

# Create Virtual Machine(VM) in Google Cloud Platform(GCP)

## Step 1: Navigating to the Google Cloud Console

1. Open your web browser and go to

<https://console.cloud.google.com/>

2. Sign in to your GCP account.

## Step 2: Creating a New Project (Optional)

1. Click on “New Project” and enter a name for your project.
2. Click “Create” to create the new project.

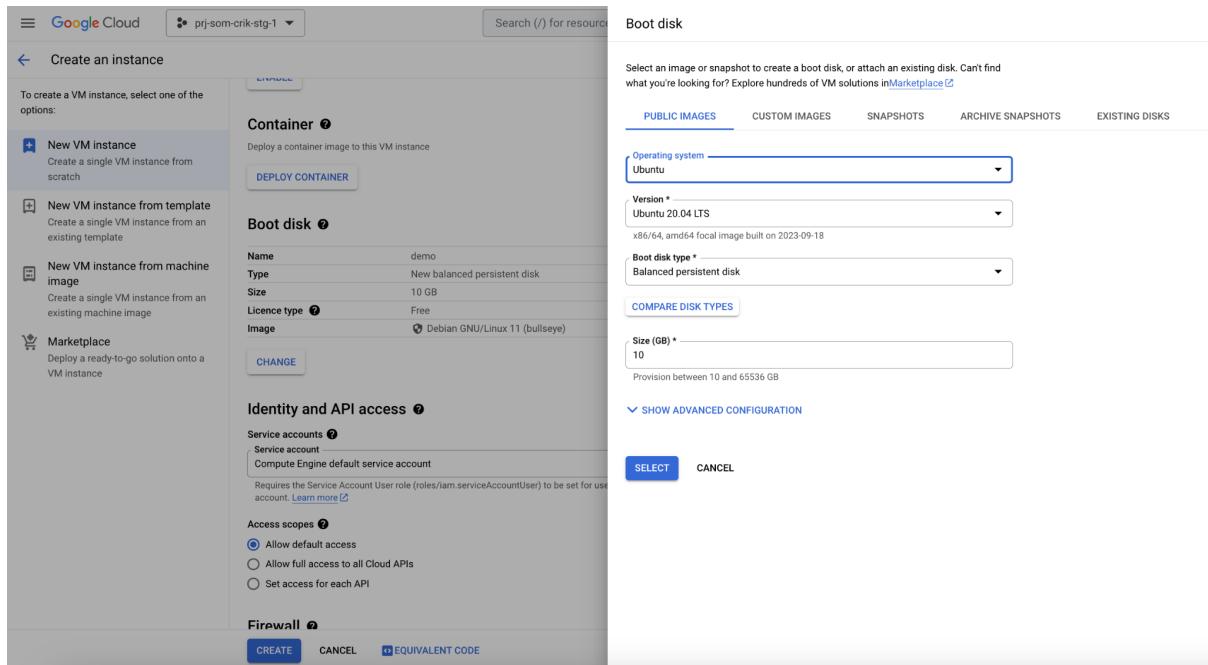
The screenshot shows the 'New Project' creation interface in the Google Cloud Console. At the top, there's a navigation bar with the Google Cloud logo and a search bar. Below it, the 'New Project' section is titled 'New Project'. A warning message states: '⚠ You have 25 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)' with a 'MANAGE QUOTAS' button. The 'Project name \*' field contains 'Crikfun'. The 'Billing account \*' dropdown is set to 'My Billing Account'. The 'Organisation \*' dropdown is set to 'real11.com'. A note at the bottom says: 'Select an organisation to attach it to a project. This selection can't be changed later.'

## Step 3: Creating a Virtual Machine Instance

1. In the Compute Engine dashboard, click on the “**Create**” button to create a new VM instance.
2. Enter a “**name**” for your VM instance.
3. Choose the “**region**” and “**zone**” where you want to host your VM.
4. Choose the “**machine type**”. You can select from various options based on CPU and memory resources.

The screenshot shows the 'Create an instance' wizard. At the top, it says 'To create a VM instance, select one of the options:' with four choices: 'New VM instance', 'New VM instance from template', 'New VM instance from machine image', and 'Marketplace'. The 'New VM instance' option is selected. Below this, there's a 'Name' field with 'demo' entered, a 'Region' dropdown set to 'asia-south1 (Mumbai)', and a 'Zone' dropdown set to 'asia-south1-c'. To the right, a 'Monthly estimate' table shows costs for a C3D machine type: US\$69.32 monthly, US\$0.09 hourly, and no upfront costs. Under 'Machine configuration', a 'TRY NOW' button is shown above a table of machine types. The 'C3D' row is highlighted with a 'PREVIEW' badge. The table columns are Series, Description, vCPUs, Memory, and Platform. Other rows include E2, N2, N2D, T2A, T2D, and N1. At the bottom, a 'Machine type' section says 'Choose a machine type with preset amounts of vCPUs and memory that suit most workloads.'

5. Under “**Boot disk**”, select the operating system and disk size for your VM.
6. Click on the “**Create**” button to create the VM instance.



## Source

- <https://cloud.google.com/compute/docs/instances/create-start-instance>
- <https://medium.com/@abogam78/how-to-create-virtual-machine-vm-in-google-cloud-platform-gcp-d6b631248523>

## Create firewall rule in Google Cloud Platform(GCP)

### Step 1. Navigate to the Firewall Rules Page

- Select "VPC network" under the "Networking" section. Then, select on "Firewall rules".

## **Step 2.Create a New Firewall Rule**

- Select the "**Create Firewall Rule**" button creating a new firewall rule.

## **Step 3. Configure the Firewall Rule**

- **Name:** Give your **firewall rule** a name.
- **Network:** Select the .
- **Priority:** Set a priority for the rule.
- **Direction of traffic:** Specify if the rule is for **Ingress (incoming)** or **Egress (outgoing) traffic**.
- **Action:** Choose whether to **allow** or **deny** traffic.
- **Targets:** Define the **instances or tags** that this rule applies to.
- **Source Filter:** Specify the source IP ranges or IP blocks for the traffic.
- **Protocols and Ports:** Define the **protocols (TCP, UDP, etc.)** and port ranges.

