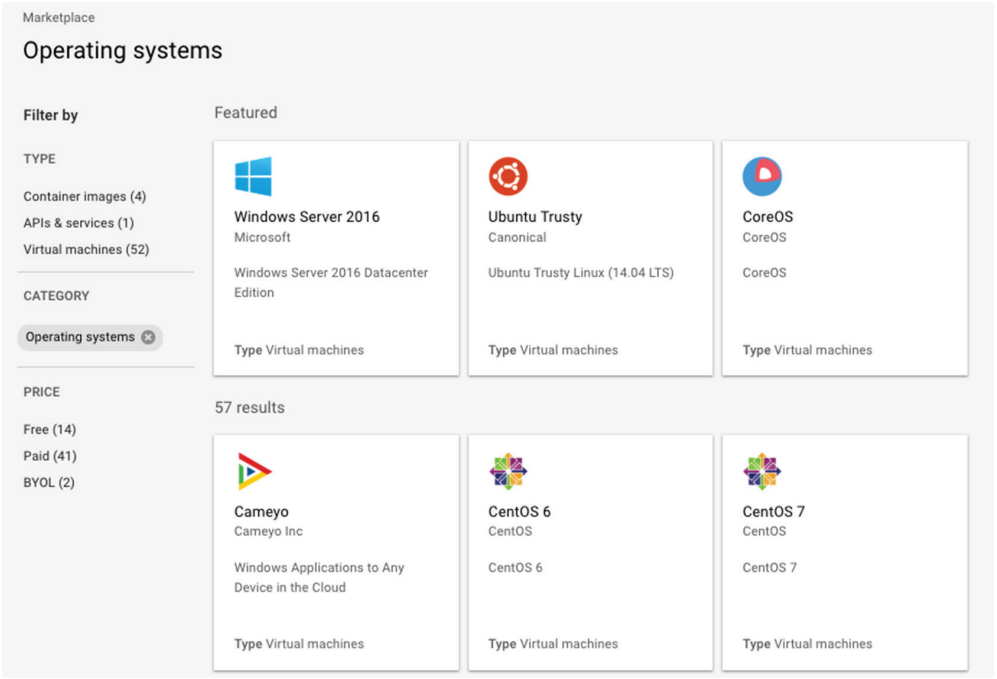
You can see a list of available operating systems in Figure 16.4.

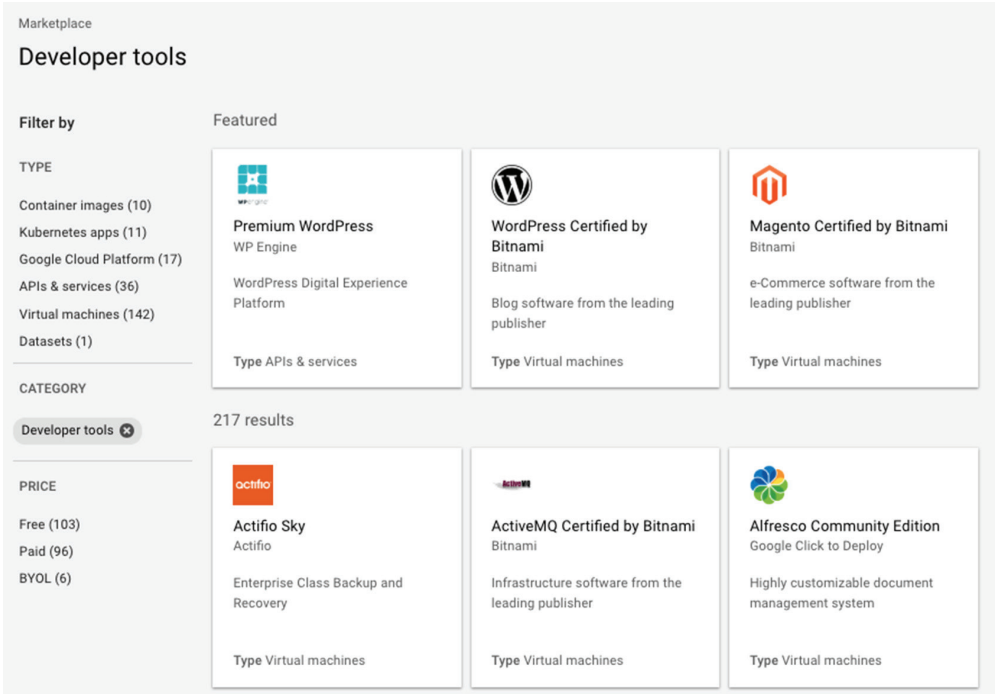
FIGURE 16.4 Operating systems available in Cloud Launcher



Notice that you can further filter the list of operating systems by license type. The license types are free, paid, and bring your own license (BYOL). Free operating systems include Linux and FreeBSD options. The paid operating systems include Windows operating systems and enterprise-supported Linux. You will be charged a fee based on your usage, and that charge will be included in your GCP billing. The BYOL option includes two supported Linux operating systems that require you to have a valid license to run the software. You are responsible for acquiring the license before running the software.

Figure 16.5 shows a sample of developer tools available in Cloud Launcher. These include WordPress, a backup and recovery application, and a document management system.

FIGURE 16.5 Developer tools available in Cloud Launcher



Notice that there are two WordPress options. Cloud Launcher can have the same application provided by multiple vendors. It is best to review the description of each option to find the one best suited for your needs.

Let's take a look at the kind of information provided along with the solutions listed in Cloud Launcher. Figure 16.6 shows the bulk of the information available. It includes an overview, pricing information, and details about the contents of the package. There is also information on where the solution will run within the GCP.

The left side of the page lists the details of the contents of the solution. Figure 16.7 shows the contents of a WordPress package, which include Apache web server, MySQL, and PHP components. The list also specifies the operating system and the types of resources it will use.

FIGURE 16.6 Overview page of a WordPress solution

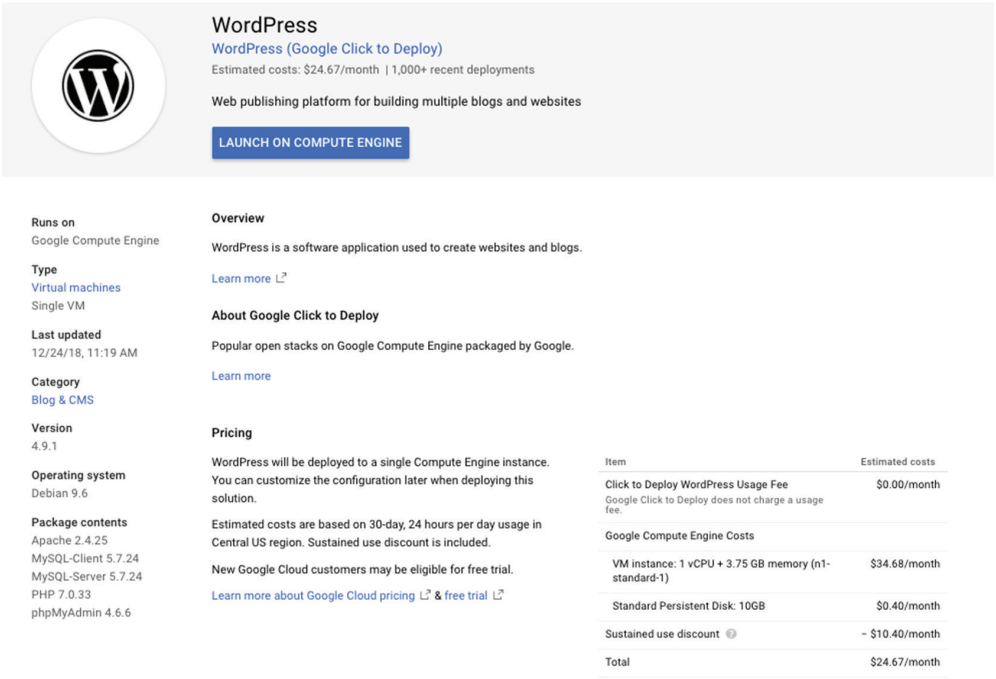
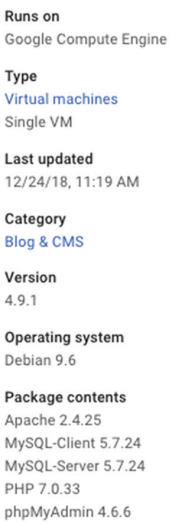


FIGURE 16.7 Details of the contents of the solution package



On the right side of the page is pricing information (see Figure 16.8). These are estimated costs for running the solution, as configured, for one month, which includes the costs of VMs, persistent disks, and any other resources. The price estimate also includes discounts for sustained usage of GCP resources, which are applied as you reach a threshold based on the amount of time a resource is used.

FIGURE 16.8 Pricing estimates for the WordPress solution

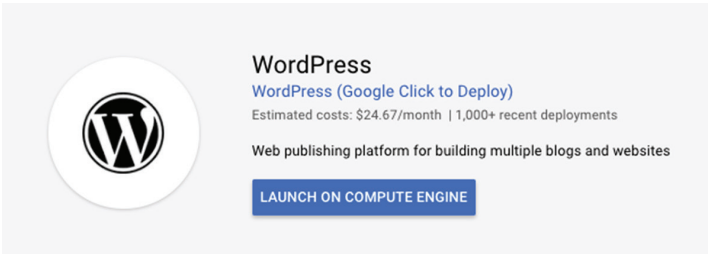
Item	Estimated costs
Click to Deploy WordPress Usage Fee Google Click to Deploy does not charge a usage fee.	$0.00/month
Google Compute Engine Costs	
VM instance: 1 vCPU + 3.75 GB memory (n1-standard-1)	$34.68/month
Standard Persistent Disk: 10GB	$0.40/month
Sustained use discount	- $10.40/month
Total	$24.67/month

Deploying Cloud Launcher Solutions

After you identify a solution that meets your needs, you can launch it from Cloud Launcher.

Go to the overview page of the product you would like to launch, as shown in Figure 16.9.

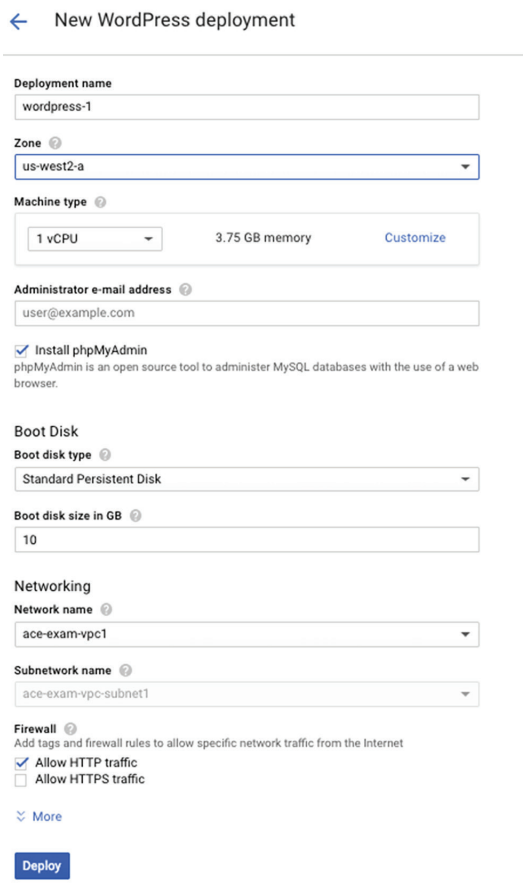
FIGURE 16.9 Launch a Cloud Launcher solution from the overview page of the product.



This will generate a form like the one shown in Figure 16.10.

The contents of the form will vary by application, but many parameters are common across solutions. In this form, you specify a name for the deployment, a zone, and the machine type, which is preconfigured. You must also specify an administrator email. You can optionally install a PHP tool called phpMyAdmin, which is helpful for administering WordPress and other PHP applications.

FIGURE 16.10 The launch form for a WordPress solution in Cloud Launcher



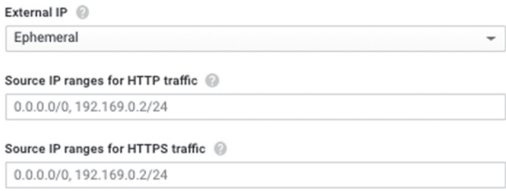
You can choose the type and size of the persistent disk. In this example, the solution will deploy to a 1 vCPU server with 3.75GB of memory and a 10GB boot disk using standard persistent disks. If you wanted, you could opt for an SSD disk for the boot disk. You can also change the size of the boot disk.

In the Networking section, you can specify the network and subnet to launch the VM. You can also configure firewall rules to allow HTTP and HTTPS traffic.

If you expand the More link below the Networking section, you will see options for configuring IP addresses (see Figure 16.11). You can choose to have an ephemeral external

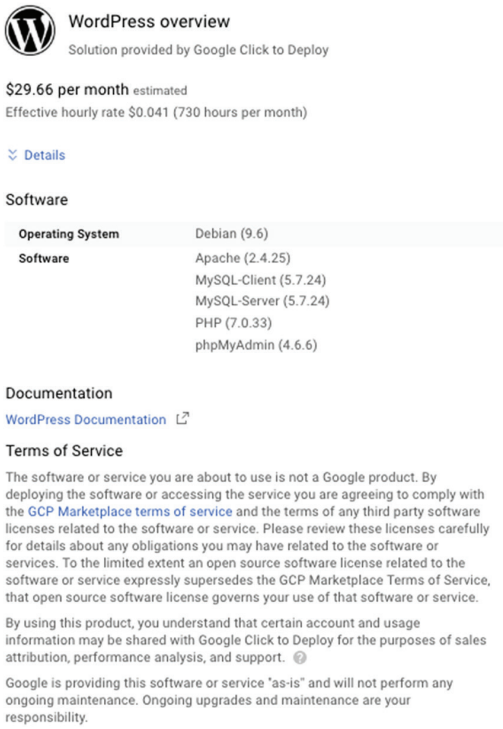
IP or no external IP. If you are hosting a website, choose an external address so the site is accessible from outside the GCP project. Static IP is not an option. You can also specify source IP ranges for HTTP and HTTPS traffic.

FIGURE 16.11 Additional parameters for IP configuration



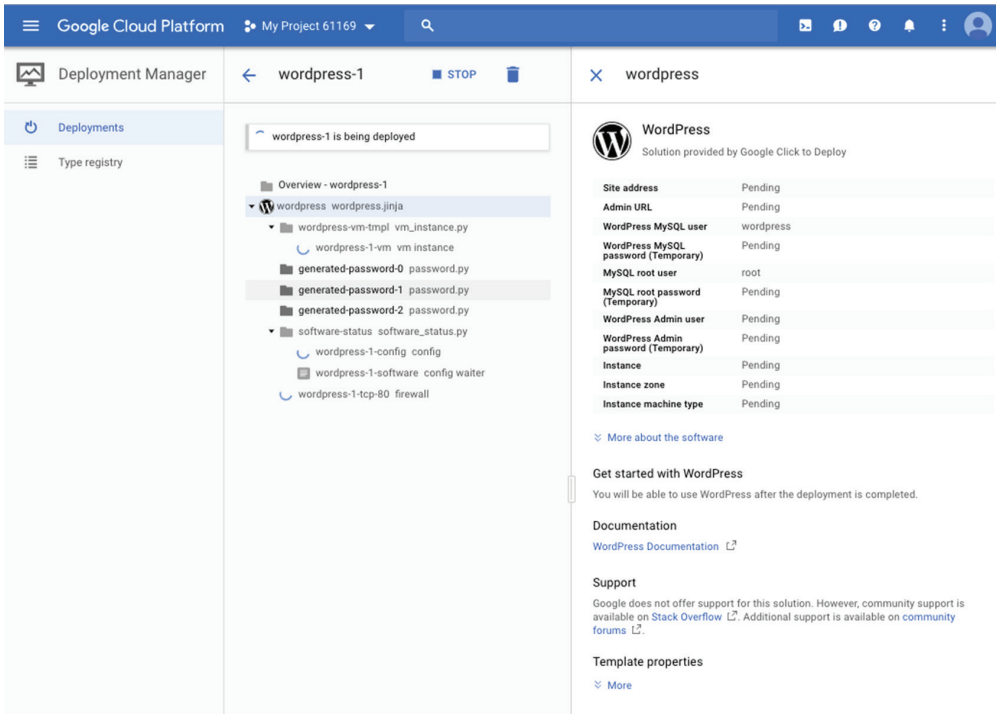
In addition to the parameters described earlier, the launch page will also display overview information, as shown in Figure 16.12.

FIGURE 16.12 Solution overview shown in the Launch form



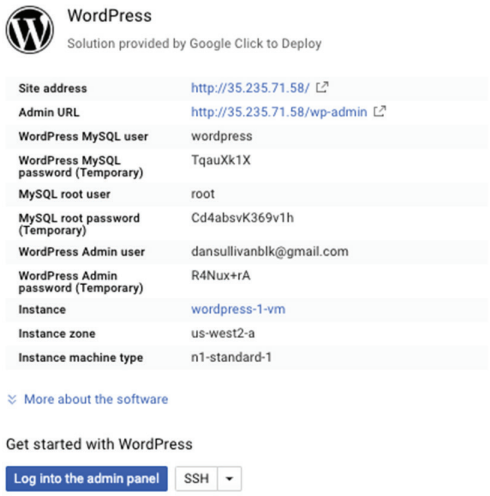
Click the Deploy button to launch the deployment. That will open Deployment Manager and show the progress of the deployment (see Figure 16.13).

FIGURE 16.13 Cloud Deployment Manager launching WordPress



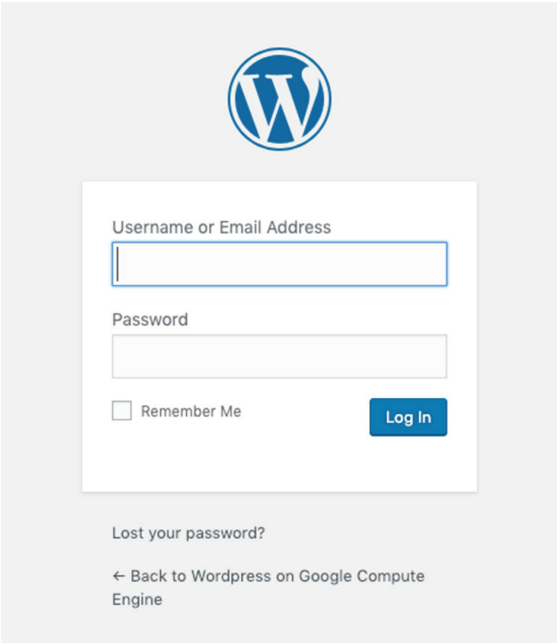
When the launching process completes, you will see summary information about the deployment and a button to launch the admin panel, as shown in Figure 16.14.

