Exam Session - Knowledge Check: Encryption (SAA-C03)

cloudacademy.com/quiz/exam/3795299/results

#1

The AWS Secrets Manager is used for _____.

✗

autoscaling of EC2 instances

✓

storing secrets such as database credentials in a secure store

✗

the encryption and decryption of data

✗

assigning permissions and roles to users and resources

Explanation

You should always avoid embedding and hard-coding credentials in an application. This problem is alleviated with the introduction of AWS Secrets Manager, a service which allows you to store the secret such as database credentials in a secure store.

/course/using-aws-secrets-manager-manage-rotate-retrieve-secrets-1602/using-aws-secrets-manager-to-manage-rotate-and-retrieve-secrets/

#2

Which statement regarding CloudHSM and AWS KMS is correct?

✗

AWS KMS does not use HSMs while CloudHSM does.

✗

AWS KMS and CloudHSM only support asymmetric encryption.

✗

AWS KMS provides more key management options than AWS CloudHSM.

✓

AWS KMS manages HSM devices while CloudHSM provides customer-managed HSM devices.

Explanation

AWS CloudHSM is not the only encryption service available with AWS, you may have also heard of the Key Management Service, known as KMS. KMS is a managed service used to store and generate encryption keys that can be used by other AWS services and applications to encrypt your data.

Much like CloudHSM, KMS uses HSMs, but with KMS, these are managed by AWS, as a result, you have less management control of the keys and key material. Later in this course, I shall explain the integrations that exist between the 2 services.

🔗/course/get-started-with-aws-cloudhsm/what-is-cloudhsm/

#3

Your team has two KMS keys, KMS key1 and KMS key2.The policy for KMS key1 allows access to the AWS account (root user). The policy for KMS key2 allows access to you and your coworker, River. River currently has no IAM policy. Which keys, if any, does River have access to?

✗

River has no access to either KMS key.

✗

River has access to only KMS key1.

✓

River has access to only KMS key2.

✗

River has access to both KMS key1 and KMS key2.

Explanation

KMS key1's key policy allows access to the AWS account (root user) and thereby enables IAM policies to allow access to KMS key1. Unfortunately, River cannot access KMS key1 because KMS key1's key policy does not explicitly allow her access and she has no IAM policy that allows access. She can, however, access KMS key2 because the KMS key2's key policy explicitly allows her access.

🔗http://docs.aws.amazon.com/kms/latest/developerguide/deleting-keys-determining-usage.html

#4

What are the general steps in SSE-S3 data encryption? (Choose 2 answers)

✓

Encrypt the data with a data key

✗

Create a copy of a data key from the master key.

✓

Encrypt the data key with a master key.

✗

Encrypt the data with a master key.

Explanation

With SSE-S3, a multifactor encryption process was used by first encrypting the object data with a data key and then this data key was encrypted with a master key.

🔗/course/aws-big-data-security-encryption/

Covered in this lecture

Amazon S3 and Amazon Athena Encryption

Course:AWS Encryption for Data Analytics

12m

🔖

#5

Key Management Service (KMS) is used to manage encryption keys in your AWS environment. How can you audit the changes made on KMS?

✗

KMS provides full audit details as part of KMS console which can be accessed through web interface and APIs.

✗

KMS provides history to each key changes; you can track the changes done on each key using key history.

✗

KMS will log all changes in a special S3 bucket that is created the first time KMS service is being used.

✓

KMS has full audit and compliance integration with CloudTrail; this is where you can audit all changes performed on KMS.

Explanation

KMS is fully integrated with CloudTrail which provides audit and compliance features on all actions performed in KMS.

/amazon-web-services/amazon-web-services-key-management-service-kms-course/key-management-service-basics.html

Covered in this lecture

Course Summary

Course:How to Use KMS Key Encryption to Protect Your Data

10m

#6

AWS Secrets Manager rotates secrets automatically with backend support from ______.

✗

Redshift

✓

built-in Lambda functions

✗

Python scripts

✗

DocumentDB

Explanation

AWS Secrets Manager supports RDS, DocumentDB, and Redshift and rotates these secrets automatically with backend support from built-in Lambda functions.

