules of file shares. Therefore, understanding the specific type of Azure storage and the corresponding operations is crucial for correctly managing access and security configurations in Azure.

**What I should Know about Azure Recovery Services**

1. Define Stored Access Policy

* Purpose: Stored access policies provide additional control over service-level SAS without regenerating the SAS. You can modify or revoke permissions via a stored access policy.
* Components: A stored access policy can include up to five identifiers and can define permissions, start time, expiry time, and IP address range for a SAS.
* Flexibility: Changing a stored access policy automatically impacts all associated SAS tokens, enabling easier and centralized management of permissions.

1. Set Share ACL

* Application: This API is used specifically for Azure File shares.
* Function: Allows you to set the ACL for a file share, which includes creating or modifying stored access policies.
* Usage: Essential for controlling access to file shares using both shared keys and SAS tokens. The ACL defines who can access the share and what operations they can perform.

1. Set Table ACL

* Application: This operation is specific to Azure Table storage.
* Function: Used to manage ACLs for table storage, which can include setting stored access policies similar to file shares and blob containers.
* Usage: Important for defining and restricting access to data in Table storage, particularly useful in scenarios involving large amounts of structured data.

1. Set Container ACL

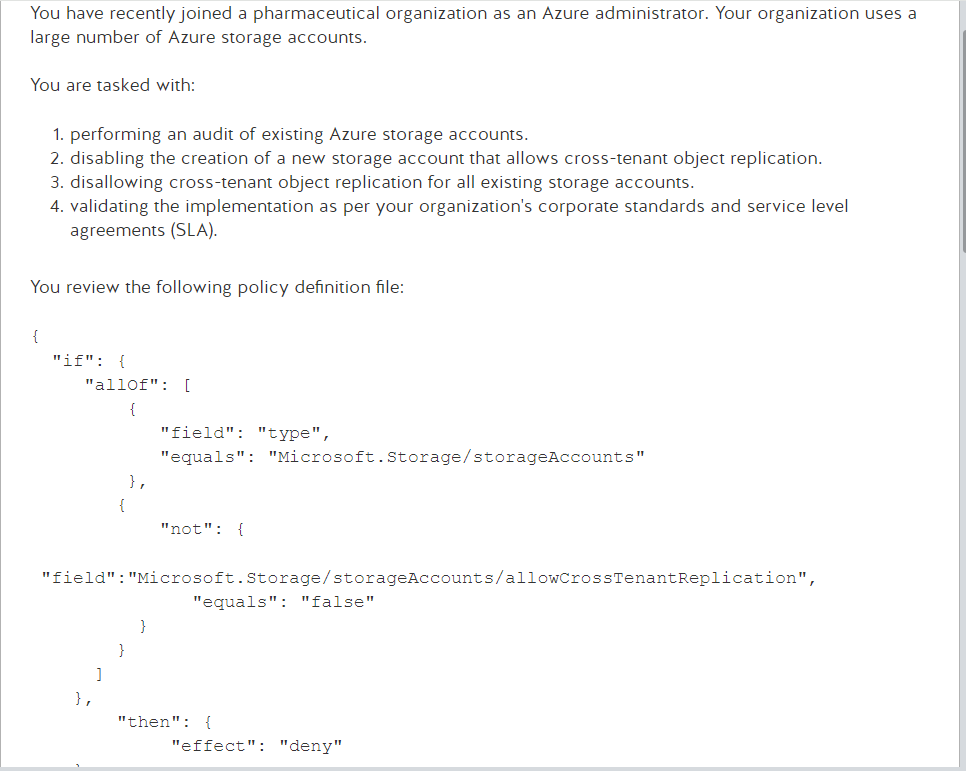
* Application: Pertains to blob containers in Azure Blob storage.
* Function: Allows setting the ACL for a blob container, including stored access policies that manage SAS permissions.
* Usage: Critical for managing and securing blob data, which is often used for storing large unstructured data like videos, backups, and logs.

**General Tips for AZ-104:**

* Understand Access Control Lists (ACLs): Know how ACLs control access to resources and how they can be managed via stored access policies.
* Know the Differences: Be clear on the differences in applying ACLs across different storage types (file shares, blob containers, and table storage) and how they affect access management.
* Stored Access Policies: Learn how modifying a stored access policy affects SAS tokens and thus the security and accessibility of the storage resources.
* Practical Usage: Familiarize yourself with practical use-cases where setting or modifying ACLs might be necessary, such as in highly secure environments or where access needs to be regularly rotated or updated.

These key points align with Azure administration tasks related to managing security and access control, which are crucial components of the AZ-104 exam. Understanding these will help you grasp broader Azure security and management concepts, thereby aiding in effective preparation for the exam.

Problem: Azure Policy



The JSON policy definition you provided is an example of an Azure Policy definition used to enforce specific conditions on Azure resources—in this case, Azure storage accounts. Here's how to read and understand this policy:

Structure of the Policy:

1. "if" Block:
   * This block defines the conditions under which the policy will apply. It uses a logical "allof" operator, meaning all listed conditions must be true for the policy to take effect.
   * Condition 1:
     + "field": "type"
     + "equals": "Microsoft.Storage/storageAccounts"
     + This condition specifies that the policy applies only to resources of the type Microsoft.Storage/storageAccounts, which are Azure storage accounts.
   * Condition 2:
     + Wrapped in a "not" operator, meaning the condition inside it should not be true for the allof condition to hold.
     + "field": "Microsoft.Storage/storageAccounts/allowCrossTenantReplication"
     + "equals": "false"
     + This condition checks if the allowCrossTenantReplication property of a storage account is set to false. The not wrapper inverts this, so the condition inside the allof statement is checking that allowCrossTenantReplication should not be false (i.e., it should be true).
2. "then" Block:
   * This block defines what action should be taken when the conditions in the "if" block are met.
   * "effect": "deny"
   * This action tells Azure to deny any operation that would result in a situation where the conditions in the "if" block are true. In this case, it means denying operations that would create or maintain a storage account with allowCrossTenantReplication enabled.

Purpose of the Policy:

* Disabling Cross-Tenant Object Replication: The policy is designed to prevent the creation or updating of storage accounts that have cross-tenant object replication enabled. This aligns with organizational policies that may require data isolation between tenants for security or compliance reasons.

**How it Helps with Your Tasks:**

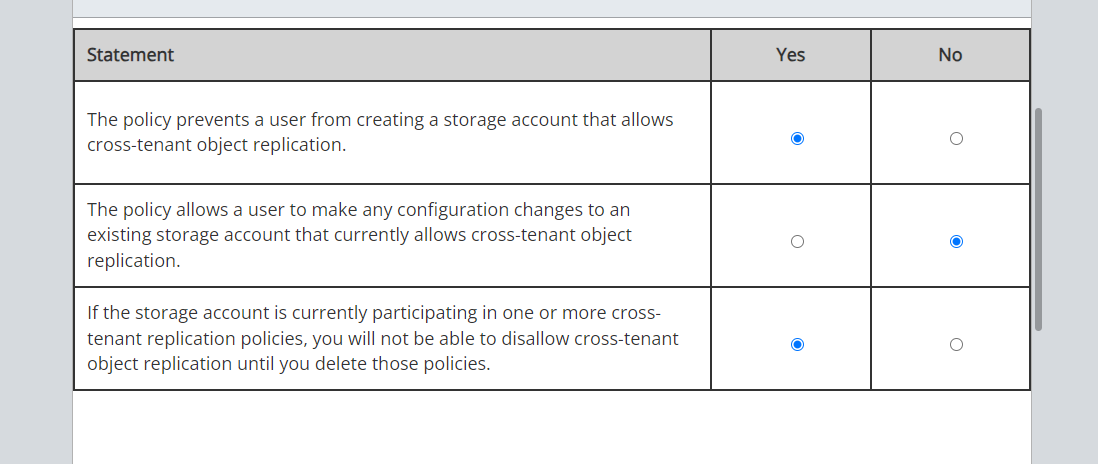
Performing an Audit: While this policy itself does not perform an audit, it ensures compliance by preventing unwanted configuration changes. For auditing, you would need to run compliance scans or use Azure monitoring tools to review existing configurations.

Disabling Creation of New Storage Account with Cross-Tenant Replication: This policy directly helps with this task by denying operations that would enable cross-tenant replication.

Disallowing Cross-Tenant Replication for All Existing Storage Accounts: Again, while this policy prevents changes that would enable such settings, ensuring existing accounts do not have it enabled would require additional auditing and potentially separate remediation actions.

Validating Implementation According to SLA: By enforcing security best practices and compliance standards, this policy helps maintain the standards specified in Service Level Agreements related to security and data isolation.

This policy definition effectively manages and restricts storage account configurations to align with specific organizational standards and compliance needs related to cross-tenant data replication.



In this scenario, you are required to review, validate and ensure that appropriate Azure policies are in place. Azure Policy supports effects that determine what happens when a policy rule is evaluated against a resource. To ensure that storage accounts in your organization disallow cross-tenant replication, you need to create a policy that prevents the creation of a new storage account that allows cross-tenant object replication policies.

The policy shown in this scenario prevents a user from creating a storage account that allows cross-tenant object replication. The AllowCrossTenantReplication property for the storage account is set to false in order to disable account creation or configuration updates, in compliance with the policy. An important point to note is that there are two possibilities:

* If a storage account does not currently participate in any cross-tenant object replication policies, then setting the AllowCrossTenantReplication property to false prevents future configuration of cross-tenant object replication policies with this storage account as the source or destination.
* If a storage account currently is in one or more cross-tenant object replication policies, then you cannot set the AllowCrossTenantReplication property to false until you delete the existing cross-tenant policies.

The policy does not allow a user to make any configuration changes to an existing storage account that currently allows cross-tenant object replication. Since our enforcement policy uses "effect": "deny", this prevents a request that would create or modify a storage account to allow cross-tenant object replication. The Deny policy prevents all configuration changes to an existing account if the enforcement policy uses "effect": "deny".

If a storage account currently participates in one or more cross-tenant object replication policies, then setting the AllowCrossTenantReplication property to false is not permitted until you delete the existing cross-tenant policies. Only if you have deleted the existing cross-tenant policies, can you set the AllowCrossTenantReplication property to false.

1. **Qu'est-ce que AllowCrossTenantReplication?**

AllowCrossTenantReplication est une propriété que l'on peut définir dans la configuration d'un compte de stockage sur Azure. Cette propriété détermine si un compte de stockage peut participer à la réplication d'objets entre différents locataires (tenants). Un "tenant" est une instance d'utilisateur ou de groupe d'utilisateurs qui partage un accès commun à certains services cloud.

Si AllowCrossTenantReplication est réglé sur true, cela signifie que le compte de stockage peut envoyer ou recevoir des données avec des comptes appartenant à d'autres locataires, facilitant ainsi la réplication des données entre eux.

Si elle est réglée sur false, le compte de stockage ne peut pas participer à de telles activités de réplication croisée.

1. **La réplication avec AllowCrossTenantReplication**

La réplication d'objets cross-tenant avec AllowCrossTenantReplication activé permet essentiellement à un compte de stockage de dupliquer automatiquement ses données à un autre compte de stockage situé dans un tenant différent. Cela peut être utilisé pour des raisons de sauvegarde, de reprise après sinistre, ou simplement pour partager des données entre différentes parties d'une organisation ou entre différentes organisations.