## **Step 4. Click "Create" to Create the Rule**

### **Source**

- <https://cloud.google.com/filestore/docs/configuring-firewall>

## **Take Snapshot & Create Image In Google Cloud Platform(GCP)**

**Step 1:** Navigate to the Disks Page and Select the Disk for Snapshot.

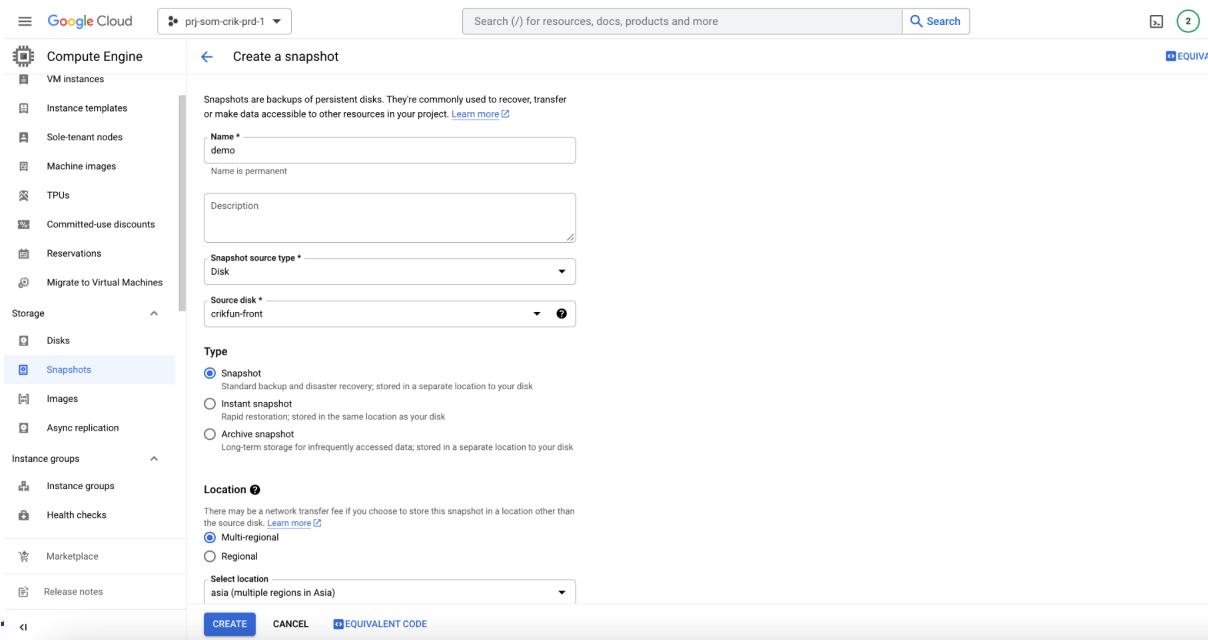
## Step 2: Create Snapshot

- Click on "**Create Snapshot**" button.

## Step 3: Set up snapshot preferences or configurations.

- **Snapshot name:** Write a **name** for your snapshot. It should be a **unique name**.
- **Storage Location:** You can choose the storage location for the snapshot.
- **Snapshot Schedule:** If you need to create snapshots regularly, you can set up a snapshot schedule.

## Step 4: Click "**Create**" to **Create the Snapshot**



## Create Instance Template Using GCP Image

### Step 1: Navigate to the Snapshots Page and Select the Snapshot.

### Step 2: Create an Image from the Snapshot

- On the snapshot details page, click the "**Create image**" button.

- **Step 3: Configure the Image**

- **Image name:** Provide a unique name for the image.
- **Image family (optional):** You can specify an image family.
- **Minimum CPU platform (optional):** Set the minimum CPU platform for the image (optional).
- **Labels (optional):** You can add labels for additional organisation.

## Step 4: Click "Create" to Create the Image.

The screenshot shows the 'Create an image' dialog in the Google Cloud Compute Engine interface. The left sidebar is collapsed, showing options like Compute Engine, VM instances, Instance templates, Sole-tenant nodes, Machine images, TPUs, Committed-use discounts, Reservations, Migrate to Virtual Machines, Storage (Disks, Snapshots), Images (selected), Async replication, Instance groups (selected), Health checks, Marketplace, Release notes, and Help.

The main form has the following fields:

- Name:** demo-image
- Source:** Snapshot (selected)
- Source snapshot:** crikfunfront-v1
- Location:** Multi-regional (selected)
- Family:** cloud-front
- Description:** (empty)
- Labels:** + ADD LABEL (button)
- Encryption:** Google-managed encryption key (selected)

At the bottom are buttons for **CREATE**, **CANCEL**, and **EQUIVALENT CODE**.

## Create Instance Template Using GCP Image

**Step 1:** Navigate to the Compute Engine and select Create an Instance Template.

## Step 2: Configure the Instance Template

You can configure various settings....

- **Name:** Write a name for your instance template.
- **Machine Type:** Choose the **machine type** for your instances.
- **Boot Disk:** Select the **boot disk** and specify the image .
- **Firewall:** Configure the **firewall rules** as needed.
- **Networking:** Set up your **network configurations**.
- **Metadata:** Add any metadata or startup scripts if you need.
- **Autoscaling:** Configure auto scaling settings if needed.

The screenshot shows the 'Create an instance template' interface in the Google Cloud Platform. The left sidebar navigation includes 'Compute Engine' (selected), 'VM instances', 'Instance templates' (selected), 'Sole-tenant nodes', 'Machine images', 'TPUs', 'Committed-use discounts', 'Reservations', and 'Migrate to Virtual Machines'. Under 'Storage', there are links for 'Disks', 'Snapshots', 'Images', and 'Sync replication'. Under 'Instance groups', there are links for 'Instance groups' and 'Health checks'. At the bottom of the sidebar are links for 'Marketplace' and 'Release notes'. The main content area has a header 'Create an instance template' with a back arrow. Below it is a 'Name' input field containing 'instance-template-1'. A 'MANAGE TAGS AND LABELS' section is present. The 'Machine configuration' section features a 'TRY NOW' button and a 'General purpose' tab selected. A table lists machine types: C3, C3D, E2, N2, N2D, T2A, T2D, and N1. The E2 row is highlighted with a blue background and labeled 'PREVIEW'. The table columns are Series, Description, vCPUs, Memory, and Platform. To the right of the table, there's a monthly estimate of US\$25.46. The entire interface is set against a light gray background with standard Google UI elements like search bars and navigation icons.

## Step 3: Review and Create

- Check your settings, and click the "**Create**" button.

The screenshot shows the 'Create an instance template' dialog in the Google Cloud Compute Engine interface. On the left, there's a sidebar with navigation links like 'Compute Engine', 'VM instances', 'Instance templates', etc. The main form has the following fields:

- Name:** instance-template-1
- Type:** New balanced persistent disk
- Size:** 10 GB
- Licence type:** Free
- Image:** Debian GNU/Linux 11 (bullseye)
- Operating system:** Ubuntu (selected)
- Version:** Ubuntu 20.04 LTS
- Boot disk type:** Balanced persistent disk
- Size (GB):** 10

Below the main form, there are sections for 'Identity and API access', 'Firewall', and 'Advanced options'. At the bottom, there are 'CREATE', 'CANCEL', and 'EQUIVALENT COMMAND LINE' buttons.

## Step 4: Use the Instance Template

- Go to the "**Instance groups**" section and create an instance group.
- During the creation of the instance group, you can use the **instance template**.

