

GUIDE - serve your domain (e.g. machinelearning.000.pe) from a GCP bucket

To configure your static website with a custom domain, e.g. "machinelearning.000.pe", mostly using only CLI commands, including `gsutil` and `gcloud`, follow these steps.

0. Verify Domain Ownership with Google

<https://search.google.com/search-console/>

See sub-guide for that.

At the DNS tool of your DNS provider, create a CNAME record "go" to "c.storage.googleapis.com"

In your gcloud environemnt, set a project with a valid billing account.
`gcloud config set project <your project id>

Example:
`gcloud config set project prj-id-am-test-240312`

1. Create a Bucket, with name based on your domain name

Create the bucket:
`gsutil mb -p <project id> -c STANDARD -l <location> -b on gs://go.<your domain>

Example:
`gsutil mb -p prj-id-am-test-240312 -c STANDARD -l us-east1 -b on gs://go.machinelearning.000.pe`

If you did NOT do step 0, it will fail with a 403 error.

2. Make the Bucket Publicly Accessible

`gsutil iam ch allUsers:objectViewer gs://go.<your domain>`

Example:
`gsutil iam ch allUsers:objectViewer gs://go.machinelearning.000.pe`

This command allows public access to the objects within your bucket.

3. Create and upload files to your Website

For example, working oon the Google Cloud shell, or wherever you can issue gcloud commands:

```
mkdir site
cd site
touch index.html
touch 404.html
Edit index.html in the Cloud Shell editor, save.
Edit 404.html in the Cloud Shell editor, save.
```

Use the following command to upload content:
`gsutil -m cp -r ./ gs://go.<your domain>`

Example:
`gsutil -m cp -r ./ gs://go.machinelearning.000.pe`

4. Set Main and Error Pages for the Website

`gsutil web set -m index.html -e 404.html gs://go.<your domain>`

Example:
`gsutil web set -m index.html -e 404.html gs://go.machinelearning.000.pe`

This sets `index.html` as the main page and `404.html` as the error page.

5. That is it, for HTTP!

From now on, you can visit your domain and the static website should work via HTTP (only).

<http://go.machinelearning.000.pe/>

6. Adding SSL + HTTPS support

HTTPS requires a load balancer and a SSL certificate.

At some stage, you might be prompted to enable the "Compute Engine API" - do it, if the prompt dialog appears.

6.1. Create a new SSL certificate

Note:

In the end, this will probably NOT work if you can't edit DNS records of type A. But you can execute all the steps and understand all the parts.

Create a new SSL certificate:

```
gcloud compute ssl-certificates create <your cert name> \
  --domains=<your domain>,go.<your domain> \
  --global
```

For example:

```
gcloud compute ssl-certificates create cert-ml-000-pe \
  --domains=go.machinelearning.000.pe \
  --global
```

6.2. Get a new static IP address

Reserve a static IP for the domain, via gcloud:

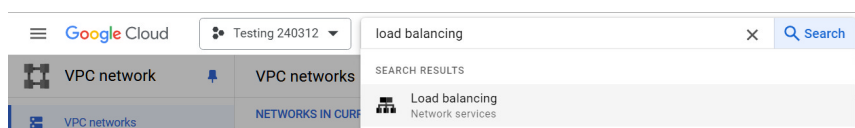
```
gcloud compute addresses create <name-for-the-ip> --project=<project id> --global
```

For example:

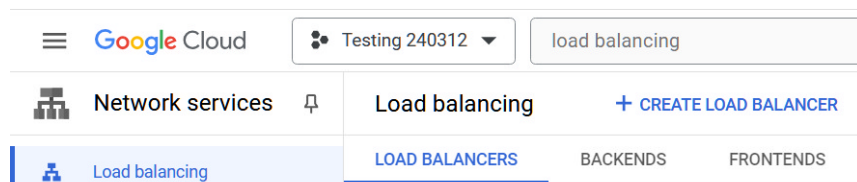
```
gcloud compute addresses create ip-for-machinelearning-000-pe --project=prj-id-am-test-240312 --global
```

6.3 Create a "load balancer"

Search and pick "load balancing".



Pick "create load balancer".