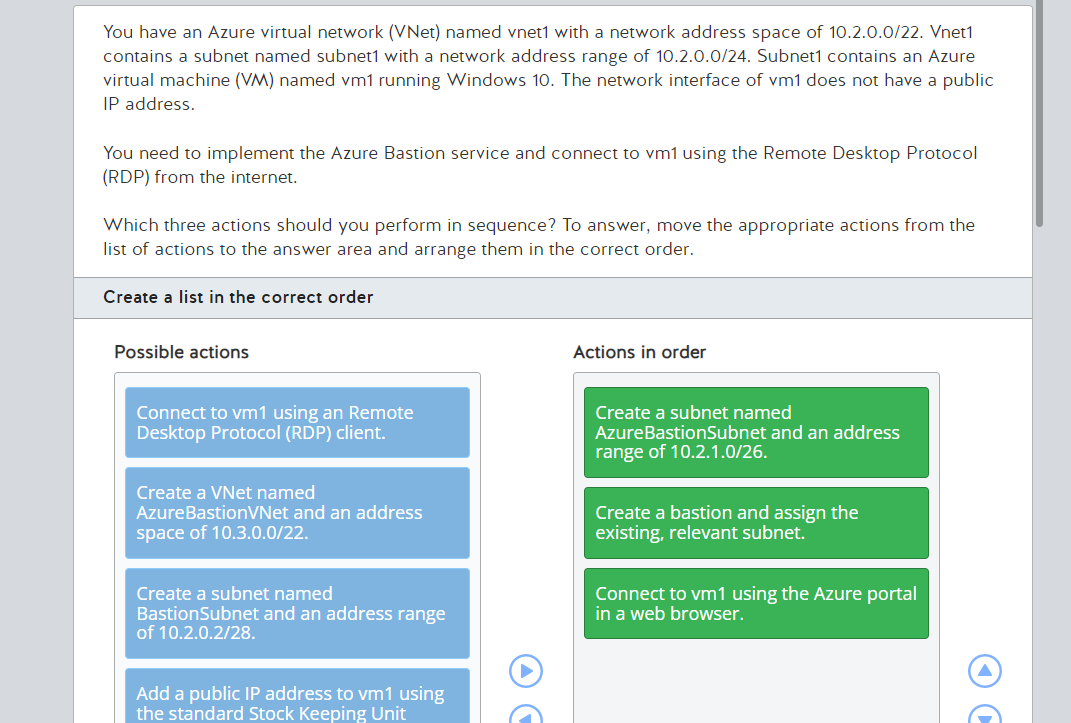
1. **Gestion des comptes de stockage participant déjà à la réplication cross-tenant**

Si un compte de stockage participe déjà à une ou plusieurs politiques de réplication cross-tenant, vous ne pouvez pas simplement désactiver AllowCrossTenantReplication. Cela est dû au fait que des politiques et des configurations existantes pourraient déjà être en place pour gérer ces échanges de données. Pour désactiver la réplication croisée, vous devez d'abord supprimer toutes les politiques de réplication cross-tenant existantes associées à ce compte de stockage. Ce n'est qu'après la suppression de ces politiques que vous pouvez régler AllowCrossTenantReplication sur false.

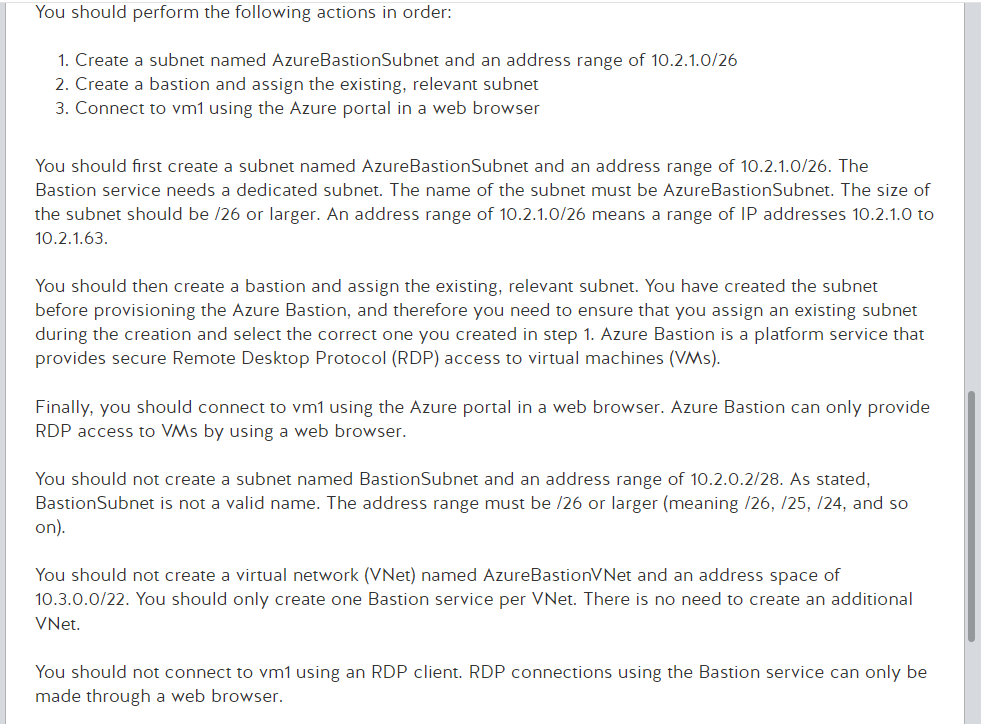
Cela signifie que tant que des politiques actives permettent la réplication croisée, les modifications pour désactiver cette fonctionnalité sont bloquées pour éviter des interruptions ou des perturbations inattendues dans la gestion des données.

Azure bastion

**Problem 20 \*\*\*: Azure Policy and Service Provider**



**Answer:**



**Problem 21 \*\*\*: Availability Set**

You have a Microsoft Azure subscription named Sub1. You deployed a Windows Server 2016 virtual machine (VM) named VM1 to Sub1.

The question is what you need to do to change the availability set assignment for VM1. The answer choices are:

* Redeploy VM1 from a recovery point.
* Assign VM1 to the new availability set.
* Migrate VM1 to another Azure region.
* Move VM1 to a different availability zone.

**Answer:**

You should redeploy VM1 from a recovery point. In Azure, you can assign a virtual machine (VM) to an availability set only during initial deployment. Therefore, to reassign the VM to another availability set, you would need to perform the following actions:

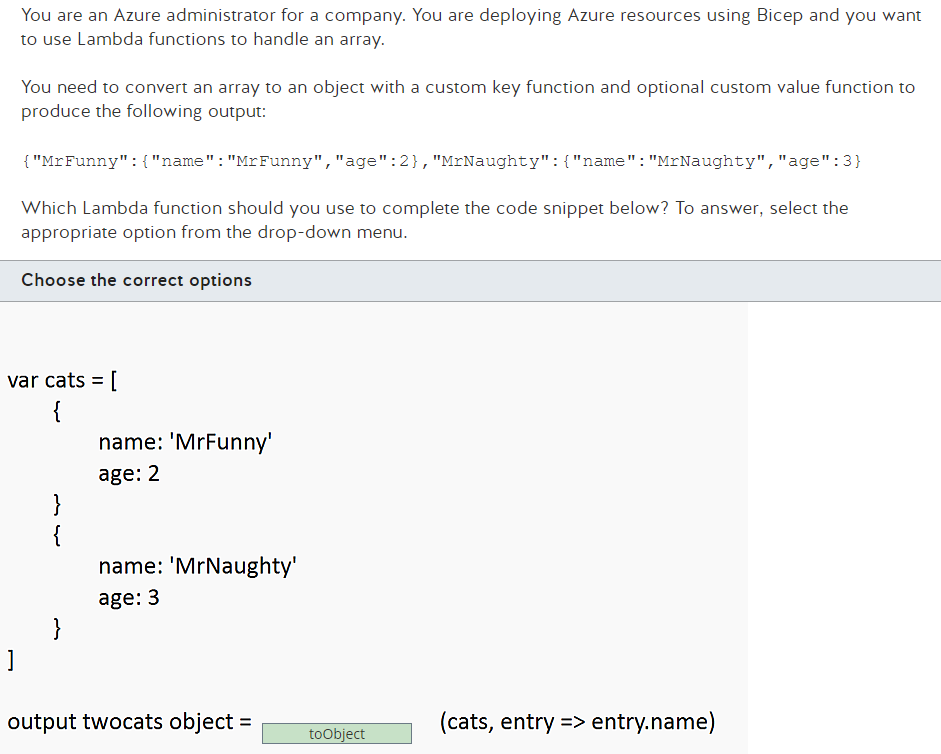
* Make a backup of the current VM
* Delete the current VM
* Deploy a new VM based on the most recent restore point to the correct availability set

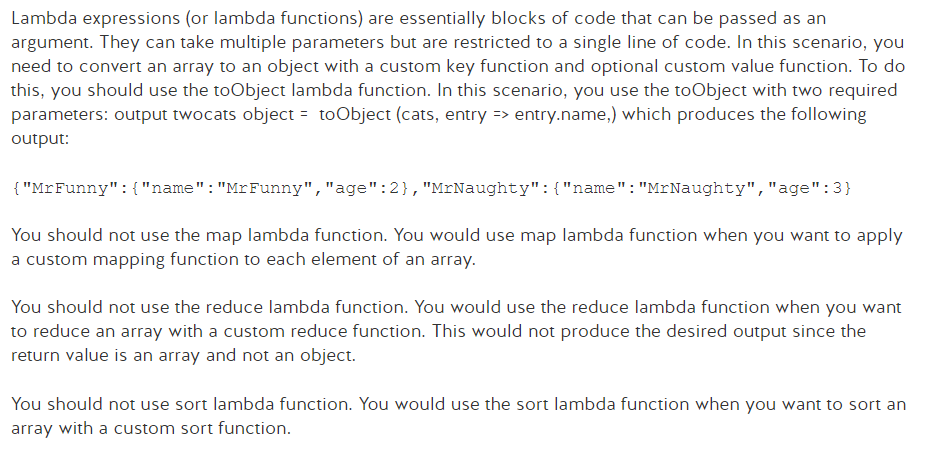
You should not move VM1 to a different availability zone because availability zones are mutually exclusive from availability sets.

You should not assign VM1 to the new availability set because, as mentioned previously, this is not a supported action in the Azure service fabric.

You should not migrate VM1 to another Azure region because by definition members of the same availability set must reside in the same region.

**Problem 22 \*\*\*: BICEP**





**Problem 23 \*\*\*: Azure Backup & PowerShell**

You have an Azure subscription. You are not currently using Azure Backup. You plan to back up a virtual machine (VM) named VM1 using Azure Backup. The backup should use the default backup policy to run daily backups.

You need to identify the PowerShell cmdlets required to configure recurring backups using the default backup policy

**Answer:**

1. Register-AzResourceProvider
   * You should first run the Register-AzResourceProvider cmdlet. This is required the first time you use Azure Backup. This registers the Azure Recovery Service provider in your subscription.
2. New-AzRecoveryServicesVault
   * You should then run the New-AzRecoveryServicesVault cmdlet to create a Recovery Services vault to store the backups.
3. Enable-AzRecoveryServicesBackupProtection
   * Finally, you should run the Enable-AzRecoveryServicesBackupProtection cmdlet to enable backups to run using the default backup policy. Once enabled, backups will run automatically.

Additional Notes:

* You should not run the Backup-AzRecoveryServicesBackupItem cmdlet. This cmdlet is used to run an on-demand backup based on your backup policy.
* You should not run the Enable-AzRecoveryServicesBackupAutoProtection cmdlet. This cmdlet is used to let users automatically protect all unprotected and new databases.
* You should not run the Set-AzRecoveryServicesBackupProperty cmdlet. This cmdlet sets backup storage properties for a Backup Services vault. This is not required when initially setting up Azure Backup.

Explanation for Needing New-AzRecoveryServicesVault:

The New-AzRecoveryServicesVault cmdlet is crucial because it creates a Recovery Services vault, which is a management entity where your backup data is stored. The vault holds the backup policies and the recovery points created over time. Each vault is associated with one or more Azure subscriptions, and the data stored in a vault is specific to a particular geographical region.

Why it's necessary:

* Central Management: It enables centralized management of backup and recovery operations across your resources, which is critical for maintaining governance and compliance with data protection policies.
* Scalability and Isolation: The vault allows for scalable storage solutions that can handle a large number of backup operations and data sizes while providing isolation between different organizational entities or environments.
* Data Security and Compliance: The Recovery Services vault provides built-in features to secure backup data, such as encryption, and helps in complying with various data protection regulations specific to the region where the vault is located.

