VirtuLearn

TRUSTED, ACCURATE AND RELIABLE!

The most comprehensive IT certification preparation materials in the industry!

https://www.virtulearner.com
support@virtulearner.com

Amazon

SCS-C03

AWS Certified Security - Specialty Exam

QUESTION: 1
A security administrator is setting up a new AWS account. The security administrator wants to secure the data that a company stores in an Amazon S3 bucket. The security administrator also wants to reduce the chance of unintended data exposure and the potential for misconfiguration of objects that are in the S3 bucket.

Which solution will meet these requirements with the LEAST operational overhead?

A. Configure the S3 Block Public Access feature for the AWS account.
B. Configure the S3 Block Public Access feature for all objects that are in the bucket.
C. Deactivate ACLs for objects that are in the bucket.
D. Use AWS PrivateLink for Amazon S3 to access the bucket.

Answer(s): A

Explanation:
Amazon S3 Block Public Access configured at the AWS account level is the recommended and most effective approach to protect data stored in Amazon S3 while minimizing operational overhead. AWS Security Specialty documentation explains that S3 Block Public Access provides centralized,

preventative controls designed to block public access to S3 buckets and objects regardless of individual bucket policies or object-level ACL configurations.
When enabled at the account level, these controls automatically apply to all existing and newly created buckets, significantly reducing the risk of accidental exposure caused by misconfigured permissions.

The AWS Certified Security - Specialty Study Guide emphasizes that public access misconfiguration is a leading cause of data leaks in cloud environments. Account-level S3 Block Public Access acts as a guardrail by overriding any attempt to grant public permissions through bucket policies or ACLs. This eliminates the need to manage security settings on a per-bucket or per-object basis, thereby reducing administrative complexity and human error.

Configuring Block Public Access at the object level, as in option B, requires continuous monitoring and manual configuration, which increases operational overhead. Disabling ACLs alone, as described in option C, does not fully prevent public access because bucket policies can still allow public permissions. Using AWS PrivateLink, as in option D, controls network access but does not protect against public exposure through misconfigured S3 policies.

AWS security best practices explicitly recommend enabling S3 Block Public Access at the account level as the primary mechanism for preventing unintended public data exposure with minimal management effort.

Referenced AWS Specialty Documents:

AWS Certified Security - Specialty Official Study Guide

Amazon S3 Security Best Practices Documentation

Amazon S3 Block Public Access Overview

AWS Well-Architected Framework - Security Pillar

QUESTION: 2
A company's developers are using AWS Lambda function URLs to invoke functions directly. The company must ensure that developers cannot configure or deploy unauthenticated functions in production accounts. The company wants to meet this requirement by using AWS Organizations. The solution must not require additional work for the developers.

Which solution will meet these requirements?

A. Require the developers to configure all function URLs to support cross-origin resource sharing (CORS) when the functions are called from a different domain.
B. Use an AWS WAF delegated administrator account to view and block unauthenticated access to function URLs in production accounts, based on the OU of accounts that are using the functions.
C. Use SCPs to allow all lambda:CreateFunctionUrlConfig and lambda:UpdateFunctionUrlConfig actions that have a lambda:FunctionUrlAuthType condition key value of AWS_IAM.
D. Use SCPs to deny all lambda:CreateFunctionUrlConfig and lambda:UpdateFunctionUrlConfig actions that have a lambda:FunctionUrlAuthType condition key value of NONE.

Answer(s): D

Explanation:
AWS Organizations service control policies (SCPs) are designed to enforce preventive guardrails across accounts without requiring application-level changes. According to the AWS Certified Security - Specialty documentation, SCPs can restrict specific API actions or require certain condition keys to enforce security standards centrally. AWS Lambda function URLs support two authentication modes:
AWS_IAM and NONE. When the authentication type is set to NONE, the function URL becomes publicly accessible, which introduces a significant security risk in production environments.

By using an SCP that explicitly denies the lambda:CreateFunctionUrlConfig and lambda:UpdateFunctionUrlConfig actions when the lambda:FunctionUrlAuthType condition key equals NONE, the organization ensures that unauthenticated function URLs cannot be created or modified in production accounts. This enforcement occurs at the AWS Organizations level and applies automatically to all accounts within the specified organizational units (OUs). Developers are not required to change their workflows or add additional controls, satisfying the requirement of no additional developer effort.

Option A relates to browser-based access controls and does not provide authentication or authorization enforcement. Option B is not valid because AWS WAF cannot be attached directly to AWS Lambda function URLs. Option C is incorrect because SCPs do not grant permissions; they only limit permissions. AWS documentation clearly states that SCPs define maximum available permissions and are evaluated before IAM policies.