PROFESSIONAL-CLOUD-DEVOPS-ENGINEER Q&As

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.certbus.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

QUESTION 1

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.

QUESTION 2

The new version of your containerized application has been tested and is ready to be deployed to production on Google Kubernetes Engine (GKE). You could not fully load-test the new version in your pre-production environment, and you need to ensure that the application does not have performance problems after deployment. Your deployment must be automated. What should you do?

A. Deploy the application through a continuous delivery pipeline by using canary deployments. Use Cloud Monitoring to look for performance issues, and ramp up traffic as supported by the metrics.

B. Deploy the application through a continuous delivery pipeline by using blue/green deployments. Migrate traffic to the new version of the application and use Cloud Monitoring to look for performance issues.

C. Deploy the application by using kubectl and use Config Connector to slowly ramp up traffic between versions. Use Cloud Monitoring to look for performance issues.

D. Deploy the application by using kubectl and set the spec.updateStrategy.type field to RollingUpdate. Use Cloud Monitoring to look for performance issues, and run the kubectl rollback command if there are any issues.

Correct Answer: A

QUESTION 3

You support a Node.js application running on Google Kubernetes Engine (GKE) in production. The application makes several HTTP requests to dependent applications. You want to anticipate which dependent applications might cause performance issues. What should you do?

A. Instrument all applications with Stackdriver Profiler.

B. Instrument all applications with Stackdriver Trace and review inter-service HTTP requests.

C. Use Stackdriver Debugger to review the execution of logic within each application to instrument all applications.

D. Modify the Node.js application to log HTTP request and response times to dependent applications. Use Stackdriver Logging to find dependent applications that are performing poorly.

Correct Answer: B

Stackdriver Trace allows you to collect and analyze performance data for all the applications that make up your system, including the applications running on GKE. By instrumenting your application with Stackdriver Trace, you can see detailed performance information for each request, including the time spent in each component of your system, as well as any inter-service HTTP requests. This will allow you to identify which dependent applications might be causing performance issues, so that you can focus on optimizing those services specifically.

QUESTION 4

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 Kubernetest Engine Monitoring (https://cloud.google.com/kubernetes-engine-monitoring).

QUESTION 5

You are configuring a CI pipeline. The build step for your CI pipeline integration testing requires access to APIs inside your private VPC network. Your security team requires that you do not expose API traffic publicly. You need to implement a solution that minimizes management overhead. What should you do?

A. Use Cloud Build private pools to connect to the private VPC.

B. Use Spinnaker for Google Cloud to connect to the private VPC.

C. Use Cloud Build as a pipeline runner. Configure Internal HTTP(S) Load Balancing for API access.

D. Use Cloud Build as a pipeline runner. Configure External HTTP(S) Load Balancing with a Google Cloud Armor policy for API access.

Correct Answer: A

https://cloud.google.com/build/docs/private-pools/private-pools-overview

QUESTION 6

You need to enforce several constraint templates across your Google Kubernetes Engine (GKE) clusters. The constraints include policy parameters, such as restricting the Kubernetes API. You must ensure that the policy parameters are stored in a GitHub repository and automatically applied when changes occur. What should you do?

A. Set up a GitHub action to trigger Cloud Build when there is a parameter change. In Cloud Build, run a gcloud CLI command to apply the change.

B. When there is a change in GitHub. use a web hook to send a request to Anthos Service Mesh, and apply the change.

C. Configure Anthos Config Management with the GitHub repository. When there is a change in the repository, use Anthos Config Management to apply the change.

D. Configure Config Connector with the GitHub repository. When there is a change in the repository, use Config Connector to apply the change.

Correct Answer: C

https://medium.com/@kasiarun/introduction-to-anthos-config-management-1a43917c26ae

QUESTION 7

You support a web application that runs on App Engine and uses CloudSQL and Cloud Storage for data storage. After a short spike in website traffic, you notice a big increase in latency for all user requests, increase in CPU use, and the

number of processes running the application. Initial troubleshooting reveals:

After the initial spike in traffic, load levels returned to normal but users still experience high latency.

Requests for content from the CloudSQL database and images from Cloud Storage show the same high latency.

No changes were made to the website around the time the latency increased.

There is no increase in the number of errors to the users.

You expect another spike in website traffic in the coming days and want to make sure users don\\'t experience latency. What should you do?

A. Upgrade the GCS buckets to Multi-Regional.

B. Enable high availability on the CloudSQL instances.

C. Move the application from App Engine to Compute Engine.

D. Modify the App Engine configuration to have additional idle instances.

Correct Answer: D

Scaling App Engine scales the number of instances automatically in response to processing volume. This scaling factors in the automatic_scaling settings that are provided on a per-version basis in the configuration file. A service with basic scaling is configured by setting the maximum number of instances in the max_instances parameter of the basic_scaling setting. The number of live instances scales with the processing volume. You configure the number of instances of each version in that service\\'s configuration file. The number of instances usually corresponds to the size of a dataset being held in memory or the desired throughput for offline work. You can adjust the number of instances of a manually-scaled version very quickly, without stopping instances that are currently running, using the Modules API set_num_instances function.

https://cloud.google.com/appengine/docs/standard/python/how-instances-are-managed

---

QUESTION 8

You recently deployed your application in Google Kubernetes Engine (GKE) and now need to release a new version of the application. You need the ability to instantly roll back to the previous version of the application in case there are issues with the new version. Which deployment model should you use?

