DUMPSHQ

Free Questions for Professional-Cloud-DevOps-Engineer by dumpshq

Shared by Mann on 12-12-2023

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You work for a global organization and are running a monolithic application on Compute Engine You need to select the machine type for the application to use that optimizes CPU utilization by using the fewest number of steps You want to use historical system metncs to identify the machine type for the application to use You want to follow Google-recommended practices What should you do?

Options:

A- Use the Recommender API and apply the suggested recommendations

B- Create an Agent Policy to automatically install Ops Agent in all VMs

C- Install the Ops Agent in a fleet of VMs by using the gcloud CLI

D- Review the Cloud Monitoring dashboard for the VM and choose the machine type with the lowest CPU utilization

Answer:

A

Explanation:

The best option for selecting the machine type for the application to use that optimizes CPU utilization by using the fewest number of steps is to use the Recommender API and apply the suggested recommendations. The Recommender API is a service that provides recommendations for optimizing your Google Cloud resources, such as Compute Engine instances, disks, and firewalls. You can use the Recommender API to get recommendations for changing the machine type of your Compute Engine instances based on historical system metrics, such as CPU utilization. You can also apply the suggested recommendations by using the Recommender API or Cloud Console. This way, you can optimize CPU utilization by using the most suitable machine type for your application with minimal effort.

Question 2

Question Type: MultipleChoice

You are reviewing your deployment pipeline in Google Cloud Deploy You must reduce toil in the pipeline and you want to minimize the amount of time it takes to complete an end-to-end deployment What should you do?

Choose 2 answers

Options:

A- Create a trigger to notify the required team to complete the next step when manual intervention is required

B- Divide the automation steps into smaller tasks

C- Use a script to automate the creation of the deployment pipeline in Google Cloud Deploy

D- Add more engineers to finish the manual steps.

E- Automate promotion approvals from the development environment to the test environment

Answer:

A, E

Explanation:

The best options for reducing toil in the pipeline and minimizing the amount of time it takes to complete an end-to-end deployment are to create a trigger to notify the required team to complete the next step when manual intervention is required and to automate promotion approvals from the development environment to the test environment. A trigger is a resource that initiates a deployment when an event occurs, such as a code change, a schedule, or a manual request. You can create a trigger to notify the required team to complete the next step when manual intervention is required by using Cloud Build or Cloud Functions. This way, you can reduce the waiting time and human errors in the pipeline. A promotion approval is a process that allows you to approve or reject a deployment from one environment to another, such as from development to test. You can automate promotion approvals from the development environment to the test environment by using Google Cloud Deploy or Cloud Build. This way, you can speed up the deployment process and avoid manual steps.

Question 3

Question Type: MultipleChoice

Your team is building a service that performs compute-heavy processing on batches of data The data is processed faster based on the speed and number of CPUs on the machine These batches of data vary in size and may arrive at any time from multiple third-party sources You need to ensure that third parties are able to upload their data securely. You want to minimize costs while ensuring that the data is processed as quickly as possible What should you do?

Options:

A- * Provide a secure file transfer protocol (SFTP) server on a Compute Engine instance so that third parties can upload batches of data and provide appropriate credentials to the server
* Create a Cloud Function with a google.storage, object, finalize Cloud Storage trigger Write code so that the function can scale up a Compute Engine autoscaling managed instance group
* Use an image pre-loaded with the data processing software that terminates the instances when processing completes

B- * Provide a Cloud Storage bucket so that third parties can upload batches of data, and provide appropriate Identity and Access Management (1AM) access to the bucket
* Use a standard Google Kubernetes Engine (GKE) cluster and maintain two services one that processes the batches of data and one that monitors Cloud Storage for new batches of data
* Stop the processing service when there are no batches of data to process

C- * Provide a Cloud Storage bucket so that third parties can upload batches of data, and provide appropriate identity and Access Management (1AM) access to the bucket
* Create a Cloud Function with a google, storage, object .finalise Cloud Storage trigger Write code so that the function can scale up a Compute Engine autoscaling managed instance group

