Professional-Cloud-DevOps-Engineer Exam Materials - Professional-Cloud-DevOps-Engineer Valid Exam Sims

DOWNLOAD the newest DumpsQuestion Professional-Cloud-DevOps-Engineer PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=14MflRM2AgBAKQY3-7iOqHcD2dR2gbyA-

The pass rate is 98% for Professional-Cloud-DevOps-Engineer exam bootcamp, if you choose us, we can ensure you that you can pass the exam just one time. In addition, we offer you free demo to have a try before buying, so that you can know what the complete version is like. In order to strengthen your confidence for Professional-Cloud-DevOps-Engineer training materials, we are pass guarantee and money back guarantee, and we will refund your money if you fail to pass the exam. We have a professional service team and they have the professional knowledge for Professional-Cloud-DevOps-Engineer Exam Bootcamp, if you have any questions, you can contact with them.

Topics of Google Professional Cloud DevOps Engineer Exam

Candidates must know the exam topics before they start preparation because it will help them in hitting the core. Our **Professional Cloud DevOps Engineer Dumps** will include the following topics:

Applying site reliability engineering principles to a service

- Balance change, velocity, and reliability of the service:
- Agree to consequences of not meeting the error budget
- Manage a service (e.g., introduce a new service, deploy it, maintain and retire it)
- Define SLOs and understand SLAs
- 1.2 Manage service life cycle:
- Toil automation
- Discover SLIs (availability, latency, etc.)

Building and implementing CI/CD pipelines for a service

- Deployment strategies with Cloud Build, Spinnaker
- Artifact repositories with Container Registry
- Design CI/CD pipelines:
- Configure deployment processes (e.g., approval flows)

Implementing service monitoring strategies

- Set ACL to restrict access to audit logs with IAM, Stackdriver Logging
- Set ACL to allow metric writing for custom metrics with IAM, Stackdriver Monitoring
- Sending logs to an external logging platform
- Collecting GKE/Kubernetes metrics

- Collecting third-party and structured logs with Stackdriver Logging, Fluentd
- Selecting the options for logging export
- Manage application logs:
- Enabling VPC flow logs
- Use metric explorer for ad hoc metric analysis
- Viewing logs in the GCP Console
- Set ACL to restrict export configuration with IAM, Stackdriver Logging
- Understanding the logging exclusion vs. logging export
- Sending application logs directly to Stackdriver API with Stackdriver Logging
- Using basic vs. advanced logging filters

Optimizing service performance

- identify resource costs
- Utilize Stackdriver to identify cloud resource utilization
- Develop a plan to optimize areas of greatest cost or lowest utilization
- Identify resource utilization levels
- Troubleshoot issues with the image/OS
- Identify service performance issues:
- Troubleshoot network issues (e.g., VPC flow logs, firewall logs, latency, view network details)

Managing service incidents

- Establish communications channels (email, IRC, Hangouts, Slack, phone, etc.)
- Identify probable causes of service failure
- Record major changes in incident state (When mitigated? When all clear? etc.)
- Provide regular status updates, internal and external
- Rotate/hand over roles
- Coordinate roles and implement communication channels during a service incident:
- Avoid exhaustion/burnout
- Manage stakeholder relationships
- Scaling response team and delegation

Professional-Cloud-DevOps-Engineer Valid Exam Sims, Latest Professional-Cloud-DevOps-Engineer Exam Answers

The clients can try out and download our Professional-Cloud-DevOps-Engineer study materials before their purchase. They can immediately use our Professional-Cloud-DevOps-Engineer training guide after they pay successfully. Our expert team will update the study materials periodically to make sure that our worthy customers can always have the latest and valid information. And if the clients encounter the problems in the course of using our Professional-Cloud-DevOps-Engineer Learning Engine, our online customer service staff will enthusiastically solve their problems.

Google Cloud Certified - Professional Cloud DevOps Engineer Exam Sample Questions (Q79-Q84):

NEW QUESTION # 79
You support a popular mobile game application deployed on Google Kubernetes Engine (GKE) across several Google Cloud regions. Each region has multiple Kubernetes clusters. You receive a report that none of the users in a specific region can connect to the application. You want to resolve the incident while following Site Reliability Engineering practices. What should you do first?

- A. Add an extra node pool that consists of high memory and high CPU machine type instances to the cluster.
- B. Use Stackdriver Monitoring to check for a spike in CPU or memory usage for the affected region.
- C. Reroute the user traffic from the affected region to other regions that don't report issues.
- D. Use Stackdriver Logging to filter on the clusters in the affected region, and inspect error messages in the logs.

