Google Discussions

EXAM PROFESSIONAL DATA ENGINEER TOPIC 1 QUESTION 162 DISCUSSION

Actual exam question from Google's Professional Data Engineer
Question #: 162
Topic #: 1
[All Professional Data Engineer Questions]

You want to archive data in Cloud Storage. Because some data is very sensitive, you want to use the `Trust No One` (TNO) approach to encrypt your data to prevent the cloud provider staff from decrypting your data. What should you do?

A. Use gcloud kms keys create to create a symmetric key. Then use gcloud kms encrypt to encrypt each archival file with the key and unique additional authenticated data (AAD). Use gsutil cp to upload each encrypted file to the Cloud Storage bucket, and keep the AAD outside of Google Cloud.

B. Use gcloud kms keys create to create a symmetric key. Then use gcloud kms encrypt to encrypt each archival file with the key. Use gsutil cp to upload each encrypted file to the Cloud Storage bucket. Manually destroy the key previously used for encryption, and rotate the key once.

C. Specify customer-supplied encryption key (CSEK) in the .boto configuration file. Use gsutil cp to upload each archival file to the Cloud Storage bucket. Save the CSEK in Cloud Memorystore as permanent storage of the secret.

D. Specify customer-supplied encryption key (CSEK) in the .boto configuration file. Use gsutil cp to upload each archival file to the Cloud Storage bucket. Save the CSEK in a different project that only the security team can access.

Show Suggested Answer

by rickywck at *March 18, 2020, 2:11 a.m.*

Comments

Type your comment...

Submit

dhs227 Highly Voted 5 years, 1 month ago

The correct answer must be D
A and B can be eliminated immediately since kms generated keys are considered potentially accessible by CSP.
C is incorrect because memory store is essentially a cache service.

Additional authenticated data (AAD) acts as a "salt", it is not a cipher.

upvoted 45 times

[Removed] 3 years, 3 months ago

The trust no one design philosophy requires that the keys for encryption should always be, and stay, in the hands of the user that applies them. This implies that no external party can access the encrypted data (assumed that the encryption is strong enough).
https://en.wikipedia.org/wiki/Trust_no_one_(Internet_security)

upvoted 3 times

mikey007 4 years, 8 months ago

AAD is bound to the encrypted data, because you cannot decrypt the ciphertext unless you know the AAD, but it is not stored as part of the ciphertext. AAD also does not increase the cryptographic strength of the ciphertext. Instead it is an additional check by Cloud KMS to authenticate a decryption request.

upvoted 4 times

[Removed] Highly Voted 5 years, 1 month ago

Answer: A
Description: AAD is used to decrypt the data so better to keep it outside GCP for safety

upvoted 15 times

aaaaaaaasdasdasfs Most Recent 2 weeks, 4 days ago

Selected Answer: D

Based on the requirement for a "Trust No One" (TNO) approach where even the cloud provider cannot decrypt your data, the correct answer is:
D. Specify customer-supplied encryption key (CSEK) in the .boto configuration file. Use gsutil cp to upload each archival file to the Cloud Storage bucket. Save the CSEK in a different project that only the security team can access.
This is the best option because:

Customer-supplied encryption keys (CSEKs) are encryption keys that you manage and provide to Google Cloud, rather than using Google-managed keys.
When you use CSEKs, Google Cloud uses your key to encrypt your data but does not store the key. This means Google Cloud cannot decrypt your data without you providing the key again.
Storing the CSEK in a different project that only the security team can access ensures that the key is securely stored but separated from the encrypted data.

upvoted 1 times

aaaaaaaasdasdasfs 3 weeks, 4 days ago

Selected Answer: D

This is the correct option because:

Customer-supplied encryption keys (CSEKs) provide client-side encryption where you fully control the keys.
By specifying the CSEK in the .boto configuration file, the data is encrypted before it reaches Google's servers.
Storing the keys in a different project with restricted access ensures proper separation.
This approach keeps the encryption keys entirely under your control, following the TNO principle.

upvoted 2 times

Anudeep58 10 months, 3 weeks ago

Selected Answer: A

Keep AAD Outside of Google Cloud:

Keeping the AAD outside of Google Cloud ensures that Google cannot access the additional context required to decrypt the files, thus implementing the TNO approach.