In summary, the New-AzRecoveryServicesVault cmdlet is fundamental for initiating Azure Backup because it sets up the infrastructure required to store and manage the backups securely and efficiently.

**Problem 24 \*\*\*: ARM and Vault**

You download an Azure Resource Manager (ARM) template that was created from an existing virtual machine (VM). You plan to use the template to deploy 50 new VMs.

You need to modify the template to reference an administrative password. The password should not be stored in clear text.

What should you do first?

**Answer:**

Creating a virtual machine in Azure, exporting its ARM template, and then updating the template to securely handle passwords using Azure Key Vault is a multi-step process. Below, I'll guide you through each of these steps using Azure CLI and PowerShell to manage resources effectively.

### Step 1: Set Up Azure CLI and Log In

First, ensure you have the Azure CLI installed on your system. You can download it from [Azure's official site](https://docs.microsoft.com/en-us/cli/azure/install-azure-cli). Once installed, open your command line interface and log in to your Azure account:

```bash

az login

```

### Step 2: Create a Virtual Machine

Let's create a resource group and then a virtual machine. Replace placeholders like `<resource-group>`, `<location>`, `<vm-name>`, and `<admin-username>` with your actual values.

```bash

# Create a resource group

az group create --name <resource-group> --location <location>

# Create a VM

az vm create \

--resource-group <resource-group> \

--name <vm-name> \

--image UbuntuLTS \

--admin-username <admin-username> \

--generate-ssh-keys

```

This command creates a basic Linux VM with SSH keys for access. If you need a Windows VM or different settings, adjust the parameters accordingly.

### Step 3: Export ARM Template of the VM

After the VM is created, you can export its ARM template:

```bash

az vm show \

--resource-group <resource-group> \

--name <vm-name> \

--query "properties.storageProfile" \

--output json

```

This command outputs the storage profile part of the VM configuration. For a complete template, you can use:

```bash

az resource export \

--resource-group <resource-group> \

--name <vm-name> \

--resource-type "Microsoft.Compute/virtualMachines" \

--output json

```

### Step 4: Create an Azure Key Vault and Store the Password

Now, let's set up Azure Key Vault to store the administrative password securely.

```bash

# Create a Key Vault

az keyvault create \

--name <vault-name> \

--resource-group <resource-group> \

--location <location>

# Set a secret in the Key Vault

az keyvault secret set \

--vault-name <vault-name> \

--name "VmAdminPassword" \

--value "<your-password>"

```

### Step 5: Update ARM Template to Use the Secret from Key Vault

You will need to manually edit the exported ARM template to reference the password stored in Azure Key Vault. In the parameters section of the ARM template, you can add a reference to the Key Vault like this:

```json

"adminPassword": {

"reference": {

"keyVault": {

"id": "/subscriptions/<subscription-id>/resourceGroups/<resource-group>/providers/Microsoft.KeyVault/vaults/<vault-name>"

},

"secretName": "VmAdminPassword"

}

}

```

Replace `<subscription-id>`, `<resource-group>`, `<vault-name>`, and `<secretName>` with your actual values.

### Step 6: Deploy the Updated ARM Template

Finally, you can deploy the updated template using Azure CLI:

```bash

az deployment group create \

--resource-group <resource-group> \

--template-file <path-to-your-template-file> \

--parameters @<path-to-your-parameters-file>

```

Replace `<path-to-your-template-file>` and `<path-to-your-parameters-file>` with the paths to your ARM template and its parameter file, respectively.

### Recap

This step-by-step process involves:

1. Logging into Azure and setting up a VM.

2. Exporting the VM's configuration as an ARM template.

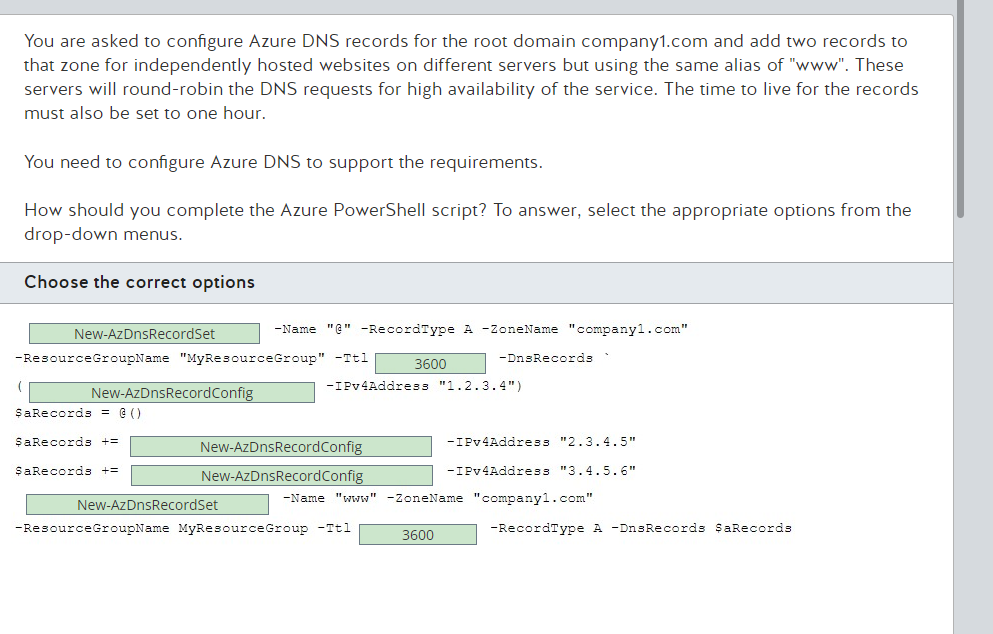
3. Setting up Azure Key Vault and storing a secret password.

4. Updating the ARM template to retrieve the password from Azure Key Vault.

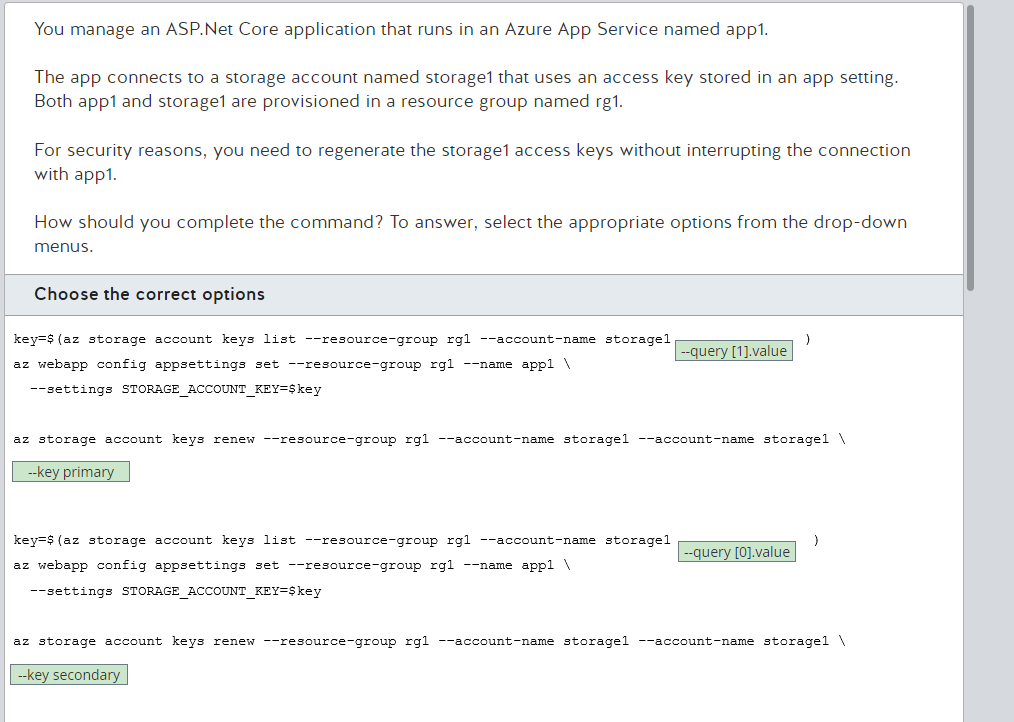
5. Deploying new VMs using the updated template.

This approach ensures your administrative passwords are managed securely, avoiding the risks associated with storing sensitive information in plain text within your deployment scripts or templates.

**Problem 25 \*\*\*: PowerShell & DNS**



**Problem 26 \*\*\*: AZURE CLI & Storage Account Keys Renewal**



**Problem 27 \*\*\*: Azure Resource Group & ARM**

You have an Azure Resource Manager (ARM) template for creating a Windows virtual machine. You got this template from an existing resource group with a single virtual machine, using the automation script option.

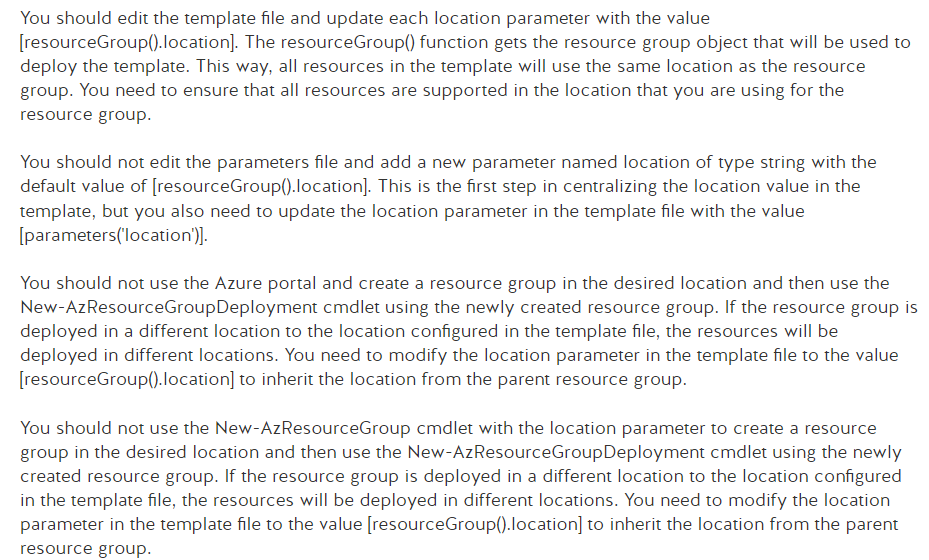
You want to reuse this template for other deployments. You need all the resources in the resource group to be in the same location.

What should you do?

Choose the correct answer:

* Edit the template file and update each location parameter with the value [resourceGroup().location].
* Use the Azure portal and create a resource group in the desired location. Then use the New-AzResourceGroupDeployment cmdlet using the newly created resource group.
* Use the New-AzResourceGroup cmdlet with the location parameter to create a resource group in the desired location. Then use the New-AzResourceGroupDeployment cmdlet using the newly created resource group. (selected answer)
* Edit the parameters file and add a new parameter named location of type string with the default value of [resourceGroup().location]

**Answer:**



**Problem 28 \*\*\*: Azure Backup Agent on VM**

Your company has an Azure subscription. You create an Azure Recovery Services vault named RSV1. You have a virtual machine (VM) named VM1 that is deployed in the East US region. You create a backup policy to back up VM1 to RSV1 on a recurring schedule. You are preparing to run your first backup and find the Backup Pre-Check status displays a Warning status.

You need to determine the possible cause of this status.

Which condition would result in a Warning status?

**Answer:**

One possible reason for a Warning status during the Backup Pre-check is that the most recent VM agent has not been installed on VM1. A Warning status indicates that the backup process might fail. The report status provides recommended steps to ensure successful backups.

A Critical status would be reported if VM1 was unable to communicate with the Azure Backup service. A Critical status indicates that the current VM configuration will result in a backup failure.