FIGURE 16.14 Information about the deployed WordPress instance



Clicking the Log Into The Admin Panel button brings you to the WordPress login (see Figure 16.15). You can log in using the username and temporary password provided in the information form after the deployment completes.

FIGURE 16.15 Logging into WordPress



Deploying an Application Using Deployment Manager

In addition to launching the solutions listed in Cloud Launcher, you can create your own solution configuration files so users can launch preconfigured solutions.

Deployment Manager Configuration Files

Deployment Manager configuration files are written in YAML syntax. The configuration files start with the word resources, followed by resource entities, which are defined using three fields:

- `name`, which is the name of the resource
- `type`, which is the type of the resource, such as compute.v1.instance

- `properties`, which are key-value pairs that specify configuration parameters for the resource. For example, a VM has properties to specify machine type, disks, and network interfaces.

For information on YAML syntax, see the official documentation at `https://yaml.org/start.html`.

A simple example defining a virtual machine called ace-exam-deployment-vm starts with the following:

```
resources:
- type: compute.v1.instance
  name: ace-exam-deployment-vm
```

Next, you can add properties, such as the machine type, disk configuration, and network interfaces.

The properties section of the configuration file starts with the word properties. For each property, there is a single key-value pair or a list of key-value pairs. The machine type property has a single key-value pair, with the key being machineType. Disks have multiple properties, so following the term disks, there is a list of key-value pairs. Continuing the example of ace-exam-deployment-vm, the structure is as follows:

```
resources:
- type: compute.v1.instance
  name: ace-exam-deployment-vm
  properties:
     machineType: [MACHINE_TYPE_URL]
     disks:
[KEY]:[VALUE]
[KEY]:[VALUE]
```

In this example, machine type would be a URL to a Google API resource specification, such as the following:

```
https://www.googleapis.com/compute/v1/projects/[PROJECT_ID]/zones/
us-central1-f/machineTypes/f1-micro
```

Note that there is a reference to `[PROJECT_ID]`, which would be replaced with an actual project ID in a configuration file. Disks have properties such as a `deviceName`, `type`, and Booleans indicating whether the disk is a boot disk or should be autodeleted. Let's continue the previous example by adding the machine type specification and some disk properties:

```
resources:
- type: compute.v1.instance
  name: ace-exam-deployment-vm
  properties:
```

```
    machineType: https://www.googleapis.com/compute/v1/projects/[PROJECT_ID]/
zones/us-central1-f/machineTypes/f1-micro
    disks:
    - deviceName: boot
      type: PERSISTENT
      boot: true
      autoDelete: true
```

Listing 16.1 shows the full configuration file from the Google Deployment Manager documentation. The following code is available at `https://cloud.google.com/deployment-manager/docs/quickstart` (source: `https://github.com/GoogleCloudPlatform/deploymentmanager-samples/blob/master/examples/v2/quick_start/vm.yaml`).

Listing 16.1: examples/v2/quick_start/vm.yaml

```
# Copyright 2016 Google Inc. All rights reserved.#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#     http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.


# Put all your resources under 'resources:'. For each resource, you need:
# - The type of resource. In this example, the type is a Compute VM instance.
# - An internal name for the resource.
# - The properties for the resource. In this example, for VM instances, you add
#   the machine type, a boot disk, network information, and so on.
#
# For a list of supported resources,
# see https://cloud.google.com/deployment-manager/docs/configuration/
supported-resource-types

resources:
- type: compute.v1.instance
  name: quickstart-deployment-vm
  properties:
```

```
    # The properties of the resource depend on the type of resource. For a list
    # of properties, see the API reference for the resource.
    zone: us-central1-f
    # Replace [MY_PROJECT] with your project ID
    machineType: https://www.googleapis.com/compute/v1/projects/[MY_PROJECT]/
zones/us-central1-f/machineTypes/f1-micro
    disks:
    - deviceName: boot
      type: PERSISTENT
      boot: true
      autoDelete: true
      initializeParams:
        # Replace [FAMILY_NAME] with the image family name.
        # See a full list of image families at
        # https://cloud.google.com/compute/docs/images#os-compute-support
        sourceImage: https://www.googleapis.com/compute/v1/projects/debian-
cloud/global/images/family/[FAMILY_NAME]
    # Replace [MY_PROJECT] with your project ID
    networkInterfaces:
    - network: https://www.googleapis.com/compute/v1/projects/[MY_PROJECT]/
global/networks/default
      # Access Config required to give the instance a public IP address
      accessConfigs:
      - name: External NAT
        type: ONE_TO_ONE_NAT
```

This configuration specifies a deployment named `quickstart-deployment-vm`, which will run in the us-central1-f zone. The deployment will use a f1-micro running a Debian distribution of Linux. An external IP address will be assigned.

Before executing this template, you would need to replace `[MY_PROJECT]` with your project ID and `[FAMILY_NAME]` with the name of a Debian image family, such as debian-9. You can find a list of images in the Compute Engine section of Cloud Console in the Images tab. You can also list images using the `gcloud compute images list` command.

Deployment Manager Template Files

If your deployment configurations are becoming complicated, you can use deployment templates. Templates are another text file you use to define resources and import those resources into configuration files. This allows you to reuse resource definitions in multiple places. Templates can be written in Python or Jinja2, a templating language.

For information on Jinja2 syntax, see the official documentation at `http://jinja.pocoo.org/docs/2.10/`.

As an Associate Cloud Engineer, you should know that Google recommends using Python to create template files unless the templates are relatively simple, in which case it is appropriate to use Jinja2.

Launching a Deployment Manager Template

You can launch a deployment template using the `gcloud deployment-manager deployments create` command. For example, to deploy the template from the Google documentation, use the following:

```
gcloud deployment-manager deployments create quickstart-deployment --config
vm.yaml
```

You can also describe the state of a deployment with the `describe` command, as follows:

```
gloud deployment-manager deployments describe quickstart-deployment
```

Real World Scenario

Providing a Deployable Service

In large enterprises, different groups often want to use the same service, such as a data science application, to understand customer purchasing patterns. Product managers across the organization may want to use this. Software developers could create a single instance of the applications resources and have multiple users work with that one instance. This is a co-hosted structure. This has some advantages if you have a single DevOps team supporting all users.

Alternatively, you could allow each user or small group of users to have their own application instance. This has several advantages. Users could run the application in their own projects, simplifying allocating charges for resources, since the project would be linked to the users' billing accounts. Also, users could scale the resources up or down as needed for their use case.

A potential disadvantage is that users may not be comfortable configuring Google Cloud resources. Deployment Manager addresses that problem by making it relatively simple to deploy an application and resources in a repeatable process. Someone who can run a `gcloud deployment-manager` command could deploy application resources similar to the way users deploy applications from Cloud Launcher.

Summary

Cloud Launcher and Cloud Deployment Manager are designed to make it easier to deploy resources in GCP. Cloud Launcher is where third-party vendors can offer deployable applications based on proprietary or open source software. When an application is

deployed from Cloud Launcher, resources such as VMs, storage buckets, and persistent disks are created automatically without additional human intervention. Deployment Manager gives cloud engineers the ability to define configuration files that describe the resources they would like to deploy. Once defined, cloud engineers can use `gcloud` commands to deploy the resources and list their status. The Deployment Manager is especially useful in organizations where you want to easily deploy resources without requiring users of those resources to understand the details of how to configure GCP resources.

Exam Essentials

Understand how to browse for solutions using the Cloud Launcher section of Cloud Console. You can use filters to narrow your search to specific kinds of solutions, such as operating systems and developer tools. There may be multiple options for a single application, such as WordPress. This is because multiple vendors provide configurations. Review the description of each to understand which best fits your needs.

Know how to deploy a solution in Cloud Launcher. Understand how to configure a Cloud Launcher deployment in Cloud Console. Understand that when you launch a solution, you may be prompted for application specific configurations. For example, with WordPress you may be prompted to install phpMyAdmin. You may also have the opportunity to configure common configuration attributes, such as the machine type and boot disk type.

Understand how to use the Deployment Manager section of the console to monitor deployment. It may be a few minutes from the time you launch a configuration to the time it is ready to use. Note that once the application is ready, you may be provided additional information, such as a username and password to log in.

Know that Deployment Manager is a GCP service for creating configuration files that define resources to use with an application. These configuration files use YAML syntax. They are made up of resource specifications that use key-value pairs to define properties of the resource.

Know that resources in a configuration file are defined using a name, type, and set of properties. The properties vary by type. The machine type can be defined using just a URL that points to a type of machine available in a region. Disks have multiple properties, including a device name, a type, and whether the disk is a boot disk.

If your configuration files are getting long or complicated, you can modularize them using templates. Templates define resources and can be imported into other templates. Templates are text files written in Jinja2 or Python.

Know that you can launch a deployment configuration file using `gcloud deployment-manager` deployments create. You can review the status of a deployment using `gcloud deployment-manager deployments-describe`.

Review Questions

1. What are the categories of Cloud Launcher solutions?
   - **A.** Data sets only
   - **B.** Operating systems only
   - **C.** Developer tools and operating systems only
   - **D.** Data sets, operating systems, and developer tools
2. What is the other name of Cloud Launcher?
   - **A.** Cloud Deployment Manager
   - **B.** Marketplace
   - **C.** Cloud Tools
   - **D.** Cloud Solutions: Third Party
3. Where do you navigate to launch a Cloud Launcher solution?
   - **A.** Overview page of the solution
   - **B.** Main Cloud Launcher page
   - **C.** Network Services
   - **D.** None of the above
4. You want to quickly identify the set of operating systems available in Cloud Launcher. Which of these steps would help with that?
   - **A.** Use Google Search to search the Web for a listing.
   - **B.** Use filters in Cloud Launcher.
   - **C.** Scroll through the list of solutions displayed on the start page of Cloud Launcher.
   - **D.** It is not possible to filter to operating systems.
5. You want to use Cloud Launcher to deploy a WordPress site. You notice there is more than one WordPress option. Why is that?
   - **A.** It's a mistake. Submit a ticket to Google support.
   - **B.** Multiple vendors may offer the same application.
   - **C.** It's a mistake. Submit a ticket to the vendors.
   - **D.** You will never see such an option.
6. You have used Cloud Launcher to deploy a WordPress site and would now like to deploy a database. You notice that the configuration form for the databases is different from the form used with WordPress. Why is that?
   - **A.** It's a mistake. Submit a ticket to Google support.
   - **B.** You've navigated to a different subform of Cloud Launcher.
   - **C.** Configuration properties are based on the application you are deploying and will be different depending on what application you are deploying.
   - **D.** This cannot happen.

7. You have been asked by your manager to deploy a WordPress site. You expect heavy traffic, and your manager wants to make sure the VM hosting the WordPress site has enough resources. Which resources can you configure when launching a WordPress site using Cloud Launcher?
   A. Machine type
   B. Disk type
   C. Disk size
   D. All of the above

8. You would like to define as code the configuration of a set of application resources. The GCP service for creating resources using a configuration file made up of resource specifications defined in YAML syntax is called what?
   A. Compute Engine
   B. Deployment Manager
   C. Marketplace Manager
   D. Marketplace Deployer

9. What file format is used to define Deployment Manager configuration files?
   A. XML
   B. JSON
   C. CSV
   D. YAML

10. A Deployment Manager configuration file starts with what term?
   A. Deploy
   B. Resources
   C. Properties
   D. YAML

11. Which of the following are used to define a resource in a Cloud Deployment Manager configuration file?
   A. `type only`
   B. `properties only`
   C. `name and type only`
   D. `type, properties, and name`

12. What properties may be set when defining a disk on a VM?
   A. A device name only
   B. A Boolean indicating a boot disk and a Boolean indicating autodelete
   C. A Boolean indicating autodelete only
   D. A device name, a Boolean indicating a boot disk, and a Boolean indicating autodelete

13. You need to look up what images are available in the zone in which you want to deploy a VM. What command would you use?
    A. `gcloud compute images list`
    B. `gcloud images list`
    C. `gsutil compute images list`
    D. `gcloud compute list images`

14. You want to use a template file with Deployment Manager. You expect the file to be complicated. What language would you use?
    A. Jinja2
    B. Ruby
    C. Go
    D. Python

15. What command launches a deployment?
    A. `gcloud deployment-manager deployments create`
    B. `gcloud cloud-launcher deployments create`
    C. `gcloud deployment-manager deployments launch`
    D. `gcloud cloud-launcher deployments launch`

16. A DevOps engineer is noticing a spike in CPU utilization on your servers. You explain you have just launched a deployment. You'd like to show the DevOps engineer the details of a deployment you just launched. What command would you use?
    A. `gcloud cloud-launcher deployments describe`
    B. `gcloud deployment-manage deployments list`
    C. `gcloud deployment-manager deployments describe`
    D. `gcloud cloud-launcher deployments list`

17. If you expand the More link in the Networking section when deploying a Cloud Launcher solution, what will you be able to configure?
    A. IP addresses
    B. Billing
    C. Access controls
    D. Custom machine type

18. What are the license types referenced in Cloud Launcher?
    A. Free only
    B. Free and Paid only
    C. Free and Bring your own license (BYOL) only
    D. Free, paid, and bring your own license

19. Which license type will add charges to your GCP bill when using Cloud Launcher with this type of license?

A. Free

B. Paid

C. BYOL

D. Chargeback

20. You are deploying a Cloud Launcher application that includes a LAMP stack. What software will this deploy?

A. Apache server and Linux only

B. Linux only

C. MySQL and Apache only

D. Apache, MySQL, Linux, and PHP

Chapter 17

Configuring Access and Security

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ 5.1 Managing Identity and Access Management (IAM)
- ✓ 5.2 Managing service accounts
- ✓ 5.3 Viewing audit logs for project and managed services

Official Google Cloud Certified: Associate Cloud Engineer: Study Guide
By Dan Sullivan

Google Cloud engineers can expect to spend a significant amount of time working with access controls. This chapter provides instruction on how to perform several common tasks, including managing identity and access management (IAM) assignments, creating custom roles, managing service accounts, and viewing audit logs.

It is important to know that the preferred way of assigning permissions to users, groups, and service accounts is through the IAM system. However, Google Cloud did not always have IAM. Before that, permissions were granted using what are now known as primitive roles, which are fairly coarse-grained. Primitive roles may have more permissions than you want an identity to have. You can constrain permissions using scopes. In this chapter, we will describe how to use primitive roles and scopes as well as IAM. Going forward, it is a best practice to use IAM for access control.

Managing Identity and Access Management

When you work with IAM, there are a few common tasks you need to perform:

- Viewing account IAM assignments
- Assigning IAM roles
- Defining custom roles

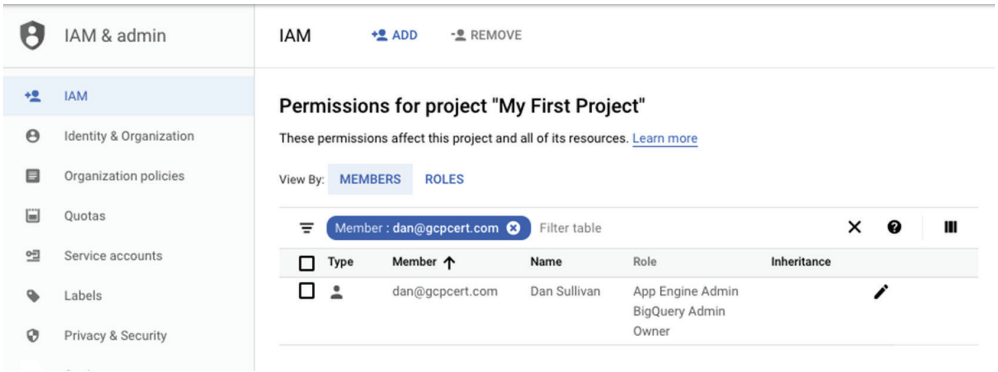
Let's look at how to perform each of these tasks.

Viewing Account Identity and Access Management Assignments

You can view account IAM assignments in Cloud Console by navigating to the IAM & Admin section. In that section, select IAM from the navigation menu to display a form such as the one shown in Figure 17.1. The example in the figure shows a list of identities filtered by member name.

In this example, the user `dan@gcpcert.com` has three roles: App Engine Admin, BigQuery Admin, and Owner. App Engine Admin and BigQuery Admin are predefined IAM roles. Owner is a primitive role.

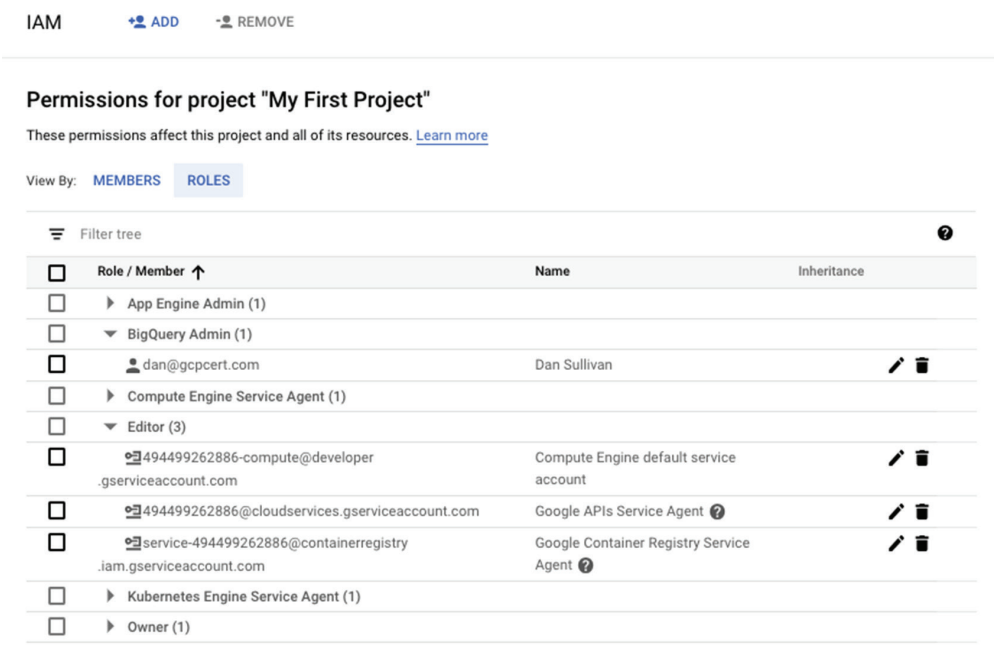
FIGURE 17.1 Permissions listing filtered by member



Primitive roles were used prior to IAM. There are three primitive roles: owner, editor, and viewer. Viewers have permission to perform read-only operations. Editors have viewer permissions and permission to modify an entity. Owners have editor permissions and can manage roles and permission on an entity. Owners can also set up billing for a project.

