

🗨️ 👤 **newuser111** 2 years, 1 month ago

Selected Answer: D

D

<https://cloud.google.com/kubernetes-engine/docs/tutorials/autoscaling-metrics#pubsub>

upvoted 2 times

🗨️ 👤 **bossdellacert** 2 years, 3 months ago

Selected Answer: D

This seems relevant

<https://cloud.google.com/kubernetes-engine/docs/tutorials/autoscaling-metrics#pubsub>

even if it uses Deployment + HorizontalPodAutoscaler which is not mentioned in the context of the question/answer

upvoted 2 times

🗨️ 👤 **tycho** 2 years, 4 months ago

I think it wrong for application to be in pod it should be a deployment, and deployment would scale

upvoted 2 times

Question #141

Topic 1

Your company is developing a web-based application. You need to make sure that production deployments are linked to source code commits and are fully auditable. What should you do?

- A. Make sure a developer is tagging the code commit with the date and time of commit.
- B. Make sure a developer is adding a comment to the commit that links to the deployment.
- C. Make the container tag match the source code commit hash.
- D. Make sure the developer is tagging the commits with latest.

🗨️ 👤 **djosani** Highly Voted 🏆 3 years, 3 months ago

Developer shouldn't tag or comment every commit with some specific data, like timestamps or something else. There might be an app version but it's not mentioned. I'd go with C as it's an automated, error-less approach that answers the question.

upvoted 35 times

🗨️ 👤 **Urban_Life** 3 years ago

@Kopper2019- what do you think about ans C?

upvoted 2 times

🗲️ 👤 **victory108** Highly Voted 👍 3 years, 3 months ago

C. Make the container tag match the source code commit hash.

upvoted 16 times

🗲️ 👤 **amxexam** 3 years, 3 months ago

Not sure how the container tag match with the commit will help to audit, can someone explain?

upvoted 2 times

🗲️ 👤 **ynoot** 3 years ago

if you got the commit hash from the container you can check the corresponding commit in the git repository. So the change, that was made and deployed into your environment can be audited.

upvoted 10 times

🗲️ 👤 **Sephetus** Most Recent 🕒 6 months ago

Selected Answer: C

Linking Deployments to Commits: By tagging the container image with the source code commit hash, you create a direct link between the deployed container and the specific state of the source code. This provides a clear and auditable trail from the deployed application back to the exact source code that was used to build it.

Auditability: Using the commit hash as the container tag ensures that each deployment can be traced back to a unique and immutable source code commit. This makes it easy to audit deployments and verify which version of the code is running in production.

upvoted 1 times

🗲️ 👤 **RaviRS** 1 year, 3 months ago

Selected Answer: C

Can't fathom A. This is what ChatGPT says about A - I agree to this.

Option A (tagging with date and time): Using date and time as tags may not be precise enough to identify the exact code version associated with a deployment, especially if multiple commits occurred within the same time window.

upvoted 1 times

🗲️ 👤 **BiddlyBdoyn** 1 year, 6 months ago

Really C should say image?

We have to separate systems: source code repo & container repo.

How do we link the two together? C is the only attempt at solving the problem.

upvoted 1 times

🗲️ 👤 **WFCheong** 1 year, 11 months ago

Selected Answer: C

Agreed with C instead of A with them.

upvoted 1 times

🗲️ 👤 **surajkrishnamurthy** 2 years ago

Selected Answer: C

C is the correct answer

upvoted 1 times

🗲️ 👤 **KumarSelvaraj** 2 years ago

Answer is C

upvoted 1 times

🗲️ 👤 **megumin** 2 years, 1 month ago

Selected Answer: C

C is ok

upvoted 2 times

🗲️ 👤 **Mahmoud_E** 2 years, 2 months ago

Selected Answer: C

C is correct "By design, the Git commit hash is immutable and references a specific version of your software." as per https://cloud.google.com/architecture/best-practices-for-building-containers#tagging_using_the_git_commit_hash

upvoted 4 times

🗨️ **zelck** 2 years, 3 months ago

Selected Answer: C

C is the answer.

https://cloud.google.com/architecture/best-practices-for-building-containers#tagging_using_the_git_commit_hash

You can use this commit hash as a version number for your software, but also as a tag for the Docker image built from this specific version of your software. Doing so makes Docker images traceable: because in this case the image tag is immutable, you instantly know which specific version of your software is running inside a given container.

upvoted 5 times

🗨️ **AzureDP900** 2 years, 5 months ago

Every Git commit with timestamp A doesn't make since. C is right

upvoted 3 times

🗨️ **munnysh** 2 years, 6 months ago

Selected Answer: C

No manual intervention is preferred in automatic deployments. Only automating the container tag to match the commit hash will be fully audit with the help of the scm.

upvoted 4 times

🗨️ **ridyr** 2 years, 7 months ago

Selected Answer: C

From: <https://cloud.google.com/architecture/best-practices-for-building-containers>

Under: Tagging using the Git commit hash (bottom of page almost)

"In this case, a common way of handling version numbers is to use the Git commit SHA-1 hash (or a short version of it) as the version number design, the Git commit hash is immutable and references a specific version of your software.

You can use this commit hash as a version number for your software, but also as a tag for the Docker image built from this specific version of your software. Doing so makes Docker images traceable: because in this case the image tag is immutable, you instantly know which specific version of your software is running inside a given container."

upvoted 7 times

🗨️ **SCVinod** 2 years, 9 months ago

It's got to be A. Option C talks about containers whereas there is no mention of containers in the question.

upvoted 4 times

🗨️ **[Removed]** 2 years, 10 months ago

Selected Answer: C

I got similar question on my exam. Answered C.

upvoted 5 times

🗨️ **Narinder** 2 years, 11 months ago

I think answer is A.