🔗/course/using-aws-secrets-manager-manage-rotate-retrieve-secrets-1602/using-aws-secrets-manager-to-manage-rotate-and-retrieve-secrets/
#7

What is the difference between default and custom KMS key stores?

✗
Default key stores secure KMS keys within an S3 bucket but custom key stores secure KMS keys within a CloudHSM device.

✗
Default key stores secure KMS keys within an AWS-managed HSM device, while custom key stores secure KMS keys within a CloudHSM device.

✓
Both default and custom key stores secure KMS keys in AWS-managed HSM devices, but custom key stores allow key material to be stored in CloudHSM devices.

✗
Both default and custom key stores secure KMS keys in AWS-managed HSM devices, but a custom key stores allow users to create customer-managed KMS keys.

Explanation

Customers can create customer-managed KMS keys without using a custom key store.

The custom key store is a resource managed from within KMS but allows you to store your key material within your managed HSMs of your CloudHSM cluster. This allows you to use the key material located within your HSM cluster to create the KMS keys that KMS uses to implement encryption across different AWS services. KMS keys created from your custom key store are 256-bit, non-exportable AES symmetric keys that never leave the HSM unencrypted. All cryptographic operations made with the KMS key happen within the HSM cluster.

So the main difference between the store is how keys are created, and where the key material is stored.

🔗/course/get-started-with-aws-cloudhsm/using-cloudhsm-custom-key-store-kms/
#8

Which of the following is true about AWS KMS keys managed by AWS?

✗

You can share AWS-managed keys between accounts.

✗

Key policy configuration can be performed by the AWS customer.

✓

These keys are used by other AWS services that have the ability to interact with KMS directly to perform encryption against data.

✗

You, as an AWS customer, can disable the key when it is no longer required.

Explanation

KMS keys can be managed either by AWS or by you and me as customers of AWS. KMS keys managed by AWS are used by other AWS services that have the ability to interact with KMS directly to perform encryption against data. An example is Amazon S3, in particular SSE-KMS, which is server-side encryption using the Key Management Service. KMS keys that are created and generated by you and me rather than AWS provide the ability to implement greater flexibility, such as being able to manage the key, including rotation, governing access, and key policy configuration, along with being able to both enable and disable the key when it is no longer required.

/course/how-to-share-cmks-across-multiple-accounts-using-kms/sharing-cmks-across-multiple-aws-accounts/
Covered in this lecture
Sharing CMKs Across Multiple AWS Accounts
Course:How to Share CMKs Across Multiple Accounts Using AWS KMS

14m

#9

Key rotation is an important concept of key management. How does Key Management Service (KMS) implement key rotation?

✗

KMS supports manual Key Rotation only; you can create new keys any time you want and all data will be re-encrypted with the new key.

✕

Key rotation is the process of synchronizing keys between configured regions; KMS will synchronize key changes in near-real time once keys are changed.

✕

Key rotation is supported through the re-importing of new KMS keys; once you import a new key all data keys will be re-encrypted with the new KMS key.

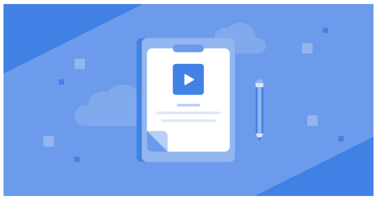
✓

KMS creates new cryptographic material for your KMS keys every rotation period, and uses the new keys for any upcoming encryption; it also maintains old keys to be able to decrypt data encrypted with those keys.

Explanation

When you enable *automatic key rotation* for a customer-managed KMS key, AWS KMS generates new cryptographic material for the KMS key every year. AWS KMS also saves the KMS key's older cryptographic material so it can be used to decrypt data that it has encrypted.

🔗/amazon-web-services/amazon-web-services-key-management-service-kms-course/key-management-service-basics.html
Covered in this lecture
Amazon Kinesis
Course:Designing Secure Applications and Architectures



5m

#10

Which of the following statements about key policies in AWS Key Management Service is true?

✓

Both an IAM identity-based policy in the AWS account that wants to access the KMS key and a resource-based key policy in the AWS account where the KMS key resides are required to access and use a KMS key from a different AWS account.