A situation in which VM1 has a non-premium storage account would not result in a Warning status. This is a supported configuration.

Having VM1 as an unmanaged Azure VM encrypted with BEK would not result in a Warning status. Backups of managed and unmanaged VMs encrypted with BEK are supported by Azure Backup.

**Problem 29 \*\*\*: Microsoft Entra ID and Licences**

Your company has a Microsoft Entra ID Governance subscription.

You are assigning licenses to individual users using group-based licensing in Microsoft Entra ID. You are using Microsoft Graph PowerShell cmdlets.

Your PowerShell script fails with a MutuallyExclusiveViolation error.

You need to troubleshoot this issue.

What is the underlying cause for this issue?

**Answer:**

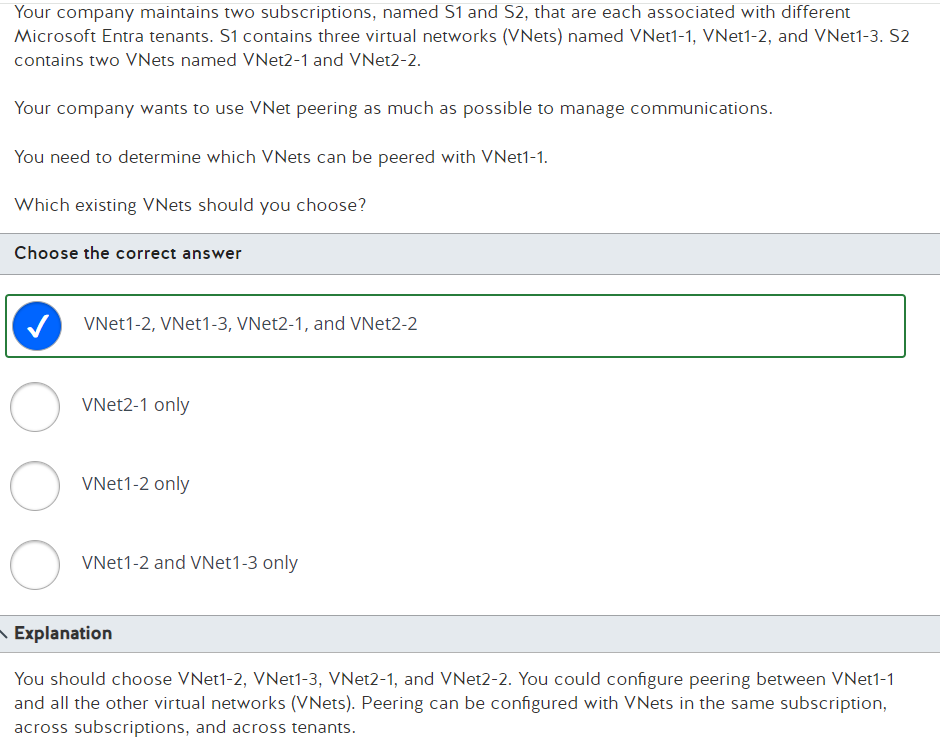
The underlying cause for the MutuallyExclusiveViolation error is that one of the products that is specified in the group contains a service plan that conflicts with another service plan that is already assigned to the user via a different product.

The underlying cause for the MutuallyExclusiveViolation error is not that there are not enough available licenses for one of the products specified in the group. If this was the underlying cause, the PowerShell script would fail with a CountViolation error.

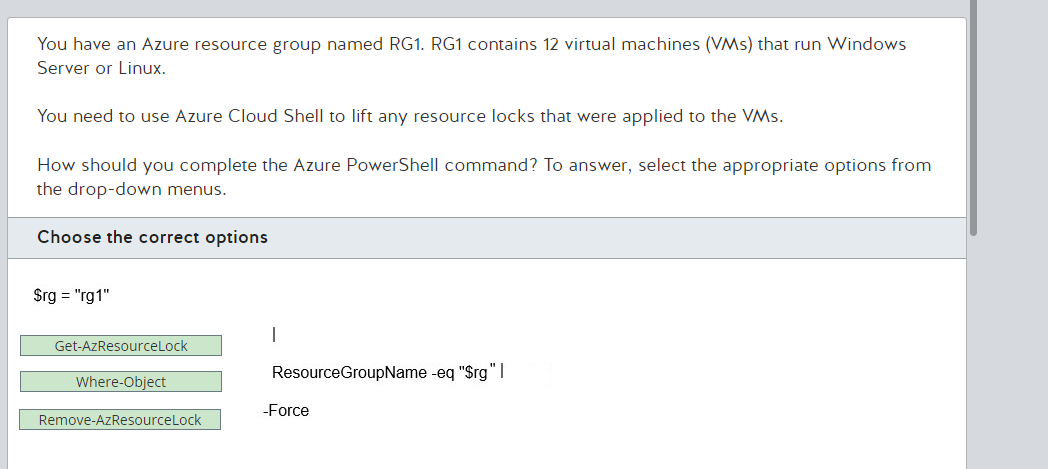
The underlying cause for the MutuallyExclusiveViolation error is not that one of the products that is specified in the group contains a service plan that must be enabled for another service plan, in another product, to function. If this was the underlying cause, the PowerShell script would fail with a DependencyViolation error.

The underlying cause for the MutuallyExclusiveViolation error is not that one of the products that is specified in the group is not available in all locations because of local laws and regulations. If this was the underlying cause, the PowerShell script would fail with a ProhibitedInUsageLocationViolation error.

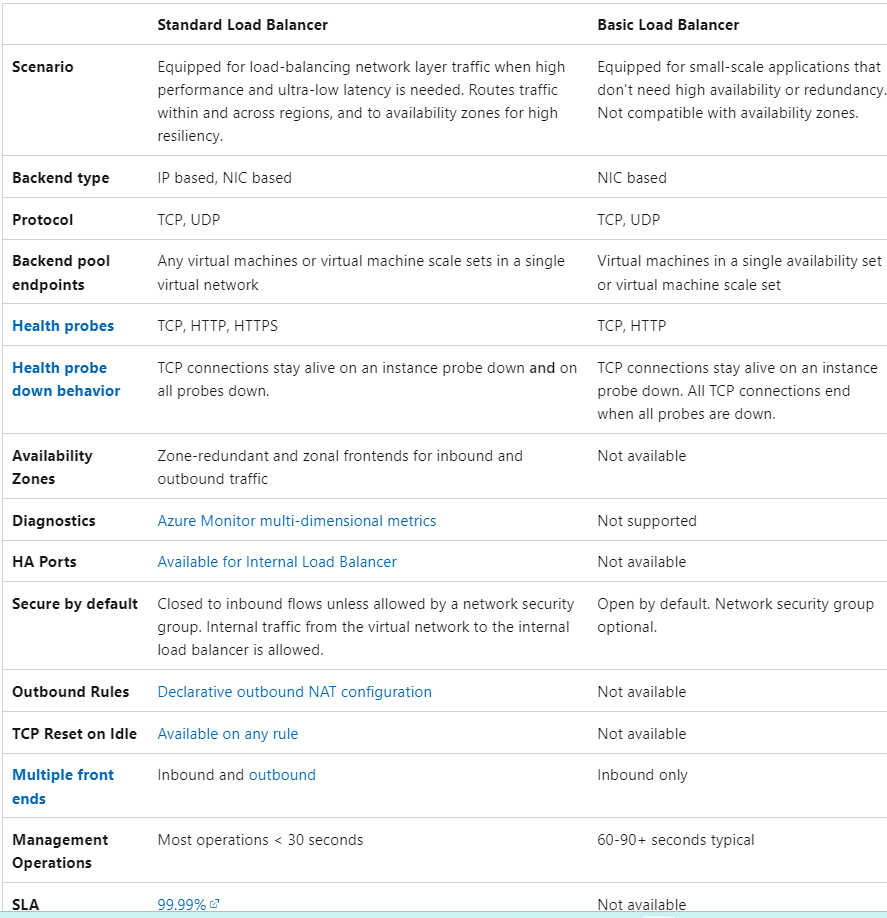
**Problem 30\*\*\*: Vnet Peering over multiple subscription/Tenants**

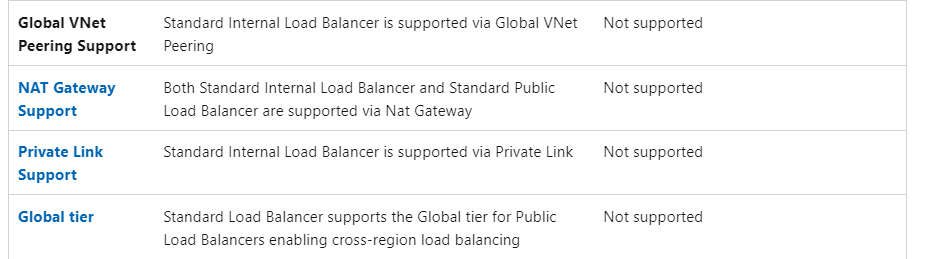


**Problem 31\*\*\*: Resource Lock**



**Problem 32\*\*\*: Load Balancer**





**Problem 33\*\*\*: SMB**

You work for a company that hosts resources in Azure, including Azure file shares. The support team asked you to create a new virtual network (VNet) and Azure file share, which is to be connected to a new Windows Server 2019 virtual machine (VM) that you also need to provision. You successfully provision the VNet, the Windows Server and the Azure file share. However, when you try to add the Azure file share to the Windows server you get the following error message:

'System error 67 has occurred'

You need to find the cause of this error and fix it before you can add the Azure file share to the Windows Server.

What fix do you need to implement to resolve the error?

**Answer:**

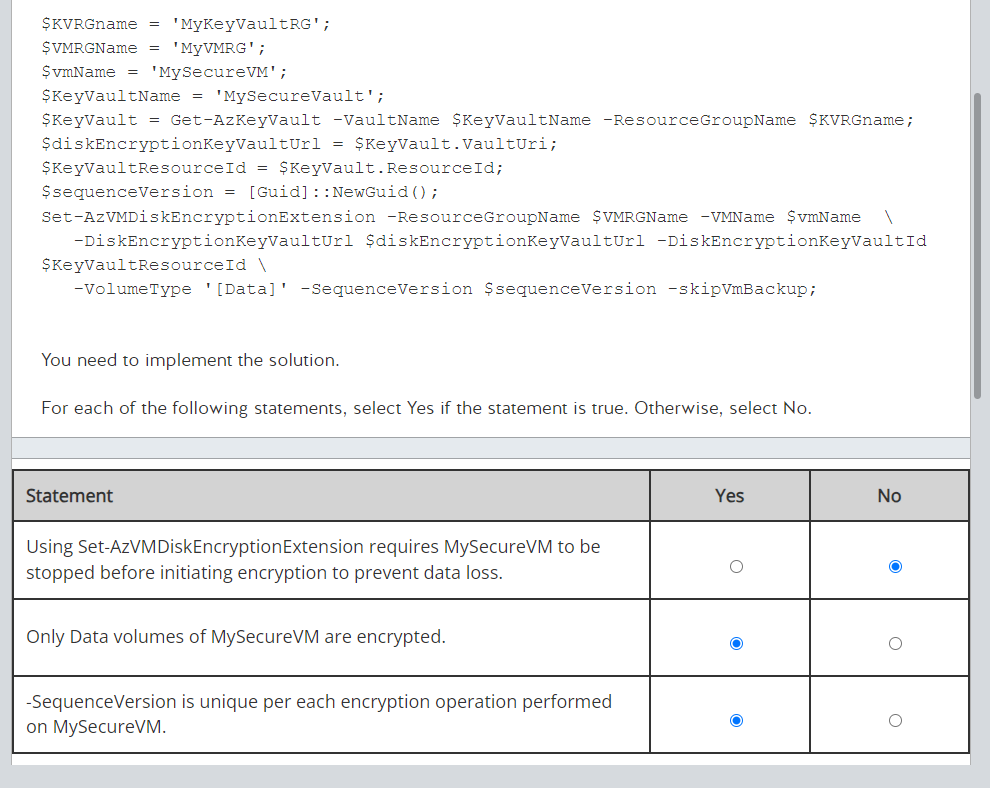
You should unblock port 445 on the Windows Server virtual machine (VM). Azure file shares use the Server Message Block (SMB) protocol on port 445. Error 67 is returned when this port is blocked. It will stop the Azure file share from mapping until it has been unblocked.