## **// instance template**

```
gcloud compute instance-templates create crikfun-front-template-v5
--project=prj-som-crik-stg-1 --machine-type=n2d-highcpu-4
--network-interface=subnet=sn-som-crik-stg-1,no-address
--metadata=startup-script=\#\! /bin/bash$'\n'cd \
/home/ubuntu/LiveScoring$\n'redis-server\ --daemonize\ yes\
$'\n'pm2\ start\ server/server.js\ \ --name\ \ \"crcikfun_front\"$'\n'pm2\
save$\n'pm2\ \ startup\ \& --maintenance-policy=MIGRATE
--provisioning-model=STANDARD
--service-account=477063799383-compute@developer.gserviceaccount.c
om
--scopes=https://www.googleapis.com/auth/devstorage.read_only,https:
//www.googleapis.com/auth/logging.write,https://www.googleapis.com/
auth/monitoring.write,https://www.googleapis.com/auth/servicecontrol,
https://www.googleapis.com/auth/service.management.readonly,https:/
/www.googleapis.com/auth/trace.append --region=asia-south1
--create-disk=auto-delete=yes,boot=yes,device-name=crikfun-front-tem
plate,image-family=criktok-front,mode=rw,size=30,type=pd-balanced
--no-shielded-secure-boot --shielded-vtpm
--shielded-integrity-monitoring --reservation-affinity=any
```

## **// Instance Template v6**

```
gcloud compute instance-templates create crikfun-front-template-v7
--project=prj-som-crik-stg-1 --machine-type=n2d-highcpu-4
--network-interface=subnet=sn-som-crik-stg-1,no-address
--metadata=startup-script-url=https://storage.cloud.google.com/crikfun-
stage/startup-script.sh --maintenance-policy=MIGRATE
--provisioning-model=STANDARD
--service-account=477063799383-compute@developer.gserviceaccount.c
om
--scopes=https://www.googleapis.com/auth/devstorage.read_only,https:
//www.googleapis.com/auth/logging.write,https://www.googleapis.com/
auth/monitoring.write,https://www.googleapis.com/auth/servicecontrol,
https://www.googleapis.com/auth/service.management.readonly,https:/
/www.googleapis.com/auth/trace.append --region=asia-south1
--create-disk=auto-delete=yes,boot=yes,device-name=crikfun-front-tem
plate,image-family=criktok-front,mode=rw,size=30,type=pd-balanced
--no-shielded-secure-boot --shielded-vtpm
--shielded-integrity-monitoring --reservation-affinity=any
```

## Source

- <https://cloud.google.com/compute/docs/instance-templates/create-instance-templates>
- <https://www.easydeploy.io/blog/how-to-create-an-instance-using-instance-template-in-gce/>

# Creating a Managed Instance Group (MIG) On GCP

**Step 1:** Go to the Navigation Menu and select Create Instance Group

**Step 2:** Select a Group Type

- For load balancing, select "**Unmanaged instance group.**"
- For managed instance groups (which are typically used for auto-scaling and load balancing), then select "**Managed instance group.**"

**Step 3:** Configure the Instance Group

- **Name:** Write a name for your managed instance group.
- **Location:** Choose the region or zone.
- **Instance Template:** Select or create an instance template, which specifies the instance configuration (e.g., machine type, image, etc.) for the group.

[Create Instance Group](#)

**New managed instance group (stateless)**  
Automatically manage groups of VMs that do stateless serving and batch processing.

**New managed instance group (stateful)**  
Automatically manage groups of VMs that have persistent data or configurations (such as databases or legacy applications).

**New unmanaged instance group**  
Manually manage groups of load balancing VMs.

Name \* demo-mig  
Name is permanent

Description

Instance template \* criktfun-front-template-prod-v1  
n2d-highcpu-4, criktfun-front-prd.global

ⓘ You've selected a global instance template. To lower impact of outages outside your region, use a regional instance template instead. [Learn more](#)

Number of instances  
Based on auto-scaling configuration

**Location**  
For higher availability, select multiple zones in a region instead of a single zone. [Learn more](#)

Single zone  
 Multiple zones

Region \* asia-south1 (Mumbai) Zones asia-south1-c, asia-south1-b and asia-south1-a

Target distribution shape Even

Instance redistribution ?  
 Allow instance redistribution

## Step 4: Additional Configuration (Optional)

- You can also configure additional settings, like the initial number of instances, target utilisation, auto-healing, and rolling updates.

## Step 5: Health Check

- Configure a health check for your instances. This is used for the load balancer to determine the health of each instance.

## =>> How to create Health Check

The screenshot shows the 'Create a health check' interface in the Google Cloud Compute Engine. The left sidebar lists various services: Storage (Disks, Snapshots, Images, Async replication), Instance groups (Instance groups, Health checks), VM Manager (Patch, OS policies), and Bare Metal Solution (Servers, Networks, Volumes, NFS shares). The 'Health checks' option under 'Instance groups' is selected. The main form fields include:

- Name \***: demo (with validation note: Lowercase, no spaces)
- Description**: (empty field)
- Scope**: Global (selected) / Regional
- Protocol**: TCP (selected) / HTTP / HTTPS / SSL / SSLv3 / TLS / TLSv1 / TLSv1.1 / TLSv1.2 / TLSv1.3
- Port \***: 80
- Proxy protocol**: NONE (selected) / TCP / UDP
- Request**: (empty field)
- Response**: (empty field)
- Logs**: On (disabled) / Off (selected)
- Health criteria**:
  - Check interval \*: 5 seconds
  - Timeout \*: 5 seconds
  - Healthy threshold \*: 2 consecutive successes

At the bottom are 'CREATE' (highlighted in blue), 'CANCEL', and 'EQUIVALENT COMMAND LINE' buttons.

## Step 6: Autoscaling (Optional)

- If you want your instance group to automatically adjust the number of instances based on traffic or usage, configure auto scaling options.

## Step 7: Review and create

- Click the "**Create**" button to create the managed instance group

**Create Instance Group**

**New managed instance group (stateless)**  
Automatically manage groups of VMs that do stateless serving and batch processing.

**New managed instance group (stateful)**  
Automatically manage groups of VMs that have persistent data or configurations (such as databases or legacy applications).

**New unmanaged instance group**  
Manually manage groups of load balancing VMs.

**Autoscaling signals**  
Use signals to help determine when to scale the group. [Learn more](#)

CPU utilisation: 60% (default)  
Predictive auto-scaling is off  
[ADD A SIGNAL](#)

**Autoscaling schedules**

**Initialisation period**  
Specify how long it takes for your app to initialise from boot time until it is ready to serve. [Learn more](#)

Initialisation period \*  
60 seconds

**Scale-in controls**

**VM instance lifecycle**  
Configure what happens when VMs are created, repaired and deleted.

**Auto-healing**  
Autohealing recreates VM instances if your application cannot be reached by the health check. [Learn more](#)

Health check  
crikfun-admin-hc (TCP)  
port: 3000, timeout: 5s, check interval: 10s, unhealthy threshold: 3 attempts  
Initial delay \*  
300 Seconds