Answer: C

NEW QUESTION # 80
Your organization is using Helm to package containerized applications Your applications reference both public and private charts Your security team flagged that using a public Helm repository as a dependency is a risk You want to manage all charts uniformly, with native access control and VPC Service Controls What should you do?

- A. Store public and private charts in OCI format by using Artifact Registry
- B. Store public and private charts by using GitHub Enterprise with Google Workspace as the identity provider
- C. Store public and private charts by using Git repository Configure Cloud Build to synchronize contents of the repository into a Cloud Storage bucket Connect Helm to the bucket by using https: // [bucket] .srorage.googleapis.com/ [holnchart] as the Helm repository
- D. Configure a Helm chart repository server to run in Google Kubernetes Engine (GKE) with Cloud Storage bucket as the storage backend

Answer: A

Explanation:
The best option for managing all charts uniformly, with native access control and VPC Service Controls is to store public and private charts in OCI format by using Artifact Registry. Artifact Registry is a service that allows you to store and manage container images and other artifacts in Google Cloud. Artifact Registry supports OCI format, which is an open standard for storing container images and other artifacts such as Helm charts. You can use Artifact Registry to store public and private charts in OCI format and manage them uniformly. You can also use Artifact Registry's native access control features, such as IAM policies and VPC Service Controls, to secure your charts and control who can access them.

NEW QUESTION # 81
You are designing a system with three different environments: development, quality assurance (QA), and production.
Each environment will be deployed with Terraform and has a Google Kubernetes Engine (GKE) cluster created so that application teams can deploy their applications. Anthos Config Management will be used and templated to deploy infrastructure level resources in each GKE cluster. All users (for example, infrastructure operators and application owners) will use GitOps. How should you structure your source control repositories for both Infrastructure as Code (laC) and application code?

- A. Cloud Infrastructure (Terraform) repository is shared: different branches are different environmentsGKE Infrastructure (Anthos Config Management Kustomize manifests) repository is shared: differentoverlay directories are different environmentsApplication (app source code) repository is shared: different directories are different features
- B. Cloud Infrastructure (Terraform) repository is shared: different directories are different environmentsGKE Infrastructure (Anthos Config Management Kustomize manifests) repositories are separated:different branches are different environmentsApplication (app source code) repositories are separated: different branches are different features
- C. Cloud Infrastructure (Terraform) repositories are separated: different branches are different environmentsGKE Infrastructure (Anthos Config Management Kustomize manifests) repositories are separated:different overlay directories are different environmentsApplication (app source code) repositories are separated: different branches are different features
- D. Cloud Infrastructure (Terraform) repository is shared: different directories are different environmentsGKE Infrastructure (Anthos Config Management Kustomize manifests) repository is shared: differentoverlay directories are different environmentsApplication (app source code) repositories are separated: different branches are different features

Answer: B

Explanation:
The correct answer is B. Cloud Infrastructure (Terraform) repository is shared: different directories are different environments. GKE Infrastructure (Anthos Config Management Kustomize manifests) repositories are separated: different branches are different environments. Application (app source code) repositories are separated: different branches are different features.
This answer follows the best practices for using Terraform and Anthos Config Management with GitOps, as described in the following sources:
For Terraform, it is recommended to use a single repository for all environments, and use directories to separate them. This way, you can reuse the same Terraform modules and configurations across environments, and avoid code duplication and drift.You can also use Terraform workspaces to isolate the state files for each environment12.
For Anthos Config Management, it is recommended to use separate repositories for each environment, and use branches to separate the clusters within each environment. This way, you can enforce different policies and configurations for each environment, and use pull requests to promote changes across environments.You can also use Kustomize to create overlays for each cluster that apply specific patches or customizations34.
For application code, it is recommended to use separate repositories for each application, and use branches to separate the features or bug fixes for each application. This way, you can isolate the development and testing of each application, and use pull requests to

merge changes into the main branch.You can also use tags or labels to trigger deployments to different environments5.
References:
1:Best practices for using Terraform | Google Cloud
2: Terraform Recommended Practices - Part 1 | Terraform - HashiCorp Learn
3:Deploy Anthos on GKE with Terraform part 1: GitOps with Config Sync | Google Cloud Blog
4: Using Kustomize with Anthos Config Management | Anthos Config Management Documentation | Google Cloud
5: Deploy Anthos on GKE with Terraform part 3: Continuous Delivery with Cloud Build | Google Cloud Blog
6: GitOps-style continuous delivery with Cloud Build | Cloud Build Documentation | Google Cloud

NEW QUESTION # 82
You are designing a new Google Cloud organization for a client. Your client is concerned with the risks associated with long-lived credentials created in Google Cloud. You need to design a solution to completely eliminate the risks associated with the use of JSON service account keys while minimizing operational overhead. What should you do?