Option C:
Customer-Supplied Encryption Key (CSEK) in .boto File:
Storing the CSEK in Cloud Memorystore or any cloud service introduces a risk where the key could be potentially accessed by cloud provider staff.
Option D:
Customer-Supplied Encryption Key (CSEK) in a Different Project:
While storing the CSEK in a different project adds some security, it still leaves the keys within the Google Cloud environment,

which does not fully meet the TNO approach.

upvoted 2 times

emmylou 1 year, 5 months ago

I just cannot understand this question. If you can't trust the provider, in this case Google, then how can you use the KMS approach. In my mind you have to generate the key locally and upload but I'm clearly wrong and don't get why.

upvoted 1 times

shanwford 1 year, 7 months ago

Selected Answer: D

IMO must be (D) : to reach TNO goal keys must be customer supplied.

upvoted 4 times

barnac1es 1 year, 7 months ago

Selected Answer: D

Customer-Supplied Encryption Key (CSEK): CSEK allows you to provide your encryption keys, ensuring that the cloud provider staff does not have access to the keys and cannot decrypt your data.

Separate Project for Key Management: Saving the CSEK in a different project that only the security team can access adds an additional layer of security. It isolates the encryption keys from the project where the data is stored, ensuring that even within the same cloud provider, only authorized personnel can access the keys.

Use of .boto Configuration: Specifying the CSEK in the .boto configuration file ensures that it is applied consistently when interacting with Cloud Storage through tools like gsutil. This way, every archival file is encrypted using your keys.

Options A and B involve using Google Cloud Key Management Service (KMS) to manage keys, which does not align with the TNO approach because cloud provider staff could potentially access the keys stored in Google Cloud KMS.

upvoted 2 times

[Removed] 1 year, 7 months ago

Selected Answer: A

The answer is A
The question tells us that "prevent the cloud provider staff from decrypting", so we cannot keep anything that helps decrypt on GCP, not even in a different project. so the answer cannot be D.

upvoted 4 times

NewDE2023 1 year, 9 months ago

Selected Answer: D

CSEKs are used when an organization needs complete control over key management.

upvoted 4 times

tavva_prudhvi 1 year, 9 months ago

Option A is not the best choice for the "Trust No One" (TNO) approach because it involves using Google Cloud's Key Management Service (KMS) to create and manage encryption keys. This means that the cloud provider will have access to the keys, which could potentially enable their staff to decrypt the data.

upvoted 2 times

midgoo 2 years, 1 month ago

Selected Answer: A

D may work, but 'Trust No One' = do not trust GCP too. So D cannot be the answer.

upvoted 3 times

musumusu 2 years, 2 months ago

answer A: KMS + AAD is more secure than CSEK

upvoted 2 times

zellck 2 years, 5 months ago

Selected Answer: A

A is the answer.

https://cloud.google.com/kms/docs/additional-authenticated-data
Additional authenticated data (AAD) is any string that you pass to Cloud Key Management Service as part of an encrypt or decrypt request. AAD is used as an integrity check and can help protect your data from a confused deputy attack. The AAD string must be no larger than 64 KiB.

Cloud KMS will not decrypt ciphertext unless the same AAD value is used for both encryption and decryption.

AAD is bound to the encrypted data, because you cannot decrypt the ciphertext unless you know the AAD, but it is not stored as part of the ciphertext. AAD also does not increase the cryptographic strength of the ciphertext. Instead it is an additional check by Cloud KMS to authenticate a decryption request.

upvoted 8 times

AzureDP900 2 years, 4 months ago

Agree with A

upvoted 2 times

Jay_Krish 2 years, 5 months ago

Selected Answer: D

CSEK with only security team having access seems to be right approach. Not sure how A can be better.

upvoted 4 times

cloudmon 2 years, 6 months ago

Selected Answer: A

It's A, because you cannot decrypt the ciphertext unless you know the AAD (https://cloud.google.com/kms/docs/additional-authenticated-data)

upvoted 2 times

devaid 2 years, 7 months ago

Selected Answer: A

Answer: A

upvoted 1 times

Load full discussion...