**CREATE**   **CANCEL**   **EQUIVALENT COMMAND LINE**

## Source

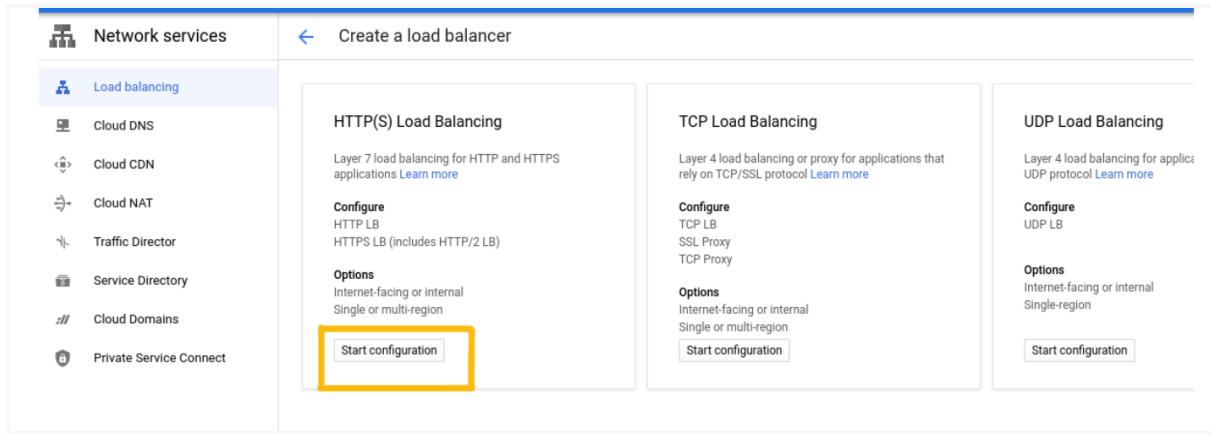
- <https://cloud.google.com/compute/docs/instance-groups>
- <https://cloud.google.com/compute/docs/quickstart-mig>

# Create a Load Balancer On GCP

**Step 1:** Go to the Navigation Menu and select Create a Load Balancer.

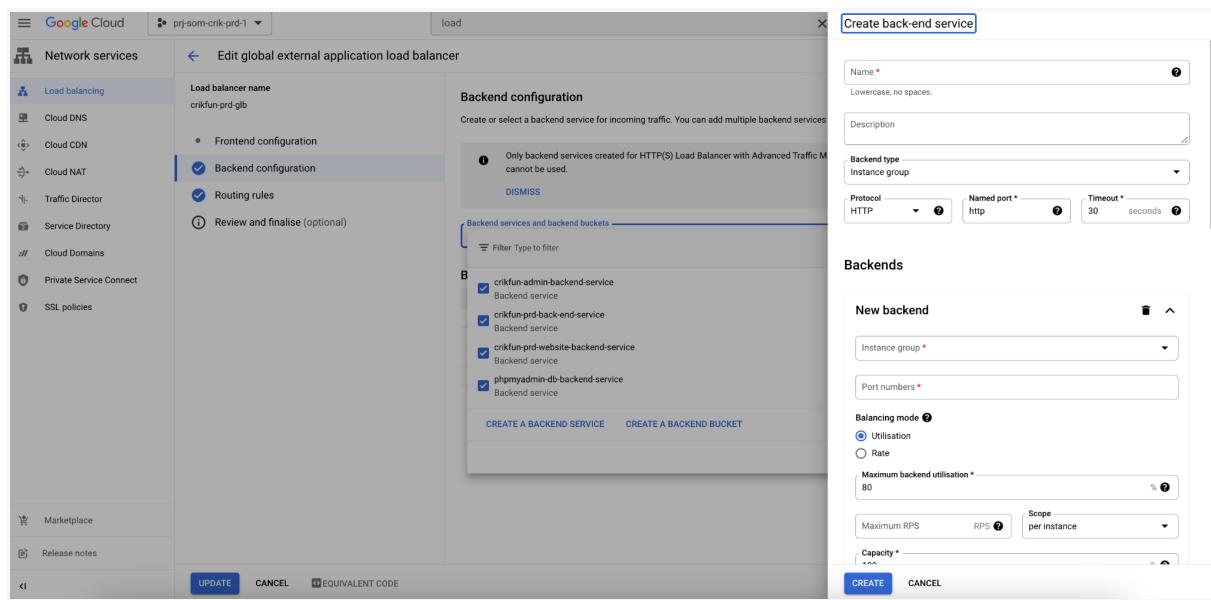
## Step 2: Choose a Load Balancer Type

- Select "**HTTP(S) Load Balancing**" for web applications.



## Step 3: Configure the Backend Service.

- Backend Service:** Create or select an existing backend service. This service defines how traffic is distributed to your instances.
- Health Check:** Configure a health check to determine the health of your instances.
- Backend Configuration:** Specify the instance group that your load balancer will route traffic to.



## Step 4: Configure the Frontend Configuration

- **IP and Port:** Define the IP address and port for the load balancer.
- **Certificate:** If you're setting up HTTPS, provide or select an SSL certificate.
- **Protocol:** Choose HTTP or HTTPS, depending on your application's requirements.

The screenshot shows the 'Edit global external application load balancer' interface. On the left sidebar, under 'Network services', 'Load balancing' is selected. The main panel displays the configuration for a load balancer named 'crikfun-http-redirect'. Under 'Frontend configuration', the 'Protocol' dropdown is set to 'HTTP'. Other options like 'HTTPS (includes HTTP/2)' are also visible. The 'Network Service Tier' section indicates 'Premium' support. At the bottom, there are 'UPDATE', 'CANCEL', and 'EQUIVALENT CODE' buttons, along with a 'DONE' button in the bottom right corner.

## Step 5: Routing rules

- Routing rules determine how your traffic will be directed. You can direct traffic to a backend service or a storage bucket. Using advanced mode, you can also submit a custom YAML routing config.

## Step 6: Review and Create

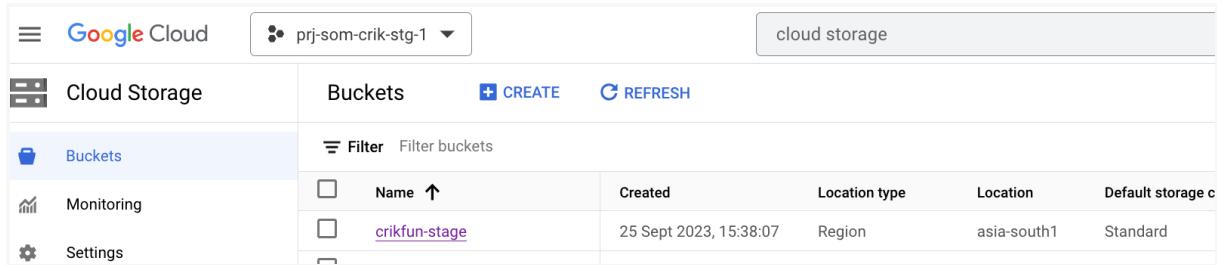
- Review your load balancer configuration, and then click the "Create" button.

