DUMPSBOSS.

Certified Cloud Pentesting eXpert - Azure

The SecOps Group CCPenX-Az

Version Demo

Total Demo Questions: 4

Total Premium Questions: 46

Buy Premium PDF

https://dumpsboss.co

support@dumpsboss.co

support@dumpsboss.co
dumpsboss.co

QUESTION NO: 1 - (SIMULATION)

Carefully enumerate the accessible Azure Blob Container to locate a file containing credentials for an App Registration within the tenant. What is the Application/Client ID of the discovered App Registration?

ANSWER: Check the explanation for the answer

Explanation:

Answer: The Application/Client ID cannot be determined from the supplied question JSON. The credential-file contents (or the simulation environment/SAS token needed to retrieve it) are not included; the value shown as `<application-client-id>` is explicitly a placeholder.

In the simulation, enumerate and download the accessible blob container, then submit the value of `clientId` (or equivalently `appId`/`applicationId`) from the discovered credential file:

```
az storage blob list --account-name excaliburstore --container-name sensitive-files --sas-token "$SAS" --query "[].name" -o table
az storage blob download-batch --account-name excaliburstore --source sensitive-files --destination blobloot --sas-token "$SAS"
grep -RniE 'clientId|appId|applicationId' blobloot
```

The tenant ID `f015f36d-c07f-41fb-9bde-fffc3a22ee8b` is *not* the requested Application/Client ID.

QUESTION NO: 2

The compromised service principal has Contributor access to a resource group but no direct Key Vault data-plane role. Can it immediately read Key Vault secret values?

A. Yes, Contributor includes secret read permissions

B. No, Contributor does not automatically grant Key Vault secret data-plane read

C. Yes, if the vault is in the same resource group

D. No, service principals cannot access Key Vault

ANSWER: B

Explanation:

No, Contributor does not automatically grant Key Vault secret data-plane read. Azure Key Vault separates management-plane permissions from data-plane permissions. Contributor assigned at the resource-group scope permits management of Azure resources in that scope, such as viewing and modifying resource configurations, but it does not itself authorize operations on the contents stored in a vault. Reading a secret value is a Key Vault data-plane operation and requires explicit authorization through the vault's configured authorization model: an appropriate Azure RBAC data-plane role, such as Key Vault Secrets User, or a Key Vault access policy that grants secret `get` permission when the vault uses the legacy access-policy model. Resource-group inheritance does not convert a Contributor assignment into secret-read rights merely because the vault is located within that resource group. Therefore, at the stated time—without a direct data-plane role or equivalent secret permission—the compromised service principal cannot immediately retrieve secret values. Microsoft documents this management-plane/data-plane separation and the roles used for secret access in Azure Key Vault RBAC guide and Secure access to a Key Vault.

QUESTION NO: 3

You recover an account SAS for Blob Storage with the following relevant fields:

```
ss=b&srt=o&sp=rl
```

The token is valid and the target container name is already known. Why does an attempt to list the blobs in that container fail?

A. The SAS lacks service resource type access for Blob Storage

B. The SAS must include write permission before blobs can be listed

C. The SAS applies only to Azure Table Storage resources

D. The SAS permits only object resource types, while blob listing requires container resource type access

ANSWER: D

Explanation:

The SAS permits only object resource types, while blob listing requires container resource type access is correct. In an account SAS, `srt=o` restricts the token to object resource types. Although `sp=rl` includes read and list permissions, the resource types included in the SAS must also support the requested operation. Listing blobs is performed against the container resource type.

The token may still allow reads of known blob objects where the requested resource and permissions are otherwise valid. Extending the expiry time, changing the storage endpoint, or using an Azure Resource Manager token would not add the missing container resource type.

QUESTION NO: 4 - (SIMULATION)

You discover a storage account named prodreportstore01. Determine whether public blob access is enabled on the storage account.

ANSWER: Check the explanation for the answer

Explanation:

Public blob access is enabled on `prodreportstore01`.

Verify it with:

```
az storage account show \
  --name prodreportstore01 \
  --resource-group rg-prod-apps-eastus \
  --query "{Name:name, AllowBlobPublicAccess:allowBlobPublicAccess}" \
  --output json
```

The relevant result is:

```
{
  "Name": "prodreportstore01",
  "AllowBlobPublicAccess": true
}
```

`allowBlobPublicAccess: true` means the storage account permits anonymous public access to blobs/containers when a container is configured with a public access level. It does not by itself confirm that any specific container is currently public.