Module 5 – Worksheet

AWS Certified Solutions Architect – Associate

Module 5: Overview

Domain 3 – Data Security

Build security into every layer:

- Compute and Networking
- Storage and Content Delivery
- Database
- Deployment and Management
- App Services

Sample Questions

QUESTION 1

You need a secure way to distribute your AWS credentials to an application running on Amazon Elastic Compute Cloud (Amazon EC2) instances in order to access supplementary AWS cloud services. What approach provides your application with access to use short-term credentials for signing requests, while protecting those credentials from other users? *(Choose the best answer)*

a. ( ) Add your credentials to the UserData parameter of each Amazon EC2 instance.

b. ( ) Use a configuration file to store your access and secret keys on the Amazon EC2 instances.

c. ( ) Specify your access and secret keys directly in your application.

d. ( ) Provision the Amazon EC2 instances with an instance profile that has the appropriate privileges.

**See solution at end of document*

QUESTION 2

Which of the following are best practices for managing AWS Identity and Access Management (IAM) user access keys? *(Choose three answers)*

a. ( ) Embed access keys directly into application code.

b. ( ) Use different access keys for different applications.

c. ( ) Rotate access keys periodically.

d. ( ) Keep unused access keys for an indefinite period of time.

e. ( ) Configure Multi-Factor Authentication (MFA) for your most sensitive operations.

**See solution at end of document*

QUESTION 3

Your application stores critical data in Amazon Simple Storage Service (Amazon S3), which must be protected against inadvertent or intentional deletion. How can this data be protected? *(Choose two answers)*

a. ( ) Use cross-region replication to copy data to another bucket automatically.

b. ( ) Set a vault lock.

c. ( ) Enable versioning on the bucket.

d. ( ) Use a lifecycle policy to migrate data to Amazon Glacier.

e. ( ) Enable MFA Delete on the bucket.

**See solution at end of document*

Personal Preparation Plan

Check off the items you've already completed. Mark items to complete before your exam:

Identify personal knowledge gaps:

The certification exam validates the following proficiencies. When you are ready for the certification exam, you should feel comfortable with the following concepts. Based on your self-assessment of your own knowledge gaps, mark those items on which you should build proficiency before your exam:

❑ Be familiar with the shared responsibility model and how responsibility is allocated between AWS and the customer.

- ❑ Understand how the principle of least privilege and types of identities may drive security recommendations. Be able to identify and differentiate between the Master account, IAM entities, and AWS Security Token Service (STS).
- ❑ Be familiar with best practices for securing your compute/network architecture layer using subnets and route tables with VPCs.
- ❑ Be able to differentiate between security groups and network ACLs.
- ❑ Know the various methods to access your VPCs, such as using an Internet gateway, VPN, AWS Direct Connect, or VPC peering.
- ❑ Know how to secure your data both in transit and at rest. Understand the implications of access-granting mechanisms and different ways to encrypt data on Amazon S3 or Amazon EBS.
- ❑ Be familiar with the different types of identities that exist in your applications and infrastructure, what each protects, and how they differ.
- ❑ Be aware of the customer's responsibility in securing their applications running on AWS infrastructure.

Resources

Related labs:
❑ Intro to AWS Identity and Access Management (IAM)
❑ Building Your First Amazon Virtual Private Cloud (VPC)
https://qwiklabs.com/learning_paths/10/lab_catalogue?locale=en

Related whitepapers:
❑ Introduction to AWS Security
❑ Overview of Security Processes
❑ Securing Data at Rest with Encryption
❑ AWS Security Best Practices
http://aws.amazon.com/whitepapers/

See blog entry:
IAM policies and Bucket Policies and ACLs! Oh My! (November 19, 2013)
http://blogs.aws.amazon.com/security/post/TxPOJBY6FE360K/IAM-policies-and-Bucket-Policies-and-ACLs-Oh-My-Controlling-Access-to-S3-Resourc

AWS Certified Solutions Architect – Associate website:
http://aws.amazon.com/certification/certified-solutions-architect-associate/

Sample Question Solutions

QUESTION 1

You need a secure way to distribute your AWS credentials to an application running on Amazon Elastic Compute Cloud (Amazon EC2) instances in order to **access supplementary AWS cloud services**. What approach provides your application with access to use **short-term credentials** for signing requests, while **protecting those credentials from other users**?

a. ( ) Add your credentials to the UserData parameter of each Amazon EC2 instance.
Incorrect: UserData is available within the Amazon EC2 instance metadata which is not protected by cryptographic methods. Anyone who can access the instance can view its metadata. Therefore, you should not store sensitive data, such as passwords or credentials, as user data.