IAM roles are collections of permissions. They are tailored to provide identities with just the permissions they need to perform a task and no more. To see a list of users assigned a role, click the Roles tab in the IAM form. This will show a display similar to Figure 17.2.

FIGURE 17.2 List of identities assigned to App Engine Admin and Editor



This form shows a list of roles with the number of identities assigned to that role in parentheses. Click the arrow next to the name of a role to display a list of identities with that role. Notice that both primitive and IAM predefined roles are included in this list.

You can also see a list of users and roles assigned across a project using the command `gcloud projects get-iam-policy`. For example, to list roles assigned to users in a project with the project ID `ace-exam-project`, use this:

```
gcloud projects get-iam-policy ace-exam-project
```

Predefined roles are grouped by service. For example, App Engine has five roles:

- App Engine Admin, which grants read, write, and modify permission to application and configuration settings. The role name used in `gcloud` commands is `roles/appengine.appAdmin`.
- App Engine Service Admin, which grants read-only access to configuration settings and write access to module-level and version-level settings. The role name used in `gcloud` commands is `roles/appengine.serviceAdmin`.
- App Engine Deployer, which grants read-only access to application configuration and settings and write access to create new versions. Users with only the App Engine Deployer role cannot modify or delete existing versions. The role name used in `gcloud` commands is `roles/appengine.deployer`.
- App Engine Viewer, which grants read-only access to application configuration and settings. The role name used in `gcloud` commands is `roles/appengine.appViewer`.
- App Engine Code Viewer, which grants read-only access to all application configurations, settings, and deployed source code. The role name used in `gcloud` commands is `roles/appengine.codeViewer`.

Although you do not have to know all of them, it helps to review predefined roles to understand patterns of how they are defined. For more details, see the Google Cloud documentation at: `https://cloud.google.com/iam/docs/understanding-roles`.

Assigning Identity and Access Management Roles to Accounts and Groups

To add IAM roles to accounts and groups, navigate to the IAM & Admin section of the console. Select IAM from the menu. Click the Add link at the top to display a form like that shown in Figure 17.3.

Specify the name of a user or group in the parameter labeled New Members. Click Select A Role to add a role. You can add multiple roles. When you click the down arrow in the Role parameter, you will see a list of services and their associated roles. You can choose the roles from that list. See Figure 17.4 for an example of a subset of the list, showing the roles for BigQuery.

FIGURE 17.3 The Add option in IAM is where you can assign users or groups one or more roles.

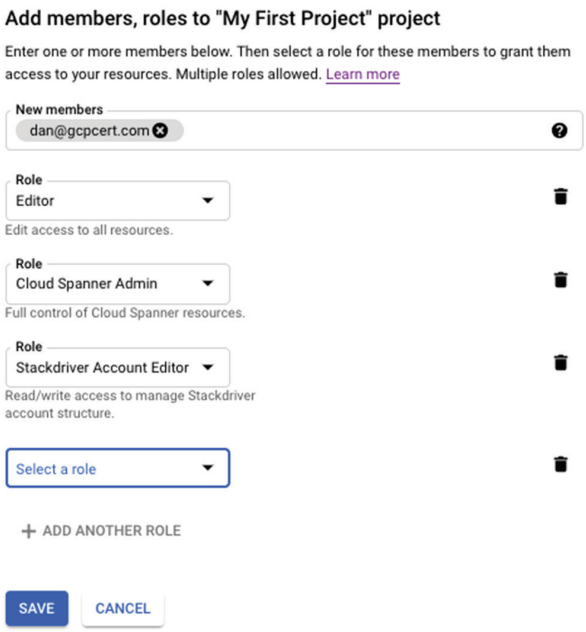
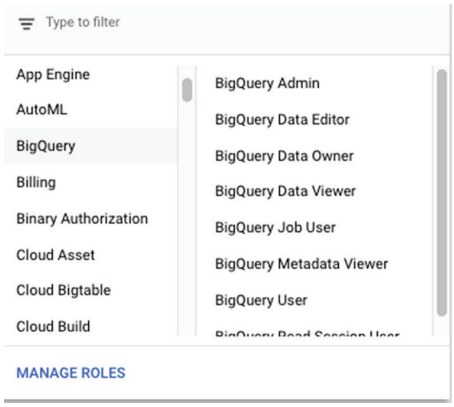


FIGURE 17.4 The drop-down list in the Roles parameters shows available roles grouped by service.



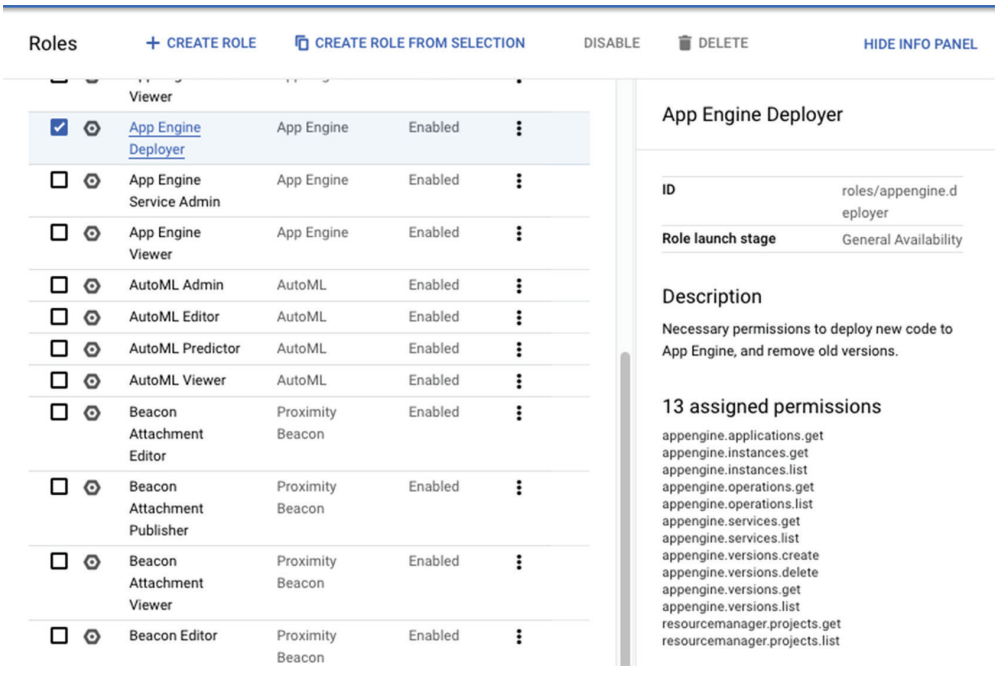
If you want to know which of the fine-grained permissions are granted when you assign a role, you can list those permissions at the command line or in the console. You can also see what permissions are assigned to a role using the command `gcloud iam roles describe`. For example, Figure 17.5 shows the list of permissions in the App Engine Deployer role.

FIGURE 17.5 An example listing permissions using the `gcloud iam roles describe` command

```
$gcloud iam roles describe roles/appengine.deployer
description: Necessary permissions to deploy new code to App Engine, and remove old
  versions.
etag: AA==
includedPermissions:
- appengine.applications.get
- appengine.instances.get
- appengine.instances.list
- appengine.operations.get
- appengine.operations.list
- appengine.services.get
- appengine.services.list
- appengine.versions.create
- appengine.versions.delete
- appengine.versions.get
- appengine.versions.list
- resourcemanager.projects.get
- resourcemanager.projects.list
name: roles/appengine.deployer
stage: GA
title: App Engine Deployer
```

You can also use Cloud Console to view permissions. Navigate to the IAM & Admin section and select Roles from the menu. This displays a list of roles. Click the checkbox next to a role name to display a list of permissions on the right, as shown in Figure 17.6 for App Engine Deployer.

FIGURE 17.6 An example listing of permissions available for App Engine Deployer using Cloud Console



You can assign roles to a member in a project using the following command:

```
gcloud projects add-iam-policy-binding [RESOURCE-NAME] --member user:[USER-
EMAIL] --role [ROLE-ID]
```

For example, to grant the role App Engine Deployer to a user identified by `jane@acexam.com`, you could use this:

```
gcloud projects add-iam-policy-binding ace-exam-project --member user:jane@
aceexam.com --role roles/appengine.deployer
```

Real World Scenario

IAM Roles Support Least Privilege and Separation of Duties

Two security best practices are assigning least privileges and maintaining a separation of duties. The principle of least privileges says you grant only the smallest set of permissions that is required for a user or service account to perform their required tasks. For example, if users can do everything they need to do with only read permission to a database, then they should not have write permission.

In the case of separation of duties, the idea is that a single user should not be able to perform multiple sensitive operations that together could present a risk. In high-risk domains, such as finance or defense, you would not want a developer to be able to modify an application and deploy that change to production without review. A malicious engineer, for example, could modify code in a finance application to suppress application logging when funds are transferred to a bank account controlled by the malicious engineer. If that engineer were to put that code in production, it could be some time before auditors discover that logging has been suppressed and there may have been fraudulent transactions.

IAM roles support least privilege by assigning minimal permissions to predefined roles. It also supports separation of duties by allowing some users to have the ability to change code and others to deploy code.

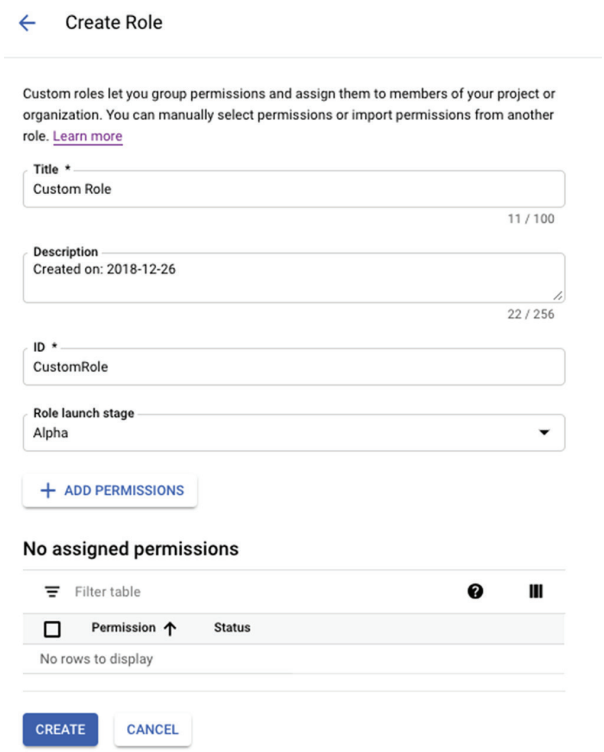
Another common security practice is defense in depth, which applies multiple, overlapping security controls. That is also a practice that should be adopted. IAM can be applied as one of the layers of defense.

Defining Custom Identity and Access Management Roles

If the set of predefined IAM roles does not meet your needs, you can define a custom role.

To define a custom role in Cloud Console, navigate to the Roles option in the IAM & Admin section of the console. Click the Create Role link at the top of the page. This will display a form like that shown in Figure 17.7.

FIGURE 17.7 Creating a role in Cloud Console



In this form you can specify a name for the custom role, a description, an identifier, a launch stage, and a set of permissions. The launch stage options are as follows: Alpha, Beta, General Availability, and Disabled.

You can click Add Permissions to display a list of permissions. The list in Figure 17.8 is filtered to include only permissions in the App Engine Admin role.

Although the list includes all permissions in the role, not all permissions are available for use in a custom role. For example, `appenngine.runtimes.actAsAdmin` is not available for custom roles. When a permission is not available, its status is listed as Not Supported. Permissions that are available for use are listed as Supported, so in the example all other permissions are available. Check the boxes next to the permissions you want to include in your custom role. Click Add to return to the Create Role form, where the list of permissions will now include the permissions you selected (see Figure 17.9).

FIGURE 17.8 List of available permissions filtered by role

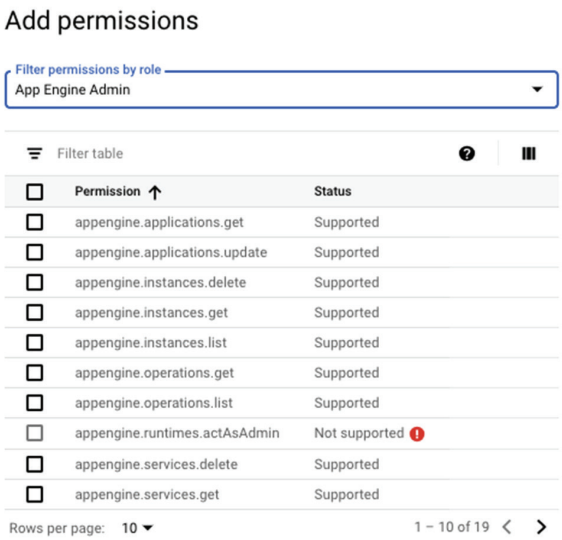
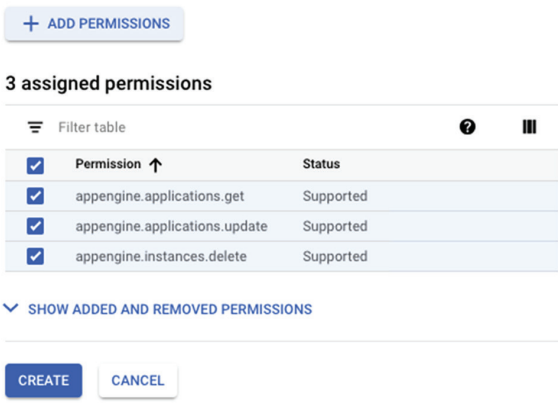


FIGURE 17.9 The permissions section of the Create Role form with permissions added



You can also define a custom role using the `gcloud iam roles create` command. The structure of that command is as follows:

```
gcloud iam roles create [ROLE-ID] --project [PROJECT-ID] --title [ROLE-TITLE] \
--description [ROLE-DESCRIPTION] --permissions [PERMISSIONS-LIST] --stage [LAUNCH-STAGE]
```

For example, to create a role that has only App Engine application update permission, you could use the following command:

```
gcloud iam roles create customAppEngine1 --project ace-exam-project
--title='Custom Update App Engine' \
--description='Custom update' --permissions=appengine.applications.update
--stage=alpha
```

Managing Service Accounts

Service accounts are used to provide identities independent of human users. Service accounts are identities that can be granted roles. Service accounts are assigned to VMs, which then use the permissions available to the service accounts to carry out tasks.

Three things cloud engineers are expected to know how to do are working with scopes, assigning service accounts to VMs, and granting access to a service account to another project.

Managing Service Accounts with Scopes

Scopes are permissions granted to a VM to perform some operation. Scopes authorize the access to API methods. The service account assigned to a VM has roles associated with it. To configure access controls for a VM, you will need to configure both IAM roles and scopes. We have discussed how to manage IAM roles, so now we will turn our attention to scopes.

A scope is specified using a URL that starts with `https://www.googleapis.com/auth/` and is then followed by permission on a resource. For example, the scope allowing a VM to insert data into BigQuery is as follows:

```
https://www.googleapis.com/auth/bigquery.insertdata
```

The scope that allows viewing data in Cloud Storage is as follows:

```
https://www.googleapis.com/auth/devstorage.read_only
```

And to write to Compute Engine logs, use this:

```
https://www.googleapis.com/auth/logging.write
```

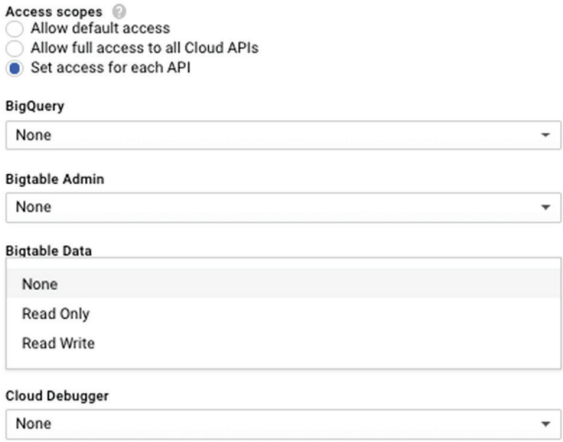
An instance can only perform operations allowed by both IAM roles assigned to the service account and scopes defined on the instance. For example, if a role grants only read-only access to Cloud Storage but a scope allows write access, then the instance will not be able to write to Cloud Storage.

To set scopes in an instance, navigate to the VM instance page in Cloud Console. Stop the instance if it is running. On the Instance Detail page, click the Edit link. At the bottom of the Edit page, you will see the Access Scopes section, as shown in Figure 17.10.

FIGURE 17.10 Access Scopes section in VM instance details edit page

The options are Allow Default Access, Allow Full Access To All Cloud APIs, and Set Access For Each API. Default access is usually sufficient. If you are not sure what to set, you can choose Allow Full Access, but be sure to assign IAM roles to limit what the instance can do. If you want to choose scopes individually, choose Set Access For Each API. This will display a list of services and scopes like that shown in Figure 17.11.