You should not unblock port 2049 on the Windows Server VM. Port 2049 is used for Network File Share (NFS) protocol and not for SMB. Error 67 only occurs if the SMB port is blocked.

You should not enable New Technology LAN Manager version 1 (NTLMv1) on the Windows Server VM. If NTLMv1 is not enabled on the Windows Server where you are trying to map an Azure file share, you would get the following error message: Error 53 has occurred. Azure File shares only support NTLMv2, therefore if NTLMv1 is enabled the share would not connect to the Windows Server.

You should not check that the permissions are configured correctly. If there was an issue with the permissions on the Azure file share, you would get a No Access error when attempting to map the share to a Windows server. You can add permissions via the Azure portal by going to the Access Control (IAM) blade in the storage account that stores the Azure file share.

**Problem 34\*\*: Disk Encryption**



**Answer:**

Based on your image, you're asking about certain conditions related to using the Set-AzVMDiskEncryptionExtension command for encrypting a virtual machine's disks in Azure. Let's go through the statements and validate them:

* Using Set-AzVMDiskEncryptionExtension requires MySecureVM to be stopped before initiating encryption to prevent data loss.
  + False: The Azure VM does not need to be stopped to enable encryption with the Set-AzVMDiskEncryptionExtension. This operation can be performed while the VM is running. Microsoft documentation confirms that Azure Disk Encryption can be enabled on running VMs without needing to stop them, thereby preventing any operational downtime.
* Only Data volumes of MySecureVM are encrypted.
  + False: This statement depends on how you configure the disk encryption. The Set-AzVMDiskEncryptionExtension can be used to encrypt both OS and data disks. In the PowerShell script provided, the -VolumeType 'Data' parameter specifies that only data volumes are being encrypted in this specific case, but it's not a limitation of the command itself; OS disks can also be encrypted if specified. => **It is TRUE BECAUSE IT IS SPECIFIED IN THE IMAGE**
* SequenceVersion is unique per each encryption operation performed on MySecureVM.
  + True: The SequenceVersion parameter, which appears to accept a new GUID with each operation ([Guid]::NewGuid()), ensures that each encryption operation is treated uniquely. This is useful for managing and distinguishing between different encryption operations, especially in automated or scripted deployments.

In summary:

* Stopping the VM is not required for encryption.
* The command can encrypt both OS and data disks, though only data disks are specified in your script.
* Each encryption operation is uniquely identified by a new SequenceVersion.

**What I should learn from Disk Encryption**

1. Understand Azure Disk Encryption (ADE) Basics
   * What it Does: Encrypts virtual machine disks using industry-standard encryption algorithms to help secure data.
   * Encryption Technologies: Uses BitLocker for Windows VMs and DM-Crypt for Linux VMs.
   * Supported VMs: Understand which types of VMs and OS disks are supported (e.g., Managed and unmanaged disks, premium and standard storage).
2. Key Management
   * Azure Key Vault: Disk encryption keys and secrets are stored in Azure Key Vault.
   * Key Vault Integration: How ADE integrates with Key Vault, including the setup of Key Vault and its linkage to the VM that will be encrypted.
   * Dual Protocols: Use of RSA and KEK (Key Encryption Key) for protecting the secret.
3. Configuration Steps
   * Enabling Encryption: Commands and steps needed to enable encryption on both Windows and Linux VMs using Azure CLI and PowerShell:
   * Azure CLI Example: az vm encryption enable --resource-group [ResourceGroupName] --name [VMName] --disk-encryption-keyvault [KeyVaultName] --volume-type [all | os | data]
   * PowerShell Example: Set-AzVMDiskEncryptionExtension -ResourceGroupName [ResourceGroupName] -VMName [VMName] -DiskEncryptionKeyVaultUrl [DiskEncryptionKeyVaultUrl] -DiskEncryptionKeyVaultId [KeyVaultResourceId] -VolumeType [ALL | OS | DATA]
   * Monitoring the Encryption Status: Using commands to check the status of disk encryption.
   * Encryption for Data Disks: Specifics on encrypting data disks versus OS disks.
4. Operational Management
   * Role-Based Access Control (RBAC): Ensuring that appropriate permissions are in place for managing disk encryption through Key Vault.
   * Handling Encryption During VM Operations: Implications of disk encryption on VM operations like start, stop, and deprovision.
   * Troubleshooting and Common Errors: Such as the ones involving VM agents or configuration issues.
5. Compliance and Security Features
   * Compliance: How Azure Disk Encryption helps meet compliance requirements.
   * Security Features: Additional security features like the integration of Azure Security Center for monitoring the encryption status.
6. Best Practices and Limitations
   * VM Availability: Best practices regarding VM availability during the encryption process.
   * Backup and Disaster Recovery: How encryption affects backup and disaster recovery processes.
7. Command Memorization
   * Commands and Parameters: Memorize the key PowerShell cmdlets and Azure CLI commands used for configuring and managing VM disk encryption, as this can often appear in tasks or simulations in exams.

By focusing on these areas, you'll develop a comprehensive understanding of Azure Disk Encryption that will not only aid in passing the AZ-104 but also provide practical knowledge for managing Azure environments securely.

**Problem 35\*\*: Azure Alerts**

Your company has a line-of-business (LOB) application that uses Azure SQL Database to store transactional information. The LOB application also uses Windows and Linux virtual machines (VMs) for the business and presentation application layers.

Some users are reporting errors in the application.

You need to be alerted every time an exception arises in any part of the application. Your solution should require minimal administrative effort.

Which two actions should you perform? Each correct answer presents part of the solution.

Choose the correct answers:

* Create an alert using a search query that looks for exceptions in application layer servers.
* Create an alert using a search query that looks for exceptions in business and presentation layer virtual machines (VMs).
* Create an alert using a search query that looks for exceptions in business layer servers.
* Create an alert using a search query that looks for exceptions in Windows servers.
* Create an alert using a search query that looks for exceptions in Linux servers.

**Answer:**

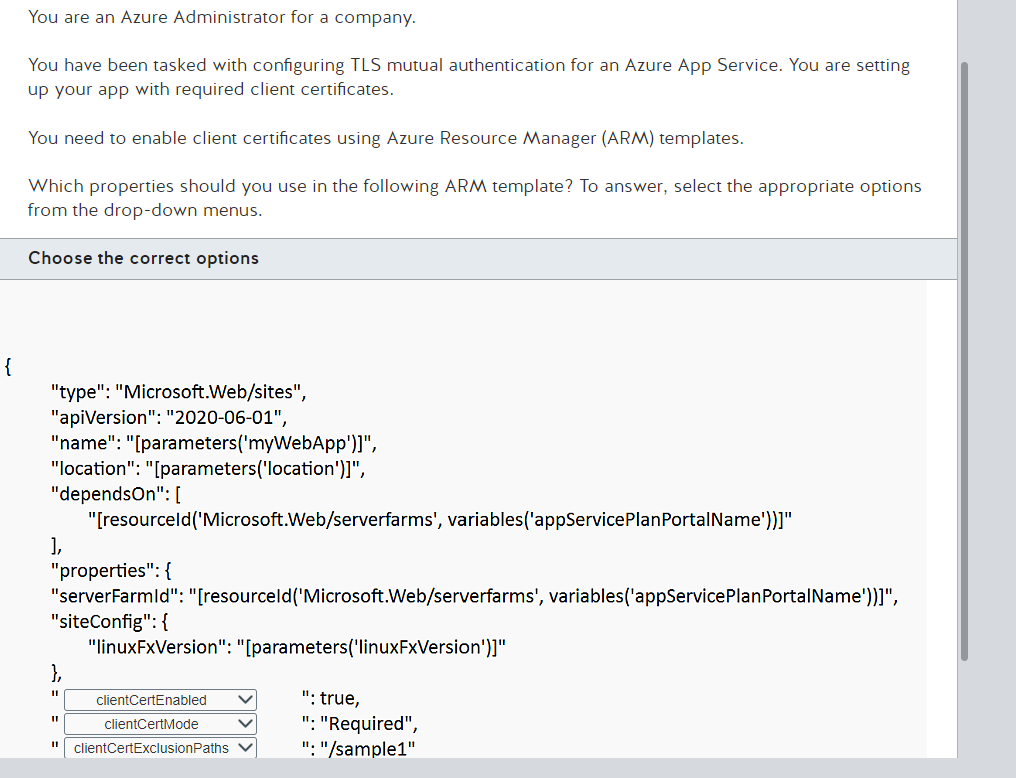
You should create an alert using a search query that looks for exceptions in Windows servers. You need to use Log Analytics or Application Insight resource types and Log signal types. Then, you can write a search query that gets all messages from Windows Events that contains the word "Exception".

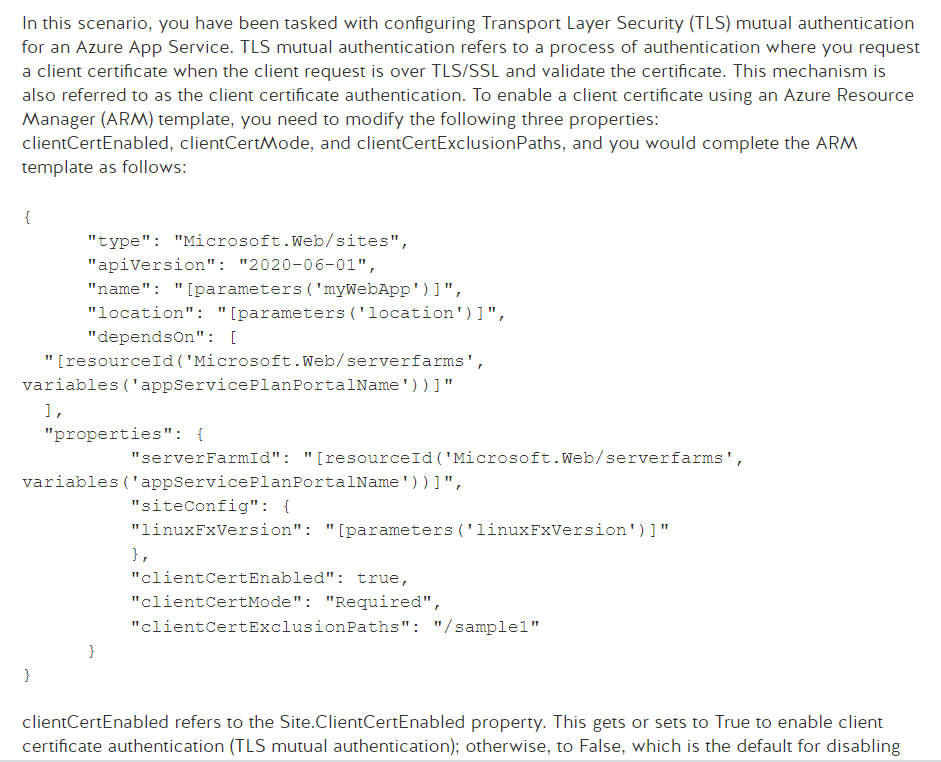
You should also create an alert using a search query that looks for exceptions in Linux servers. You use the same configuration as for Windows Events, but you will use Syslog messages.