b. ( ) Use a configuration file to store your access and secret keys on the Amazon EC2 instances.
Incorrect: While storing access and secret keys in a configuration file that is protected by cryptographic methods is an option. It does not address the requirement to use short-term credentials for signing requests.

c. ( ) Specify your access and secret keys directly in your application.
Incorrect: Specifying your access and secret keys directly in your application is not secure given that anyone who can who can access the instance can potentially view the credentials. Additionally, it does not address the requirement to use short-term credentials for signing requests.

d. (•) Provision the Amazon EC2 instances with an instance profile that has the appropriate privileges.
Correct: You should use an IAM role to manage temporary credentials for applications that run on an EC2 instance. When you use a role, you don't have to distribute long-term credentials to an EC2 instance. Instead, the role supplies temporary permissions that applications can use when they make calls to other AWS resources.

QUESTION 2

Which of the following are best practices for managing AWS Identity and Access Management (IAM) user access keys? (*Choose three answers*)

a. ( ) Embed access keys directly into application code.
Incorrect: Embedding access keys in application code is not a best practice. The AWS SDKs and the AWS Command Line Tools allow you to put access keys in known locations so that you do not have to keep them in code.

b. (•) Use different access keys for different applications.
Correct: This is a best practice. You can isolate the permissions and revoke the access keys for individual applications if an access key is exposed. Having separate access keys for different applications also generates distinct entries in AWS CloudTrail log files, which makes it easier for you to determine which application performed specific actions.

c. (•) Rotate access keys periodically.
Correct: This is a best practice. Change your own access keys regularly, and make sure that all IAM users in your account do as well. That way, if an access key is compromised without your knowledge, you limit how long the credentials can be used to access your resources.

d. ( ) Keep unused access keys for an indefinite period of time.
Incorrect: You should be diligent about removing IAM user credentials (that is, passwords and access keys) that are not needed. If a user leaves your organization, remove the corresponding IAM user so that the user's access to your resources is removed. Keeping unused access keys for an indefinite period of time is not a best practice.

e. (•) Configure Multi-Factor Authentication (MFA) for your most sensitive operations.
Correct: This is a best practice. For increased security, you should configure multi-factor

authentication (MFA) to help protect your AWS resources. MFA adds extra security because it requires users to enter a unique authentication code from an approved authentication device or SMS text message when they access AWS websites or services.

Question 3

Your application stores critical data in Amazon Simple Storage Service (Amazon S3), which must be **protected against inadvertent or intentional deletion**. How can this data be protected? (*Choose two answers*)

a. ( ) Use cross-region replication to copy data to another bucket automatically.
Incorrect: While enabling cross-region replication helps with keeping copies of your critical data in locations that are hundreds of miles apart, it does not help protect against inadvertent or intentional deletion.

b. ( ) Set a vault lock.
Incorrect: This feature is for Amazon Glacier. Amazon Glacier Vault Lock allows you to easily deploy and enforce compliance controls for individual Amazon Glacier vaults. This does not pertain to inadvertent or intentional deletion in Amazon S3.

c. (•) Enable versioning on the bucket.
Correct: Versioning allows you to preserve, retrieve, and restore every version of every object stored in an Amazon S3 bucket. Once you enable Versioning for a bucket, Amazon S3 preserves existing objects anytime you perform a PUT, POST, COPY, or DELETE operation on them. By default, GET requests will retrieve the most recently written version. Older versions of an overwritten or deleted object can be retrieved by specifying a version in the request.

d. ( ) Use a lifecycle policy to migrate data to Amazon Glacier.
Incorrect: S3 Lifecycle management provides the ability to define the lifecycle of your object with a predefined policy and reduce your cost of storage. You can set lifecycle transition policy to automatically migrate Amazon S3 objects to Standard - Infrequent Access (Standard - IA) and/or Amazon Glacier based on the age of the data. You can also set lifecycle expiration policies to automatically remove objects based on the age of the object. S3 Lifecycle management does not help prevent inadvertent or intentional deletion in Amazon S3.

e. (•) Enable MFA Delete on the bucket.
Correct: The MFA Delete capability, which uses multi-factor authentication, can be used to provide an additional layer of security. By default, all requests to your Amazon S3 bucket require your AWS account credentials. If you enable Versioning with MFA Delete on your Amazon S3 bucket, two forms of authentication are required to permanently delete a version of an object: your AWS account credentials and a valid six-digit code and serial number from an authentication device in your physical possession.