In Git, tag is used to mark release points (v1.0, v2.0 and so on). You can tag the release based on the time stamp and using git show <tag-name> command, you can see the commit detailed history.

Reference: <https://git-scm.com/book/en/v2/Git-Basics-Tagging>

C could be the correct answer for the case if you are going with container based solution which is not mentioned anywhere in the question.

upvoted 5 times

Question #142

Topic 1

An application development team has come to you for advice. They are planning to write and deploy an HTTP(S) API using Go 1.12. The API will have a very unpredictable workload and must remain reliable during peaks in traffic. They want to minimize operational overhead for this application. Which approach should you recommend?

- A. Develop the application with containers, and deploy to Google Kubernetes Engine.
- B. Develop the application for App Engine standard environment.
- C. Use a Managed Instance Group when deploying to Compute Engine.
- D. Develop the application for App Engine flexible environment, using a custom runtime.

  **vladik820** Highly Voted 2 years, 9 months ago

B is ok
upvoted 21 times

  **SweetieS** Highly Voted 2 years, 9 months ago

B is ok.
<https://cloud.google.com/appengine/docs/the-appengine-environments>
upvoted 12 times

  **cugena** 2 years, 9 months ago

Source code is written in specific versions of the supported programming languages:
Python 2.7, Python 3.7, Python 3.8, Python 3.9
Java 8, Java 11
Node.js 10, Node.js 12, Node.js 14, Node.js 16 (preview)
PHP 5.5, PHP 7.2, PHP 7.3, and PHP 7.4
Ruby 2.5, Ruby 2.6, and Ruby 2.7
Go 1.11, Go 1.12, Go 1.13, Go 1.14, Go 1.15, and Go 1.16 (preview)
upvoted 7 times

  **cugena** 2 years, 9 months ago

Intended to run for free or at very low cost, where you pay only for what you need and when you need it. For example, your application can scale to 0 instances when there is no traffic.

Experiences sudden and extreme spikes of traffic which require immediate scaling.
upvoted 4 times

  **nairj** Most Recent 3 months, 1 week ago

Both B and D are okay, however, the need for App engine Flexible environment is not required unless you want to run docker containers, have more control over the instance used and so on, hence in this case B works well.
<https://cloud.google.com/appengine/docs/the-appengine-environments>
upvoted 3 times

  **thewalker** 6 months, 3 weeks ago

Selected Answer: A

A
GKE is much reliable compared to the other options provided here.
upvoted 1 times



  **rakp** 9 months ago

Selected Answer: B

B is correct. It supports Go 1.12, and can handle sudden spikes.
<https://cloud.google.com/appengine/docs/the-appengine-environments>
upvoted 1 times

  **gotcertified** 11 months, 2 weeks ago

Can someone explain why we cannot use AppEngine Flexible environment ?
upvoted 1 times

  **anjanc** 6 months, 1 week ago

Bcs They want to minimize operational overhead for this application
upvoted 1 times

  **edoo** 4 months, 1 week ago

I guess it can't scale down to 0.
upvoted 1 times

🗨️ 👤 **sampon279** 11 months, 3 weeks ago

Selected Answer: B

App engine standard provides go env.
upvoted 1 times

🗨️ 👤 **AugustoKras011111** 1 year, 3 months ago

Selected Answer: B

App Engine Std. Can run this Go version and Scales to 0.
upvoted 1 times

🗨️ 👤 **zerg0** 1 year, 4 months ago

Selected Answer: B

Standard AppEngine Environment supports Go 1.2. The AppEngine can be low cost if no or low traffic. It has free quotas.
upvoted 1 times

🗨️ 👤 **zerg0** 1 year, 4 months ago

Selected Answer: B

AppEngine scales well, only dev effort. No infrastructure. go is supported in the standard distribution.
upvoted 1 times

🗨️ 👤 **NodummyIQ** 1 year, 5 months ago

The answer is A. B option is not correct. It is not recommended to use App Engine Standard environment for an HTTP(S) API with a very unpredictable workload because App Engine Standard environment has certain limitations and constraints that may not be suitable for an API with an unpredictable workload. For example, App Engine Standard environment has a maximum request timeout of 60 seconds, which may not be sufficient for an API with a very unpredictable workload.
upvoted 1 times

🗨️ 👤 **megumin** 1 year, 7 months ago

Selected Answer: B

B is ok
upvoted 1 times

🗨️ 👤 **AzureDP900** 1 year, 11 months ago

B is correct ..<https://cloud.google.com/appengine/docs/the-appengine-environments>

Experiences sudden and extreme spikes of traffic which require immediate scaling.
upvoted 2 times

🗨️ 👤 **munnys** 2 years ago

Selected Answer: B

<https://cloud.google.com/appengine/docs/the-appengine-environments> App engine standard environment support go 1.13 and also handles unpredictable load.
upvoted 3 times

🗨️ 👤 **TitaniumBurger** 2 years, 4 months ago

B. Unpredictable traffic & low overhead.
upvoted 2 times

🗨️ 👤 **tmnd91** 2 years, 5 months ago

Selected Answer: B

App Engine standard has autoscaling out of the box, supports Go 1.12 and can scale down to 0 to save money
upvoted 6 times

🗨️ 👤 **lxgywil** 2 years, 5 months ago

B is ok.
upvoted 1 times

Question #143

Topic 1

Your company is designing its data lake on Google Cloud and wants to develop different ingestion pipelines to collect unstructured data from different sources.