## Source

- <https://cloud.google.com/iap/docs/load-balancer-howto>
- <https://www.easydeploy.io/blog/how-to-create-a-load-balancer-in-google-cloud-platform/>

# Create Cloud Storage (Bucket)

**Step 1:** Go to Google Cloud Storage and click on the “**Create Bucket**” button

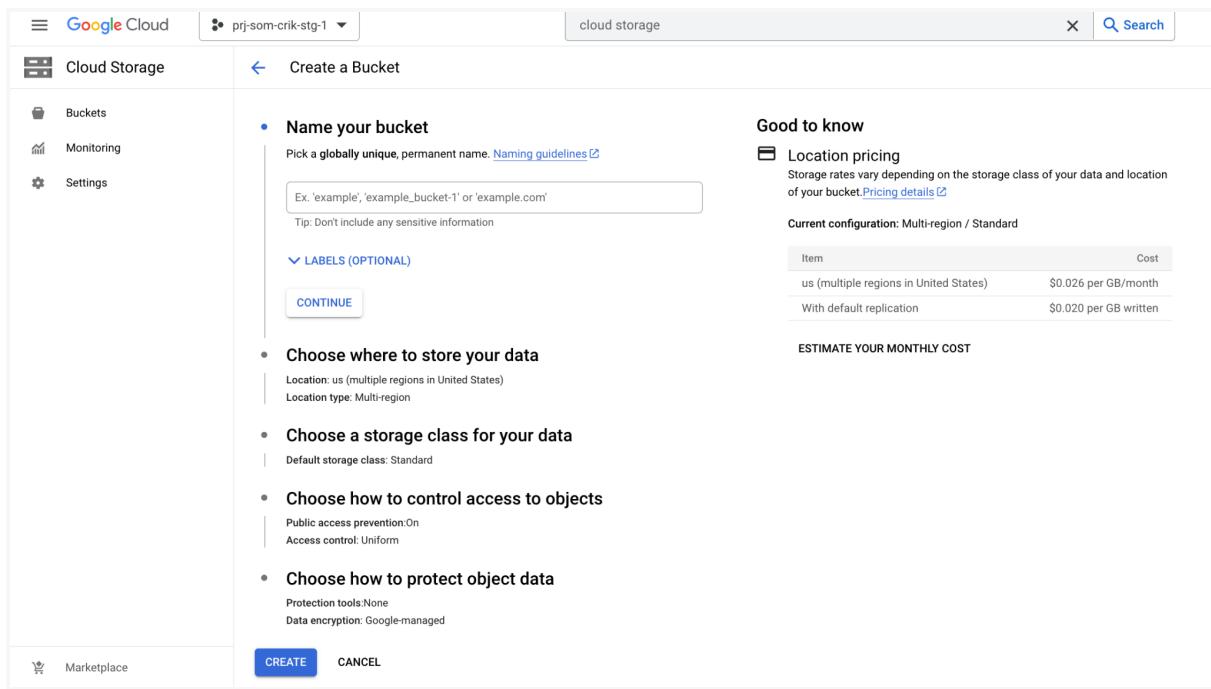


The screenshot shows the Google Cloud Storage Buckets interface. At the top, there's a navigation bar with 'Google Cloud' and a dropdown for 'prj-som-crik-stg-1'. A search bar on the right contains the text 'cloud storage'. Below the navigation is a table with the following columns: 'Buckets', '+ CREATE', and 'REFRESH'. The 'Buckets' column lists one item: 'crikfun-stage'. The table has a header row with columns: 'Name ↑', 'Created', 'Location type', 'Location', and 'Default storage class'. The 'crikfun-stage' entry shows 'Name ↑' as 'crikfun-stage', 'Created' as '25 Sept 2023, 15:38:07', 'Location type' as 'Region', 'Location' as 'asia-south1', and 'Default storage class' as 'Standard'. On the left, a sidebar menu includes 'Cloud Storage' (selected), 'Buckets' (selected), 'Monitoring', and 'Settings'.

Buckets	+ CREATE	REFRESH		
<input type="checkbox"/> <a href="#">crikfun-stage</a>	Created 25 Sept 2023, 15:38:07	Location type Region	Location asia-south1	Default storage class Standard

**Step 2:** Configure Bucket Settings.

- **Name:** Choose a unique name for your bucket (bucket names must be globally unique across all of GCP).
- **Location:** Select the location where you want your bucket's data to be stored. (Choose the region or multi-region)
- **Default storage class:** Choose the storage class that meets your data storage requirements (e.g., Standard, Nearline, Coldline, etc.).
- **Access control:** Configure your bucket's access permissions and select publicly accessible or keep it private.



**Step 3:** Review the bucket configuration after that click the "Create" button.

## Deployment Pipeline on Google Cloud with Cloud Build (CI/CD)

**Step 1:** Go to **Cloud Build** and select **Triggers** then click the "create trigger" button.

The screenshot shows the Google Cloud Build Triggers page. The left sidebar has 'Cloud Build' selected. The main area shows a table of triggers. One trigger is listed:

Name	Description	Repository	Event	Build configuration	Status
crikfun-codepipeline-trigger	-	Amityadavv/LiveScoring	Push to branch	cloudbuild.yaml	Enabled

## Step 2: Set Up the Cloud Build Trigger

The screenshot shows the 'Edit trigger' form for the 'crikfun-codepipeline-trigger'. The left sidebar has 'Triggers' selected. The main form includes fields for Name, Region, Description, and Tags, along with sections for Event and Source.

**Trigger Details:**

- Name:** crikfun-codepipeline-trigger
- Region:** asia-south1 (Mumbai)
- Description:** (empty)
- Tags:** (empty)

**Event:**

Repository event that invokes trigger

Push to a branch

Push new tag

Pull request  
Not available for Cloud Source Repositories

Or in response to

Manual invocation

Pub/Sub message

Webhook event

**Source:**

Repository generation

1st gen

2nd gen

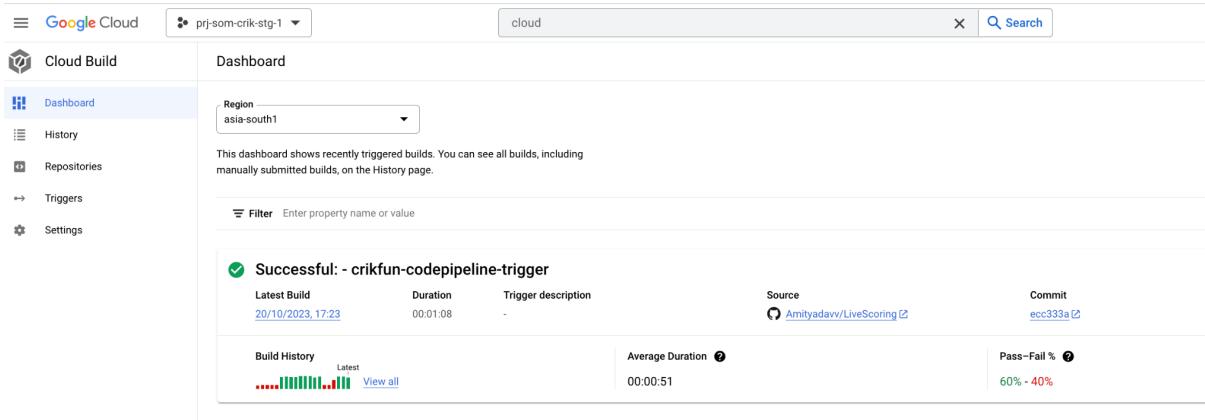
**Repository:** Amityadavv-LiveScoring

## Step 3: Triggering build

1. Commit and push any changes to the master branch of your repository.
2. Run the trigger manually by clicking the **Run trigger** button.

