Google Discussions

Exam Professional Cloud Security Engineer All Questions

View all questions & answers for the Professional Cloud Security Engineer exam

Go to Exam

EXAM PROFESSIONAL CLOUD SECURITY ENGINEER TOPIC 1 QUESTION 135 DISCUSSION

Actual exam question from Google's Professional Cloud Security Engineer

Question #: 135

Topic #: 1

[All Professional Cloud Security Engineer Questions]

You need to implement an encryption-at-rest strategy that protects sensitive data and reduces key management complexity for non-sensitive data. Your solution has the following requirements:

✑ Schedule key rotation for sensitive data.

✑ Control which region the encryption keys for sensitive data are stored in.

✑ Minimize the latency to access encryption keys for both sensitive and non-sensitive data.

What should you do?

A. Encrypt non-sensitive data and sensitive data with Cloud External Key Manager.

B. Encrypt non-sensitive data and sensitive data with Cloud Key Management Service.

C. Encrypt non-sensitive data with Google default encryption, and encrypt sensitive data with Cloud External Key Manager.

D. Encrypt non-sensitive data with Google default encryption, and encrypt sensitive data with Cloud Key Management Service.

Show Suggested Answer

by GHOST1985 at *Sept. 10, 2022, 3:56 p.m.*

Comments

Type your comment...

Submit

GHOST1985 Highly Voted 2 years, 4 months ago

Selected Answer: D

Answer D
because "Minimize the latency to access encryption keys"

upvoted 12 times

> **GHOST1985** 2 years, 3 months ago
>
> Sorry answer is B
>
> upvoted 3 times

marmar11111 Highly Voted 2 years, 2 months ago

Selected Answer: D

The default already has low latency! "Because of the high volume of keys at Google, and the need for low latency and high availability, DEKs are stored near the data that they encrypt. DEKs are encrypted with (wrapped by) a key encryption key (KEK), using a technique known as envelope encryption. These KEKs are not specific to customers; instead, one or more KEKs exist for each service."

We need less complexity and low latency so use default on non-sensitive data!

upvoted 6 times

> **adb4007** 1 year ago
>
> And keep KMS to be complience with sensitive data strategy
>
> upvoted 1 times

shayke Most Recent 2 years, 1 month ago

Selected Answer: D

D- the ans refers to both types of data:sensitive and non sensitive

upvoted 4 times

TonytheTiger 2 years, 2 months ago

Answer D
https://cloud.google.com/docs/security/encryption/default-encryption

upvoted 6 times

AzureDP900 2 years, 2 months ago

B. Encrypt non-sensitive data and sensitive data with Cloud Key Management Service.

upvoted 2 times

coco10k 2 years, 2 months ago

Selected Answer: D

keeps complexity low

upvoted 3 times

AwesomeGCP 2 years, 3 months ago

Selected Answer: B

B. Encrypt non-sensitive data and sensitive data with Cloud Key Management Service.

upvoted 1 times

GHOST1985 2 years, 3 months ago

Selected Answer: B

✑ Schedule key rotation for sensitive data. :
=> Cloud KMS allows you to set a rotation schedule for symmetric keys to automatically generate a new key version at a fixed time interval. Multiple versions of a symmetric key can be active at any time for decryption, with only one primary key version used for encrypting new data. With EKM, create an externally managed key directly from the Cloud KSM console.

✑ Control which region the encryption keys for sensitive data are stored in.
=> If using Cloud KMS, your cryptographic keys will be stored in the region where you deploy the resource. You also have the option of storing those keys inside a physical Hardware Security Module located in the region you choose with Cloud HSM.

✑ Minimize the latency to access encryption keys for both sensitive and non-sensitive data :
=> Cloud KMS is available in several global locations and across multi-regions, allowing you to place your service where you want for low latency and high availability.

https://cloud.google.com/security-key-management

upvoted 3 times

adb4007 1 year ago

You right and you need "reduces key management complexity for non-sensitive data" that why I go for D

upvoted 1 times