FIGURE 17.11 A partial list of services and scopes that can be individually configured



You can also set scopes using the `gcloud compute instances set-service-account` command. The structure of the command is as follows:

```
gcloud compute instances set-service-account [INSTANCE_NAME] \
     [--service-account [SERVICE_ACCOUNT_EMAIL] | [--no-service-account] \
     [--no-scopes | --scopes [SCOPES,...]]
```

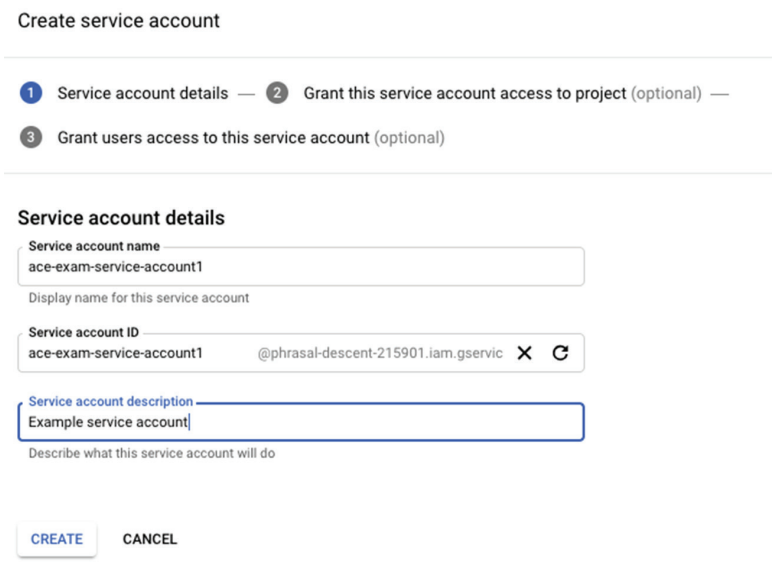
An example scope assignment using `gcloud` is as follows:

```
gcloud compute instances set-service-account ace-instance \
     --service-account examadmin@ace-exam-project.iam.gserviceaccount.com \
     --scopes compute-rw,storage-ro
```

Assigning a Service Account to a Virtual Machine Instance

You can assign a service account to a VM instance. First, create a service account by navigating to the Service Accounts section of the IAM & Admin section of the console. Click Create Service Account to display a form like that shown in Figure 17.12.

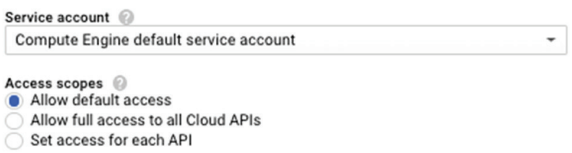
FIGURE 17.12 Creating a service account in the console



After specifying a name, identifier, and description, click Create. Next, you can assign roles as described earlier, using the console or `gcloud` commands. Once you have assigned the roles you want the service account to have, you can assign it to a VM instance.

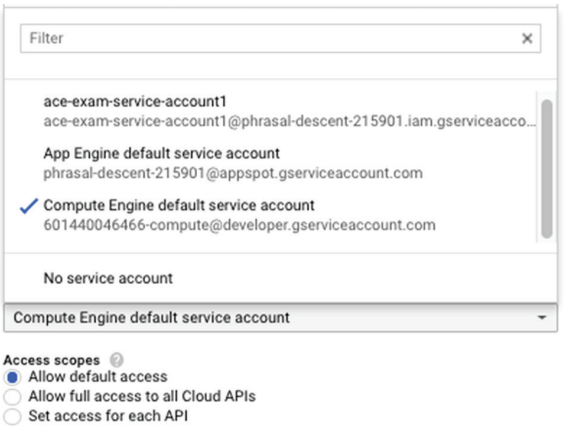
Navigate to the VM Instances page in the Compute Engine section of the console. Select a VM instance and click Edit. This will display a form with a parameter for the instance. Scroll down to see the parameter labeled Service Account (see Figure 17.13).

FIGURE 17.13 Section of Edit Instance page showing the Service Account parameter



From the drop-down list, select the service account you want assigned to that instance, as shown in Figure 17.14.

FIGURE 17.14 List of service accounts that can be assigned to the instance



You can also specify a service instance at the command line when you create an instance using the `gcloud compute instance create` command. It has the following structure:

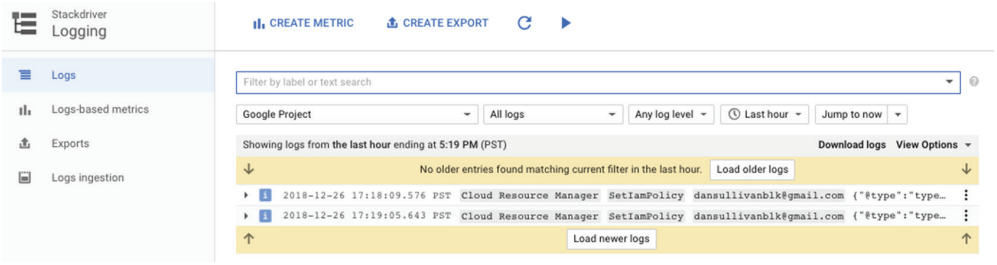
```
gcloud compute instances create [INSTANCE_NAME]  --service-account [SERVICE_
ACCOUNT_EMAIL]
```

To grant access to a project, navigate to the IAM page of the console and add a member. Use the service accounts email as the entity to add.

Viewing Audit Logs

To view audit logs, navigate to the Stackdriver Logging page in Cloud Console. This will show a listing like that in Figure 17.15.

FIGURE 17.15 Default listing of the Stackdriver Logging page



Notice you can select the resource, types of logs to display, the log level, and the period of time from which to display entities.

For additional information on logging, see Chapter 18.

Summary

Access controls in GCP are managed using IAM, primitive roles, and scopes. The three primitive roles are owner, editor, and viewer. They provide coarse-grained access controls to resources. Scopes are access controls that apply to instances of VMs. They are used to limit operations that can be performed by an instance. The set of operations that an instance can perform is determined by the scopes assigned and the roles assigned to a service account used by the instance. IAM provides predefined roles. These roles are grouped by service. The roles are designed to provide the minimal set of permissions needed to carry out a logical task, such as writing to a bucket or deploying an App Engine application. When predefined roles do not meet your needs, you can define custom roles.

Service accounts are used to enable VMs to perform operations with a set of permissions. The permissions are granted to service accounts through the roles assigned to the service account. You can use the default service account provided by GCP for an instance or you can assign your own.

Exam Essentials

Know the three types of roles: primitive, predefined, and custom. Primitive roles include owner, editor, and viewer. These were developed prior to the release of IAM. Predefined roles are IAM roles. Permissions are assigned to these roles, and then the roles are assigned to users, groups, and service accounts. Custom roles include permissions selected by the user creating the custom role.

Understand that scopes are a type of access control applied to VM instances. The VM can only perform operations allowed by scopes and IAM roles assigned to the service account of the instance. You can use IAM roles to constrain scopes and use scopes to constrain IAM roles.

Know how to view roles assigned to identities. You can use the Roles tab in the IAM & Admin section of the console to list the identities assigned particular roles. You can also use `gcloud projects get-iam-policy` command to list roles assigned to users in a project.

Understand that IAM roles support separation of duties and the principle of least privilege. Primitive roles did not support least privilege and separation of duties because they are too coarse-grained. Separation of duties ensures that two or more people are required to complete a sensitive task.

Know how to use `gcloud iam roles describe` to view details of a role, including permissions assigned to a role. You can also view users granted roles by drilling down into a role in the Roles page of the IAM & Admin section of the console. When working with IAM, you will be using the `gcloud` command when working from the command line.

Understand the different options for accessing scopes when creating an instance. The options are Default Access, Full Access, and Set Access for Each API. If you aren't sure which to use, you can grant full access, but be sure to limit what the instance can do by assigning roles that constrain allowed operations.

Know that Stackdriver Logging collects logging events. They can be filtered and displayed in the Logging section of Cloud Console. You can filter by resource, type of log, log level, and period of time to display.

Review Questions

You can find the answers in the Appendix.

1. What does IAM stand for?
   - **A.** Identity and Authorization Management
   - **B.** Identity and Access Management
   - **C.** Identity and Auditing Management
   - **D.** Individual Access Management
2. When you navigate to IAM & Admin in Cloud Console, what appears in the main body of the page?
   - **A.** Members and roles assigned
   - **B.** Roles only
   - **C.** Members only
   - **D.** Roles and permissions assigned
3. Why are primitive roles classified in a category in addition to IAM?
   - **A.** They are part of IAM.
   - **B.** They were created before IAM.
   - **C.** They were created after IAM.
   - **D.** They are not related to access control.
4. A developer intern is confused about what roles are used for. You describe IAM roles as a collection of what?
   - **A.** Identities
   - **B.** Permissions
   - **C.** Access control lists
   - **D.** Audit logs
5. You want to list roles assigned to users in a project called ace-exam-project. What `gcloud` command would you use?
   - **A.** `gcloud iam get-iam-policy ace-exam-project`
   - **B.** `gcloud projects list ace-exam-project`
   - **C.** `gcloud projects get-iam-policy ace-exam-project`
   - **D.** `gcloud iam list ace-exam-project`

6. You are working in the form displayed after clicking the Add link in the IAM form of IAM & Admin in Cloud Console. There is a parameter called New Members. What items would you enter in that parameter?
   - **A.** Individual users only
   - **B.** Individual users or groups
   - **C.** Roles or individual users
   - **D.** Roles or groups

7. You have been assigned the App Engine Deployer role. What operations can you perform?
   - **A.** Write new versions of an application only
   - **B.** Read application configuration and settings only
   - **C.** Read application configuration and settings and write new configurations
   - **D.** Read application configuration and settings and write new versions

8. You want to list permissions in a role using Cloud Console. Where would you go to see that?
   - **A.** IAM & Admin; select Roles. All permissions will be displayed.
   - **B.** IAM & Admin; select Roles. Check the box next to a role to display the permissions in that role.
   - **C.** IAM & Admin; select Audit Logs.
   - **D.** IAM & Admin; select Service Accounts and then Roles.

9. You are meeting with an autidor to discuss security practices in the cloud. The auditor asks how you implement several best practices. You describe how IAM predefined roles help to implement which security best practice(s)?
   - **A.** Least privilege
   - **B.** Separation of duties
   - **C.** Defense in depth
   - **D.** Options A and B

10. What launch stages are available when creating custom roles?
    - **A.** Alpha and beta only
    - **B.** General availability only
    - **C.** Disabled only
    - **D.** Alpha, beta, general availability, and disabled

11. The `gcloud` command to create a custom role is what?
    - **A.** `gcloud project roles create`
    - **B.** `gcloud iam roles create`
    - **C.** `gcloud project create roles`
    - **D.** `gcloud iam create roles`

12. A DevOps engineer is confused about the purpose of scopes. Scopes are access controls that are applied to what kind of resources?
    A. Storage buckets
    B. VM instances
    C. Persistent disks
    D. Subnets

13. A scope is identified using what kind of identifier?
    A. A randomly generated ID
    B. A URL beginning with `https://www.googleserviceaccounts/`
    C. A URL beginning with `https://www.googleapis.com/auth/`
    D. A URL beginning with `https://www.googleapis.com/auth/PROJECT_ID]`

14. A VM instance is trying to read from a Cloud Storage bucket. Reading the bucket is allowed by IAM roles granted to the service account of the VM. Reading buckets is denied by the scopes assigned to the VM. What will happen if the VM tries to read from the bucket?
    A. The application performing the read will skip over the read operation.
    B. The read will execute because the most permissive permission is allowed.
    C. The read will not execute because both scopes and IAM roles are applied to determine what operations can be performed.
    D. The read operation will succeed, but a message will be logged to Stackdriver Logging.

15. What are the options for setting scopes in a VM?
    A. Allow Default Access and Allow Full Access only
    B. Allow Default Access, Allow Full Access, and Set Access for Each API
    C. Allow Full Access or Set Access For Each API only
    D. Allow Default Access and Set Access For Each API only

16. What `gcloud` command would you use to set scopes?
    A. `gcloud compute instances set-scopes`
    B. `gcloud compute instances set-service-account`
    C. `gcloud compute service-accounts set-scopes`
    D. `gcloud compute service-accounts define-scopes`

17. What `gcloud` command would you use to assign a service account when creating a VM?
    A. `gcloud compute instances create [INSTANCE_NAME] --service-account [SERVICE_ACCOUNT_EMAIL]`
    B. `gcloud compute instances create-service-account [INSTANCE_NAME][SERVICE_ACCOUNT_EMAIL]`
    C. `gcloud compute instances define-service-account [INSTANCE_NAME][SERVICE_ACCOUNT_EMAIL]`
    D. `gcloud compute create instances-service-account [INSTANCE_NAME][SERVICE_ACCOUNT_EMAIL]`

18. What GCP service is used to view audit logs?

A. Compute Engine

B. Cloud Storage

C. Stackdriver Logging

D. Custom logging

19. What options are available for filtering log messages when viewing audit logs?

A. Period time and log level only

B. Resource, type of log, log level, and period of time only

C. Resource and period of time only

D. Type of log only

20. An auditor needs to review audit logs. You assign read-only permission to a custom role you create for auditors. What security best practice are you following?

A. Defense in depth

B. Least privilege

C. Separation of duties

D. Vulnerability scanning

Chapter

18

Monitoring, Logging, and Cost Estimating

THIS CHAPTER COVERS THE FOLLOWING OBJECTIVES OF THE GOOGLE ASSOCIATE CLOUD ENGINEER CERTIFICATION EXAM:

- ✓ **4.6 Monitoring and logging**
- ✓ **2.1 Planning and estimating GCP product use using the Pricing Calculator**

Official Google Cloud Certified: Associate Cloud Engineer: Study Guide
By Dan Sullivan

Monitoring system performance is an essential part of cloud engineering. In this chapter, you will learn about Stackdriver, a GCP service for resource monitoring, logging, tracing, and debugging. You will start by creating alerts based on resource metrics and custom metrics. Next, you will turn your attention to logging with a discussion of how to create log sinks to store logging data outside of Stackdriver. You'll also see how to view and filter log data. Stackdriver includes diagnostic tools such as Cloud Trace and Cloud Debugger, which you'll learn about as well. We'll close out the chapter with a review of the Pricing Calculator for estimating the cost of GCP resources and services.

Monitoring with Stackdriver

Stackdriver is a service for collecting performance metrics, logs, and event data from our resources. Metrics include measurements such as the average percent CPU utilization over the past minute and the number of bytes written to a storage device in the last minute. Stackdriver includes many predefined metrics. Some examples are shown in Table 18.1 that you can use to assess the health of your resources and, if needed, trigger alerts to bring your attention to resources or services that are not meeting service-level objectives.

TABLE 18.1 Example Stackdriver metrics

GCP Product	Metric
Compute Engine	CPU utilization
Compute Engine	Disk bytes read
BigQuery	Execution times
Bigtable	CPU load
Cloud Functions	Execution count

Stackdriver works in hybrid environments with support for GCP, Amazon Web Services, and on-premise resources.

Creating Alerts Based on Resource Metrics

Metrics are defined measurements on a resource collected at regular intervals. Metrics return aggregate values, such as the maximum, minimum, or average value of the item measured, which could be CPU utilization, amount of memory used, or number of bytes written to a network interface.

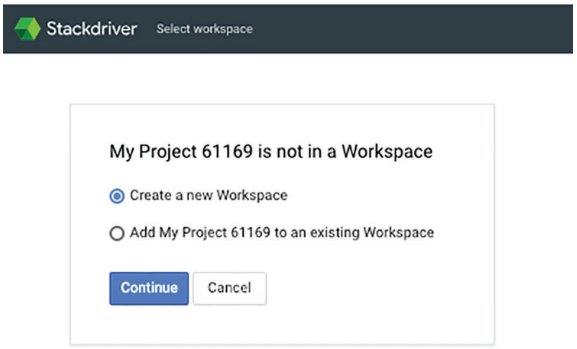
For this example, assume you are working with a VM that has Apache Server and PHP installed. To monitor and collect metrics, you need to install the Stackdriver agent for monitoring. Since you are installing the monitoring agent, you'll install the logging agent at the same time because you'll need that later. To install the Stackdriver monitoring and logging agents on a Linux VM, execute the following command at the shell prompt (note, these are not `gcloud` commands):

```
curl -sSO https://dl.google.com/cloudagents/install-monitoring-agent.sh
sudo bash install-monitoring-agent.sh
curl -sSO https://dl.google.com/cloudagents/install-logging-agent.sh
sudo bash install-logging-agent.sh --structured
```

VMs with agents installed collect monitoring and logging data and send it to Stackdriver. Stackdriver needs a Workspace to store the data.

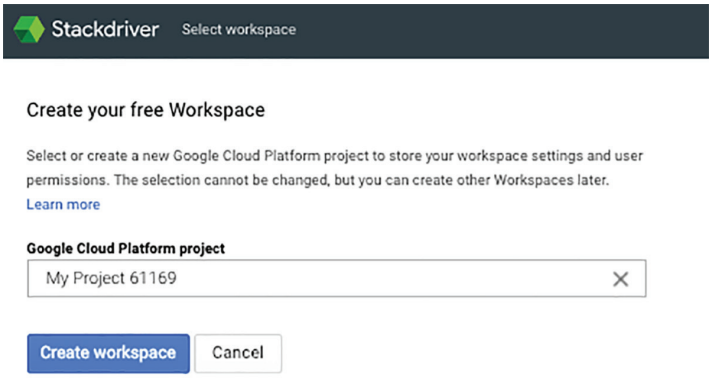
To create a Workspace and initialize it, navigate to the Stackdriver Monitoring section of Cloud Console. If a Workspace does not exist for your project, a form such as that shown in Figure 18.1 will appear.

FIGURE 18.1 Initial form used to create a Workspace in Stackdriver



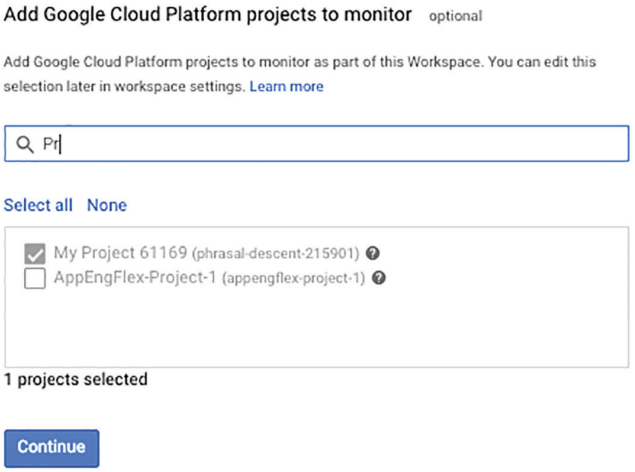
Next, select a project to monitor, as shown in Figure 18.2.