✕

Only an IAM identity-based policy in the AWS account that wants to access the KMS key is required to access and use a KMS key from a different AWS account.

✕

Only a resource-based key policy in the AWS account where the KMS key resides is required to access and use a KMS key from a different AWS account.

✕

Neither an IAM identity-based policy nor a resource-based key policy are required to access and use a KMS key from a different AWS account.

Explanation

Permissions to allow you to access and use a KMS key from a different AWS account can't be given and generated using IAM alone. As a result, you have to use and edit a resource-based key policy in the AWS account where the KMS key resides, in addition to an IAM identity-based policy in the AWS account that wants to access the KMS key.

/course/how-to-share-cmks-across-multiple-accounts-using-kms/sharing-cmks-across-multiple-aws-accounts/

Covered in this lecture

Sharing CMKs Across Multiple AWS Accounts

Course:How to Share CMKs Across Multiple Accounts Using AWS KMS

14m

#11

A user has enabled server-side encryption with S3 (SSE-S3) for an object. The user downloads the encrypted object from S3. How can the user decrypt it?

✕

The user must provide a KMS data key.

✕

The user needs to decrypt the object using their own account credentials.

✕

S3 provides an object key to decrypt the object.

✓

S3 manages encryption and decryption automatically

Explanation

If the user is using the server-side encryption feature, Amazon S3 encrypts the object data before saving it on disks in its data centres and decrypts it when the user downloads the objects. Thus, the user is free from the tasks of managing encryption, encryption keys, and related tools.

http://docs.aws.amazon.com/AmazonS3/latest/dev/UsingEncryption.html

Covered in this lecture

Server-Side Encryption with S3 Managed Keys (SSE-S3)

Course:Understanding S3 Encryption Mechanisms to Secure your Data

1m

#12

The AWS CloudHSM service provides HSMs that are validated to Federal Information Processing Standards (FIPS) 140-2 Level 3. This validation is often required to offer which of the following services? (Choose 2 answers)

✓

Offer document signing

✓

Run a public certificate authority

✗

Encrypt a personal computer board

✗

Password encryption

Explanation

FIPS 140-2 defines four levels of security, simply named "Level 1" to "Level 4". It does not specify in detail what level of security is required by any particular application.

FIPS 140-2 Level 1 the lowest, imposes very limited requirements; loosely, all components must be "production-grade" and various egregious kinds of insecurity must be absent. This applies to encryption of a personal computer board, which requires lower standards in physical security.

Password encryption is based more on algorithmic security rather than physical security, and certified password encryption can be provided using AES-128 encryption.

🔗/course/get-started-with-aws-cloudhsm/what-is-cloudhsm/

#13

When transmitting sensitive data using encryption algorithms, _____ refers to the input to an encryption algorithm, meaning that the data is in its unprotected, or unencrypted form.

✓

plaintext

✗

rawtext

✗

ciphertext

✗

usertext

Explanation

Plaintext refers to information or data in an unencrypted, or unprotected, form. *Ciphertext* refers to the output of an encryption algorithm operating on plaintext. Ciphertext is unreadable without knowledge of the algorithm and a secret key.

🔗http://docs.aws.amazon.com/kms/latest/developerguide/crypto-intro.html

Covered in this lecture

Client-Side Encryption with KMS Managed Keys (CSE-KMS)

Course:Understanding S3 Encryption Mechanisms to Secure your Data

2m

#14

What is AWS CloudHSM?

✓

A cloud-based hardware device that stores cryptographic keys

✗

A cryptographic key creation and storage service hosted in the AWS cloud

✗

An on-premise hardware device that manages identity and access management

✗

An AWS service that stores secrets in the cloud

Explanation

What is CloudHSM? Cloud HSM is a FIPS 140 level two validated hardware device for secure cryptographic key storage. I can't stress this enough, CloudHSM is a hardware appliance, it is not a virtualized service.

🔗/course/get-started-with-aws-cloudhsm/what-is-cloudhsm/

Covered in this lecture
What is CloudHSM?
Course:Manage Your Own Encryption Keys Using AWS CloudHSM

5m

#15

You typically use KMS keys in AWS KMS to encrypt your _____.