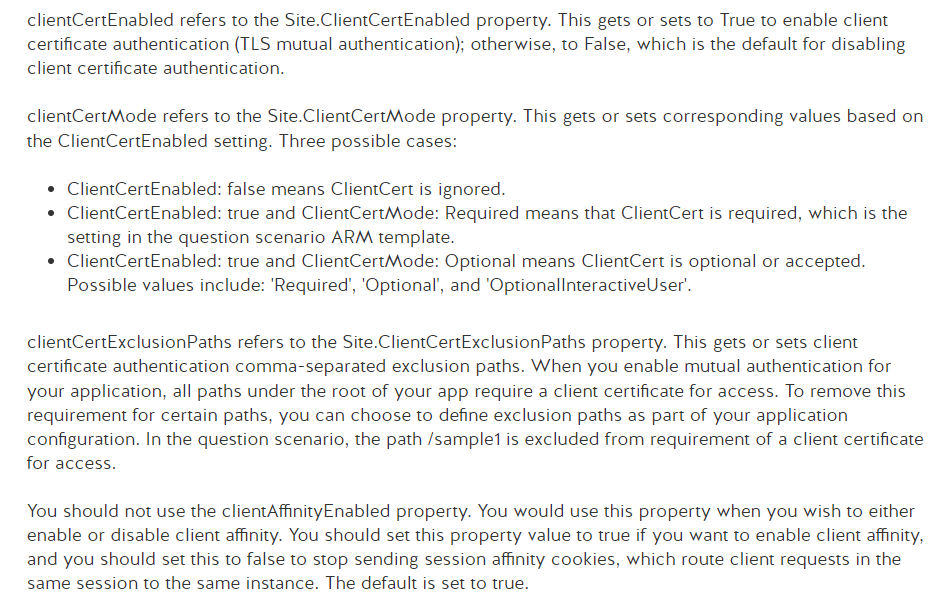
You should not create an alert using a search query that looks for exceptions in business and presentation layer virtual machines (VMs). When creating an alert, you can only select one target type. This means that you can only get information from Windows Events or Syslog, so you will not be able to query both data sources at the same time.

You should not create an alert using a search query that looks for exceptions in application layer servers or in the business layer. You need to query for specific log data sources. Application and business layers are concepts of a design pattern for applications that can be a compound of Windows, Linux VMs, or other Azure services.

**Problem 36\*\*: Azure Webapp Client Certificates**







**Problem 37\*\*: Azure Instance Metdata Service**

You are tasked with retrieving Load Balancer information using Azure Instance Metadata Service (IMDS).

You see the following error message:

Error code: 404; No load balancer metadata is found.

You need to troubleshoot this issue.

Which two things can you infer from the error message? Each correct answer presents a complete solution.

* There is a rate limit.
* The load balancer has the Basic instead of the Standard SKU.
* The path is misconfigured.
* The virtual machine is not associated with a load balancer.

**Answer:**

The Azure Instance Metadata Service (IMDS) provides information about currently running virtual machine instances. You can use the SKU, storage, network configurations, and upcoming maintenance events-related information effectively for managing and configuring the virtual machines. In this scenario, since the retrieved data from IMDS displays the No load balancer metadata is found error message, this could be owing to either of the following two reasons:

* The virtual machine is not associated with a load balancer.
* The load balancer has the Basic instead of Standard SKU.
* The error code does not indicate a misconfiguration of the path. In such a case, you would see Error 404, but with a different error message displayed:

404; API is not found: Path = "<UrlPath>", Method = "<Method>".

The error code does not indicate a rate limit. In this case, you would see Error 429 displayed with the "Too many requests" message

**What I should learn**

l'Azure Instance Metadata Service (IMDS) permet de savoir si une machine virtuelle (VM) est configurée pour utiliser un load balancer. IMDS fournit des métadonnées sur une instance de VM en cours d'exécution, y compris des informations sur la configuration réseau de l'instance, ce qui inclut les détails sur le load balancer si celui-ci est configuré.

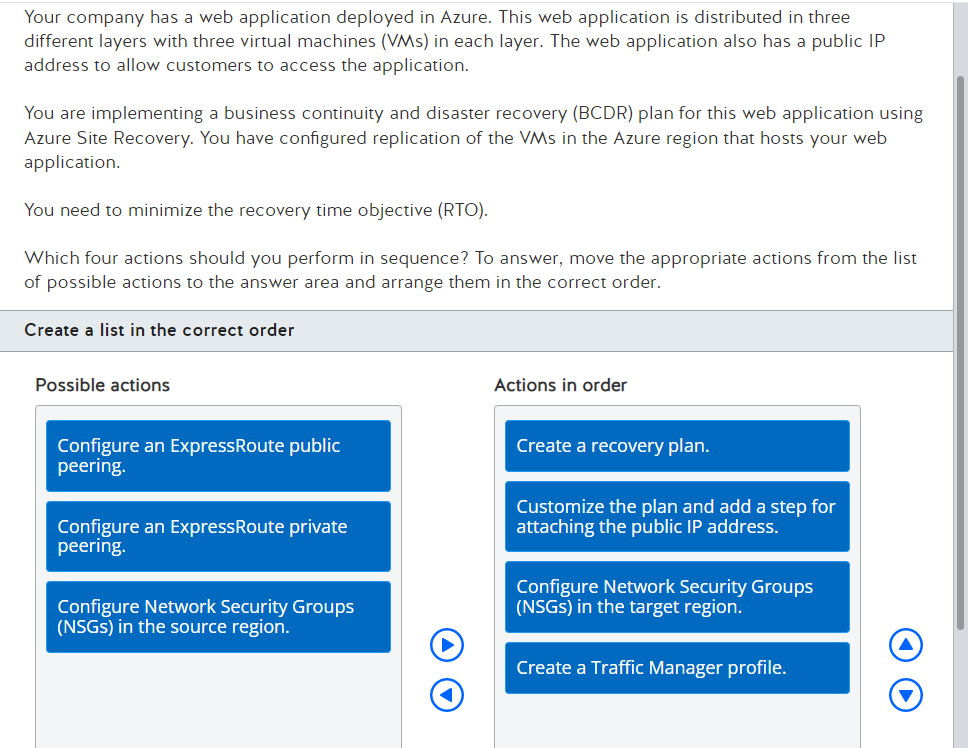
Pour obtenir ces informations, vous pouvez interroger IMDS depuis la VM elle-même en utilisant une API spécifique. Voici un exemple de requête que vous pourriez utiliser pour obtenir des informations sur le load balancer configuré pour une VM Azure :

curl -H Metadata:true "http://169.254.169.254/metadata/instance/network?api-version=2021-02-01"

Il est important de spécifier Metadata:true dans l'en-tête de la requête pour s'assurer que la requête est traitée par IMDS. L'URL doit inclure la dernière version de l'API (api-version=2021-02-01 dans cet exemple), et vous pouvez ajuster cette valeur selon la dernière version disponible.

La réponse à cette requête inclura des détails sur la configuration réseau de la VM, y compris des informations sur le load balancer si applicable. Notez que ces détails seront disponibles uniquement si la VM est effectivement configurée pour utiliser un Azure Load Balancer, que ce soit un SKU Basic ou Standard. Si la VM n'est pas associée à un load balancer ou si le load balancer est de type Basic (qui ne supporte pas certaines intégrations avec IMDS), les informations pertinentes pourraient ne pas apparaître ou l'erreur mentionnée précédemment (404; No load balancer metadata is found) peut survenir.

**Problem 37\*\*: Azure Recovery & BCDR**



Votre entreprise dispose d'une application web déployée sur Azure, répartie en trois couches avec trois machines virtuelles (VM) pour chaque couche. Cette application dispose également d'une adresse IP publique pour permettre aux clients d'accéder à l'application.

Vous travaillez à la mise en place d'un plan de continuité et de reprise d'activité (BCDR) pour cette application en utilisant Azure Site Recovery. Vous avez configuré la réplication des VM dans la région Azure qui héberge votre application web.

L'objectif est de minimiser l'objectif de temps de récupération (RTO). Voici les quatre actions à réaliser en séquence pour atteindre cet objectif :

1. Créer un plan de récupération (Create a recovery plan): Cela permet de structurer et d'automatiser le processus de reprise après sinistre pour garantir une récupération rapide et ordonnée des applications et des données dans l'ordre requis pour respecter les interdépendances entre les couches de l'application.
2. Personnaliser le plan et ajouter une étape pour attacher l'adresse IP publique (Customize the plan and add a step for attaching the public IP address): Cette étape garantira que l'adresse IP publique est correctement reconfigurée ou réassociée à l'application dans le cadre du processus de récupération pour permettre l'accès des clients sans délai.
3. Configurer les groupes de sécurité réseau (NSG) dans la région cible (Configure Network Security Groups (NSGs) in the target region): Les NSG doivent être configurés pour garantir que la sécurité du réseau est maintenue après la récupération, protégeant ainsi l'application contre les accès non autorisés tout en permettant le trafic légitime.
4. Créer un profil de Traffic Manager (Create a Traffic Manager profile): Cela permet de gérer le trafic entrant vers l'application, en assurant une bascule transparente et la distribution du trafic entre les instances actives de l'application, notamment entre les régions en cas de reprise après sinistre.

Ces étapes, lorsqu'elles sont exécutées correctement et dans l'ordre, vous aideront à minimiser le RTO en assurant que tous les aspects de la reprise après sinistre - de la récupération des données à la gestion du trafic et de la sécurité - sont bien coordonnés.

You should perform the following steps in sequence:

* Configure Network Security Groups (NSGs) in the target region.
* Create a recovery plan.
* Customize the plan and add a step for attaching the public IP address.
* Create a Traffic Manager profile.

First, you should configure NSGs in the target region. When you set up Azure Site Recovery, the service can automatically configure virtual networks and subnets that you have configured in your source site. Unfortunately, the Azure Site Recovery service does not automatically replicate any NSGs that you have set in your source site. You need to re-create in the target region any NSG that you configured in the source region. Once you have configured replication of the VMs in the source region and you have performed the first replication, you can create the target NSG.

Next, you should create a recovery plan. When you design a business continuity and disaster recovery plan, you write the tasks that are required to bring your web application online in the event of a critical failure in the Azure region where you are hosting your application. An Azure Site Recovery plan is a list of tasks that allows you to automate the execution of all jobs in your plan. Before you can create a recovery plan, you need to configure the replication of your VMs.

Next, you should customize the plan and add a step for attaching the public IP address. By default, Azure Site Recovery does not migrate public IP addresses, and these types of addresses cannot be retained during failover. If you need your target site to have a public IP address, you need to add the IP address to the target region. You can do this manually or by adding a step to the Azure Automation runbook to add this public IP address to the target region.

Finally, you should create a Traffic Manager profile. Traffic Manager allows you to monitor the status of the source and target sites. In the case of a failure of the source site, Traffic Manager automatically redirects the traffic to the target site while Azure Site Recovery handles the failover to the target region, making your web application available again and minimizing downtime.

You should not configure NSGs in the source region. Because you already configured the replication of your VMs from the source site to the destination region, the NSG in the source region is already correctly set to allow the replication traffic.

You should not configure an ExpressRoute public or private peering. ExpressRoute allows you to connect your on-premises infrastructure with Azure. In this scenario, your application is hosted in Azure, so there is no need to configure an ExpressRoute peering.

**Problem 38\*\*: Azure Network Watcher**

You deploy virtual machines (VMs) in your Azure subscription. The VMs are connected to different virtual networks (VNets). You configure custom network security groups (filtering rules) on the VNets. Communication between the VMs is working correctly.

You deploy an additional VM named VM01 in a new VNet named VNet01. This new VM is experiencing some connectivity issues. You decide to use Network Watcher to troubleshoot the connectivity issues.

You need to determine which filtering rule is causing the issue.

Which two cmdlets should you use? Each correct answer presents part of the solution.

Choose the correct answers

* Get-AzNetworkSecurityGroup
* Test-AzNetworkWatcherConnectivity
* Test-AzNetworkWatcherNextHop (selected)
* Test-AzNetworkWatcherIPFlow (selected)
* Get-AzEffectiveNetworkSecurityGroup

**Answer:**

You should use the Test-AzNetworkWatcherIPFlow cmdlet. This cmdlet tests the outbound connection from the source virtual machine (VM) and source port to the destination IP address and port using the Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) protocols. If any of the configured filtering rules block traffic between the endpoints that you configured in the invocation of the cmdlet, it returns the name of the network security group (NSG) that contains the filtering rule that denied the communication.

