EXAM ASSOCIATE CLOUD ENGINEER TOPIC 1 QUESTION 69 DISCUSSION

Actual exam question from Google's Associate Cloud Engineer
Question #: 69
Topic #: 1
[All Associate Cloud Engineer Questions]

You are using Deployment Manager to create a Google Kubernetes Engine cluster. Using the same Deployment Manager deployment, you also want to create a
DaemonSet in the kube-system namespace of the cluster. You want a solution that uses the fewest possible services. What should you do?

A. Add the cluster's API as a new Type Provider in Deployment Manager, and use the new type to create the DaemonSet.

B. Use the Deployment Manager Runtime Configurator to create a new Config resource that contains the DaemonSet definition.

C. With Deployment Manager, create a Compute Engine instance with a startup script that uses kubectl to create the DaemonSet.

D. In the cluster's definition in Deployment Manager, add a metadata that has kube-system as key and the DaemonSet manifest as value.

Show Suggested Answer

by [deleted] at *June 12, 2020, 5:18 p.m.*

Comments

Type your comment...

Submit

ESP_SAP Highly Voted 4 years, 1 month ago

Correct Answer is (A):

Adding an API as a type provider
This page describes how to add an API to Google Cloud Deployment Manager as a type provider. To learn more about types and type providers, read the Types overview documentation.

A type provider exposes all of the resources of a third-party API to Deployment Manager as base types that you can use in your configurations. These types must be directly served by a RESTful API that supports Create, Read, Update, and Delete (CRUD).

If you want to use an API that is not automatically provided by Google with Deployment Manager, you must add the API as a type provider.

https://cloud.google.com/deployment-manager/docs/configuration/type-providers/creating-type-provider

upvoted 76 times

magistrum 3 years, 9 months ago

very good find, sounds like you hit the nail in the head

upvoted 7 times

kishoredeena Highly Voted 4 years, 4 months ago

Option A is the right answer

upvoted 10 times

Captain1212 Most Recent 1 year, 1 month ago

Selected Answer: A

A is the correct , bcoz it help you contact directly to the gke cluster to create daemon

upvoted 2 times

sthapit 1 year, 2 months ago

Should have been D

upvoted 1 times

sakdip66 1 year, 6 months ago

Selected Answer: A

option A is the right answer because it lets you directly interact with the Kubernetes API to create the Daemonset using the same deployment Manager Deployment

upvoted 1 times

Buruguduystunstugudunstuy 1 year, 8 months ago

Selected Answer: A

I would say both Answer A and Answer D are valid solutions, and it depends on your preference and requirements.

Answer A involves adding the cluster's API as a new Type Provider in Deployment Manager and using the new type to create the DaemonSet. This solution would allow you to create and manage the DaemonSet and the cluster in the same Deployment Manager deployment.

Answer D involves adding a metadata block to the Deployment Manager deployment of the cluster, which will create the DaemonSet in the kube-system namespace of the cluster. This solution would allow you to create the DaemonSet in a simple way and avoid the need to create a new Type of Provider.

In conclusion, I would choose Answer A to be considered the answer that uses the fewest possible services, as it only involves adding the cluster's API as a new Type Provider in Deployment Manager, which is a lightweight solution.

upvoted 7 times

Bobbybash 1 year, 8 months ago

Selected Answer: D

D. In the cluster's definition in Deployment Manager, add a metadata that has kube-system as key and the DaemonSet manifest as value.

This approach involves adding the DaemonSet manifest directly as a metadata entry in the cluster's definition in Deployment Manager. When the cluster is created, the DaemonSet is automatically created in the kube-system namespace. This approach is the simplest and requires the fewest number of services. Option A is also a viable solution but requires more work to set up a Type Provider. Option B is not suitable because it involves a separate service (Runtime Configurator)

more work to set up a Type Provider. Option B is not suitable because it involves a separate service (Runtime Configurator). Option C is also not recommended because it involves creating a Compute Engine instance and using kubectl to create the DaemonSet, which is more complicated and less efficient than the other options.

upvoted 3 times

vkamlesh0205 1 year, 10 months ago

Selected Answer: A

Option A is the right answer

upvoted 1 times

RanjithK 2 years, 3 months ago

Selected Answer: A

Answer is A.

upvoted 1 times

AzureDP900 2 years, 4 months ago

go with A as per ESP_SAP explanations..

upvoted 1 times

haroldbenites 2 years, 4 months ago

Go for A

upvoted 1 times

luciorifa 2 years, 8 months ago

Selected Answer: A

A is the correct answe, the API need to be added as a type provider

upvoted 1 times

ArunTaneja 2 years, 8 months ago

Selected Answer: A

A should be correct one

upvoted 1 times

thuvh 2 years, 10 months ago

Selected Answer: A

https://medium.com/google-cloud/cloud-deployment-manager-kubernetes-2dd9b8124223

upvoted 2 times

axantroff 1 year ago

Good reference. Thanks for it. Recomended

upvoted 1 times

MCMS 2 years, 10 months ago

Selected Answer: A

Correct Answer is (A)

upvoted 2 times

PR0704 2 years, 11 months ago

couldn't be more confusing

upvoted 10 times

vishnukumartr 2 years, 11 months ago

A. Add the clusterא€™s API as a new Type Provider in Deployment Manager, and use the new type to create the DaemonSet.

upvoted 1 times

Load full discussion...