Name	Description	Repository	Event	Build configuration	Status
crikfun-codepipeline-trigger	-	Amityadavv/LiveScoring	Push to branch	cloudbuild.yaml	Enabled

## Step 4: Select Dashboard and see cloud build update.

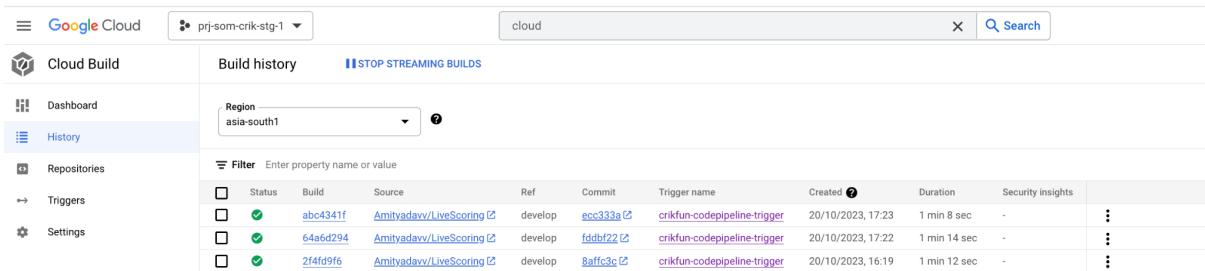


The screenshot shows the Google Cloud Cloud Build dashboard. The left sidebar has links for Dashboard, History, Repositories, Triggers, and Settings. The main area shows a successful build for the trigger 'crikfun-codepipeline-trigger'. The build details are as follows:

Latest Build	Duration	Trigger description	Source	Commit
20/10/2023, 17:23	00:01:08	-	Amityadavv/LiveScoring	ecc333a

Below this, there's a 'Build History' section with a chart showing build status over time, and summary metrics for Average Duration (00:00:51) and Pass-Fail % (60% - 40%).

## Step 5: Check cloud build history.



The screenshot shows the Google Cloud Cloud Build History page. The left sidebar has links for Dashboard, History, Repositories, Triggers, and Settings. The main area shows three recent builds:

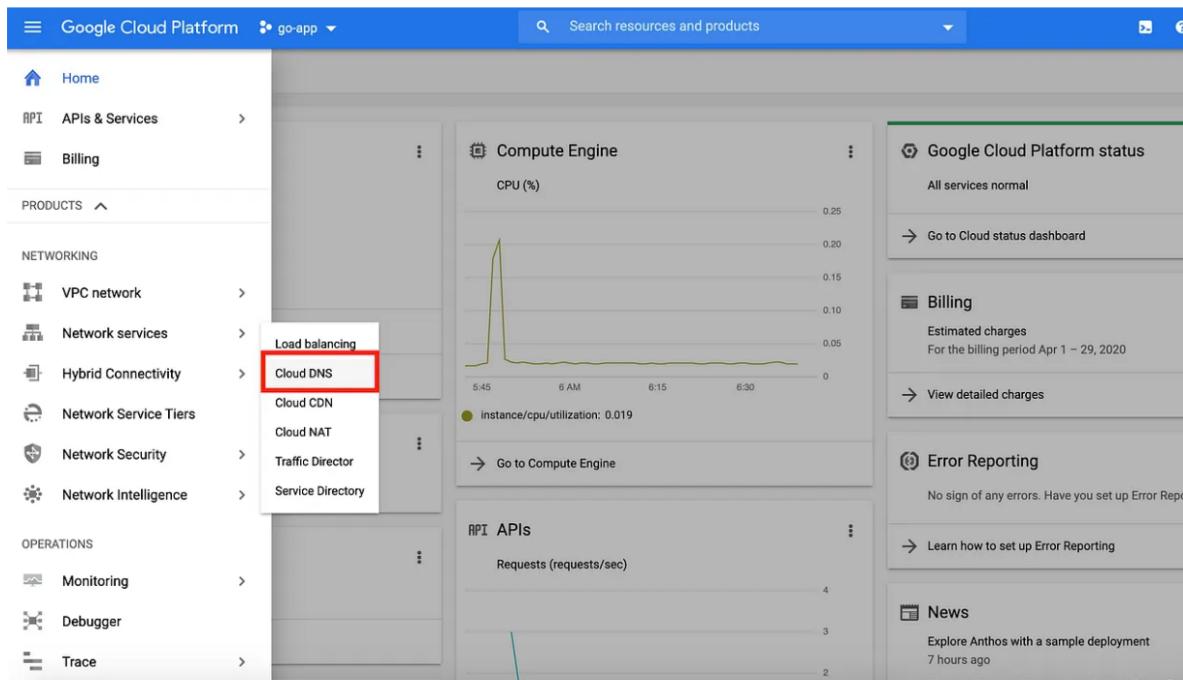
Status	Build	Source	Ref	Commit	Trigger name	Created	Duration	Security insights
Success	abc4341f	Amityadavv/LiveScoring	develop	ecc333a	crikfun-codepipeline-trigger	20/10/2023, 17:23	1 min 8 sec	-
Success	64a6d294	Amityadavv/LiveScoring	develop	f0dbf22	crikfun-codepipeline-trigger	20/10/2023, 17:22	1 min 14 sec	-
Success	2f4fd9f6	Amityadavv/LiveScoring	develop	8affc3c	crikfun-codepipeline-trigger	20/10/2023, 16:19	1 min 12 sec	-

## Source

- <https://medium.com/ci-t/how-to-set-up-a-deployment-pipeline-on-gcp-with-cloud-build-container-registry-and-cloud-run-73391f5b77e4>