After the data is stored in Google Cloud, it will be processed in several data pipelines to build a recommendation engine for end users on the website. The structure of the data retrieved from the source systems can change at any time. The data must be stored exactly as it was retrieved for reprocessing purposes in case the data structure is incompatible with the current processing pipelines. You need to design an architecture to support the use case after you retrieve the data. What should you do?

- A. Send the data through the processing pipeline, and then store the processed data in a BigQuery table for reprocessing.
- B. Store the data in a BigQuery table. Design the processing pipelines to retrieve the data from the table.
- C. Send the data through the processing pipeline, and then store the processed data in a Cloud Storage bucket for reprocessing.
- D. Store the data in a Cloud Storage bucket. Design the processing pipelines to retrieve the data from the bucket.

 **vladik820** Highly Voted 2 years, 9 months ago

D is ok

The data needs to be stored as it is retrieved. This would mean that any processing should be done after it is stored.

upvoted 28 times

 **MaxNRG** Highly Voted 2 years, 8 months ago

D, store RAW unstructured data as-is in Cloud Storage, and then define how to process it.

Classical Data Lake ELT (Extract -> Load -> Transform)

upvoted 6 times

 **Gino17m** Most Recent 1 month, 3 weeks ago


Selected Answer: D

D

Unstructured data - GCS

Data stored exactly as it was retrieved - store before processing

upvoted 1 times

 **anjanc** 6 months, 1 week ago

Key word is "The data must be stored exactly as it was retrieved for reprocessing purposes in case the data structure is incompatible with the current processing pipelines." and hence D


upvoted 1 times

 **devnul** 10 months ago

D. It aligns with an example in the Cloud Architecture Framework

<https://cloud.google.com/architecture/big-data-analytics/analytics-lakehouse>

upvoted 2 times

 **BeCalm** 1 year, 3 months ago

What is the point of data being in a lake and then being dumped into GCS without processing. What purpose is served with GCS being a copy lake?

upvoted 1 times

 **jlambdan** 1 year, 2 months ago

here gcs is the lake. Not a copy.

The data warehouse will be what comes out of the pipelines.

upvoted 3 times

- 
 **megumin** 1 year, 7 months ago
 Selected Answer: D
 D is ok
 upvoted 2 times
- 
 **jmbiancof** 1 year, 7 months ago
 D is ok
 upvoted 2 times
- 
 **Nirca** 1 year, 9 months ago
 Selected Answer: D
 D is ok
 The data needs to be stored as it is retrieved. This would mean that any processing should be done after it is stored in GCS.
 upvoted 2 times
- 
 **AzureDP900** 1 year, 11 months ago
 storing and retrieving data in cloud storage solve the purpose of this use case. D is perfect answer.
 upvoted 1 times
- 
 **snwbr** 2 years, 2 months ago
 Although... wouldn't be Bigtable or Datastore better than GCS?
 upvoted 1 times
- 
 **wykofo** 1 year, 11 months ago
 Both BigTable and DataStore are NoSQL Databases, qns mentioned that data structure may change anytime
 upvoted 2 times
- 
 **[Removed]** 2 years, 4 months ago
 Selected Answer: D
 I got similar question on my exam. Answered D.
 upvoted 4 times
- 
 **technodev** 2 years, 5 months ago
 Got this question in my exam, answered D
 upvoted 4 times
- 
 **lxgywil** 2 years, 5 months ago
 D is ok
 upvoted 2 times
- 
 **vincy2202** 2 years, 6 months ago
 D is the correct answer
 upvoted 1 times
- 
 **pakilodi** 2 years, 6 months ago
 Selected Answer: D
 D is correct
 upvoted 1 times
- 
 **TheCloudBoy77** 2 years, 7 months ago
 D - Data must be stored as it is before and after so use Cloud storage and then build pipelines as needed.
 upvoted 2 times

Question #144

Topic 1

You are responsible for the Google Cloud environment in your company. Multiple departments need access to their own projects, and the members within each department will have the same project responsibilities. You want to structure your Google Cloud environment for minimal maintenance and maximum overview of

maintenance and maximum overview of

IAM permissions as each department's projects start and end. You want to follow Google-recommended practices. What should you do?

- A. Grant all department members the required IAM permissions for their respective projects.
- B. Create a Google Group per department and add all department members to their respective groups. Create a folder per department and grant the respective group the required IAM permissions at the folder level. Add the projects under the respective folders.
- C. Create a folder per department and grant the respective members of the department the required IAM permissions at the folder level. Structure all projects for each department under the respective folders.
- D. Create a Google Group per department and add all department members to their respective groups. Grant each group the required IAM permissions for their respective projects.

🗲️ 👤 **Manh** Highly Voted 2 years, 9 months ago

it's B

upvoted 16 times

🗲️ 👤 **victory108** Highly Voted 2 years, 9 months ago

B. Create a Google Group per department and add all department members to their respective groups. Create a folder per department and grant the respective group the required IAM permissions at the folder level. Add the projects under the respective folders.

upvoted 10 times

🗲️ 👤 **afsarkhan** Most Recent 1 month, 1 week ago

it's B

upvoted 1 times

🗲️ 👤 **megumin** 1 year, 7 months ago

Selected Answer: B

B is ok

upvoted 1 times

🗲️ 👤 **zelck** 1 year, 9 months ago

Selected Answer: B

B is the answer.

<https://cloud.google.com/resource-manager/docs/access-control-folders#best-practices-folders-iam>
Use groups whenever possible to manage principals.