* Use an image pre-loaded with the data processing software that terminates the instances when processing completes

D- * Provide a Cloud Storage bucket so that third parties can upload batches of data, and provide appropriate Identity and Access Management (1AM) access to the bucket
* Use Cloud Monitoring to detect new batches of data in the bucket and trigger a Cloud Function that processes the data
* Set a Cloud Function to use the largest CPU possible to minimize the runtime of the processing

Answer:

C

Explanation:

The best option for ensuring that third parties are able to upload their data securely and minimizing costs while ensuring that the data is processed as quickly as possible is to provide a Cloud Storage bucket so that third parties can upload batches of data, and provide appropriate Identity and Access Management (IAM) access to the bucket; create a Cloud Function with a google.storage.object.finalize Cloud Storage trigger; write code so that the function can scale up a Compute Engine autoscaling managed instance group; use an image pre-loaded with the data processing software that terminates the instances when processing completes. A Cloud Storage bucket is a resource that allows you to store and access data in Google Cloud. You can provide a Cloud Storage bucket so that third parties can upload batches of data securely and conveniently. You can also provide appropriate IAM access to the bucket by using roles and policies to control who can read or write data to the bucket. A Cloud Function is a serverless function that executes code in response to an event, such as a change in a Cloud Storage bucket. A google.storage.object.finalize trigger is a type of trigger that fires when a new object is created or an existing object is overwritten in a Cloud Storage bucket. You can create a Cloud Function with a google.storage.object.finalize trigger so that the function runs whenever a new batch of data is uploaded to the bucket. You can write code so that the function can scale up a Compute Engine autoscaling managed instance group, which is a group of VM instances that automatically adjusts its size based on load or custom metrics. You can use an image pre-loaded with the data processing software that terminates the instances when processing completes, which means that the instances only run when there is data to process and stop when they are done. This way, you can minimize costs while ensuring that the data is processed as quickly as possible.

Question 4

Question Type: **MultipleChoice**

Your company's security team needs to have read-only access to Data Access audit logs in the _Required bucket You want to provide your security team with the necessary permissions following the principle of least privilege and Google-recommended practices. What should you do?

Options:

A- Assign the roles/logging, viewer role to each member of the security team

B- Assign the roles/logging. viewer role to a group with all the security team members

C- Assign the roles/logging.privateLogViewer role to each member of the security team

D- Assign the roles/logging.privateLogviewer role to a group with all the security team members

Answer:

D

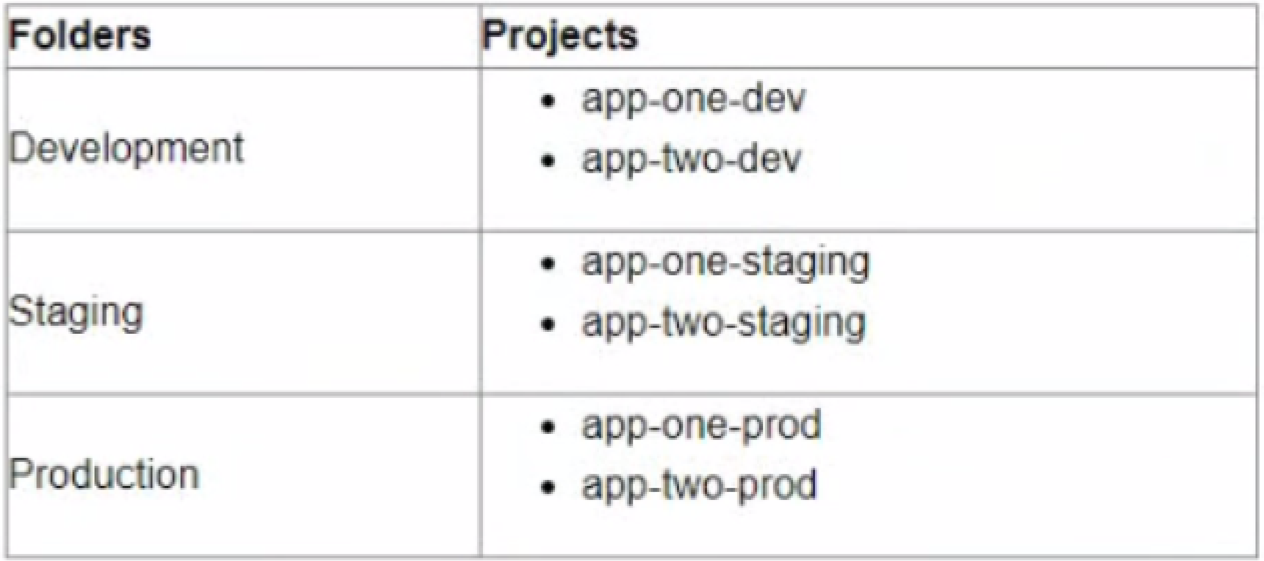
Explanation:

The best option for providing your security team with the necessary permissions following the principle of least privilege and Google-recommended practices is to assign the roles/logging.privateLogViewer role to a group with all the security team members. The roles/logging.privateLogViewer role is a predefined role that grants read-only access to Data Access audit logs and other private logs in Cloud Logging. A group is a collection of users that can be assigned roles and permissions as a single unit. You can assign the roles/logging.privateLogViewer role to a group with all the security team members by using IAM policies. This way, you can provide your security team with the minimum level of access they need to view Data Access audit logs in the _Required bucket.

Question 5

Question Type: **MultipleChoice**

You are currently planning how to display Cloud Monitoring metrics for your organization's Google Cloud projects. Your organization has three folders and six projects:



Folders	Projects
Development	• app-one-dev • app-two-dev
Staging	• app-one-staging • app-two-staging
Production	• app-one-prod • app-two-prod

You want to configure Cloud Monitoring dashboards lo only display metrics from the projects within one folder You need to ensure that the dashboards do not display metrics from projects in the other folders You want to follow Google-recommended practices What should you do?

Options:

A- Create a single new scoping project

B- Create new scoping projects for each folder

C- Use the current app-one-prod project as the scoping project

D- Use the current app-one-dev, app-one-staging and app-one-prod projects as the scoping project for each folder

Answer:

B

Explanation:

The best option for configuring Cloud Monitoring dashboards to only display metrics from the projects within one folder is to create new scoping projects for each folder. A scoping project is a project that defines which resources are monitored by Cloud Monitoring. You can create new scoping projects for each folder by using the gcloud monitoring register-project command. This way, you can associate each scoping project with a folder and only monitor the resources within that folder. You can then configure Cloud Monitoring dashboards to use the scoping projects as data sources and only display metrics from the projects within one folder.

Question 6

Question Type: MultipleChoice

You are building an application that runs on Cloud Run The application needs to access a third-party API by using an API key You need to determine a secure way to store and use the API key in your application by following Google-recommended practices What should you do?

Options:

A- Save the API key in Secret Manager as a secret Reference the secret as an environment variable in the Cloud Run application

B- Save the API key in Secret Manager as a secret key Mount the secret key under the /sys/api_key directory and decrypt the key in the Cloud Run application

C- Save the API key in Cloud Key Management Service (Cloud KMS) as a key Reference the key as an environment variable in the Cloud Run application

D- Encrypt the API key by using Cloud Key Management Service (Cloud KMS) and pass the key to Cloud Run as an environment variable Decrypt and use the key in Cloud Run

Answer:

A

Explanation:

The best option for storing and using the API key in your application by following Google-recommended practices is to save the API key in Secret Manager as a secret and reference the secret as an environment variable in the Cloud Run application. Secret Manager is a service that allows you to store and manage sensitive data, such as API keys, passwords, and certificates, in Google Cloud. A secret is a resource that represents a logical secret, such as an API key. You can save the API key in Secret Manager as a secret and use IAM policies to control who can access it. You can also reference the secret as an environment variable in the Cloud Run application by using the ${SECRET_NAME} syntax. This way, you can securely store and use the API key in your application without exposing it in your code or configuration files.

20%
DISCOUNT