**Implement Cloud Security Fundamentals on Google Cloud: Challenge Lab**

**Overview**

In a challenge lab you’re given a scenario and a set of tasks. Instead of following step-by-step instructions, you will use the skills learned from the labs in the course to figure out how to complete the tasks on your own! An automated scoring system (shown on this page) will provide feedback on whether you have completed your tasks correctly.

When you take a challenge lab, you will not be taught new Google Cloud concepts. You are expected to extend your learned skills, like changing default values and reading and researching error messages to fix your own mistakes.

To score 100% you must successfully complete all tasks within the time period!

This lab is recommended for students who have enrolled in the [Implement Cloud Security Fundamentals on Google Cloud](https://www.cloudskillsboost.google/course_templates/645) skill badge. Are you ready for the challenge?

Topics tested

* Create a custom security role.
* Create a service account.
* Bind IAM security roles to a service account.
* Create a private Kubernetes Engine cluster in a custom subnet.
* Deploy an application to a private Kubernetes Engine cluster

**Challenge scenario**

You have started a new role as a junior member of the security team for the Orca team in Jooli Inc. Your team is responsible for ensuring the security of the Cloud infrastructure and services that the company's applications depend on.

You are expected to have the skills and knowledge for these tasks, so don't expect step-by-step guides to be provided.

Your challenge

You have been asked to deploy, configure, and test a new Kubernetes Engine cluster that will be used for application development and pipeline testing by the Orca development team.

As per the organization's security standards you must ensure that the new Kubernetes Engine cluster is built according to the organization's most recent security standards and thereby must comply with the following:

* The cluster must be deployed using a dedicated service account configured with the least privileges required.
* The cluster must be deployed as a Kubernetes Engine private cluster, with the public endpoint disabled, and the master authorized network set to include only the ip-address of the Orca group's management jumphost.
* The Kubernetes Engine private cluster must be deployed to the orca-build-subnet in the Orca Build VPC.

From a previous project you know that the minimum permissions required by the service account that is specified for a Kubernetes Engine cluster is covered by these three built in roles:

* roles/monitoring.viewer
* roles/monitoring.metricWriter
* roles/logging.logWriter

These roles are specified in the Google Kubernetes Engine (GKE)'s Harden your cluster's security guide in the [Use least privilege Google service accounts](https://cloud.google.com/kubernetes-engine/docs/how-to/hardening-your-cluster#use_least_privilege_sa) section.

You must bind the above roles to the service account used by the cluster as well as a custom role that you must create in order to provide access to any other services specified by the development team. Initially you have been told that the development team requires that the service account used by the cluster should have the permissions necessary to add and update objects in Google Cloud Storage buckets. To do this you will have to create a new custom IAM role that will provide the following permissions:

* storage.buckets.get
* storage.objects.get
* storage.objects.list
* storage.objects.update
* storage.objects.create

Once you have created the new private cluster you must test that it is correctly configured by connecting to it from the jumphost, orca-jumphost, in the management subnet orca-mgmt-subnet. As this compute instance is not in the same subnet as the private cluster you must make sure that the master authorized networks for the cluster includes the internal ip-address for the instance, and you must specify the --internal-ip flag when retrieving cluster credentials using the gcloud container clusters get-credentials command.

All new cloud objects and services that you create should include the "orca-" prefix.

Your final task is to validate that the cluster is working correctly by deploying a simple application to the cluster to test that management access to the cluster using the kubectl tool is working from the orca-jumphost compute instance.

For all tasks in this lab, use the europe-west4 region and the europe-west4-a zone.

In this lab, you have configured a new Kubernetes Engine private cluster that is configured to use a dedicated service account with the least privileges required to run the cluster. You have also created a custom IAM role that provides the permissions required to create and update storage objects in Google Cloud Storage buckets and bound that role to the service account used by the cluster. You also validated that the cluster is working correctly by deploying a simple application to the cluster to test that management access to the cluster using the kubectl tool is working from the orca-jumphost compute instance.