You should also use the Get-AzEffectiveNetworkSecurityGroup cmdlet. This cmdlet calculates the filters that affect a VM in a specific resource group and helps you to determine why the rules shown by the Test-AzNetworkWatcherIPFlow cmdlet allow or prevent communication.

You should not use the Get-AzNetworkSecurityGroup cmdlet. This cmdlet shows the details about a specific NSG but it does not provide details of the rules that affect the VM.

You should not use the Test-AzNetworkWatcherConnectivity cmdlet. This cmdlet tests the outbound connectivity between a VM and a specific destination. If traffic is denied, the cmdlet does not provide the name of the NSG that denied communication.

You should not use the Test-AzNetworkWatcherNextHop cmdlet. This cmdlet is useful for troubleshooting routing issues. In this scenario, you need to troubleshoot filtering issues.

**Problem 39\*\*: Azure DNS service**

You configure the companycs.com zone in Azure DNS. You have an A record set named app that points to an App Service that hosts a web application.

You need to make this application available by using the webapp.companycs.com domain name. This new domain name needs to point to the public IP address of the App Service.

You need to ensure that the Domain Name System (DNS) record for this new domain name is updated or deleted automatically in case the app.companycs.com DNS record is modified or deleted.

Which type of record set should you create?

Choose the correct answer

- A CNAME record set

- A CNAME alias record set

- An A record set

- An A alias record set

**Answer:**

You should create an address mapping (A) alias record set. An A alias record set is a special type of record set that allows you to create an alternative name for a record set in your domain zone or for resources in your subscription. This is different from a canonical name (CNAME) record type because the alias record set will be updated or deleted in case the target record set is modified or deleted. You can only create an A alias record set that points to A, AAAA (IPv6 address), or CNAME record types in an Azure DNS zone.

You should not use a CNAME alias record set. The custom domain name for your web application is represented by an A record set. A CNAME alias record set can only point to another CNAME record set. Moreover, the value returned by a CNAME alias record set is a domain name. You are required to create a Domain Name System (DNS) record that returns an IPv4 address. This means that you need an A alias record set.

You should not use an A record set. This record set type will not be automatically updated or deleted if the app.companycs.com record is modified or deleted.

You should not use a CNAME record set. This record set type will not be automatically updated or modified if the app.companycs.com record is modified or deleted. You are also required to create a DNS record that returns an IPv4 address. This means that you need an A alias record set.

**Problem 40\*\*\*: Azure Storage Account and RBAC**

You work for a company that has an existing Azure subscription and a separate storage account for each department. A member of the Marketing team has downloaded Azure Storage Explorer as they want to upload some old files into Azure Blob Storage. When they attempt to log in to Storage Explorer, they get the following error message:

"'Read: List/Get Storage Account(s)' permissions issue"

The user has Contributor access to the storage account. You need to grant them the necessary access to the subscription in order to ensure that this error does not occur again. You must apply the principle of least privilege.

What role should you assign the user at the subscription level?

**Answer:**

You should grant the user the Reader role. This particular error occurs when the user does not have the relevant permissions at the subscription level. Once they have read access, they will be able to list all storage accounts, including the storage account which they have contributor access to. This will allow them to move the necessary files, and it is also the role with the least amount of privilege.

You should not grant the user the Owner role. This role would give the user full control over all resources within the subscription, including virtual machines (VMs), networks, and storage. Although this role would allow the user to list all the storage accounts and then move their files, it does not follow least privilege best practices.

You should not grant the user the Contributor role. This role would give the user the right to modify all resources within the subscription, including VMs, networks and storage. Although it would grant the user less access than the Owner role, it still does not meet the least privilege access requirement.

You should not grant the user the Storage Blob Data Reader role. This role would cause the same error as before as it grants read access to Azure Storage containers and blobs. The user already has the relevant access to the Marketing storage account, but to resolve this error they need access at the subscription level.

**Problem 41\*\*\*: Azure Region & Backup in Paired Region**

Your company has an Azure subscription. You create an Azure Recovery Services vault named Rsv1 to support Azure Backup. You back up the virtual machine (VM) VM1 from the East US region.

You need to restore VM1 to a different region.

To which region should you restore VM1?

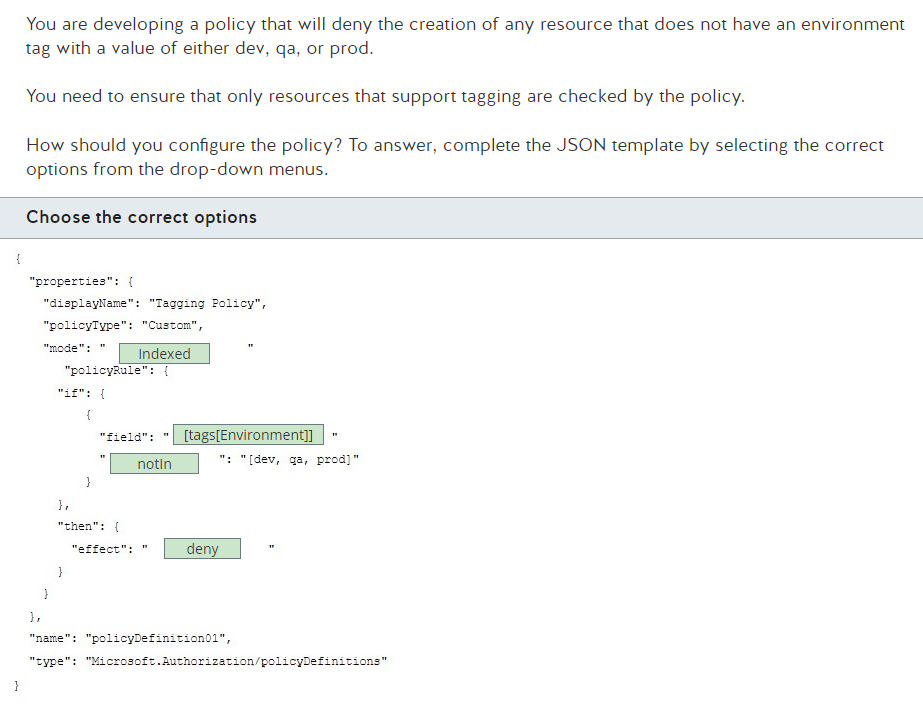
**Answer:**

You should restore VM1 to the East US paired secondary region, i.e., the West US region. Restore is supported to East US and West US only. One of the restore options is Cross Region Restore (CRR), which enables you to restore a VM to an Azure paired secondary region. The secondary region for East US is West US.

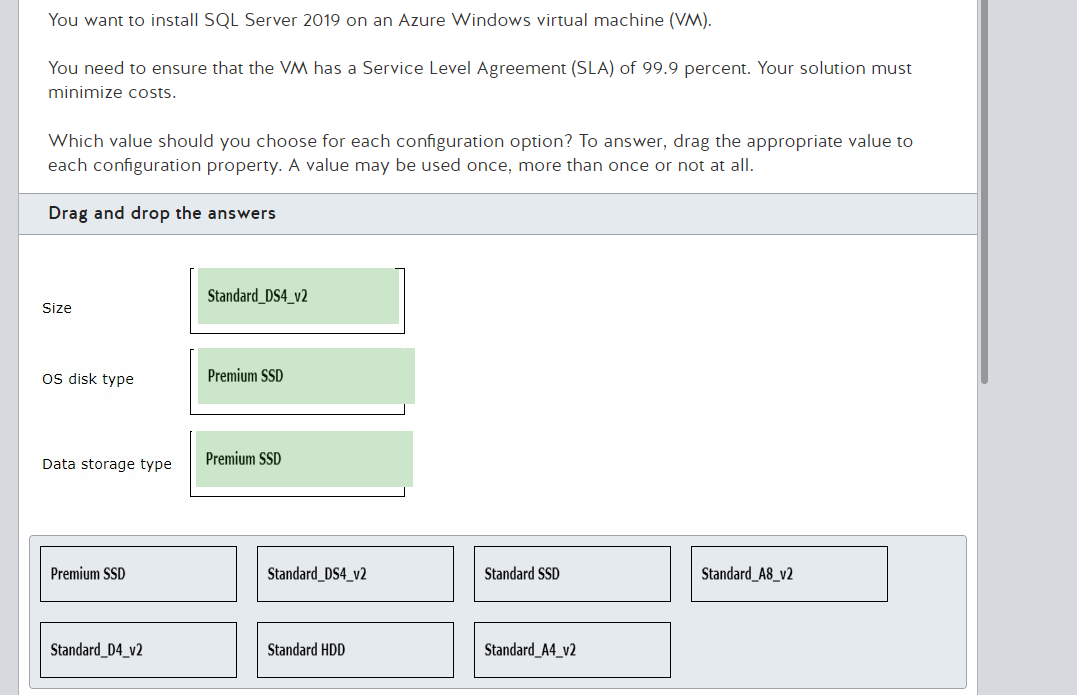
The Central US region is the secondary region for the East US 2 region.

CRR is supported across Azure regional pairs only. You cannot use any region outside the US as the restored region.

**Problem 42\*\*: Azure Policy**



**Problem 43\*\*: Azure Disk & SLA**

****

Available Options:

* Premium SSD
* Standard\_DS4\_v2
* Standard SSD
* Standard\_A8\_v2
* Standard\_D4\_v2
* Standard HDD
* Standard\_A4\_v2

Selection to Minimize Costs and Ensure an SLA of 99.9%:

1. OS Disk Type:
   * Premium SSD: This is recommended to ensure high performance and low latency, which is crucial for database applications like SQL Server.
2. Data Storage Type:
   * Premium SSD: Using Premium SSDs for data ensures optimal performance and high availability.
3. VM Size:
   * Standard\_DS4\_v2: This VM size offers a good balance between performance and cost and is suitable for medium to high SQL Server workloads.

Recommended Configuration:

* OS Disk Type: Premium SSD
* Data Storage Type: Premium SSD
* VM Size: Standard\_DS4\_v2

These choices ensure high availability (99.9% SLA) while keeping costs reasonable

**Problem 44\*\*: VMSS et Rolling Update**

You deploy a virtual machine scale set (VMSS) to support a critical application. The upgrade policy for the VMSS is set to Rolling.

You need to apply a change in the scale set OS and Data disk Profile for the VMSS to the existing VM images.

Which PowerShell cmdlet should you use?

Choose the correct answer

* Set-AzVmssVM
* Update-AzVmss
* Start-AzVmssRollingOSUpgrade
* Update-AzVmssInstance

**Answer:**

You should use the Set-AzVmssVM cmdlet. Changes to the scale set OS and Data disk Profile do not follow the scale set upgrade policy and must be applied manually to each existing instance using this cmdlet.

You should not use the Update-AzVmss cmdlet. This cmdlet is used to update the state of a virtual machine scale set (VMSS) and does not apply the specified changes to individual instances.

You should not use the Update-AzVmssInstance cmdlet. This cmdlet is used to update an instance when the VMSS upgrade policy is set to Manual. It does not include changes to the scale set OS and Data disk Profile.

You should not use the Start-AzVmssRollingOSUpgrade cmdlet. This is used to upgrade existing VM instances to the latest available Platform Image OS version.