<https://cloud.google.com/resource-manager/docs/creating-managing-folders>

A folder can contain projects, other folders, or a combination of both. Organizations can use folders to group projects under the organization node in a hierarchy. For example, your organization might contain multiple departments, each with its own set of Google Cloud resources. Folders allow you to group these resources on a per-department basis.

upvoted 4 times

🗲️ 👤 **Nirca** 1 year, 9 months ago

Selected Answer: B

B is most appropriate for the use case and principle of least privilege.

upvoted 1 times

🗲️ 👤 **AzureDP900** 1 year, 11 months ago

B is most appropriate for the use case and principle of least privilege.

upvoted 1 times

🗲️ 👤 **coutcin** 2 years, 1 month ago

Selected Answer: B









B is correct

upvoted 1 times









🗲️ 👤 **lxgywil** 2 years, 5 months ago

B is ok

upvoted 1 times

- 
 **edilramos** 2 years, 6 months ago
 B is ideal for minimal maintenance and maximum overview of IAM permissions as each department's projects start and end. Manage the users inside Groups will turn it easier.
 upvoted 5 times
- 
 **anjuagrawal** 2 years, 6 months ago
 Voted B
 upvoted 1 times
- 
 **vincy2202** 2 years, 6 months ago
 B is the correct answer
 upvoted 1 times
- 
 **nqthien041292** 2 years, 6 months ago

Selected Answer: B

 Vote B
 upvoted 2 times
- 
 **danielfootc** 2 years, 8 months ago
 I would select B.
 upvoted 2 times
- 
 **AnilKr** 2 years, 8 months ago
 B is correct, folder restructure per department and IAM permission for Group is recommended.
 upvoted 4 times
- 
 **Sonu_xyz** 2 years, 9 months ago
 Answer is B
 upvoted 2 times
- 
 **diaga2** 2 years, 9 months ago
 Yes, B
 upvoted 4 times

Question #145

Topic 1

Your company has an application running as a Deployment in a Google Kubernetes Engine (GKE) cluster. You have separate clusters for development, staging, and production. You have discovered that the team is able to deploy a Docker image to the production cluster without first testing the deployment in development and then staging. You want to allow the team to have autonomy but want to prevent this from happening. You want a Google Cloud solution that can be implemented quickly with minimal effort. What should you do?

- A. Configure a Kubernetes lifecycle hook to prevent the container from starting if it is not approved for usage in the given environment.
- B. Implement a corporate policy to prevent teams from deploying Docker images to an environment unless the Docker image was tested in an earlier environment.
- C. Configure binary authorization policies for the development, staging, and production clusters. Create attestations as part of the continuous integration pipeline.
- D. Create a Kubernetes admissions controller to prevent the container from starting if it is not approved for usage in the given environment.

- 
 **diaga2**

Highly Voted

 1 year, 9 months ago
 C is s fine.

upvoted 15 times

[Removed] Highly Voted 1 year, 4 months ago

Selected Answer: C

I got similar question on my exam. Answered C.

upvoted 11 times

Deb2293 Most Recent 3 months, 1 week ago

Selected Answer: C

C it is

upvoted 2 times

omermahgoub 5 months, 3 weeks ago

A good option for quickly implementing a solution to prevent deployments to the production cluster without first testing in development and staging would be to configure binary authorization policies for the development, staging, and production clusters. You can then create attestations as part of the continuous integration pipeline.

Option C, "Configure binary authorization policies for the development, staging, and production clusters. Create attestations as part of the continuous integration pipeline," would be the correct choice for this scenario.

Binary authorization is a feature of Google Kubernetes Engine that allows you to enforce policies on the images that are deployed to your clusters. By configuring binary authorization policies for the development, staging, and production clusters, you can ensure that only images that have been attested by an authorized entity are allowed to be deployed to those clusters. You can create the attestations as part of the continuous integration pipeline, which will allow you to verify that the image has been tested before it is deployed to the next environment.

upvoted 10 times

omermahgoub 5 months, 3 weeks ago

Option A, "Configure a Kubernetes lifecycle hook to prevent the container from starting if it is not approved for usage in the given environment," would not be a good choice because it would not prevent the deployment of the container to the cluster in the first place.

Option D, "Create a Kubernetes admissions controller to prevent the container from starting if it is not approved for usage in the given environment," would also not be a good choice because it would not prevent the deployment of the container to the cluster in the first place.

Option B, "Implement a corporate policy to prevent teams from deploying Docker images to an environment unless the Docker image was tested in an earlier environment," would be a good option, but it would not be as effective as using binary authorization policies, as it would rely on the team following the policy rather than enforcing it automatically.

upvoted 1 times

megumin 7 months, 1 week ago

Selected Answer: C

C is ok

upvoted 1 times

Thornadoo 10 months, 3 weeks ago

Why not A? Need something to be implemented quickly is what the q asks.

upvoted 1 times

AzureDP900 11 months, 2 weeks ago

C is right..

Binary Authorization implements a policy model, where a policy is a set of rules that governs the deployment of container images. Rules in a policy provide specific criteria that an image must satisfy before it can be deployed.