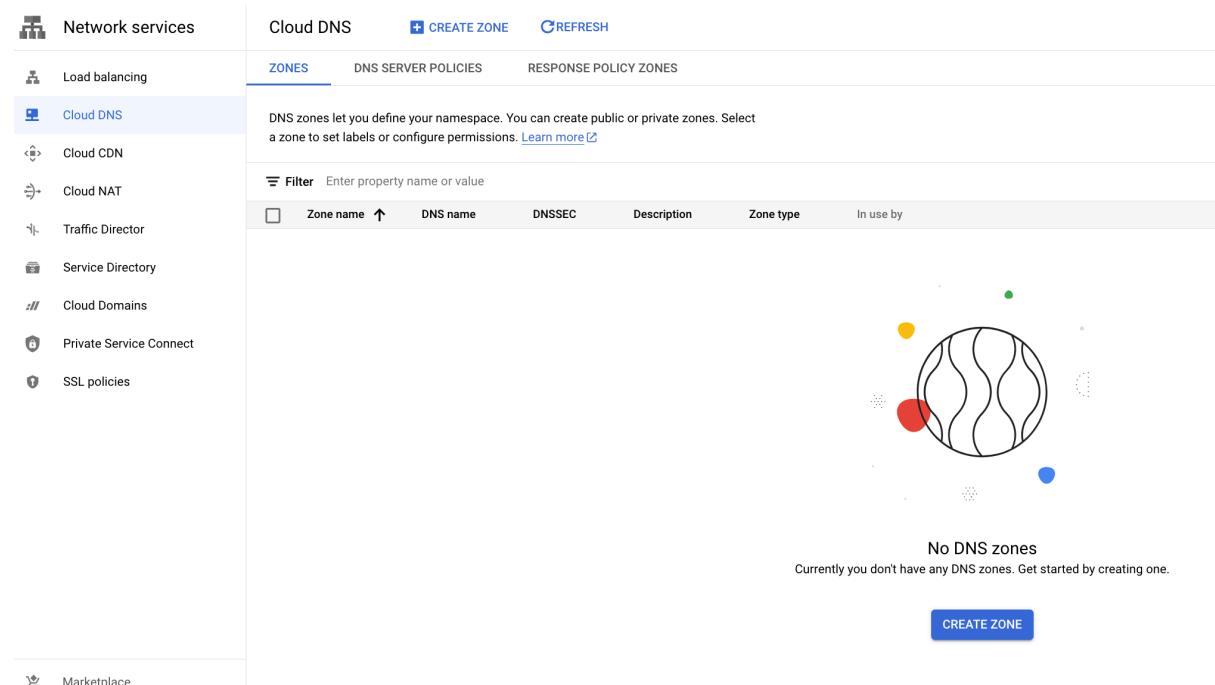
# Setup Cloud DNS

**Step 1:** Go to **Network services** and Choose **Cloud DNS** on the sub menu.



The screenshot shows the Google Cloud Platform dashboard. On the left, there's a navigation sidebar with sections like Home, APIs & Services, Billing, and PRODUCTS. Under PRODUCTS, the Network services section is expanded, showing sub-options: Load balancing, Cloud DNS (which is highlighted with a red box), Cloud CDN, Cloud NAT, Traffic Director, and Service Directory. The main content area has three cards: Compute Engine (showing CPU usage over time), API APIs (showing requests per second), and Google Cloud Platform status (showing all services normal). The status card includes links to Go to Cloud status dashboard, View detailed charges, and Learn how to set up Error Reporting.

**Step 2:** Go to the “**Cloud DNS**” page and click the “**Create zone**” button for hosting the zone.



The screenshot shows the Cloud DNS page under the Network services section. The left sidebar lists options like Load balancing, Cloud DNS (which is selected and highlighted in blue), Cloud CDN, Cloud NAT, Traffic Director, Service Directory, Cloud Domains, Private Service Connect, SSL policies, and Marketplace. The main content area has tabs for ZONES, DNS SERVER POLICIES, and RESPONSE POLICY ZONES. Below the tabs, a note says: "DNS zones let you define your namespace. You can create public or private zones. Select a zone to set labels or configure permissions. [Learn more](#)". There's a search bar labeled "Filter" and a table header with columns: Zone name, DNS name, DNSSEC, Description, Zone type, and In use by. A large, stylized globe icon is in the center. At the bottom, it says "No DNS zones" and "Currently you don't have any DNS zones. Get started by creating one." A prominent blue "CREATE ZONE" button is at the bottom right.

## Step 3: Create a Managed Zone

- Enter a unique Zone name, like "crikfun-com-zone".
- Under "DNS Name," enter your domain name (e.g., crikfun.com).
- Choose the DNSSEC options (optional).
- Click the "Create" button to create the managed zone.

The screenshot shows the Google Cloud Platform interface for creating a DNS zone. The top navigation bar includes 'Google Cloud', a dropdown for 'prj-som-crik-stg-1', a search bar with 'network service', and a close button. The main menu on the left lists 'Network services' (selected), 'Load balancing', 'Cloud DNS' (selected), 'Cloud CDN', 'Cloud NAT', 'Traffic Director', 'Service Directory', 'Cloud Domains', 'Private Service Connect', and 'SSL policies'. The 'Cloud DNS' section has a sub-header 'Create a DNS zone'. It contains fields for 'Zone type' (set to 'Public'), 'Zone name\*' (set to 'crikfun-com-zone'), 'DNS name\*' (set to 'crikfun.com'), 'DNSSEC\*' (set to 'Off'), and a 'Description' field. Below these fields is a note: 'After creating your zone, you can add resource record sets and modify the networks your zone is visible on.' At the bottom are 'CREATE' and 'CANCEL' buttons.

## Step 4: After that Configure DNS Records

- Inside your newly created managed zone, click the "Add Record Set" button.
- Add the necessary DNS records (e.g., A, CNAME, etc.) for your services.
- Click the "Create" button for each record set.

The screenshot shows the Google Cloud Network services interface. On the left sidebar, 'Cloud DNS' is selected. The main area is titled 'Create record set' and shows fields for 'DNS name' (.crikfun.com.), 'Resource record type' (A), 'TTL' (5), and 'TTL unit' (minutes). Below these are sections for 'IPv4 address' and 'IPv6 address'. A red error message 'Provide valid data for the record. Example: 192.0.2.91' is displayed next to the IPv4 address field. At the bottom are 'CREATE' and 'CANCEL' buttons.

## Step 5: Update Domain Registrar Settings

- Log in to your domain registrar's website (e.g., GoDaddy, Namecheap).
- Go to the DNS settings or DNS management section for your domain.
- Replace the existing DNS server records with the ones provided by Google Cloud DNS.

```
ns-cloud-a1.googledomains.com
ns-cloud-a2.googledomains.com
ns-cloud-a3.googledomains.com
ns-cloud-a4.googledomains.com
```