**What I should learn**

The Azure PowerShell commands related to Virtual Machine Scale Sets (VMSS) you provided have specific roles. Here's what each command does:

1. Set-AzVmssVM

* Role: This command is used to configure or update the settings of a specific virtual machine within a scale set. For example, you can use it to modify the tags or properties of an individual virtual machine within a scale set.
* Usage:

Set-AzVmssVM -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet" -InstanceId "1" -Tag @{"key"="value"}

1. Update-AzVmss

* Role: This command updates the configuration of an entire Virtual Machine Scale Set. For instance, you can use it to apply a new image or change the configuration of the entire scale set.
* Usage:

Update-AzVmss -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet"

1. Start-AzVmssRollingOSUpgrade

* Role: This command initiates a rolling upgrade of the operating system for instances in a scale set. This allows you to update the operating system of instances within a scale set without completely disrupting the services.
* Usage:

Start-AzVmssRollingOSUpgrade -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet"

1. Update-AzVmssInstance

* Role: This command is used to update a specific instance within a Virtual Machine Scale Set. This could include configuration updates or specific instance modifications.
* Usage:

Update-AzVmssInstance -ResourceGroupName "myResourceGroup" -VMScaleSetName "myScaleSet" -InstanceId "1"

Summary of Roles:

* Set-AzVmssVM: Updates the settings of a specific VM within a scale set.
* Update-AzVmss: Updates the configuration of an entire Virtual Machine Scale Set.
* Start-AzVmssRollingOSUpgrade: Initiates a rolling OS upgrade for instances in a scale set.
* Update-AzVmssInstance: Updates a specific instance within a Virtual Machine Scale Set.

You have a resource group named APP-RG that consists of several resources.

**Problem 45\*\*: Ressource Deployment : Incremental/Complete mode**

You are asked to add a storage account to the resource group. You decide to deploy the new storage account by using an Azure Resource Manager (ARM) template and the New-AzResourceGroupDeployment cmdlet. This template does not contain any linked or nested templates.

After the deployment finishes successfully, you realize that all the resources in the resource group have been deleted except for the newly created storage account.

Why did this happen?

Choose the correct answer:

* The template contains the mode parameter with the value Complete.
* You did not use the -mode parameter with the New-AzResourceGroupDeployment cmdlet.
* The template contains the mode parameter with the value Incremental.
* You used the -mode complete parameter with the New-AzResourceGroupDeployment cmdlet.

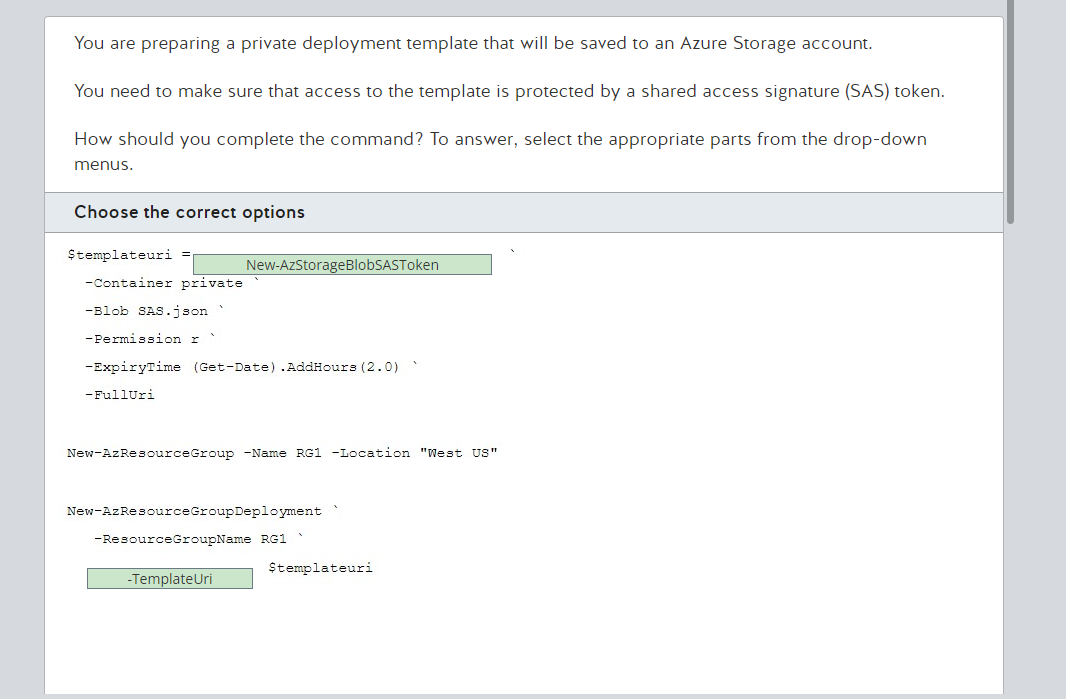
**Answer:**

Resources in the resource group were deleted except for the newly-created storage account because you used the -mode complete parameter with the New-AzResourceGroupDeployment cmdlet. The New-AzResourceGroupDeployment cmdlet has two deployment modes: incremental and complete. When you use the complete mode, all resources in the resource group that are not included in the template are deleted.

Using the -mode parameter with the New-AzResourceGroupDeployment cmdlet was not the cause. When you do not use the -mode parameter, you are using the default incremental deployment mode. In this mode, any resource that is not present in the template is maintained in the resource group. If a resource in the resource group is present in the template and any of these parameters in the template differ from the values in the resource group, those values are updated in the resource present in Azure. You should use this mode when deploying the template.

**The cause was not the template containing the mode parameter with the value Complete or Incremental. The template that you are using does not contain any linked or nested templates, so the mode parameter should not be present in the template with either value. This parameter is part of the deployment resource type and is typically used with nested or linked templates. Complete and incremental deployment modes behave the same way as in the New-AzResourceGroupDeployment cmdlet.**

**Problem 46\*\*: Azure Storage & SAS & Powershell**



**Problem 47\*\*: Azure Administrator Roles**

You are the administrator for your Azure subscription. Your company hires a new cloud engineer.

The cloud engineer needs to be able to manage other engineers' access to Azure resources. You need to follow the principle of least privilege.

Which role should you assign to the new engineer?

Choose the correct answer

* Co-Administrator
* Contributor
* Owner
* User Access Administrator
* User Administrator

**Answer:**

You should grant the User Access Administrator role. Users with this role create and manage users and groups, manage support tickets, and monitor service health.

You should not grant the Owner role. This role allows members to manage user access to Azure resources but also grants full access to all resources. The Owner role actually is a combination of User Access Administrator role, as well as the Contributor role. Granting this role would not follow the principle of least privilege in this scenario.

You should not grant the Contributor role. This role allows members to create and manage all types of resources, but it does not allow them to manage other users' access to Azure resources in the subscription.

You should not grant the Co-Administrator role. This is a classic subscription role that is equivalent to the Owner Role-Based Access Control (RBAC) role. Granting this role would not follow the principle of least privilege in this scenario.

You should not grant the User Administrator role. This is a Microsoft Entra administrator role that does not control access to any Azure resources. This role grants permissions to manage users and groups in the Microsoft Entra tenant associated with the Azure subscription.

**Problem 48\*\*\*: AzCopy & Authorization schema**

An organization is in the process of migrating its resources into Azure from on-premises. You plan on using AzCopy to migrate resources into Azure, specifically Blob Storage. Company policy prohibits the use of Shared Access Signature (SAS) tokens, which is one of the authorization methods that supports Blob Storage.

You need to use another authorization option which will support migrating resources into Blob storage when using AzCopy.

Which authorization option should you use?

Choose the correct answer:

* Microsoft Entra ID
* Anonymous public read access
* Active Directory Domain Services
* Microsoft Entra Domain Services (DS)

**Answer:**

You should use Microsoft Entra ID authorization to migrate resources into Blob Storage when using AzCopy. This option supports Blob Storage, Queue, and table resources. Microsoft recommends using Microsoft Entra credentials for authorization as it gives optimal security and is much easier to use than other authorization methods.

You cannot use Anonymous public read access as this is not a supported authorization option for Blob storage when using AzCopy. Microsoft does not recommend using Anonymous public read access as it allows clients to read the data stored in Azure without any authorization, and it is therefore not secure.

You cannot use Microsoft Entra Domain Services (DS) as this is not a supported authorization option for Blob storage when using AzCopy. This authorization option supports Azure files which is identity-based authorization over Server Message Block (SMB). You can use Azure Role Based Access Control (RBAC) to implement fine-grained control over the resources stored in the storage account.

You cannot use Active Directory Domain Services as this is not a supported authorization option for Blob storage when using AzCopy. This option supports Azure Files, which is identity-based authorization over SMB. You can use RBAC to implement fine-grained control over the resources stored in the storage account. The main difference between this and Microsoft Entra DS is that Active Directory is hosted on-premises and Microsoft Entra DS is a service managed and hosted in Azure.

**Problem 49 \*\*\*: Azure Web App Slot & Powershell**

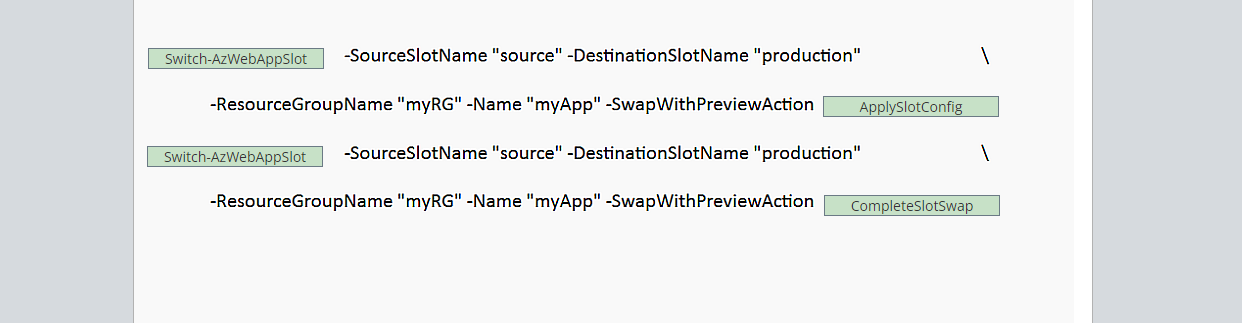
You are an Azure administrator at a company.

You are tasked with swapping deployment slots for your web app in Azure App Service.

You need to use PowerShell to swap a staging deployment with the production slot with preview.

Which cmdlets should you use? To answer, select the appropriate option from the drop-down menu.

**Answer:**



In this scenario, you are planning to do a swap of the Azure App Service from a staging deployment with the production slot with preview. This is also referred to as a multi-stage swap. When you perform a swap with preview, App Service executes the same swap operation but pauses after the first step. You can then verify the result on the staging slot before completing the swap. If you cancel the swap, App Service reapplies configuration elements to the source slot. You should complete the PowerShell cmdlets as follows:

**Switch-AzWebAppSlot -SourceSlotName "source" -DestinationSlotName "production" -ResourceGroupName "myRG" -Name "myApp" -SwapWithPreviewAction ApplySlotConfig**

**Switch-AzWebAppSlot -SourceSlotName "source" -DestinationSlotName "production" -ResourceGroupName "myRG" -Name "myApp" -SwapWithPreviewAction CompleteSlotSwap**

You should use the Switch-AzWebAppSlot PowerShell cmdlet. You should use this cmdlet to swap two slots within a web app. An important point to note is that you should use "production" as the slot name when the either the source or destination slot is intended to be the production slot.

You should use the ApplySlotConfig value for the -SwapWithPreviewAction parameter. The ApplySlotConfig value sets up the swap and initiates the process. An important note to is that, after the completion of the first stage, the process pauses for you to apply the "CompleteSlotSwap" stage. At this stage, you can verify the result on the staging slot before completing the swap.

You should use the CompleteSlotSwap value for the -SwapWithPreviewAction parameter. The CompleteSlotSwap value completes the swap. You should use this in the second stage after initiating the swap with the ApplySlotConfig value for the -SwapWithPreviewAction parameter.

You should not use the New-AzWebAppSlot PowerShell cmdlet. You should use this cmdlet for creating an Azure web app slot. In this scenario, you are tasked with swapping a staging environment with a production slot and not creating a new Azure web app slot.

You should not use the Set-AzWebAppSlot PowerShell cmdlet. You should use this cmdlet for modifying an Azure web app slot. In this scenario, you are tasked with swapping a staging environment with a production slot and not modifying an existing Azure web app slot.