For more information about the Binary Authorization policy model and other concepts, see Key concepts.

upvoted 4 times

AzureDP900 11 months, 2 weeks ago

https://cloud.google.com/binary-authorization/docs/overview#policy_model

upvoted 3 times

yogi_508 1 year, 6 months ago

where the case study questions are available in this website?

upvoted 1 times

- 🗲️ 👤 **vincy2202** 1 year, 6 months ago
C is the correct answer
<https://cloud.google.com/binary-authorization/docs/overview>
upvoted 6 times
- 🗲️ 👤 **Jimjiang** 1 year, 7 months ago
C is fine
upvoted 1 times
- 🗲️ 👤 **danielfootc** 1 year, 8 months ago
I think C is the correct answer.
upvoted 1 times
- 🗲️ 👤 **AnilKr** 1 year, 8 months ago
C is correct, binary authorization is the solution.
upvoted 2 times
- 🗲️ 👤 **victory108** 1 year, 9 months ago
C. Configure binary authorization policies for the development, staging, and production clusters. Create attestations as part of the continuous integration pipeline.
upvoted 2 times
- 🗲️ 👤 **serious_user** 1 year, 9 months ago
C is ok
upvoted 2 times
- 🗲️ 👤 **vladik820** 1 year, 9 months ago
C is ok
upvoted 2 times
- 🗲️ 👤 **SweetieS** 1 year, 9 months ago
Sorry, it's C : Configure binary authorization policies for the development, staging, and production clusters. Create attestations as part of the continuous integration pipeline.
upvoted 3 times
- 🗲️ 👤 **SweetieS** 1 year, 9 months ago
D is ok.
<https://cloud.google.com/binary-authorization/docs/overview>
upvoted 1 times
- 🗲️ 👤 **cugena** 1 year, 9 months ago
You meant C I guess
upvoted 1 times

Question #146

Topic 1

Your company wants to migrate their 10-TB on-premises database export into Cloud Storage. You want to minimize the time it takes to complete this activity, the overall cost, and database load. The bandwidth between the on-premises environment and Google Cloud is 1 Gbps. You want to follow Google-recommended practices. What should you do?

- A. Develop a Dataflow job to read data directly from the database and write it into Cloud Storage.
- B. Use the Data Transfer appliance to perform an offline migration.
- C. Use a commercial partner ETL solution to extract the data from the on-premises database and upload it into Cloud Storage.

D. Compress the data and upload it with gsutil -m to enable multi-threaded copy.

  **pr2web** Highly Voted 3 years, 3 months ago

This is pretty simple.

Time to transfer using Transfer Appliance: 1-3 weeks (I've used it twice and had a 2-3 week turnaround total)

Time to transfer using 1Gbps : 30 hours (<https://cloud.google.com/architecture/migration-to-google-cloud-transferring-your-large-datasets>)



Answer is D, using gsutil

upvoted 100 times

  **mickeythecraycray** 2 years, 8 months ago



Will that not increase the Database load?, one of the requirement is to reduce the load of the DB during this operation.

upvoted 3 times

  **Aiffone** 2 years, 11 months ago



If I can do it in 30hrs, why choose 1 week? i'd go with B

upvoted 3 times

  **Aiffone** 2 years, 11 months ago



I mean I'd go with A rather...questions says to spend minimum time and we have 1Gbps to do 10Tb in 30hrs

upvoted 2 times

  **Aiffone** 2 years, 11 months ago



Transfer appliance -A

upvoted 2 times

  **Deb2293** 1 year, 9 months ago

Go home you are drunk

upvoted 5 times

  **joe2211** 3 years ago

Not about time but "Google-recommended practices"



upvoted 8 times

  **MikeB19** 3 years, 3 months ago

This is the correct article to support this question but the article proves the transfer appliance is the correct answer. Right below the transfer calc chart is recommended amount of data for gsutil. Gsutil should be used for data transfer under 1 tb

"Your private data center to Google Cloud Enough bandwidth to meet your project deadline for less than 1 TB of data gsutil"

upvoted 3 times

  **gingerbeer** Highly Voted 3 years, 2 months ago

No perfect answer as B and D both have flaws. B is time latency as transfer appliance usually takes weeks; D gsutil applies for less than 1TB. The answer should be storage transfer service for on-premises data, which is not available here.

If have to choose one I go for B

upvoted 21 times

  **RitwickKumar** 2 years, 4 months ago

Storage transfer service is for online data. It can't serve the purpose if you don't have the connectivity established between on prem and g. Which is what we can't assume ourselves in this question.

upvoted 1 times

  **T12344223** Most Recent 1 week, 5 days ago

Selected Answer: B

B and D sounds feasible but gsutil is not recommended any more so definitely B.

<https://cloud.google.com/storage/docs/gsutil>

However, I'm not sure if D is replaced with gcloud storage cp.

upvoted 1 times

ccpmad 6 months, 1 week ago

Selected Answer: B

D says compress data, ¿in a single file? it will be more than the limit 5 TB of gsutil, so it is B.
upvoted 1 times

huuthanhdlv 6 months, 4 weeks ago

I think the answer is B.

The main consideration is between B and D. Just thinking if they want the answer to be online transfer, they should have added Online Transfer Service instead of gsutils. Just guessing Google must want us to choose B :)

upvoted 1 times

seetpt 7 months ago

Selected Answer: B

B fo sho

upvoted 1 times

afsarkhan 7 months, 1 week ago

D will be most cost effective where as B will incur cost (question asking to consider cost effective solution as well) so D is my answer
upvoted 2 times

MFay 7 months, 3 weeks ago

Selected Answer: B

Option B (Data Transfer appliance) is the best choice for efficient and cost-effective data migration while minimizing database load and transfer time. This solution bypasses network limitations and reduces the impact on the on-premises environment, making it ideal for migrating large datasets to the cloud.

upvoted 2 times

gbemimatti 7 months, 4 weeks ago

Selected Answer: B

Compressing the data and uploading it with gsutil -m can be a good optimization for your transfer, but it has limitations to consider:

Compression Overhead: While compressing the data can reduce upload size and potentially speed up transfer, the compression and decompression processes themselves take time and resources. Depending on your data type, the benefit of reduced size might be offset by the processing overhead.

Transfer Appliance: The recommended approach with the Transfer Appliance already utilizes parallel transfers for faster uploads, potentially making gsutil -m less impactful.