FIGURE 18.2 Selecting a project for the Workspace



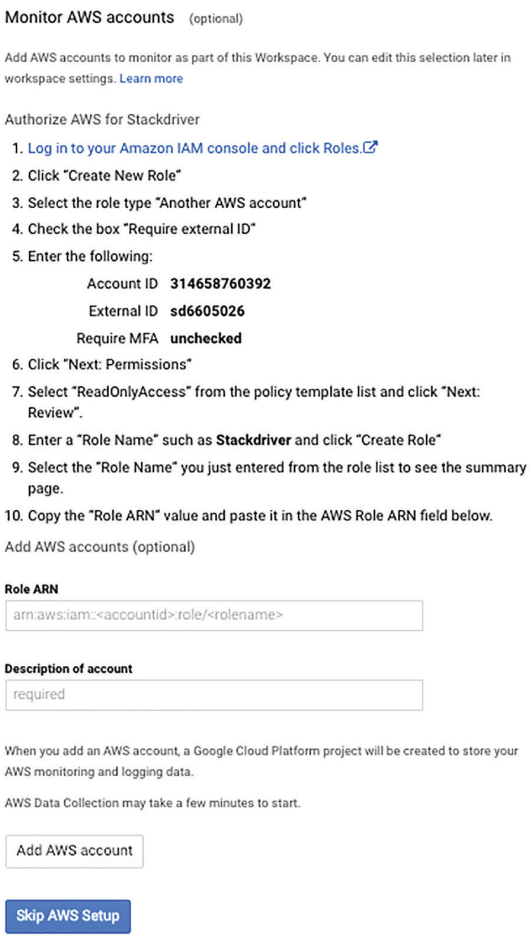
If you want to monitor multiple projects in a Workspace, you can optionally select other projects, as shown in Figure 18.3.

FIGURE 18.3 Optionally adding other projects to monitor



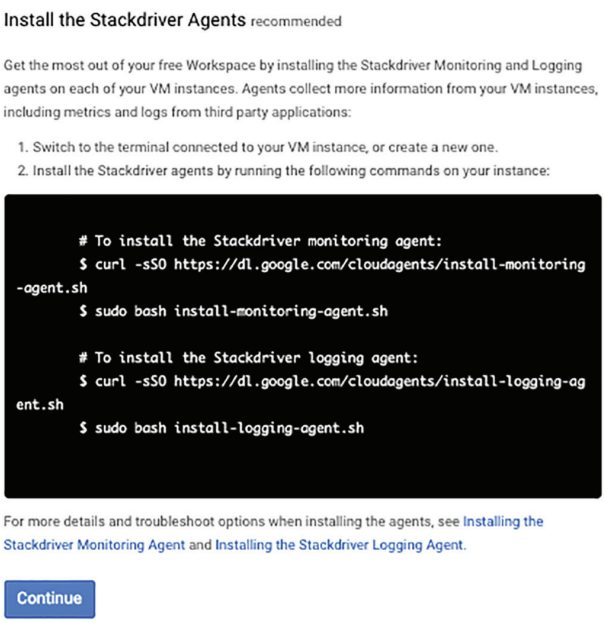
If you want to monitor AWS resources in a Workspace, you can optionally select them as well, as shown in Figure 18.4.

FIGURE 18.4 Optionally monitoring AWS resources



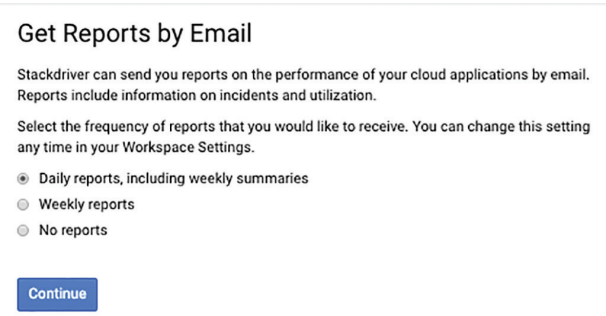
The next step in the initialization process lists commands to install agents (see Figure 18.5).

FIGURE 18.5 Listing of instructions to install agents on servers to be monitored



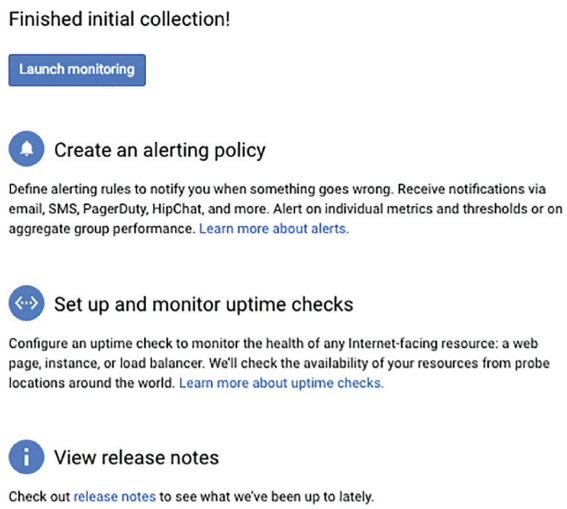
Stackdriver will mail daily or weekly reports if you opt email to have them emailed to you (see Figure 18.6).

FIGURE 18.6 Listing of email reporting options



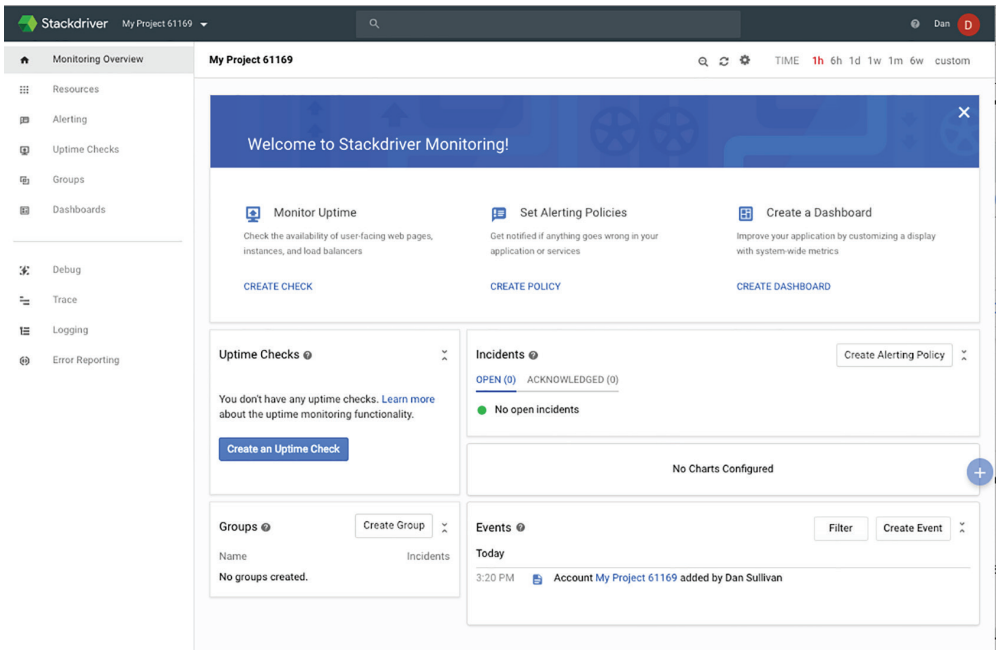
When the initialization process is done, a form similar to the one shown in Figure 18.7 appears, which lists some common tasks, such as adding metrics and viewing release notes.

FIGURE 18.7 The Stackdriver Workspace initialization is complete.



After the Workspace is initialized, navigate to Stackdriver Monitoring to display a Monitoring Overview page, similar to Figure 18.8.

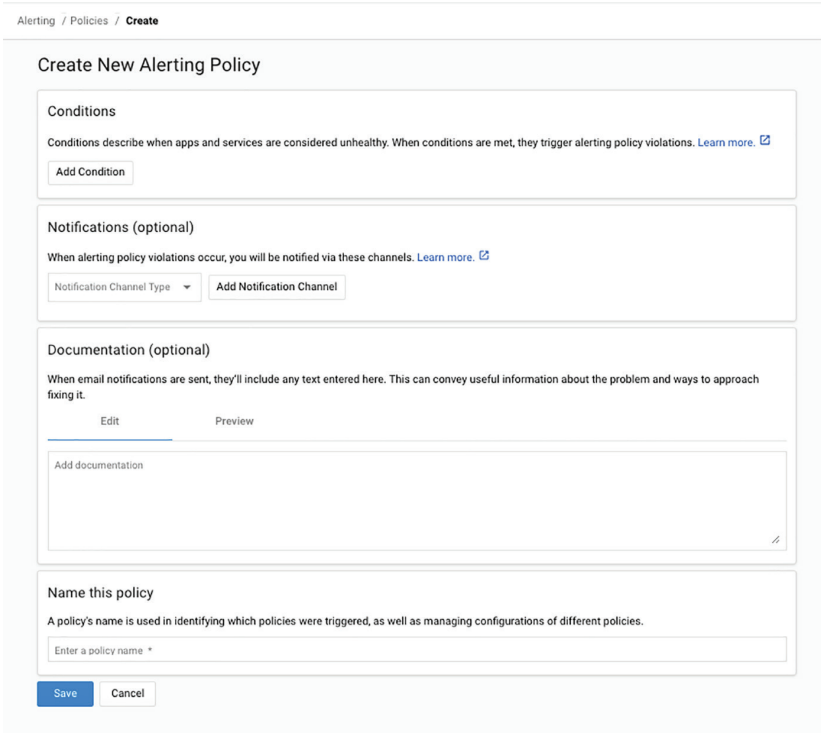
FIGURE 18.8 Monitoring Overview page in Stackdriver



At this point, Stackdriver agents are installed and you have a Workspace available.

Next, create a policy to monitor a metric. A policy consists of conditions that determine when to issue an alert or notification, for example when CPU utilization is greater than 80 percent for more than 5 minutes. Policies also include notification channels and optional documentation, as shown in Figure 18.9. This form is displayed when you click Create Policy from the Monitoring Overview page.

FIGURE 18.9 Creating a new policy for monitoring a metric



Click Add Condition to display a form where you can specify condition parameters. Figure 18.10 shows the Metric Condition form prior to selecting CPU utilization.

After selecting CPU utilization, additional parameters are displayed, as shown in Figure 18.11. The condition will check the CPU utilization status. It will be applied to VMs that match the filter criteria, for example, any VM with a label included in the filter. The filter criteria include VM features such as zone, region, project ID, instance ID, and labels. The Group By parameter allows you to group time series, or data that is produced at regular intervals and has a fixed format, for example by zone, and aggregate the values so there are fewer time series to display. This is especially helpful, for example, if you want to have a group of VMs in a cluster appear as a single time series.

FIGURE 18.10 Selecting a CPU utilization metric

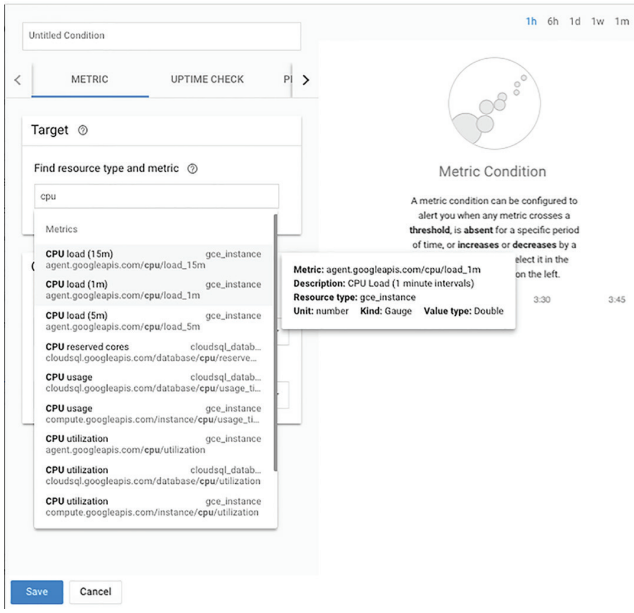
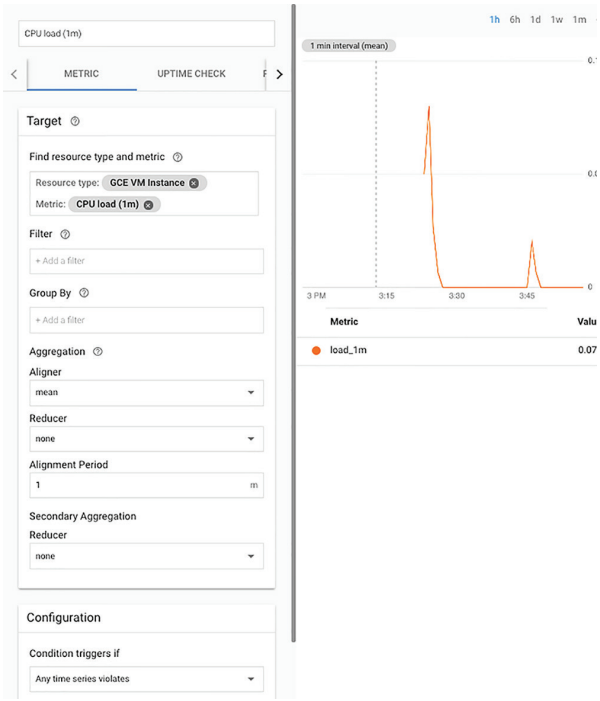
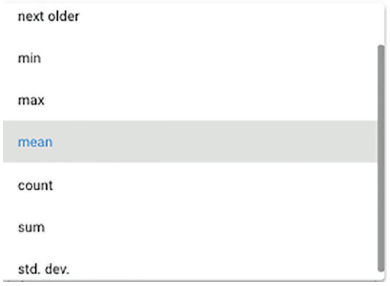


FIGURE 18.11 Additional parameters to configure CPU utilization monitoring



Agents send data from monitored resources to Stackdriver in streams. To perform checks on the streamed data, the data points need to be aggregated at specific time intervals. For example, data points may be received every 20 seconds but for monitoring purposes, we check the average CPU utilization per minute. Consider a stream of CPU utilization metrics that come in to Stackdriver over a 1-minute period. It is useful to consolidate those measures into a single value, like the average, maximum, or minimum value for the set of measures for that minute. This process of grouping data into regular-sized buckets of time is called aligning. Figure 18.12 shows some of the functions, including min, max, and mean, that can be applied to data that arrives within a time bucket.

FIGURE 18.12 Optional aggregates for Aligner

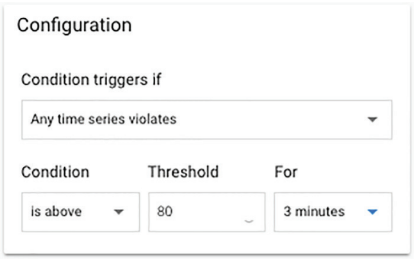


In addition to aligning time series, when you aggregate, you can specify a reducer, which is a function for combining values in a group of time series to produce a single value. The reducers include common statistics, such as sum, min, max, and count (see Figure 18.13).

FIGURE 18.13 Aggregate functions for reducing multiple values to a single value

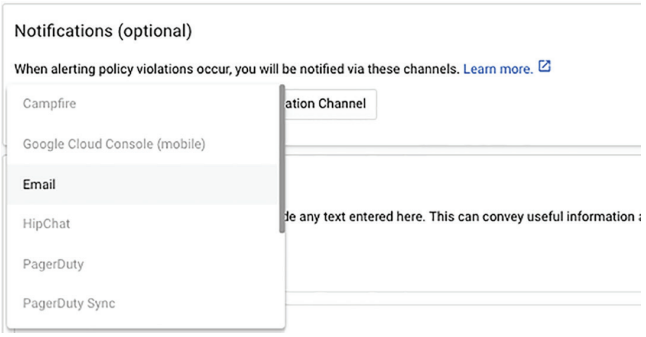
Next, you need to specify when a condition should trigger, as shown in Figure 18.14. This could be anytime you see a value that exceeds the specified threshold, or it could be only if the measured value exceeds a threshold for an extended period of time. For example, you may only want to trigger an alert on CPU utilization if it is above a threshold for more than five minutes. This can help prevent too many alerts from being generated just because there is an occasional but short-term spike in CPU utilization.

FIGURE 18.14 Specifying a threshold above which an alert is triggered



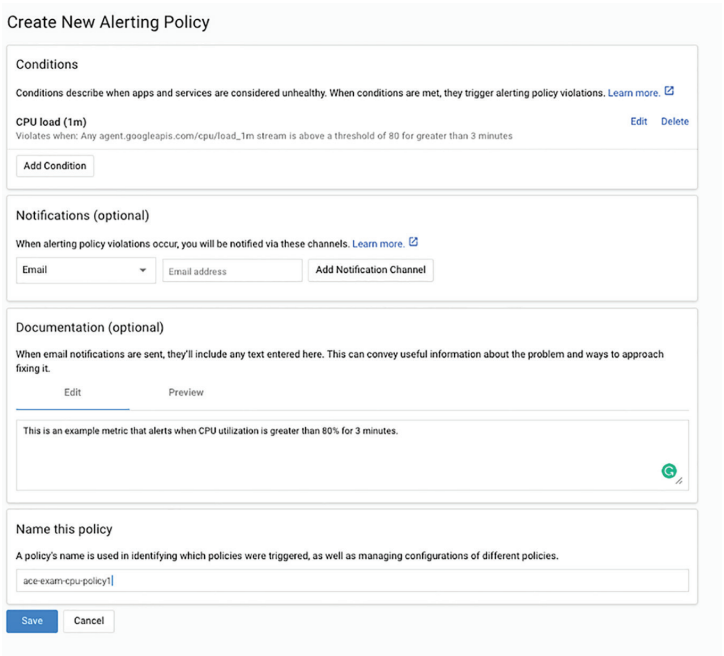
A policy can have one or more notification channels. Channels include email notification as well as Slack, Google Cloud Console (mobile), and popular DevOps tools such as PagerDuty, HipChat, and Campfire (see Figure 18.15).