- A. Apply the constraints/iam.disableserviceAccountKeycreation constraint to the organization.
- B. Apply the constraints/iam. disableServiceAccountKeyUp10ad constraint to the organization.
- C. Grant the roles/ iam.serviceAccountKeyAdmin IAM role to organization administrators only.
- D. Use custom versions of predefined roles to exclude all iam.serviceAccountKeys. * service account role permissions.

Answer: A

Explanation:
The correct answer is B. Apply the constraints/iam.disableServiceAccountKeyCreation constraint to the organization.
According to the Google Cloud documentation, the constraints/iam.disableServiceAccountKeyCreation constraint is an organization policy constraint that prevents the creation of user-managed service account keys1.User-managed service account keys are long-lived credentials that can be downloaded as JSON or P12 files and used to authenticate as a service account2.These keys pose severe security risks if they are leaked, stolen, or misused by unauthorized entities34.By applying this constraint to the organization, you can completely eliminate the risks associated with the use of JSON service account keys and enforce a more secure alternative for authentication, such as Workload Identity or short-lived access tokens12. This also minimizes operational overhead by avoiding the need to manage, rotate, or revoke user-managed service account keys.
The other options are incorrect because they do not completely eliminate the risks associated with the use of JSON service account keys. Option A is incorrect because it only restricts the IAM permissions to create, list, get, delete, or sign service account keys, but it does not prevent existing keys from being used or leaked.
Option C is incorrect because it only disables the upload of user-managed service account keys, but it does not prevent the creation or download of such keys. Option D is incorrect because it only limits the IAM role that can create and manage service account keys, but it does not prevent the keys from being distributed or exposed to unauthorized entities.

NEW QUESTION # 83
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 repositoryIf all tests are successful build a container2. Trigger Cloud Build to deploy the application container to a testing environment, and run integrationtests and acceptance tests3. 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
- B. 1. Trigger Cloud Build to build the application container and run unit tests with the container2. If unit tests are successful, deploy the application container to a testing environment, and run integration tests3. If the integration tests are successful the pipeline deploys the application container to the production environment After that, run acceptance tests
- C. 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 tests3. If all tests are successful the pipeline deploys the application to the production environment and runs smoke tests
- D. 1. Install a Git hook to require developers to run unit tests before pushing the code to a central repository2. Trigger Cloud Build to build the application container Deploy the application container to a testing environment, and run integration tests3. If the integration tests are successful deploy the application container to your production environment. and run acceptance tests

Answer: C

Explanation:
The best option for building a CI/CD pipeline for a containerized application in Google Cloud is to 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, trigger Cloud Build to deploy the container to a testing environment, and run integration tests and acceptance tests, and if all tests are successful, the pipeline deploys the application to the production environment and runs smoke tests. This option follows the best practices for CI/CD pipelines, such as running tests at different stages of the pipeline, using a central registry for storing and managing containers, deploying to different environments, and using Cloud Build as a unified tool for building, testing, and deploying.

NEW QUESTION # 84
.....

Many times getting a right method is important and more efficient than spending too much time and money in vain. Our DumpsQuestion team devote themselves to studying the best methods to help you pass Professional-Cloud-DevOps-Engineer exam certification. From the time when you decide whether to purchase our Professional-Cloud-DevOps-Engineer exam software or not, we have provided you with comprehensive guarantees, including free demo download before buying, payment guarantee in purchase process, one-year free update service after you purchased Professional-Cloud-DevOps-Engineer Exam software, and full refund guarantee of dump cost if you fail Professional-Cloud-DevOps-Engineer exam certification, which are all our promises to ensure customer interests.

robertanvq534902.csublogs.com, idaxmll507069.angelinsblog.com, Disposable vapes

P.S. Free 2026 Google Professional-Cloud-DevOps-Engineer dumps are available on Google Drive shared by DumpsQuestion: https://drive.google.com/open?id=14MflRM2AgBAKQY3-7iOqHcD2dR2gbyA-