Google Discussions

Exam Associate Cloud Engineer All Questions

View all questions & answers for the Associate Cloud Engineer exam

Go to Exam

EXAM ASSOCIATE CLOUD ENGINEER TOPIC 1 QUESTION 78 DISCUSSION

Actual exam question from Google's Associate Cloud Engineer
Question #: 78
Topic #: 1
[All Associate Cloud Engineer Questions]

You are using Google Kubernetes Engine with autoscaling enabled to host a new application. You want to expose this new application to the public, using HTTPS on a public IP address. What should you do?

A. Create a Kubernetes Service of type NodePort for your application, and a Kubernetes Ingress to expose this Service via a Cloud Load Balancer.

B. Create a Kubernetes Service of type ClusterIP for your application. Configure the public DNS name of your application using the IP of this Service.

C. Create a Kubernetes Service of type NodePort to expose the application on port 443 of each node of the Kubernetes cluster. Configure the public DNS name of your application with the IP of every node of the cluster to achieve load-balancing.

D. Create a HAProxy pod in the cluster to load-balance the traffic to all the pods of the application. Forward the public traffic to HAProxy with an iptable rule. Configure the DNS name of your application using the public IP of the node HAProxy is running on.

Show Suggested Answer

by dan80 at *June 5, 2020, 6:11 p.m.*

Comments

Type your comment...

Submit

arsav Highly Voted 3 years, 8 months ago

HAProxy is HTTP only, doesnt support HTTPS, so you can reject option D
https://www.haproxy.org/#desc

Cluster IP - is an internal IP, you cannot expose public externally. reject option B

out of option A and C

C, port 443 is https but public DNS is not going to give you a load balancing
A is the right choice,
kubernets ingress exposes HTTPS
https://kubernetes.io/docs/concepts/services-networking/ingress/

and cloud load balancer is the right choice which will help to expose the app to public

upvoted 53 times

NoniGeorge 2 years, 9 months ago

Pretty sure that option D works more from on premise then cloud because with cloud you pretty much don't have to configure your ip tables !

upvoted 1 times

dan80 Highly Voted 4 years, 4 months ago

A is correct.

upvoted 33 times

magistrum 3 years, 9 months ago

Saw this which provides good context https://medium.com/google-cloud/kubernetes-nodeport-vs-loadbalancer-vs-ingress-when-should-i-use-what-922f010849e0

upvoted 21 times

nitinz 3 years, 8 months ago

you nailed it.

upvoted 2 times

yehia2221 Most Recent 2 months, 4 weeks ago

option A is correct, but do not use it in real deployments, it is a bad practice. I am wondering why they didn't mention Cluster IP and exposing it via an ingress or at least a service of type loadbalancer

upvoted 1 times

Cynthia2023 9 months, 3 weeks ago

Selected Answer: A

In Kubernetes, a Service of type NodePort is a way to expose your applications to external traffic. It's one of the several types of Services available in Kubernetes to control how external sources can access services running within the cluster. Here's what a NodePort service entails:

Exposing Services Outside the Cluster:

A NodePort service makes your application accessible from outside the Kubernetes cluster by opening a specific port (the NodePort) on all the nodes (VMs) in your cluster. This port is randomly selected from a defined range (default: 30000-32767) unless you specify a particular port.

upvoted 2 times

Cynthia2023 9 months, 3 weeks ago

When a NodePort service is created, each node in the cluster allocates the specified NodePort. External traffic can access the service by hitting any node's IP address at the NodePort, regardless of whether that node is actually running a pod for the service.
Kubernetes internally routes that traffic to the appropriate pods, even if they are running on different nodes.

upvoted 1 times

BAofBK 11 months, 2 weeks ago

The correct answer is A

upvoted 1 times

Captain1212 1 year, 1 month ago

Selected Answer: A

option A is correct as you need load balancing and in option c dns will not give you load balancing

upvoted 2 times

frantishk 1 year, 2 months ago

I didnt know, that ClusterIP is an internal IP and you cannot expose public externally..

Thanks !

upvoted 2 times

trainingexam 1 year, 3 months ago

Selected Answer: A

A is very easy solution

upvoted 1 times

Buruguduystunstugudunstuy 1 year, 8 months ago

Selected Answer: A

To expose a new application hosted on Google Kubernetes Engine with autoscaling enabled to the public using HTTPS on a public IP address, the most appropriate option would be;

A. Create a Kubernetes Service of type NodePort for your application, and a Kubernetes Ingress to expose this Service via a Cloud Load Balancer.

upvoted 1 times

GS300 1 year, 9 months ago

Selected Answer: A

A works and is correct, but service type should be ClusterIP

upvoted 1 times

cslince 1 year, 10 months ago

Selected Answer: A

A is correct .

upvoted 2 times

gcpBeginner 2 years ago

Correct answer is A. Create a Kubernetes Service of type NodePort for your application, and a Kubernetes Ingress to expose this Service via a Cloud Load Balancer.

upvoted 1 times

Cornholio_LMC 2 years ago

had this question today

upvoted 1 times

NOOGLer 2 years, 3 months ago

A is the correct

upvoted 1 times

AzureDP900 2 years, 4 months ago

This is A without any second thought.

upvoted 1 times

haroldbenites 2 years, 4 months ago

Go for A

upvoted 1 times

BalajiMBala 2 years, 6 months ago

option A is an correct answer

upvoted 1 times

Load full discussion...

Start Learning for free

Social Media

Facebook , Twitter
YouTube , Reddit
Pinterest

We are the biggest and most updated IT certification exam material website.
Using our own resources, we strive to strengthen the IT professionals community for free.