FIGURE 18.15 Specifying notification channels



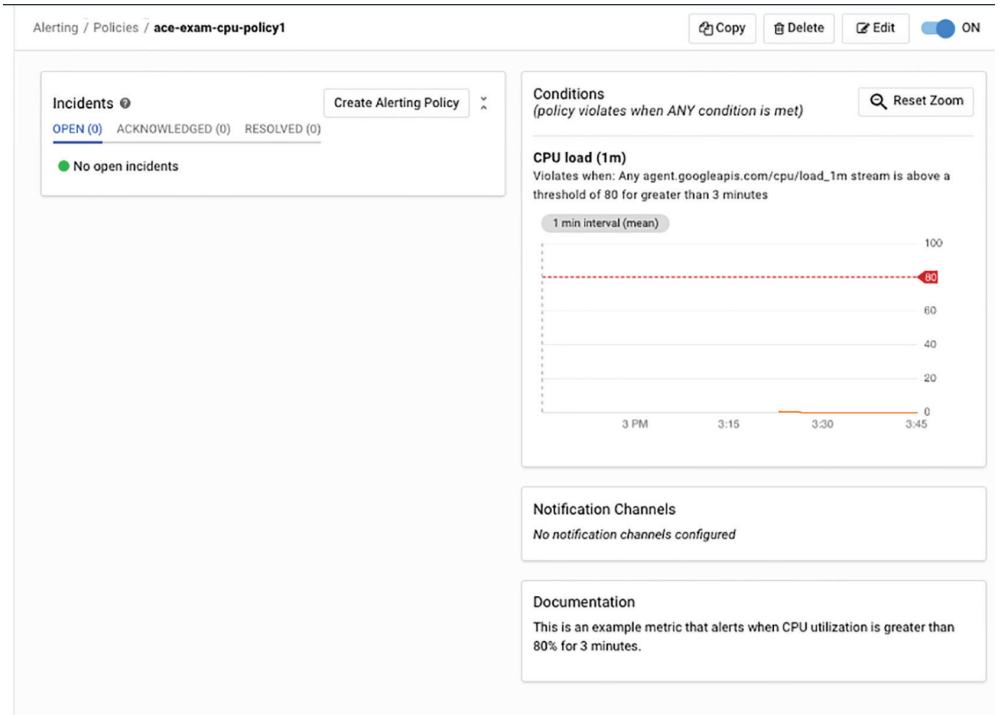
The documentation parameter, shown in Figure 18.16, is optional but recommended. The documentation will be included in notifications, which can help DevOps engineers understand the problem and provide information on how to resolve the issue.

FIGURE 18.16 Adding documentation and a policy name along with a condition and notification specifications



After policies are defined, you can view a summary of the recent history of the metric back to the time when the policy was defined. This includes visualizations such as those shown in Figure 18.17.

FIGURE 18.17 The status of the policy and a display of CPU load in the recent past



Creating Custom Metrics

If there is an application-specific metric you would like to monitor, you can create custom metrics. Custom metrics are like predefined metrics, except you create them. The names of custom metrics start with custom.googleapis.com/, so they are easy to recognize by name. The most important difference is that you can decide what time series data to write to the custom metric.

There are two ways to create custom metrics: using OpenCensus, an open source monitoring library (`https://opencensus.io/`) or using Stackdriver's Monitoring API. OpenCensus provides a higher-level, monitoring-focused API, while the Stackdriver Monitoring API is lower-level.

When you define a custom metric, you will need to specify the following:

- A type name that is unique within the project
- A project
- A display name and description
- A metric kind, such as a gauge, delta, or cumulative metric. Gauges are measures at a point in time, deltas capture the change over an interval, and cumulative are accumulated values over an interval.

- Metric labels
- Monitored resource objects to include with time series data points. These provide the context for a measurement. For example, you could include an application instance ID with an application-specific metric.

To define a custom metric, you will need to program a call to the monitoring API or use the OpenCensus library. How this is done varies by the programming language you use. See Google's Stackdriver documentation for examples using C#, Go, Java, Node.js, PHP, and Python (`https://cloud.google.com/monitoring/custom-metrics/creating-metrics`).

Real World Scenario

Too Many Monitors Are As Bad As Too Few

Be careful when crafting monitoring policies. You do not want to subject DevOps engineers to so many alerts that they begin to ignore them. This is sometimes called alert fatigue. Policies that are too sensitive will generate alerts when no human intervention is required. For example, CPU utilization may regularly spike for brief periods of time. If this is a normal pattern for your environment and it is not adversely impacting your ability to meet service level agreements, then there is little reason to alert on them. Design policies to identify conditions that actually require the attention of an engineer and are not likely to resolve on their own. Use thresholds that are long enough so conditions are not triggered on transient states that will not last long. Often by the time an engineer resolves it, the condition is no longer triggering. Designing policies for monitoring is something of an art. You should assume you will need multiple iterations to tune your policies to find the right balance of generating just the right kinds of useful alerts without also generating alerts that are not helpful.

Logging with Stackdriver

Stackdriver Logging is a service for collecting, storing, filtering, and viewing log and event data generated in GCP and in Amazon Web Services. Logging is a managed service, so you do not need to configure or deploy servers to use the service.

The Associate Cloud Engineering Exam guidelines note three logging tasks a cloud engineer should be familiar with:

- Configuring log sinks
- Viewing and filtering logs
- Viewing message details

We'll review each of these in this section.

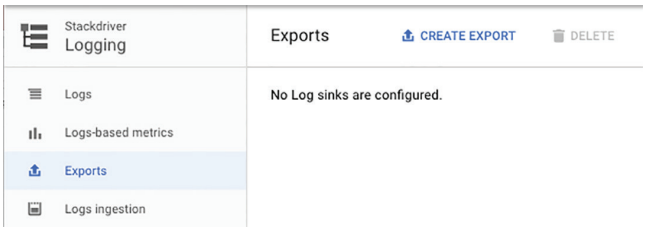
Configuring Log Sinks

Stackdriver Logging retains log data for 30 days. This is sufficient if you use logs to diagnose operational issues but rarely view the logs after a few days. This is often not enough.

Your organization may need to keep logs longer to comply with government or industry regulations. You may also want to analyze logs to gain insight into application performance. For these use cases, it is best to export logging data to a long-term storage system like Cloud Storage or BigQuery.

The process of copying data from Logging to a storage system is called exporting, and the location to which you write the log data is called a sink. You can create a log sink by navigating to the Logging section of Cloud Console and selecting the Exports option from the Logging menu, as shown in Figure 18.18.

FIGURE 18.18 Logging Export form in Cloud Console



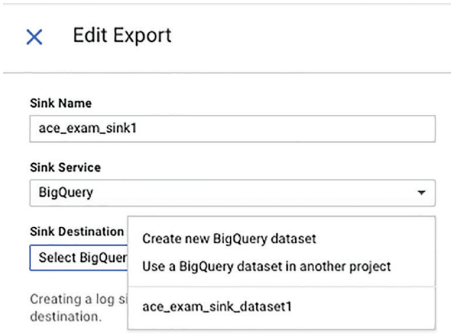
Click Create Export to open a form to create a log sink. The form prompts for three parameters:

- Sink name
- Sink service
- Sink destination

You can make up a sink name, as in Figure 18.19. The sink service is one of the following:

- BigQuery
- Cloud Storage
- Cloud Pub/Sub
- Custom Destination

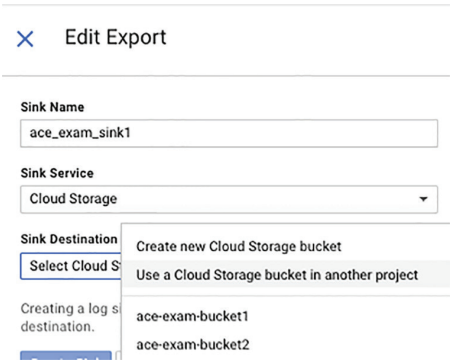
FIGURE 18.19 Creating a BigQuery log sink



If you choose BigQuery as your service, the sink destination will be an existing BigQuery data set or a new data set, as shown in Figure 18.19. When log data is exported to BigQuery, it is organized into tables based on the log name and timestamps. For example, a syslog exported on January 2, 2019, would have the name syslog_20190102. The tables have columns storing the timestamp, log name, and text payload or log message.

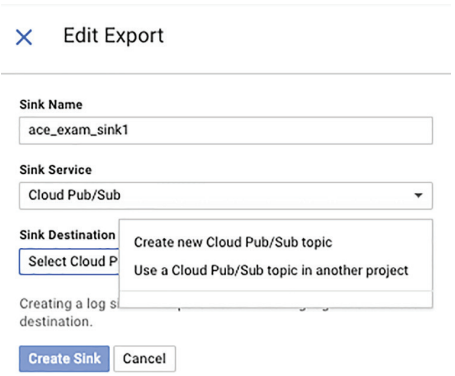
If you choose Cloud Storage, you can export logs to an existing bucket or you can create a new bucket (see Figure 18.20). When log data is exported to Cloud Storage, Logging writes a set of files to the sink bucket. Files are organized hierarchically by log type and date. For example, if a syslog is exported on January 2, 2019, to a bucket named ace-exam-log-sink1, the path to the file would be `ace-exam-log-sink1/syslog/2019/01/02/`.

FIGURE 18.20 Creating a Cloud Storage log sink



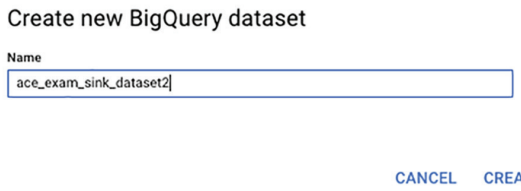
If you choose Cloud Pub/Sub, then you can choose between creating a topic or using an existing one, as shown in Figure 18.21. When log data is exported to Cloud Pub/Sub, the data is encoded in base64 in an object structure known as a LogEntry. LogEntries contain the logname, timestamp, textPayload, and resource properties, such as `type`, `instance_id`, `zone`, and `project_id`.

FIGURE 18.21 Creating a Pub/Sub log sink



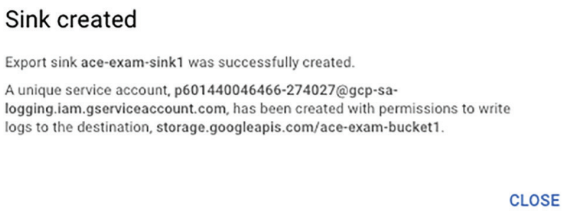
A custom destination is used to specify the name of a project, other than the current project, that is hosting the sink. If you choose to create a new object as a sink, you will be prompted to name the new sink, as shown in Figure 18.22.

FIGURE 18.22 Specifying the name of a new BigQuery data set



After a sink is created, you will receive a message such as the one shown in Figure 18.23, with details on the newly created sink.

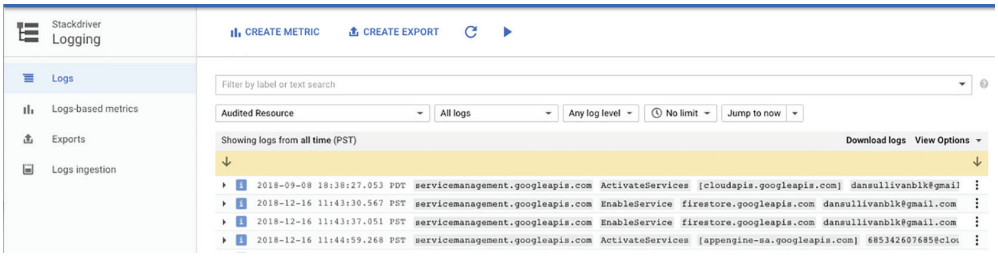
FIGURE 18.23 Confirmation that a new sink has been created



Viewing and Filtering Logs

To view the contents of logs, navigate to the Stackdriver Logging section of the console and select Logs in the Logging menu, as shown in Figure 18.24.

FIGURE 18.24 Listing of log entries in Cloud Console



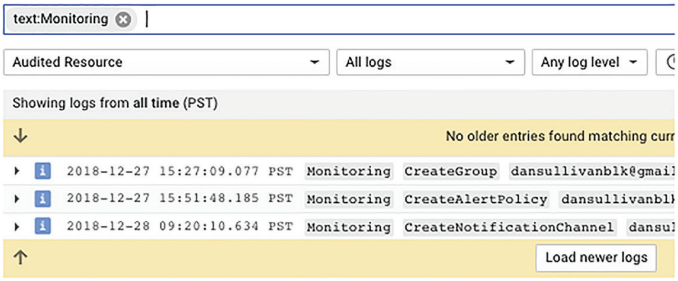
Notice that at the top of the form there are several options for filtering log messages, including filtering by the following:

- Label or text search
- Resource type
- Log type
- Log level
- Time limit

There is also an option to jump to the latest entries by selecting Jump To Now.

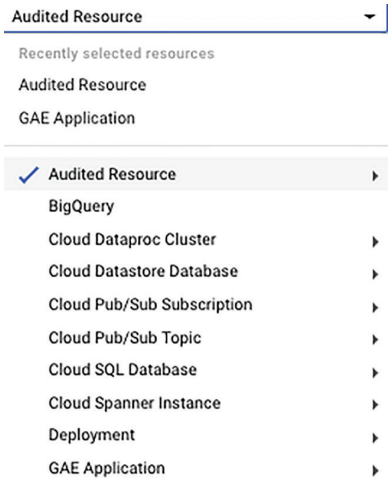
You can use the label or text search to filter on text strings or labels in log messages. For example, Figure 18.25 shows a set of log entries filtered by the text Monitoring. Stackdriver will add types such as text: as needed.

FIGURE 18.25 Log entries that contain the text string Monitoring



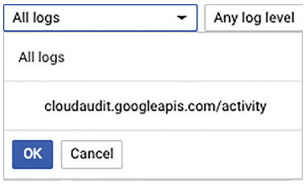
The resource type drop-down box (see Figure 18.26) provides a list of GCP resource types, including any audited resources, VM instances, subnetworks, projects, and databases.

FIGURE 18.26 Partial list of resource types for filtering logs



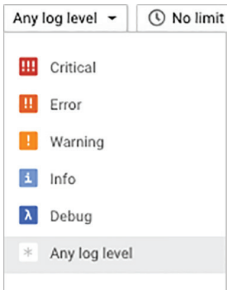
The drop-down box labeled All Logs lets you filter by log type (see Figure 18.27).

FIGURE 18.27 Example listing of logs generating entries in Stackdriver Logging



The next option, Any Log Level, shows log message levels, such as Error, Info, Warning, and Debug (see Figure 18.28).

FIGURE 18.28 A list of log levels that can be used to filter log entries displayed

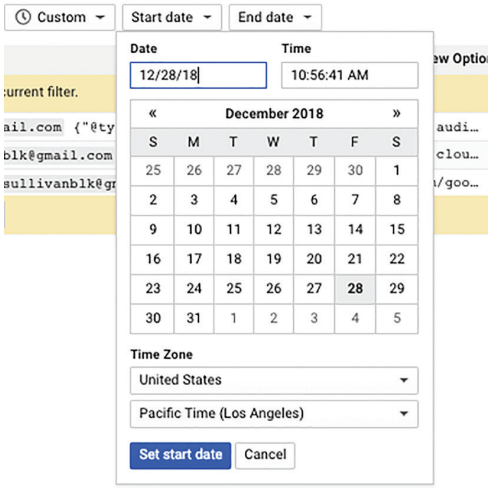


The time selection filter shows No Limit by default. The drop-down (see Figure 18.29) includes several time spans and allows for a custom time limit.

FIGURE 18.29 Predefined time span options for filtering log entries

If you choose Custom, you can select start and end dates, as shown in Figure 18.30.

FIGURE 18.30 Form for specifying a custom time range for filtering log entries



Viewing Message Details

Each log entry is displayed as a single line when you view the contents of logs. Notice the triangle icon at the left end of the line. If you click that icon, the line will expand to show additional detail. For example, Figure 18.31 shows a log entry expanded one level.

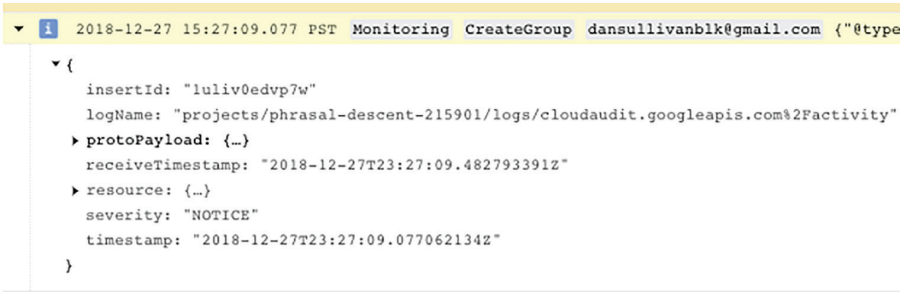
FIGURE 18.31 A log entry expanded one level

In the case of the first-level expansion, you see high-level information such as insertId, logName, and receiveTimestamp. You also see other structured data elements, such as protoPayload and resources. Figure 18.32 shows the protoPayload structure expanded.

FIGURE 18.32 A log entry with the protoPayload structure expanded

You can continue to drill down individually into each structure if there is a triangle at the left. For example, in the protoPayload structure, you could drill down into authenticationInfo, authorizationInfo, and requestMetadata, among others. Alternatively, you could click the Expand All link in the upper-right corner of the log entry listing to expand all structures (see Figure 18.33).

FIGURE 18.33 A partial listing of a fully expanded log entry



Stackdriver Logging is used to collect log data and events and store them for up to 30 days. Logs can be exported to Cloud Storage, BigQuery, and Cloud Pub/Sub. Cloud Console provides a logging interface that provides multiple ways to filter and search log entries.

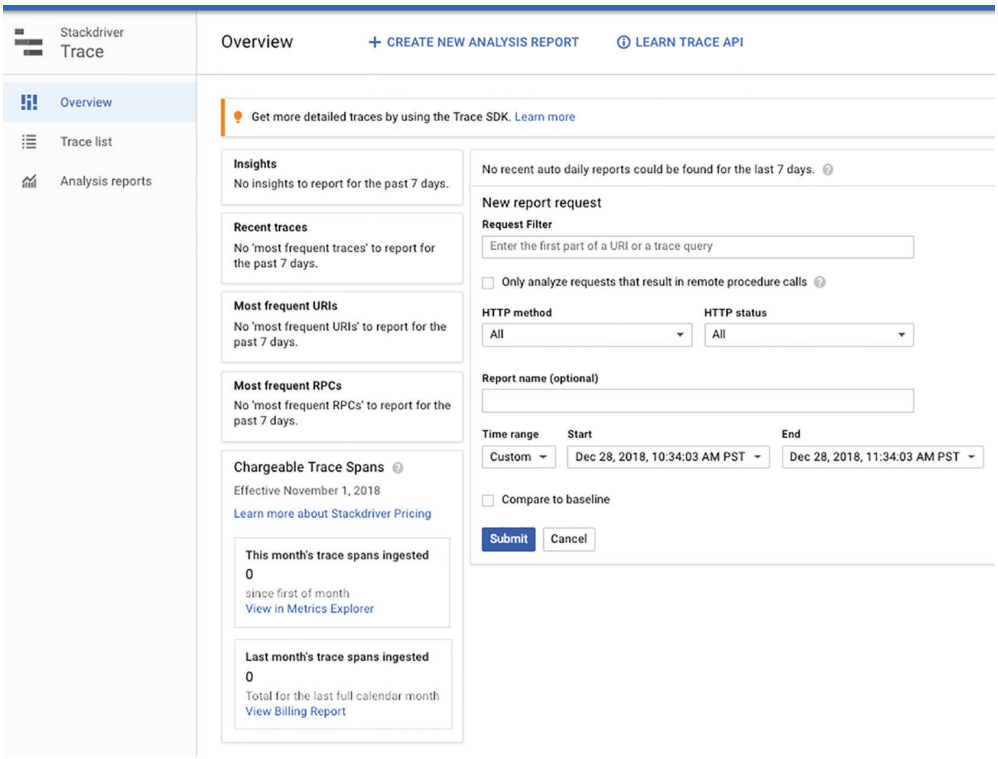
Using Cloud Diagnostics

Google Cloud Platform provides diagnostic tools that software developers can use to collect information about the performance and functioning of their applications. Specifically, developers can use Cloud Trace and Cloud Debug to collect data as their applications execute.

