Google Discussions

EXAM PROFESSIONAL CLOUD SECURITY ENGINEER TOPIC 1 QUESTION 15 DISCUSSION

Actual exam question from Google's Professional Cloud Security Engineer
Question #: 15
Topic #: 1
[All Professional Cloud Security Engineer Questions]

An application running on a Compute Engine instance needs to read data from a Cloud Storage bucket. Your team does not allow Cloud Storage buckets to be globally readable and wants to ensure the principle of least privilege.
Which option meets the requirement of your team?

A. Create a Cloud Storage ACL that allows read-only access from the Compute Engine instance's IP address and allows the application to read from the bucket without credentials.

B. Use a service account with read-only access to the Cloud Storage bucket, and store the credentials to the service account in the config of the application on the Compute Engine instance.

C. Use a service account with read-only access to the Cloud Storage bucket to retrieve the credentials from the instance metadata.

D. Encrypt the data in the Cloud Storage bucket using Cloud KMS, and allow the application to decrypt the data with the KMS key.

Show Suggested Answer

by ArizonaClassics at *Aug. 2, 2020, 12:17 a.m.*

Comments

Type your comment...

Submit

Medofree Highly Voted 2 years, 3 months ago

Selected Answer: C

Correct ans is C. The credentials are retrieved from the metedata server

upvoted 13 times

ESP_SAP Highly Voted 3 years, 8 months ago

Correct Answer is (B):
If your application runs inside a Google Cloud environment that has a default service account, your application can retrieve the service account credentials to call Google Cloud APIs. Such environments include Compute Engine, Google Kubernetes Engine, App Engine, Cloud Run, and Cloud Functions. We recommend using this strategy because it is more convenient and secure than manually passing credentials.

Additionally, we recommend you use Google Cloud Client Libraries for your application. Google Cloud Client Libraries use a library called Application Default Credentials (ADC) to automatically find your service account credentials. ADC looks for service account credentials in the following order:

https://cloud.google.com/docs/authentication/production#automatically

upvoted 13 times

GianpiGale 1 month, 2 weeks ago

Using the default service account does'nt enforce least privilege, you do the same but with a dedicated one, therefore C. Although i think C is very poorly articulated

upvoted 1 times

ChewB666 3 years, 8 months ago

Hello guys!

Does anyone have the rest of the questions to share? :(
I can't see the rest of the issues because of the subscription.

upvoted 3 times

okhascorpio Most Recent 9 months, 2 weeks ago

A. Although it would work, but it is less preferred method and are error prone.
B. Storing credentials in config is not good idea.
C. Is preferred method as applications can get credentials from instance metadata securely.
D. does not suggest controlled access, only encryption.

upvoted 2 times

ArizonaClassics 10 months, 2 weeks ago

C. Use a service account with read-only access to the Cloud Storage bucket to retrieve the credentials from the instance metadata.

upvoted 2 times

1br4in 1 year, 1 month ago

correct is B: Utilizzare un service account con accesso in sola lettura al bucket di Cloud Storage e archiviare le credenziali del service account nella configurazione dell'applicazione sull'istanza di Compute Engine.

Utilizzando un service account con accesso in sola lettura al bucket di Cloud Storage, puoi fornire all'applicazione le credenziali necessarie per leggere i dati dal bucket. Archiviando le credenziali del service account nella configurazione dell'applicazione sull'istanza di Compute Engine, garantisce che solo l'applicazione su quell'istanza abbia accesso alle credenziali e, di conseguenza, al bucket.

Questa opzione offre il principio del privilegio minimo, in quanto il service account ha solo i permessi necessari per leggere i dati dal bucket di Cloud Storage e le credenziali sono limitate all'applicazione specifica sull'istanza di Compute Engine. Inoltre, non richiede l'accesso globale ai bucket di Cloud Storage o l'utilizzo di autorizzazioni di accesso di rete basate su indirizzo IP.

upvoted 1 times

mahi9 1 year, 5 months ago

Selected Answer: C

C is the most viable option

upvoted 2 times

Meyucho 1 year, 8 months ago

Selected Answer: A

A CORRECT: It's the only answer when you use ACL to filter local IP's addresses and you can have the bucket without global access.

B INCORRET: Doesn't use the least privilege principle.
C INCORRECT: What credentials are we talking about!? To do this it's better option B.
D INCORRECT: Need global access.

upvoted 3 times

gcpengineer 1 year, 2 months ago

no.its not a soln

upvoted 1 times

dat987 1 year, 8 months ago

Selected Answer: B

meta data do not set service account

upvoted 2 times

GCP72 1 year, 11 months ago

Selected Answer: C

The correct answer is C

upvoted 2 times

AaronLee 2 years, 4 months ago

The Answer is C
If the environment variable GOOGLE_APPLICATION_CREDENTIALS is set, ADC uses the service account key or configuration file that the variable points to.

If the environment variable GOOGLE_APPLICATION_CREDENTIALS isn't set, ADC uses the service account that is attached to the resource that is running your code.
https://cloud.google.com/docs/authentication/production#passing_the_path_to_the_service_account_key_in_code

upvoted 4 times

jj_618 2 years, 10 months ago

So is it B or C?

upvoted 1 times

StanPeng 2 years, 5 months ago

B for sure. C is wrong logic

upvoted 1 times

Ishu_awsguy 1 year, 6 months ago

C is the right answer. If the service account has read permissions to cloud storage. Nothing extra is needed

upvoted 1 times

Medofree 2 years, 3 months ago

No the C is the right ans, you don't need to generate credentials into GCP since they are stored into metadata server, the application will retrieve them automatically through a Google Lib (or even manually by calling the url curl http://metadata.google.internal/computeMetadata/v1/instance/service-accounts/default/token -H "Metadata-Flavor: Google")

upvoted 3 times

bolu 3 years, 5 months ago

Answer can be either B or C due to the relevance to servicing account. But storing password in app is a worst practice and we read it several times everywhere online hence it results in C as a best answer to handle service account through metadata

upvoted 5 times

HectorLeon2099 3 years, 9 months ago

I'll go with B.
A - ACL's are not able to allow access based on IP
C - If you store the credentials in the metadata those will be public accessible by everyone with project access.
D - Too complex

upvoted 6 times

saurabh1805 3 years, 9 months ago

Yes B is best possible option. This is something google also recommnd.
https://cloud.google.com/storage/docs/authentication#libauth

upvoted 3 times

gcpengineer 1 year, 2 months ago

google never recommend that

upvoted 3 times

CHECK666 3 years, 10 months ago

c is correct

upvoted 2 times

Moe666 3 years, 10 months ago

C is the answer

upvoted 2 times

mlyu 3 years, 11 months ago

Hi guys, How do we handle the requirement "does not allow Cloud Storage buckets to be globally readable"? seems none of the answers mention about it

upvoted 1 times

rakeshvardan 3 years, 11 months ago

You most likely want to use ACLs if you need to customize access to individual objects within a bucket, since IAM permissions apply to all objects within a bucket. However, you should still use IAM for any access that is common to all objects in a bucket, because this reduces the amount of micro-managing you have to do.

A - as per the above documentation ACLs are needed for specific objects inside bucket.
B - credentials for the service account shouldn't be stored in the app
D - there is no requirement to encrypt the storage data
Hence C seems to be the correct one

upvoted 4 times

Load full discussion...