

# PROFESSIONAL-CLOUD-DEVOPS- ENGINEER<sup>Q&As</sup>

Professional Cloud DevOps Engineer

**Pass Google PROFESSIONAL-CLOUD-DEVOPS-  
ENGINEER Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass2lead.com/professional-cloud-devops-engineer.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by Google  
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



### QUESTION 1

You support a high-traffic web application with a microservice architecture. The home page of the application displays multiple widgets containing content such as the current weather, stock prices, and news headlines. The main serving thread makes a call to a dedicated microservice for each widget and then lays out the homepage for the user. The microservices occasionally fail; when that happens, the serving thread serves the homepage with some missing content. Users of the application are unhappy if this degraded mode occurs too frequently, but they would rather have some content served instead of no content at all. You want to set a Service Level Objective (SLO) to ensure that the user experience does not degrade too much. What Service Level Indicator (SLI) should you use to measure this?

- A. A quality SLI: the ratio of non-degraded responses to total responses.
- B. An availability SLI: the ratio of healthy microservices to the total number of microservices.
- C. A freshness SLI: the proportion of widgets that have been updated within the last 10 minutes.
- D. A latency SLI: the ratio of microservice calls that complete in under 100 ms to the total number of microservice calls.

Correct Answer: A

<https://cloud.google.com/architecture/adopting-slos>

---

### QUESTION 2

You support an e-commerce application that runs on a large Google Kubernetes Engine (GKE) cluster deployed on-premises and on Google Cloud Platform. The application consists of microservices that run in containers. You want to identify containers that are using the most CPU and memory. What should you do?

- A. Use Stackdriver Kubernetes Engine Monitoring.
- B. Use Prometheus to collect and aggregate logs per container, and then analyze the results in Grafana.
- C. Use the Stackdriver Monitoring API to create custom metrics, and then organize your containers using groups.
- D. Use Stackdriver Logging to export application logs to BigQuery, aggregate logs per container, and then analyze CPU and memory consumption.

Correct Answer: A

\*

<https://cloud.google.com/anthos/clusters/docs/on-prem> GKE on-prem is also called Anthos clusters on VMware

\*

<https://cloud.google.com/anthos/clusters/docs/on-prem/concepts/logging-and-monitoring> You have several logging and monitoring options for your Anthos clusters on VMware:

+

Cloud Logging and Cloud Monitoring, enabled by in-cluster agents deployed with Anthos clusters on VMware.

+

Prometheus and Grafana, disabled by default.

+

Validated configurations with third-party solutions.

=> it means, if not a special situation, the correct should be using the first option: Logging and Monitoring. In this case, we want metrics, so Monitoring (aka. Cloud Monitoring, Stackdriver Monitoring) should be used. We are talking about GKE, so we will use Kubernetes Engine Monitoring (<https://cloud.google.com/kubernetes-engine-monitoring>).

### QUESTION 3

You are creating Cloud Logging sinks to export log entries from Cloud Logging to BigQuery for future analysis. Your organization has a Google Cloud folder named Dev that contains development projects and a folder named Prod that contains production projects. Log entries for development projects must be exported to dev\_dataset, and log entries for production projects must be exported to prod\_dataset. You need to minimize the number of log sinks created, and you want to ensure that the log sinks apply to future projects. What should you do?

- A. Create a single aggregated log sink at the organization level.
- B. Create a log sink in each project.
- C. Create two aggregated log sinks at the organization level, and filter by project ID.
- D. Create an aggregated log sink in the Dev and Prod folders.

Correct Answer: D

### QUESTION 4

You support a trading application written in Python and hosted on App Engine flexible environment. You want to customize the error information being sent to Stackdriver Error Reporting. What should you do?

- A. Install the Stackdriver Error Reporting library for Python, and then run your code on a Compute Engine VM.
- B. Install the Stackdriver Error Reporting library for Python, and then run your code on Google Kubernetes Engine.
- C. Install the Stackdriver Error Reporting library for Python, and then run your code on App Engine flexible environment.
- D. Use the Stackdriver Error Reporting API to write errors from your application to ReportedErrorEvent, and then generate log entries with properly formatted error messages in Stackdriver Logging.

Correct Answer: D

If you're using the Error Reporting API, you can report error events from your application by writing them to ReportedErrorEvent. Doing this generates log entries with properly formatted error messages in Cloud Logging. <https://cloud.google.com/error-reporting/docs/formatting-error-messages>

### QUESTION 5

Your application artifacts are being built and deployed via a CI/CD pipeline. You want the CI/CD pipeline to securely access application secrets. You also want to more easily rotate secrets in case of a security breach. What should you do?

- A. Prompt developers for secrets at build time. Instruct developers to not store secrets at rest.
- B. Store secrets in a separate configuration file on Git. Provide select developers with access to the configuration file.
- C. Store secrets in Cloud Storage encrypted with a key from Cloud KMS. Provide the CI/CD pipeline with access to Cloud KMS via IAM.
- D. Encrypt the secrets and store them in the source code repository. Store a decryption key in a separate repository and grant your pipeline access to it.

Correct Answer: C

By storing secrets in Cloud Storage, you can take advantage of the security features provided by the platform and encrypt them using Cloud KMS, a GCP service that allows you to create, manage, and use encryption keys. This way you can control who has access to the secrets, and you can easily rotate the encryption keys in case of a security breach. Additionally, you can use IAM to give the CI/CD pipeline the necessary permissions to access the secrets and use them during the deployment process, without the need to store them in the source code or give access to them to specific developers.

[PROFESSIONAL-CLOUD-DEVOPS-ENGINEER VCE Dumps](#)

[PROFESSIONAL-CLOUD-DEVOPS-ENGINEER Practice Test](#)

[PROFESSIONAL-CLOUD-DEVOPS-ENGINEER Study Guide](#)