✗

passwords

✗

S3 buckets

✗

personal data

✓

data encryption keys

Explanation

The primary resources in AWS KMS are KMS keys. KMS keys are either customer-managed or AWS-managed. You can use either type of KMS key to protect up to 4 kibibytes (KiB) of data directly. Typically, you use KMS keys to protect *data encryption keys* (or *data keys*), which are then used to encrypt or decrypt larger amounts of data outside of the service. KMS keys never leave AWS KMS unencrypted, but data keys can. AWS KMS does not store, manage, or track your data keys.

http://docs.aws.amazon.com/kms/latest/developerguide/concepts.html

Covered in this lecture

Course Summary

Course:How to Use KMS Key Encryption to Protect Your Data

10m

#16

You are in charge of choosing an encryption option for a set of newly acquired storage objects containing personal data. You have recently noticed a potential access issue with some of the other encryption keys using AWS server-side encryption with managed keys (SSE-S3).Which of the following is the best scenario for choosing an encryption option to prevent key access issues?

✗

Choosing server-side encryption with managed keys (SSE -S3) because AWS provides the most secure key management by default.

✗

Choosing server-side encryption with managed keys (SSE -S3) because it requires minimal configuration providing you more time to monitor key access.

✗

Choosing server-side encryption with Key Management Service (KMS) because the the KMS monitors the encryption and decryption of objects.

✓

Choosing server-side encryption with Key Management Service (KMS) because it allows you to define policies that define how keys are used.

Explanation

Using KMS gives you far greater flexibility of how your keys are managed. For example, you are able to disable, rotate, and apply access controls to the KMS key, and audit against their usage using AWS Cloud Trail. SSE-S3 is a less appropriate option in this case because it manages the keys for you; similarly, using SSE -S3 makes the encryption process invisible to the end user, thus limiting your ability to understand or mitigate the encryption key issue. KMS allows for the monitoring in different ways of encryption and decryption processes by allowing the user access to these processes, not by doing it independently of the user.

🔗https://docs.aws.amazon.com/AmazonS3/latest/userguide/UsingKMSEncryption.html

#17

When creating a CloudHSM clustering, a physical HSM device cannot actually be placed in a VPC. Which of the following network components is used to represent each HSM device in a cluster?

✓

An Elastic Network Interface (ENI)

✕

A NAT instance

✕

A NAT gateway

✕

An Elastic IP address (EIP)

Explanation

During the deployment of your HSMs, it’s actually an Elastic Network Interface (ENI) that is placed within the subnet that you select of your VPC. The HSM itself actually resides in a different AWS-owned VPC, and is located in the same AZ as you select during its deployment. So it’s the ENI that is deployed in your VPC which acts as an interface between your network and the HSM residing in an AWS-owned VPC.

🔗/course/get-started-with-aws-cloudhsm/cloudhsm-operations/

#18

AWS CloudHSM is compatible with which type(s) of encryption keys?

✓

Both asymmetric and symmetric encryption keys

✕

Symmetric encryption keys

✕

Asymmetric encryption keys

✕

Neither symmetric nor asymmetric encryption keys

Explanation

There are a number of different operations that CloudHSM can help you provide, these include:

The creation, storage and management of cryptographic keys, allowing you to import and export both asymmetric and symmetric keys.

The ability to use cryptographic hash functions to enable you to compute message digests and hash-based message authentication codes, otherwise known as HMACs.

Cryptographic data signing and signature verification.

Using both asymmetric and symmetric encryption algorithms.

And the ability to generate cryptographically secure random data.

/course/get-started-with-aws-cloudhsm/what-is-cloudhsm

#19

What is the main key type within Amazon Key Management Service?

✕

Data Encryption Key

✓

AWS KMS Key

✕

Data Key

✗

Data Key Pair

Explanation

The AWS KMS Key is the main key type within KMS and can generate, encrypt, and decrypt data encryption keys known as the DEKs, which are used outside of the KMS service by other AWS services to perform encryption against your data.