I will go with B

upvoted 2 times

gbemimatti 7 months, 4 weeks ago

Compressing the data and uploading it with gsutil -m can be a good optimization for your transfer, but it has limitations to consider:

Compression Overhead: While compressing the data can reduce upload size and potentially speed up transfer, the compression and decompression processes themselves take time and resources. Depending on your data type, the benefit of reduced size might be offset by the processing overhead.

Transfer Appliance: The recommended approach with the Transfer Appliance already utilizes parallel transfers for faster uploads, potentially making gsutil -m less impactful.

I will go with B

upvoted 1 times

342f1c6 8 months, 3 weeks ago

Selected Answer: D

with 1 Gbps it will take only 30 hrs so best option is D
upvoted 2 times

RajSelvaraj 9 months, 1 week ago

Option B and D are most feasible options

Option B will be okay if the size of the data is too huge

Option D will be good for a few TBs of data. I am assuming 10 TB will fit in this case.

<https://cloud.google.com/blog/topics/developers-practitioners/how-transfer-your-data-google-cloud>

upvoted 1 times

🗨️ 👤 **madcloud32** 9 months, 3 weeks ago

Selected Answer: B

Answer B. Cp limit is 5 TB max
upvoted 3 times

🗨️ 👤 **OrangeTiger** 10 months, 2 weeks ago

Selected Answer: D

I chose D.
According to the link below, 10TB of data can be transferred in 30h. The light blue area is the acceptable line for online transfer.
https://cloud.google.com/architecture/migration-to-google-cloud-transferring-your-large-datasets?hl=ja#online_versus_offline_transfer
upvoted 2 times

🗨️ 👤 **ccpmad** 6 months, 1 week ago

D says compress data, in a single file? it will be more than the limit 5 TB of gsutil
upvoted 1 times

🗨️ 👤 **Pime13** 10 months, 3 weeks ago

Selected Answer: D

<https://cloud.google.com/architecture/migration-to-google-cloud-transferring-your-large-datasets>
upvoted 2 times

🗨️ 👤 **Pime13** 10 months, 3 weeks ago

Selected Answer: D

https://cloud.google.com/architecture/migration-to-google-cloud-transferring-your-large-datasets#online_versus_offline_transfer
upvoted 1 times

🗨️ 👤 **didek1986** 11 months ago

Selected Answer: B

It is B
upvoted 2 times

Question #147

Topic 1

Your company has an enterprise application running on Compute Engine that requires high availability and high performance. The application has been deployed on two instances in two zones in the same region in active-passive mode. The application writes data to a persistent disk. In the case of a single zone outage, that data should be immediately made available to the other instance in the other zone. You want to maximize performance while minimizing downtime and data loss.

What should you do?

- A. 1. Attach a persistent SSD disk to the first instance. 2. Create a snapshot every hour. 3. In case of a zone outage, recreate a persistent SSD disk in the second instance where data is coming from the created snapshot.
- B. 1. Create a Cloud Storage bucket. 2. Mount the bucket into the first instance with gcs-fuse. 3. In case of a zone outage, mount the Cloud Storage bucket to the second instance with gcs-fuse.
- C. 1. Attach a regional SSD persistent disk to the first instance. 2. In case of a zone outage, force-attach the disk to the other instance.
- D. 1. Attach a local SSD to the first instance disk. 2. Execute an rsync command every hour where the target is a persistent SSD disk attached

to the second instance. 3. In case of a zone outage, use the second instance.

🗲️ 👤 **juma_david** Highly Voted 👍 3 years, 3 months ago

Answer C

<https://cloud.google.com/compute/docs/disks/repd-failover>

upvoted 43 times

🗲️ 👤 **[Removed]** Highly Voted 👍 3 years, 1 month ago

C is right answer.

C. 1. Attach a regional SSD persistent disk to the first instance. 2. In case of a zone outage, force-attach the disk to the other instance. gcs-fuse is slower than of regional SSD PD.

**** Admin: You need to correct lots of questions. Some of the marked answers are nonsense, these must be revisited based on experts comments.

upvoted 42 times

🗲️ 👤 **Sephehus** Most Recent 🕒 6 months ago

Selected Answer: B

BigQuery cannot use customer supplied KMs keys only customer managed keys. The other options add too much complexity to the problem

upvoted 1 times

🗲️ 👤 **Sephehus** 6 months ago

Somehow I commented on the wrong answer please delete.

upvoted 1 times

🗲️ 👤 **afsarkhan** 7 months, 1 week ago

Selected Answer: C

C makes a better sense than any other option

upvoted 1 times

🗲️ 👤 **dija123** 8 months ago

Selected Answer: C

Agree with Regional SSD persistent

upvoted 1 times

🗲️ 👤 **[Removed]** 11 months, 3 weeks ago

C

In the event that the primary zone fails, you can fail over your regional Persistent Disk volume to a VM in another zone by using a force-attach operation. When there's a failure in the primary zone, you might not be able to detach the disk from the VM because the VM can't be reached perform the detach operation. Force-attach operation lets you attach a regional Persistent Disk volume to a VM even if that volume is attached to another VM

upvoted 2 times

🗲️ 👤 **RaviRS** 1 year, 3 months ago

Selected Answer: C

I don't get why B has been given as answer... GCS-FUSE brings in additional complexity and it also doesn't serve the same purpose as effectively as regional SSD does.