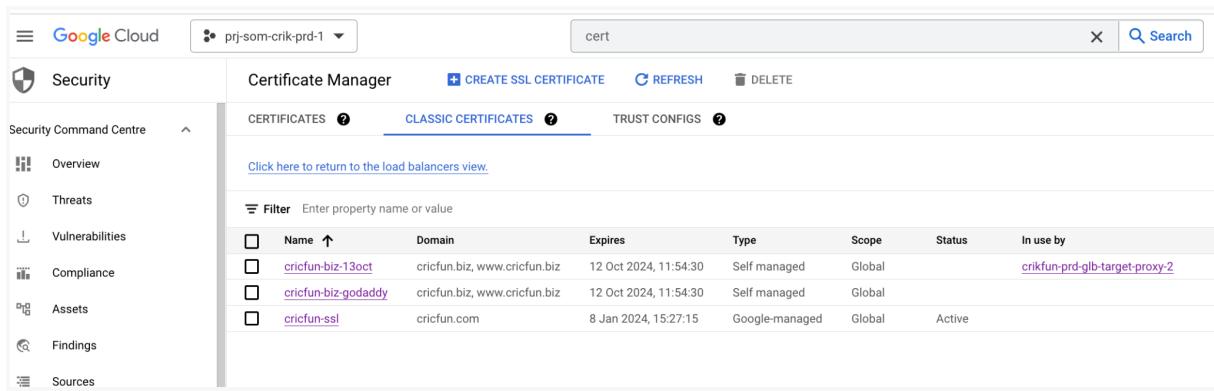
## Source

- <https://cloud.google.com/dns/docs/set-up-dns-records-domain-name>
- <https://bismobaruno.medium.com/configure-domain-to-google-compute-engine-with-google-cloud-dns-2fb0a26832af>

# Enable Ssl on Cloud DNS

**Step 1:** Go to "**Certificate Manager**" and Click the "**Create Certificate**" button.

Create a Managed SSL Certificate



The screenshot shows the Google Cloud Certificate Manager interface. The left sidebar includes sections for Security Command Centre (Overview, Threats, Vulnerabilities, Compliance, Assets, Findings, Sources), and a search bar at the top right. The main area is titled "Certificate Manager" with tabs for "CERTIFICATES", "CLASSIC CERTIFICATES" (which is selected), and "TRUST CONFIGS". A sub-header says "Click here to return to the load balancers view." Below is a table with columns: Name, Domain, Expires, Type, Scope, Status, and In use by. The table lists three entries:

	Name	Domain	Expires	Type	Scope	Status	In use by
<input type="checkbox"/>	cricfun-biz-13oct	cricfun.biz, www.cricfun.biz	12 Oct 2024, 11:54:30	Self managed	Global	Active	crikfun-prd-glb-target-proxy-2
<input type="checkbox"/>	cricfun-biz-godaddy	cricfun.biz, www.cricfun.biz	12 Oct 2024, 11:54:30	Self managed	Global	Active	
<input type="checkbox"/>	cricfun-ssl	cricfun.com	8 Jan 2024, 15:27:15	Google-managed	Global	Active	

**Step 2:** Go to create a certificate option.

- Enter "**Name**" as cricfun-website-cri
- Select Option Create Google-managed certificate in create mode option
- Enter Domain name as "**cricfun.com**".

The screenshot shows the Google Cloud Security Command Centre interface. On the left, there's a sidebar with various security modules like Overview, Threats, Vulnerabilities, Compliance, Assets, Findings, and Sources. The main area is titled 'Create a certificate' and has a 'Name' field set to 'cricfun-website-crt'. Below it, there's a 'Description' field and a 'Create mode' section where 'Create Google-managed certificate' is selected. Under 'Domains', 'Domain 1' is listed as 'cricfun.com'. At the bottom, there are 'CREATE', 'CANCEL', and 'EQUIVALENT COMMAND LINE' buttons.

**Step 3:** Verify the details and click on the Create button.

## Create Redis VM (in Memory Store)

**Step 1:** Go to Google Cloud Memorystore and click on Create a New Redis Instance button.

The screenshot shows the Google Cloud Memorystore interface. The top navigation bar includes 'Google Cloud', a project dropdown 'prj-som-crik-stg-1', and a search bar. The main area is titled 'Instances' and features a 'CREATE INSTANCE' button. Below it, there are tabs for 'REDIS CLUSTER' (selected), 'PREVIEW', 'REDIS' (underlined), and 'MEMCACHED'. A 'Filter' input field is present. A table header with columns 'Instance ID', 'Type', 'Version', 'Location', 'Primary endpoint', 'Read replica', 'Read endpoint', and 'Instance capacity' is shown, followed by a message 'No rows to display'.

**Step 2:** Configure Your Redis Instance

- Write a name for your Redis instance.
- Select the location for your instance. Memorystore is available in specific regions.

**Name your instance**

The instance ID is a permanent and unique identifier. The display name is optional and for display purposes only.

**Instance ID \***  Use lowercase letters, numbers and hyphens. Start with a letter.

**Display name**

**Tier selection**

Determines availability, cost and performance.

**Basic**  
Lower cost. Does not provide high availability.

**Standard**  
Supports automatic failover for high availability and up to five read replicas for scaling reads. [Learn more](#)

**Capacity**

GB  
Provision up to 300 GB of memory.

**Choose region and zonal availability**

Specify the region where your instance will be deployed. For basic tier, you can choose the zone where the instance is deployed. [Learn more](#)

Region *	Zone
asia-south1 (Mumbai)	Any

**Summary**

Tier	Standard
Location	asia-south1
Estimated maximum throughput (MB/s) <small>?</small>	1250 / 2000

**Cost estimate**

Based on instance tier, region and capacity. [Pricing details](#)

20 GB with 2 read replicas	US\$1,314.00/month
----------------------------	--------------------

\*\*Save up to 20% by committing to one year and 40% by committing to three years using committed-use discounts. [Learn more](#)

- Choose the Redis version that you want to use.
- Specify the amount of memory you need for your Redis cache.
- Configure your network settings, such as choosing a VPC network and subnet.

**Set up connection**  
Choose how you would like applications to access your instance. Choice is permanent. [Learn more](#)

**Network \***  
vpc-som-crik-stg-1

**Additional configurations**

**Connections**  
Choose how you would like to manage your IP range and network connection to the Memorystore network. Your choice is permanent. [Learn more](#)

**Security**  
Choose client authentication and encryption settings for your instance

**Maintenance**  
About twice a year, instances must undergo routine maintenance to fix bugs, prevent security exploits and perform upgrades. Your instance must be restarted while updates are made, but you can choose when this happens. [Learn more](#)

**Configuration**  
Customise the behaviour of your instance's Redis configuration and enable snapshots. [Learn more](#)

**Summary**

Tier	Standard
Location	asia-south1
Estimated maximum throughput (MB/s)	1250 / 2000

**Cost estimate**  
Based on instance tier, region and capacity. [Pricing details](#)

20 GB with 2 read replicas	US\$1,314.00/month
----------------------------	--------------------

\*\*Save up to 20% by committing to one year and 40% by committing to three years using committed-use discounts. [Learn more](#)

**CREATE INSTANCE** CANCEL EQUIVALENT COMMAND LINE

## Step 3: Advanced Settings (Optional)

- Configure additional settings such as access control, client export configuration, and backup schedules based on your specific use case.

## Step 4: Review the configuration details.

- If everything looks good, click the "**Create**" button to create your Google Cloud Memorystore Redis instance.

## Source

- <https://cloud.google.com/memorystore/docs/redis/create-manage-instances>
- <https://medium.com/google-cloud/memorystore-redis-access-through-vpc-peering-3bb75e1746d4>