/course/how-to-share-cmks-across-multiple-accounts-using-kms/sharing-cmks-across-multiple-aws-accounts/

Covered in this lecture

Sharing CMKs Across Multiple AWS Accounts

Course:How to Share CMKs Across Multiple Accounts Using AWS KMS

14m

#20

To manage access to your AWS KMS keys in AWS Key Management Service, you must use a(n) ______.

✗

permission

✗

data encryption key

✗

IAM policy

✓

key policy

Explanation

In all cases, to manage access to your KMS keys, you must use a key policy.

🔗/course/how-to-share-cmks-across-multiple-accounts-using-kms/sharing-cmks-across-multiple-aws-accounts/
Covered in this lecture
Sharing CMKs Across Multiple AWS Accounts
Course:How to Share CMKs Across Multiple Accounts Using AWS KMS

14m

#21

Which of the following statements about AWS Secrets Manager is false?

✗

If automatic secret rotation is enabled, when you first store a secret, it performs a rotation immediately.

✗

You can use an existing Lambda function to enable a secret rotation.

✗

You can let Secrets Manager create a new Lambda function for you to enable a secret rotation.

✓

By default, automatic secret rotation is enabled.

Explanation

Here we can decide if we want to configure automatic rotation. By default, it's disabled. Now you can let Secrets Manager create a new Lambda function for you to enable this rotation, or you can use an existing Lambda function. And when you first store your secret, it performs a rotation immediately.

🔗/course/using-aws-secrets-manager-manage-rotate-retrieve-secrets-1602/using-aws-secrets-manager-to-manage-rotate-and-retrieve-secrets/
#22

When an AWS CloudHSM device is initialized, what happens to the existing keys stored on the device?

✓

The existing keys are destroyed.

✗

The existing keys are updated.

✗

The existing keys are backed up to Amazon S3.

✗

The existing keys are unchanged.

Explanation

Two, be careful when initializing a CloudHSM. This action will destroy the keys, so either have another copy of the keys or be absolutely sure you do not and never, ever will need these keys to decrypt any data.

/course/get-started-with-aws-cloudhsm/what-is-cloudhsm/

Covered in this lecture
What is CloudHSM?
Course:Manage Your Own Encryption Keys Using AWS CloudHSM

5m

#23

Which Amazon S3 data encryption mechanism offers the highest level of control to the customer, but also requires the highest level of customer responsibility?

✗

SSE-C

✓

CSE-C

✗

SSE-S3

✗

CSE-KMS

Explanation

Using CSE-C, AWS assists in creating the keys and storing the encrypted objects. Key storage, rotation, encryption and decryption are entirely performed on the client's side.

/course/s3-encryption-mechanisms/s3-encrypt-cse-c/

Covered in this lecture
Client-Side Encryption with Customer Provided Keys (CSE-C)
Course:Understanding S3 Encryption Mechanisms to Secure your Data

2m

#24

AWS Key Management Service (KMS), makes use of _____ encryption, which is the practice of encrypting plaintext data with a unique data key, and then encrypting the data key with a key encryption (KEK).

✕
double

✕
super

✕
nested

✓
envelope

Explanation

AWS KMS uses *envelope encryption* to protect data. Envelope encryption is the practice of encrypting plaintext data with a unique data key, and then encrypting the data key with a *key encryption key* (KEK). You might choose to encrypt the KEK with another KEK, and so on, but eventually you must have a *master key*. The master key is an unencrypted (plaintext) key with which you can decrypt one or more other keys.

http://docs.aws.amazon.com/kms/latest/developerguide/concepts.html

#25

As a best practice with CloudHSM, always deploy CloudHSM in a high availability configuration with at least __________ appliances in separate availability zones.

✓

two

✗

three

✗

four

✗

five

Explanation

Always deploy CloudHSM in a high availability setup with at least two appliances in separate availability zones, and if possible, deploy a third either on-premises or in another AWS region.

/course/get-started-with-aws-cloudhsm/what-is-cloudhsm/

Covered in this lecture

Introduction to CloudHSM

Course:Manage Your Own Encryption Keys Using AWS CloudHSM

2m