upvoted 1 times

🗲️ 👤 **DS2023** 1 year, 6 months ago

Selected Answer: C

Ans: C, please check - <https://cloud.google.com/compute/docs/disks/high-availability-regional-persistent-disk>

upvoted 1 times

🗲️ 👤 **dbsmk** 1 year, 8 months ago

<https://cloud.google.com/compute/docs/disks/repd-failover>

Seems C is correct

upvoted 1 times

🗄️ 👤 **examch** 1 year, 11 months ago

C is the correct answer,

https://cloud.google.com/compute/docs/disks/repd-failover#zonal_failures

upvoted 1 times

🗄️ 👤 **surajkrishnamurthy** 2 years ago

Selected Answer: C

Answer is C

upvoted 1 times

🗄️ 👤 **zetalexg** 2 years ago

Admins please take some time and redo the answers, put them to match at least the most voted ones, would help a lot.

upvoted 7 times

🗄️ 👤 **ashrafh** 2 years, 1 month ago

Selected Answer: C

Regional persistent disk is a storage option that provides synchronous replication of data between two zones in a region. Regional persistent disks can be a good building block to use when you implement HA services in Compute Engine.

upvoted 2 times

🗄️ 👤 **megumin** 2 years, 1 month ago

Selected Answer: C

C is ok

upvoted 1 times

🗄️ 👤 **Mahmoud_E** 2 years, 2 months ago

Selected Answer: C

C is the right answer

upvoted 1 times

🗄️ 👤 **Nirca** 2 years, 3 months ago

Selected Answer: C

You want to maximize performance while minimizing downtime and data loss

upvoted 1 times

🗄️ 👤 **RitwickKumar** 2 years, 4 months ago

Selected Answer: C

Inline with the current architecture itself "The application writes data to a persistent disk."

upvoted 1 times

Question #148

Topic 1

You are designing a Data Warehouse on Google Cloud and want to store sensitive data in BigQuery. Your company requires you to generate the encryption keys outside of Google Cloud. You need to implement a solution. What should you do?

- A. Generate a new key in Cloud Key Management Service (Cloud KMS). Store all data in Cloud Storage using the customer-managed key option and select the created key. Set up a Dataflow pipeline to decrypt the data and to store it in a new BigQuery dataset.
- B. Generate a new key in Cloud KMS. Create a dataset in BigQuery using the customer-managed key option and select the created key.
- C. Import a key in Cloud KMS. Store all data in Cloud Storage using the customer-managed key option and select the created key. Set up a Dataflow pipeline to decrypt the data and to store it in a new BigQuery dataset.
- D. Import a key in Cloud KMS. Create a dataset in BigQuery using the customer-supplied key option and select the created key.

🗄️ 👤 **alexandercamacho** **Highly Voted** 2 years, 3 months ago

alexandercamacho (engraving) 2 years, 3 months ago

Selected Answer: D

The answer is easy. It says keys must be left outside of Google Cloud.

This automatically eliminates A / B.

Now the C option says decrypts before storing it in BigQuery which the point is to encrypt the data while been in BigQuery, D is the only possible answer.

upvoted 17 times

Sephethus 6 months ago

Except that BigQuery doesn't support customer supplied keys outside of GCP.

upvoted 2 times

Sweeties (Highly Voted) 3 years, 3 months ago

D is OK

upvoted 17 times

SR23222 1 year, 6 months ago

But CSEK is not supported in BigQuery

upvoted 2 times

[Removed] 1 year, 3 months ago

It is a tricky distinction because of the term collision.

However, "import key to KMS" does not mean CSEK.

CSEK does not get imported or stored in KMS at all. CSEK "customer supplied" is per-transaction uploaded by every API call by the user/client (no KMS).

This situation "customer supplied" means created from non-GCP KMS (could be on-prem or EKM). Once a key is imported to KMS it is treated as CMEK. The API client calling GCS doesn't need to upload the key. It lives in KMS. That is not the same "per-transaction" upload as CSEK.

upvoted 2 times

[Removed] 1 year, 3 months ago

I mean after being imported to KMS your key is handled like a CMEK and available to BQ service.

upvoted 1 times

25lion52 (Most Recent) 2 months, 3 weeks ago

Selected Answer: D

C - won't encrypt data in BQ with customer key.

A,B - you will generate key inside the GCP (what is also wrong by requirements)

D - looks good, but say to select CSEK... but after importing the key to KMS it becomes a customer-managed.

I would select the D

upvoted 1 times

Sephethus 6 months ago

Selected Answer: B

The answer cannot be D since BigQuery does not support customer provided keys, only customer managed keys generated in Cloud KMS. S is the only viable option that doesn't add complexity.

upvoted 1 times

Sephethus 6 months ago

It cannot be D, BigQuery does not support customer supplied KMS keys, only customer managed keys, B.

upvoted 1 times

odacir 1 year, 1 month ago

Selected Answer: D

<https://cloud.google.com/bigquery/docs/customer-managed-encryption>

upvoted 1 times

thewalker 1 year, 1 month ago

A, B, C are ruled out as they say Customer Managed keys.

Hence, D.

upvoted 2 times

🗨️ 👤 **devnul** 1 year, 3 months ago

GCP docu says "BigQuery and BigLake tables don't support Customer-Supplied Encryption Keys (CSEK)."

However, I just tested it and it worked:

1. Create Key

`openssl rand 32 > ./key2`

2. Import into KMS

`gcloud kms keys versions import --import-job csek1 --location us-west1 --keyring csek --key csek --algorithm google-symmetric-encryption target-key-file ./key2`

3. In Cloud Console: select the key when creating a new data set and table in BigQuery

upvoted 4 times

🗨️ 👤 **[Removed]** 1 year, 3 months ago

Right, term collision with "customer supplied" key. However, "import key to KMS" does not mean CSEK.

upvoted 2 times

🗨️ 👤 **jits1984** 1 year, 8 months ago

Selected Answer: C

C - as BigQuery doesn't support Customer Supplier Keys.

upvoted 2 times

🗨️ 👤 **n_nana** 1 year, 9 months ago

Selected Answer: C

BigQuery doesn't support CSEK
upvoted 1 times

🗨️ 👤 **medi01** 1 year, 8 months ago

BQ DOES support CSEK.
upvoted 1 times

🗨️ 👤 **n_nana** 1 year, 9 months ago

Sorry even C is not correct, why to store the data in bq without encryption.
data should be passed encrypted from storage to bq.
then Answer is B
upvoted 1 times

🗨️ 👤 **A21325412** 1 year, 1 month ago

I would go with C.