Overview of Cloud Trace

Cloud Trace is a distributed tracing system for collecting latency data from an application. This helps developers understand where applications are spending their time and to identify cases where performance is degrading. Figure 18.34 shows the overview page of the Cloud Trace service.

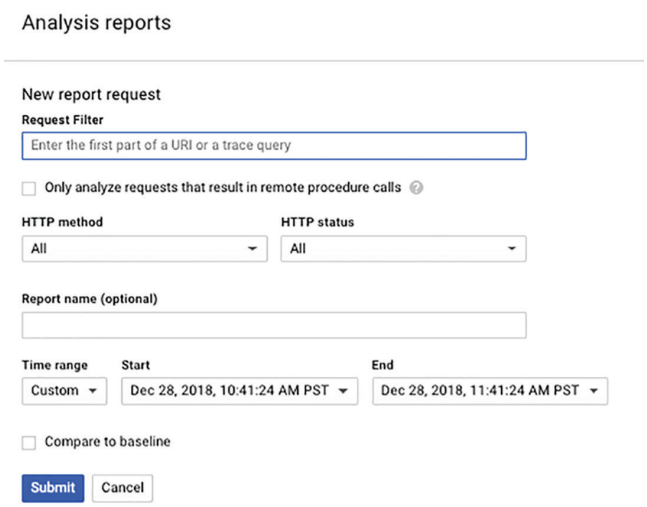
FIGURE 18.34 Overview of Cloud Trace



From the Cloud Trace console, you can list traces generated by applications running in a project. Traces are generated when developers specifically call Cloud Trace from their

applications. In addition to seeing lists of traces, you can create reports that filter trace data according to report criteria (see Figure 18.35).

FIGURE 18.35 Creating a report using Cloud Trace data



In addition to filtering on time and trace query, you can filter on HTTP method (see Figure 18.36) and return status (see Figure 18.37).

FIGURE 18.36 Filtering trace data by HTTP method

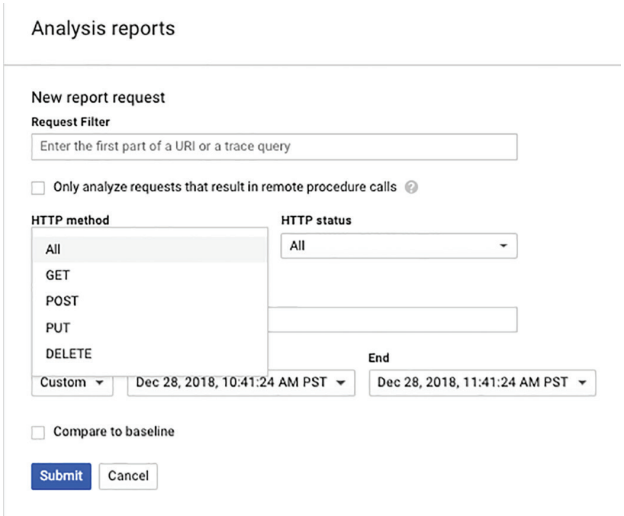
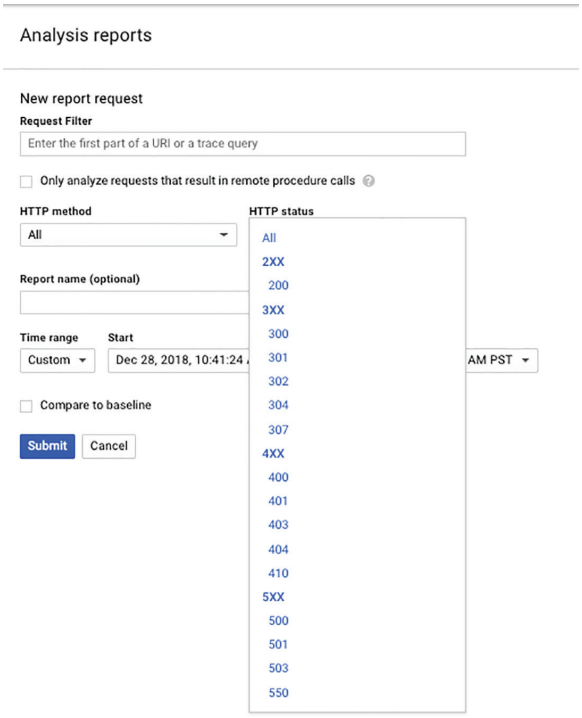


FIGURE 18.37 Filtering trace data by response code



For the purpose of the Associate Cloud Engineering exam, remember that Cloud Trace is a distributed tracing application that helps developers and DevOps engineers identify sections of code that are performance bottlenecks.

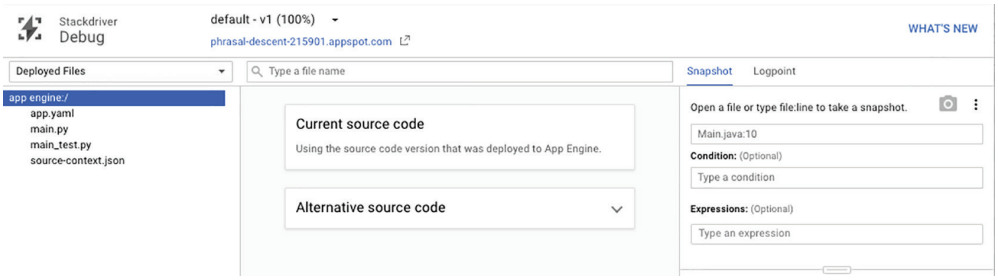
Overview of Cloud Debug

Cloud Debug is an application debugger for inspecting the state of a running program. Like Cloud Trace, this is a tool typically used by software developers, but it is helpful for cloud engineers to be familiar with Cloud Debug's capabilities.

Cloud Debug allows developers to insert log statements or take snapshots of the state of an application. The service is enabled by default on App Engine and can be enabled for Compute Engine and Kubernetes Engine.

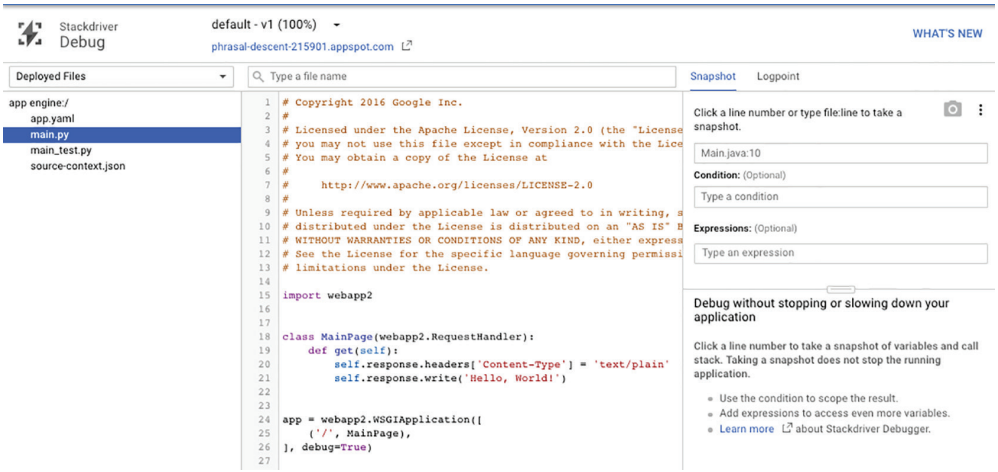
To view Cloud Debug, navigate to Cloud Debug in Cloud Console to display a page like the one shown in Figure 18.38.

FIGURE 18.38 Overview page of Cloud Debug



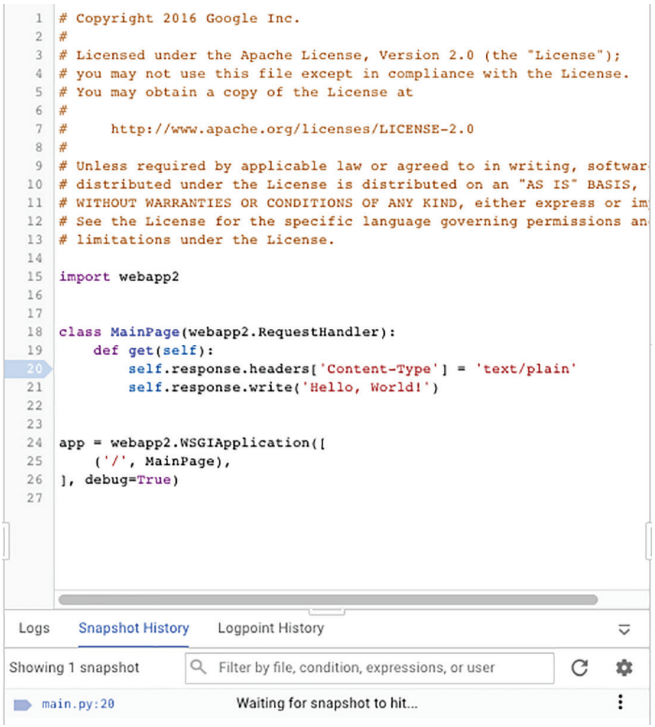
Selecting a program file displays the contents of the file. For example, Figure 18.39 shows the contents of a file called `main.py`.

FIGURE 18.39 Code listing of sample Python program provided by Google



In this interface, you can click a line of code to have a snapshot taken when that line executes. In Figure 18.40 the light blue arrow on line 20 indicates where Cloud Debug will take the snapshot.

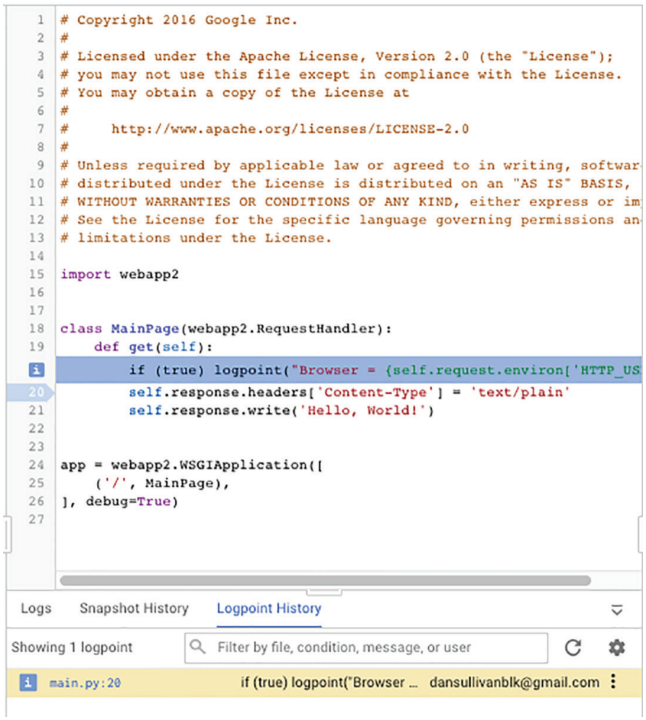
FIGURE 18.40 Setting a snapshot to be taken when line 20 executes



You can also inject a logpoint, which is a log statement that is written to the log when the statement executes. In Figure 18.41 a line of code has been added to create a logpoint and print a message.

For the purpose of the Associate Cloud Engineer exam, remember that Cloud Debug is used to take snapshots of the status of a program while it executes, and logpoints allow developers to inject log messages on the fly without altering source code.

FIGURE 18.41 Code with a logpoint injected

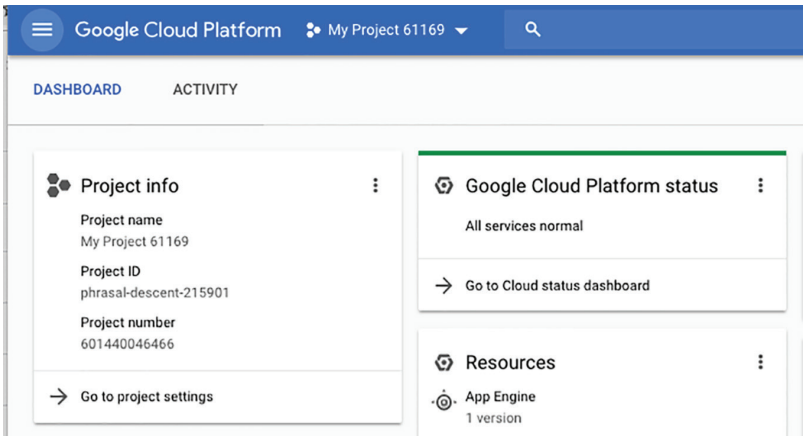


Viewing Google Cloud Platform Status

In addition to understanding the state of your applications and services, cloud engineers need to be aware of the status of GCP services. You can find this status in the Google Cloud Status Dashboard.

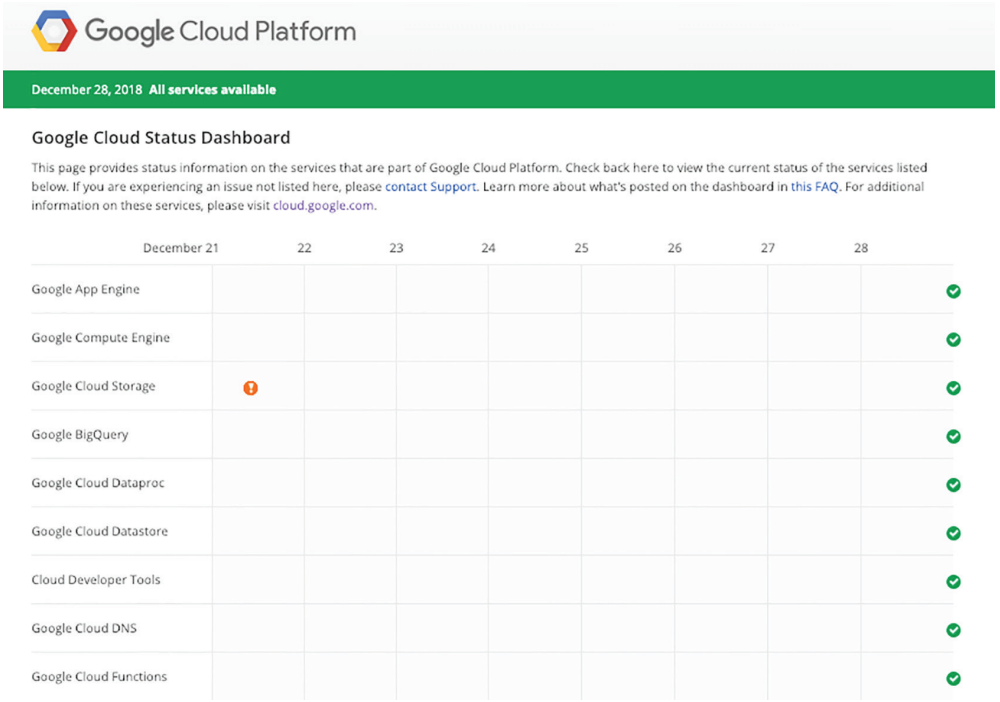
To view the status of Google Cloud services, navigate to the home page in Cloud Console and find the Google Cloud Platform Status card on the home page (see Figure 18.42). You can also find the dashboard at `https://status.cloud.google.com/`.

FIGURE 18.42 The Cloud Console home page has a card linking to the Cloud Status Dashboard.



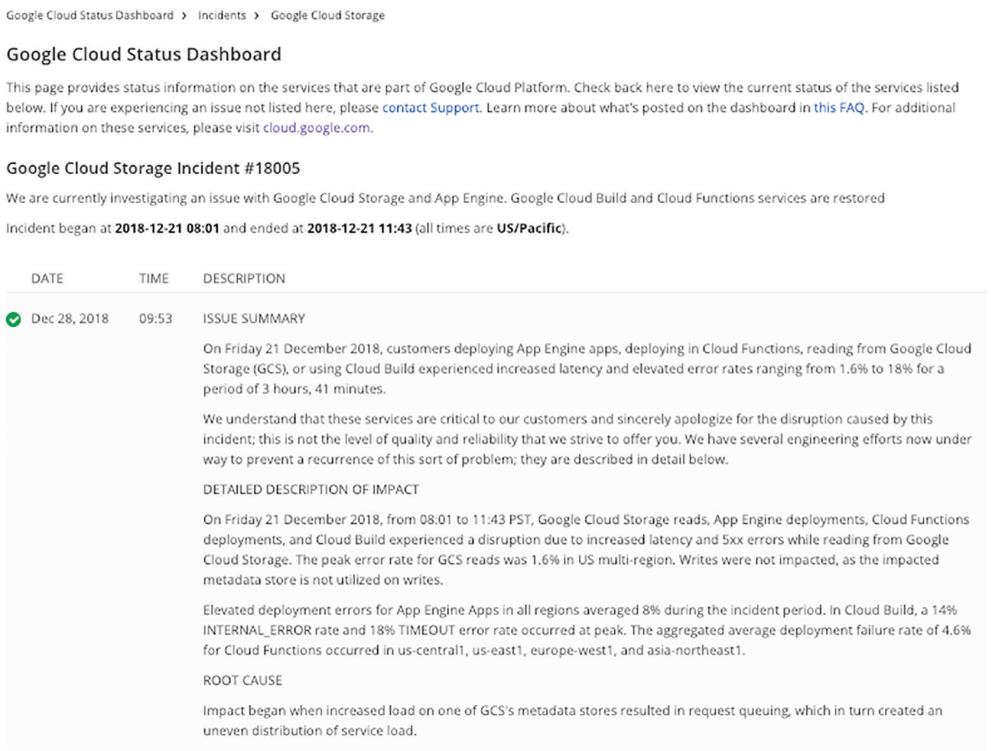
Click the Go To Cloud Status Dashboard link to display the dashboard. An example is shown in Figure 18.43.