A. Perform a rolling deployment, and test your new application after the deployment is complete.

B. Perform A/B testing, and test your application periodically after the deployment is complete.

C. Perform a canary deployment, and test your new application periodically after the new version is deployed.

D. Perform a blue/green deployment, and test your new application after the deployment is complete.

Correct Answer: D

https://cloud.google.com/architecture/application-deployment-and-testing-strategies

---

QUESTION 9

Your company follows Site Reliability Engineering principles. You are writing a postmortem for an incident, triggered by

a software change, that severely affected users. You want to prevent severe incidents from happening in the future. What should you do?

A. Identify engineers responsible for the incident and escalate to their senior management.

B. Ensure that test cases that catch errors of this type are run successfully before new software releases.

C. Follow up with the employees who reviewed the changes and prescribe practices they should follow in the future.

D. Design a policy that will require on-call teams to immediately call engineers and management to discuss a plan of action if an incident occurs.

Correct Answer: B

QUESTION 10

You are building and deploying a microservice on Cloud Run for your organization. Your service is used by many applications internally. You are deploying a new release, and you need to test the new version extensively in the staging and production environments. You must minimize user and developer impact. What should you do?

A. Deploy the new version of the service to the staging environment. Split the traffic, and allow 1% of traffic through to the latest version. Test the latest version. If the test passes, gradually roll out the latest version to the staging and production environments.

B. Deploy the new version of the service to the staging environment. Split the traffic, and allow 50% of traffic through to the latest version. Test the latest version. If the test passes, send all traffic to the latest version. Repeat for the production environment.

C. Deploy the new version of the service to the staging environment with a new-release tag without serving traffic. Test the new-release version. If the test passes, gradually roll out this tagged version. Repeat for the production environment.

D. Deploy a new environment with the green tag to use as the staging environment. Deploy the new version of the service to the green environment and test the new version. If the tests pass, send all traffic to the green environment and delete the existing staging environment. Repeat for the production environment.

Correct Answer: A

QUESTION 11

You need to build a CI/CD pipeline for a containerized application in Google Cloud. Your development team uses a central Git repository for trunk-based development. You want to run all your tests in the pipeline for any new versions of the application to improve the quality. What should you do?

A. 1. Install a Git hook to require developers to run unit tests before pushing the code to a central repository.

2.

Trigger Cloud Build to build the application container. Deploy the application container to a testing environment, and run integration tests.

3.

If the integration tests are successful, deploy the application container to your production environment, and run acceptance tests.

B. 1. Install a Git hook to require developers to run unit tests before pushing the code to a central repository. If all tests are successful, build a container.

2.

Trigger Cloud Build to deploy the application container to a testing environment, and run integration tests and acceptance tests.

3.

If all tests are successful, tag the code as production ready. Trigger Cloud Build to build and deploy the application container to the production environment.

C. 1. Trigger Cloud Build to build the application container, and run unit tests with the container.

2.

If unit tests are successful, deploy the application container to a testing environment, and run integration tests.

3.

If the integration tests are successful, the pipeline deploys the application container to the production environment. After that, run acceptance tests.

D. 1. Trigger Cloud Build to run unit tests when the code is pushed. If all unit tests are successful, build and push the application container to a central registry.

2.

Trigger Cloud Build to deploy the container to a testing environment, and run integration tests and acceptance tests.

3.

If all tests are successful, the pipeline deploys the application to the production environment and runs smoke tests

Correct Answer: D

QUESTION 12

Your organization is starting to containerize with Google Cloud. You need a fully managed storage solution for container images and Helm charts. You need to identify a storage solution that has native integration into existing Google Cloud services, including Google Kubernetes Engine (GKE), Cloud Run, VPC Service Controls, and Identity and Access Management (IAM). What should you do?

A. Use Docker to configure a Cloud Storage driver pointed at the bucket owned by your organization.

B. Configure an open source container registry server to run in GKE with a restrictive role-based access control (RBAC) configuration.

C. Configure Artifact Registry as an OCI-based container registry for both Helm charts and container images.

D. Configure Container Registry as an OCI-based container registry for container images.

Correct Answer: C

https://cloud.google.com/artifact-registry/docs/helm

QUESTION 13

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 14

You are ready to deploy a new feature of a web-based application to production. You want to use Google Kubernetes Engine (GKE) to perform a phased rollout to half of the web server pods.

What should you do?

A. Use a partitioned rolling update.

B. Use Node taints with NoExecute.

C. Use a replica set in the deployment specification.

D. Use a stateful set with parallel pod management policy.

Correct Answer: A

QUESTION 15

You use Spinnaker to deploy your application and have created a canary deployment stage in the pipeline. Your application has an in-memory cache that loads objects at start time. You want to automate the comparison of the canary

version against the production version. How should you configure the canary analysis?

A. Compare the canary with a new deployment of the current production version.

B. Compare the canary with a new deployment of the previous production version.

C. Compare the canary with the existing deployment of the current production version.

D. Compare the canary with the average performance of a sliding window of previous production versions.

Correct Answer: A

Reference: https://spinnaker.io/guides/user/canary/