<https://cloud.google.com/bigquery/docs/customer-managed-encryption>
Read that document in the link carefully.

1st paragraph: "By Default, BigQuery encrypts your content stored at rest";
1st bullet point, 2nd paragraph under the [Before you Begin] section: "BigQuery and BigLake tables don't support Customer-Supplied Encryption Keys (CSEK)"

There is also a difference between CMEK and CSEK.
CMEK: you can create and manage a key using Cloud KMS;
CSEK: you specify the contents of the key;

Ref for CMEK vs CSEK:

<https://cloud.google.com/sql/docs/mysql/cmek#:~:text=Note%3A%20Customer%2Dmanaged%20encryption%20keys,specific%20resources%20across%20Google%20Cloud.>

upvoted 1 times

🗨️ 👤 **A21325412** 1 year, 1 month ago

Even though I'll chose C for the answer over D, because of the terminology in "BQ using customer-supplied key", I have an issue with this:

To me it does not make any sense.

The data is being encrypted by some key say K1 to store in Cloud Storage, then Decrypted, to be Re-Encrypted (automatically by some K2 [a google created key]) by BigQuery when being stored. This negates the use of K1 on your Data Storage in BigQuery.

It makes no sense. If someone sees this differently, I'd love to hear it. Thanks.

upvoted 1 times

🗨️ 👤 **nandoD** 1 year, 7 months ago

If you want to control encryption yourself, you can use customer-managed encryption keys (CMEK) for BigQuery.
<https://cloud.google.com/bigquery/docs/customer-managed-encryption>

upvoted 1 times

🗨️ 👤 **SR23222** 1 year, 6 months ago

There is a difference between customer managed and customer supplied. Link that you have shared talks about customer managed and not customer supplied
upvoted 1 times

🗨️ 👤 **AugustoKras011111** 1 year, 9 months ago

Selected Answer: D

Key work: "keys outside of Google Cloud" so you have to import the key. between C and D I go with D.
upvoted 1 times

  **smachnio** 1 year, 11 months ago

Selected Answer: D

D is correct. I had this question on the exam toaday and I go with D.

Explanation is - Generate the key outside the GCP so C and D are correct.

"Set up a Dataflow pipeline to decrypt the data and to store it in a new BigQuery dataset" is not correct becuae it means that data exist on C what is not correct. Only D is correct.

upvoted 2 times

  **examch** 1 year, 11 months ago



Selected Answer: C

Yes, BigQuery and BigLake tables don't support Customer-Supplied Encryption Keys (CSEK). Answer must be either A or C, since the say generate key outside Google Cloud, import the key, hence I go for the answer C.

https://cloud.google.com/bigquery/docs/customer-managed-encryption#before_you_begin



<https://cloud.google.com/kms/docs/importing-a-key>

upvoted 2 times

  **NodummyIQ** 1 year, 11 months ago

Answer D is incorrect because BigQuery does not support the use of customer-supplied keys to encrypt data at rest. Instead, you can use customer-managed encryption keys in Cloud KMS to encrypt the data in BigQuery. To do this, you can either generate a new key in Cloud KM (answer A) or import an existing key (answer C). Once you have a key in Cloud KMS, you can create a BigQuery dataset and select the key as customer-managed key for the dataset. This will enable BigQuery to use the key to encrypt the data in the dataset.

upvoted 4 times

  **examch** 1 year, 11 months ago

Yes, BigQuery and BigLake tables don't support Customer-Supplied Encryption Keys (CSEK). Answer must be either A or C, since the say generate key outside Google Cloud, import the key, hence I go for the answer C.

upvoted 1 times

Question #149

Topic 1

Your organization has stored sensitive data in a Cloud Storage bucket. For regulatory reasons, your company must be able to rotate the encryption key used to encrypt the data in the bucket. The data will be processed in Dataproc. You want to follow Google-recommended practices for security. What should you do?

- A. Create a key with Cloud Key Management Service (KMS). Encrypt the data using the encrypt method of Cloud KMS.
- B. Create a key with Cloud Key Management Service (KMS). Set the encryption key on the bucket to the Cloud KMS key.
- C. Generate a GPG key pair. Encrypt the data using the GPG key. Upload the encrypted data to the bucket.
- D. Generate an AES-256 encryption key. Encrypt the data in the bucket using the customer-supplied encryption keys feature.

  **victory108** Highly Voted 2 years, 9 months ago

B. Create a key with Cloud Key Management Service (KMS). Set the encryption key on the bucket to the Cloud KMS key.

upvoted 32 times

  **SweetieS** Highly Voted 2 years, 9 months ago

B is OK

<https://cloud.google.com/storage/docs/encryption/using-customer-managed-keys#add-object-key>

upvoted 9 times

  **Pime13** Most Recent 4 months, 2 weeks ago

Selected Answer: B

<https://cloud.google.com/storage/docs/encryption/customer-managed-keys#key-rotation>

upvoted 1 times

  **Roro_Brother** 6 months, 1 week ago

Selected Answer: B

It's B, off course

upvoted 1 times