FIGURE 18.43 Partial listing of the Google Cloud Status Dashboard



The dashboard lists GCP services on the left side. The columns represent days in the recent past. The content of each cell indicates the status. If there is a green check mark, the service is up and running. If there is an orange icon, for example as in the Cloud Storage row and December 21 column, then there was a disruption in the service. Click the orange icon to display additional details, as shown in Figure 18.44.

FIGURE 18.44 Example description of service interruption



Using the Pricing Calculator

Google provides a Pricing Calculator to help GCP users understand the costs associated with the services and configuration of resources they choose to use. The Pricing Calculator is an online tool at `https://cloud.google.com/products/calculator/`.

With the Pricing Calculator you can specify the configuration of resources, the time they will be used, and, in the case of storage, the amount of data that will be stored. Other parameters can be specified too. Those will vary according to the service you are calculating charges for.

For example, Figure 18.45 shows some of the services available to use with the Pricing Calculator. Currently, there are almost 40 services available in the Pricing Calculator.

FIGURE 18.45 Pricing Calculator banner with a partial display of services available

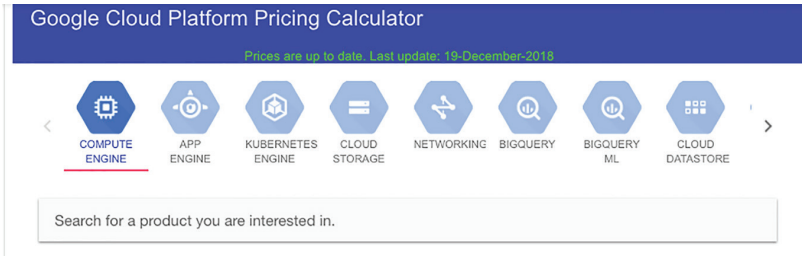
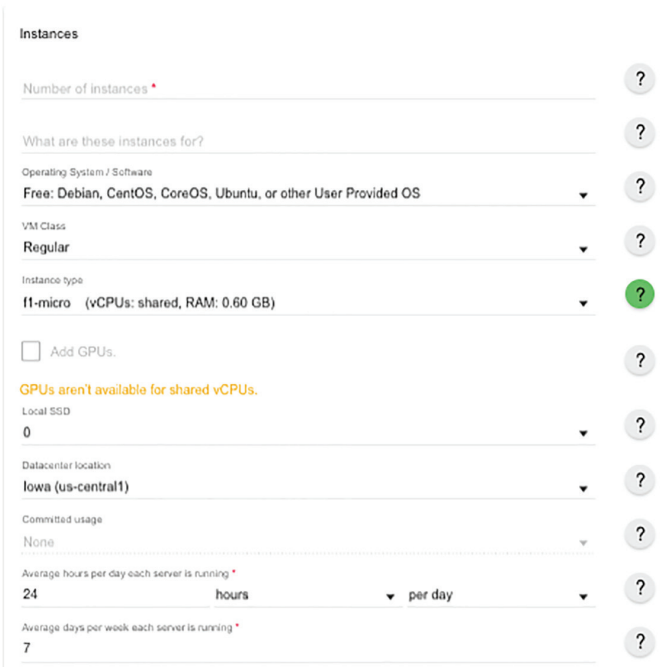


Figure 18.46 shows part of the form for estimating the cost of VMs. In this form you can specify the following:

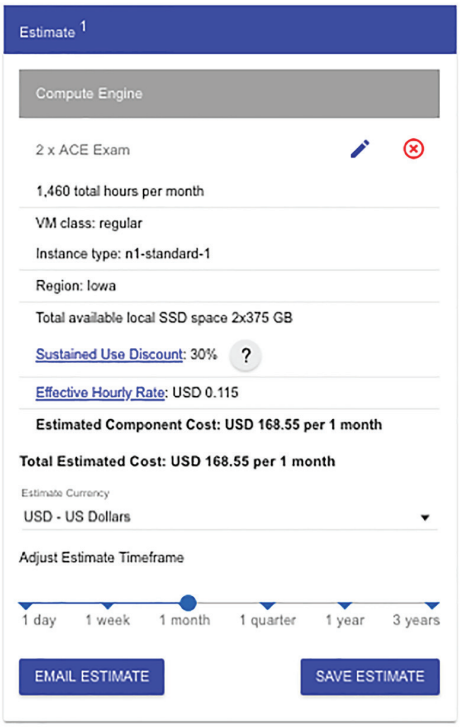
- Number of instances
- Machine types
- Operating system
- Average usage per day and week
- Persistent disks
- Load balancing
- Cloud TPUs (for machine learning applications)

FIGURE 18.46 Partial listing of pricing form for VMs



After you enter data into the form, the Pricing Calculator will generate an estimate, such as that shown in Figure 18.47.

FIGURE 18.47 Example price estimate for 2 n1-standard-1 VMs



Different resources will require different parameters for an estimate. For example, Figure 18.48 shows estimating the price of a Kubernetes cluster, which requires details about VMs, persistent disks, and load balancers.

Figure 18.49 shows a further example, this time for BigQuery. For that service, you need to specify the location of your data, the amount of data stored, the amount streamed in, and the volume of data scanned when executing queries. The table parameter is where you indicate which BiqQuery table you are querying. Storage Pricing and Query Pricing both accept numeric values for the amount of data stored in the table (GBs) and the volume scanned during queries (TBs). Also, there is an option for flat rate pricing if you spend more than $40,000 per month.

FIGURE 18.48 Form for estimating the price of a Kubernetes cluster

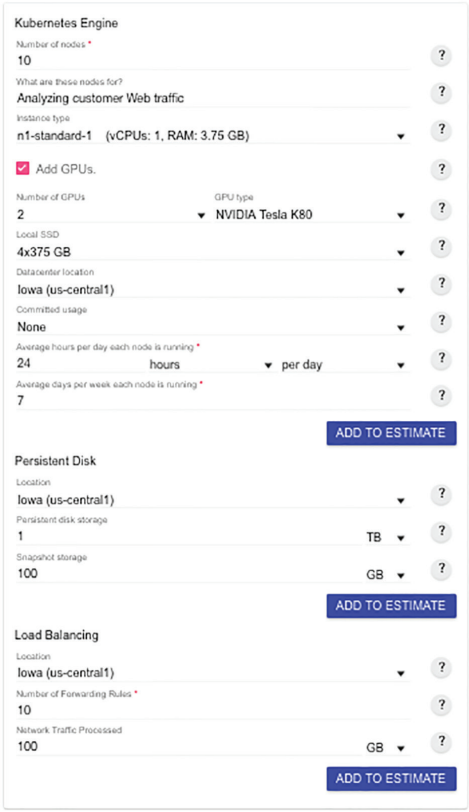
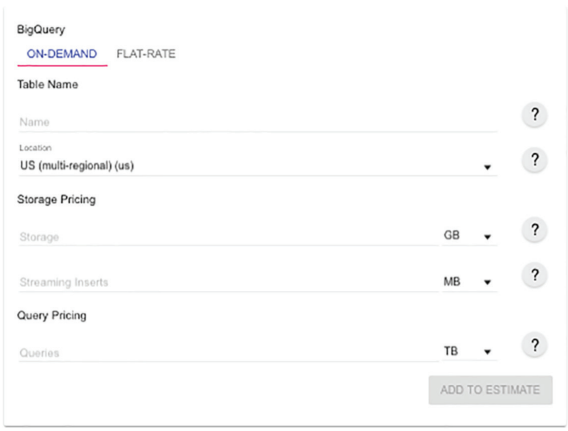


FIGURE 18.49 The parameters required to estimate the cost of storing and querying BigQuery data



The Pricing Calculator allows you to estimate the price of multiple services and then generate a total estimate for all services.

Summary

Cloud engineers are responsible for monitoring the health and performance of applications and cloud services. GCP provides multiple tools, including monitoring, logging, debugging, and tracing services.

Stackdriver Monitoring allows you to define alerts on metrics, such as CPU utilization, so that you can be notified if part of your infrastructure is not performing as expected. Stackdriver Logging collects, stores, and manages log entries. Log data that needs to be stored more than 30 days can be exported to Cloud Storage, BigQuery, or Cloud Pub/Sub. Cloud Trace provides distributed tracing services to identify slow-running parts of code. Cloud Debug provides for creating snapshots of running code and injecting log messages without altering source code.

You can always get the status of GCP services at the Google Cloud Status Dashboard at `https://status.cloud.google.com/`.

The Pricing Calculator is designed to help you estimate the cost of almost 40 services in the GCP.

Exam Essentials

Understand the need for monitoring and the role of metrics. Metrics provide data on the state of applications and infrastructure. We create conditions, like CPU exceeding 80 percent for 5 minutes, to trigger alerts. Alerts are delivered by notification channels. GCP has a substantial number of predefined metrics, but you can create custom metrics as well.

Stackdriver Logging collects, stores, filters, and displays log data. Logs can come from virtually any source. Logging keeps log data for 30 days. If you need to keep log data longer than that, then you need to export the data to a log sink. Log sinks may be a Cloud Storage bucket, a BigQuery data set, or a Cloud Pub/Sub topic.

Know how to filter logs. Logs can contain a large amount of data. Use filters to search for text or labels, limit log entries by log type and severity, and restrict the time range to a period of interest.

Log entries are hierarchical. Stackdriver Logging shows a single line summary for a log entry by default, but you can drill down into the details of a log entry. Use the Expand All and Collapse All options to quickly view or hide the full details of a log entry.

Cloud Trace is a distributed tracing service. Software developers include Cloud Tracer code in their applications to record trace data. Trace data can be viewed as individual traces, or you can create reports that include parameters specifying a subset of traces to include.

Cloud Debug is used to analyze running code by taking snapshots or injecting logpoints. Snapshots show the stack, or execution context, at a point in the execution of a program. Logpoints are log statements injected into running code but do not require changes to source code.

GCP publishes the status of services in the Google Cloud Platform Status page. It includes a list of all services, their current status, and the status over the near past. If there is an incident in a service, you will find additional details on the impact and root cause of the problem.

The Pricing Calculator is used to estimate the cost of resources and services in the GCP. It is available at `https://cloud.google.com/products/calculator/`.

There is a separate calculator for each service. Each service has its own set of parameters for estimating costs. The Pricing Calculator allows you to estimate the cost of multiple services and generate a total estimate for all those services.

Review Questions

1. What Stackdriver service is used to generate alerts when the CPU utilization of a VM exceeds 80 percent?
   - **A.** Logging
   - **B.** Monitoring
   - **C.** Cloud Trace
   - **D.** Cloud Debug
2. You have just created a virtual machine, and you'd like Stackdriver Monitoring to alert you via email whenever the CPU average utilization exceeds 75 percent for 5 minutes. What do you need to do to the VM to have this happen?
   - **A.** Install a Stackdriver workspace
   - **B.** Install the Stackdriver monitoring agent on the VM
   - **C.** Edit the VM configuration in Cloud Console and check the Monitor With Stackdriver checkbox
   - **D.** Set a notification channel
3. Stackdriver can be used to monitor resources where?
   - **A.** In Google Cloud Platform only
   - **B.** In Google Cloud Platform and Amazon Web Services only
   - **C.** In Google Cloud Platform and on premises data centers
   - **D.** In Google Cloud Platform, Amazon Web Services, and on premises data centers
4. Grouping a set of metrics that arrive in a period of time into regular-sized buckets is called what?
   - **A.** Aggregation
   - **B.** Alignment
   - **C.** Minimization
   - **D.** Consolidation
5. You have created a condition of CPU utilization, and you want to receive notifications. Which of the following are options?
   - **A.** Email only
   - **B.** PagerDuty only
   - **C.** Hipchat and PagerDuty
   - **D.** Email, PagerDuty, and Hipchat

6. When you create a policy to notify you of a potential problem with your infrastructure, you can specify optional documentation. Why would you bother putting documentation in that form?
   - **A.** It is saved to Cloud Storage for future use.
   - **B.** It can help you or a colleague understand the purpose of the policy.
   - **C.** It can contain information that would help someone diagnose and correct the problem.
   - **D.** Options B and C.

7. What is alert fatigue, and why is it a problem?
   - **A.** Too many alert notifications are sent for events that do not require human intervention, and eventually DevOps engineers begin to pay less attention to notifications.
   - **B.** Too many alerts put unnecessary load on your systems.
   - **C.** Too few alerts leave DevOps engineers uncertain of the state of your applications and infrastructure.
   - **D.** Too many false alerts

8. How long is log data stored in Stackdriver Logging?
   - **A.** 7 days
   - **B.** 15 days
   - **C.** 30 days
   - **D.** 60 days

9. You need to store log entries for a longer period of time than Stackdriver Logging retains them. What is the best option for preserving log data?
   - **A.** There is no option; once the data retention period passes, Stackdriver Logging deletes the data.
   - **B.** Create a log sink and export the log data using Stackdriver Logging's export functionality.
   - **C.** Write a Python script to use the Stackdriver API to write the data to Cloud Storage.
   - **D.** Write a Python script to use the Stackdriver API to write the data to BigQuery.

10. Which of the following are options for logging sinks?
    - **A.** Cloud Storage bucket only
    - **B.** BigQuery dataset and Cloud Storage bucket only
    - **C.** Cloud Pub/Sub topic only
    - **D.** Cloud Storage bucket, BigQuery dataset, and Cloud Pub/Sub topic

11. Which of the following can be used to filter log entries when viewing logs in Stackdriver Logging?
    - **A.** Label or text search only
    - **B.** Resource type and log type only

C. Time and resource type only

D. Label or text search, resource type, log type, and time

12. Which of the following is not a standard log level that can be used to filter log viewings?

A. Critical

B. Halted

C. Warning

D. Info

13. You are viewing log entries and spot one that looks suspicious. You are not familiar with the kind of log entry, and you want to see the complete details of the log entry as quickly as possible. What would you do?

A. Drill down one by one into each structure in the log entry.

B. Click Expand all to show all details.

C. Write a Python script to reformat the log entry.

D. Click the Show Detail link next to the log entry.

14. What Stackdriver service is best for identifying where bottlenecks exist in your application?

A. Monitoring

B. Logging

C. Trace

D. Debug

15. There is a bug in a microservice. You have reviewed application outputs but cannot identify the problem. You decide you need to step through the code. What Stackdriver service would you use to give you insight into the status of the services at particular points in execution?

A. Monitoring

B. Logging

C. Trace

D. Debug

16. You believe there may be a problem with BigQuery in the us-central zone. Where would you go to check the status of the BigQuery service for the quickest access to details?

A. Email Google Cloud Support.

B. Check `https://status.cloud.google.com/`.

C. Check `https://bigquery.status.cloud.google.com/`.

D. Call Google tech support.

17. You would like to estimate the cost of GCP resources you will be using. Which services would require you to have information on the virtual machines you will be using?
    - **A.** Compute Engine and BigQuery
    - **B.** Compute Engine and Kubernetes Engine
    - **C.** BigQuery and Kubernetes Engine
    - **D.** BigQuery and Cloud Pub/Sub

18. You are generating an estimate of the cost of using BigQuery. One of the parameters is Query Pricing. You have to specify a value in TB units. What is the value you are specifying?
    - **A.** The amount of data stored in BigQuery
    - **B.** The amount of data returned by the query
    - **C.** The amount of data scanned by the query
    - **D.** The amount of data in Cloud Storage bucket

19. Why do you need to specify the operating system to be used when estimating the cost of a VM?
    - **A.** All operating systems are charged a fixed rate.
    - **B.** Some operating systems incur a cost.
    - **C.** It's not necessary; it is only included for documentation.
    - **D.** To estimate the cost of Bring Your Own License configurations.

20. You want to create a custom metric for use in Stackdriver Monitoring but do not want to use the low-level Stackdriver API. What is an alternative open source option for working with custom metrics?
    - **A.** Prometheus
    - **B.** OpenCensus
    - **C.** Grafana
    - **D.** Nagios

Appendix

Answers to Review Questions

Official Google Cloud Certified: Associate Cloud Engineer: Study Guide
By Dan Sullivan

Chapter 1: Overview of Google Cloud Platform

1. B. The basic unit for purchasing computing resources is the virtual machine (VM). A physical server underlies VMs, but the resources of a physical server are allocated to VMs. Blocks and subnets are not relevant to the fundamental unit of computing.

2. D. When using managed clusters, the cloud provider will monitor the health of nodes in the cluster, set up networking between nodes in the cluster, and configure firewall and other security controls.

3. B. App Engine is a serverless platform for running applications, while Cloud Functions is a service for executing short-running functions in response to events. Kubernetes Engine is a managed cluster service, and both Kubernetes Engine and Compute Engine require you to configure servers.

4. B. Object storage, like Cloud Storage, provides redundantly stored objects without limits on the amount of data you can store, which makes option B correct. Since file system functionality is not required, option D is not a good option. Block storage could be used, but you would have to manage your own replication to ensure high availability. Caches are transient, in-memory storage and are not high-availability, persistent storage systems.

5. D. Block sizes in a block storage system can vary; therefore, option D is the correct answer. Block size is established when a file system is created. 4KB block sizes are commonly used in Linux.

6. C. Firewalls in Google Cloud Platform (GCP) are software-defined network controls that limit the flow of traffic into and out of a network or subnetwork, so option C is the correct answer. Routers are used to move traffic to appropriate destinations on the network. Identity access management is used for authenticating and authorizing users; it is not relevant to network controls between subnetworks. IP address tables are not a security control.

7. C. Option C is correct because specialized services in GCP are serverless. Google manages the compute resources used by the services. There is no need for a user to allocate or monitor VMs.

8. B. Option B is correct; investing in servers works well when an organization can accurately predict the number of servers and other equipment it will need for an extended period and can utilize that equipment consistently. Startups are not established businesses with histories that can guide expected needs in three to five years. It does not matter if a budget is fixed or variable; investing in servers should be based on demand for server capacity.

9. B. The characteristics of the server, such as the number of virtual servers, the amount of memory, and the region where you run the VM, influence the cost, so option B is correct. Time of day is not a factor, nor is the type of application you run on the VM.

10. D. Cloud Vision is one of GCP's specialized services. Users of the service do not need to configure any VMs to use the service.