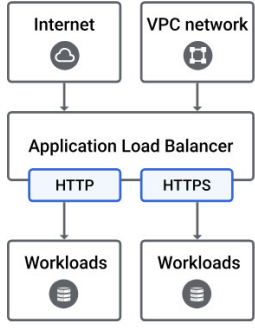
Choose (5 steps):

- "Application load balancer (http/https)"
- "Public facing (external)"
- "Best for global workloads"
- "Global external application load balancer"
- "Create load balancer"

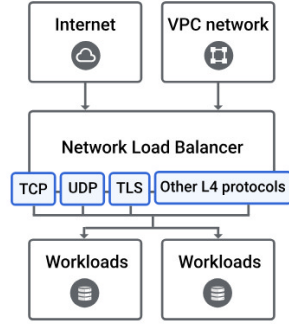
[←](#) Create a load balancer

1 Type of load balancer

- Application Load Balancer (HTTP/HTTPS)**
Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic.



- Network Load Balancer (TCP/UDP/SSL)**
Choose a Network Load Balancer when you need TLS offloading at scale, support for UDP, and exposing IP addresses to your applications.



NEXT

- 2 Public facing or internal**
Public facing (external)
- 3 Global or single region deployment**
Global workloads
- 4 Load balancer generation**
Global external Application Load Balancer
- 5 Create load balancer**

CONFIGURE CANCEL

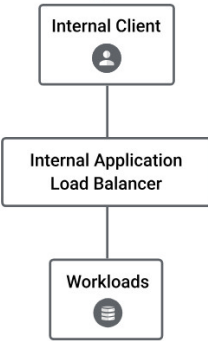
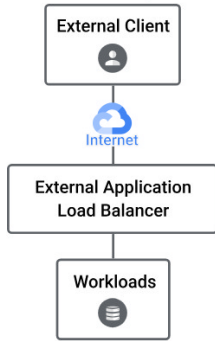
[←](#) Create a load balancer

Type of load balancer

Application Load Balancer

2 Public facing or internal

- Public facing (external)**
An internet-facing load balancer routes requests from clients over the Internet to targets.
- Internal**
An internal load balancer routes requests from clients to backends using private IP addresses.



NEXT

- 3 Global or single region deployment**
Global workloads
- 4 Load balancer generation**
Global external Application Load Balancer
- 5 Create load balancer**

CONFIGURE CANCEL

[← Create a load balancer](#)

✓ Type of load balancer

Application Load Balancer

✓ Public facing or internal

Public facing (external)

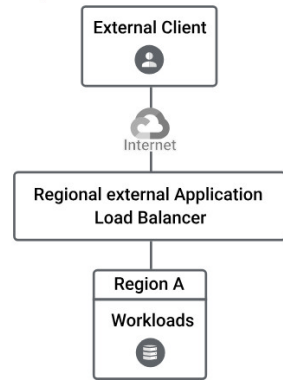
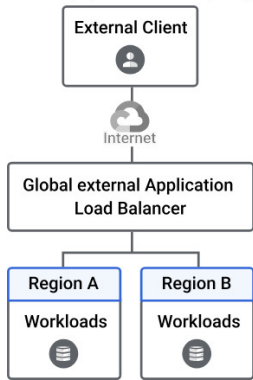
3 Global or single region deployment

Best for global workloads

Multiple regions. Use this for better performance if you have clients distributed globally (with a global anycast IP) or if you want to deploy backends in multiple regions.

Best for regional workloads

Single region. Use this if you want traffic to remain in a region. For example, for workloads with jurisdictional compliance.



NEXT

4 Load balancer generation

Global external Application Load Balancer

5 Create load balancer

[← Create a load balancer](#)

✓ Type of load balancer

Application Load Balancer

✓ Public facing or internal

Public facing (external)

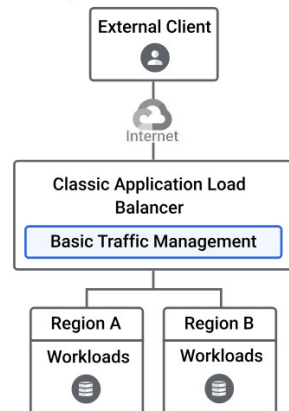
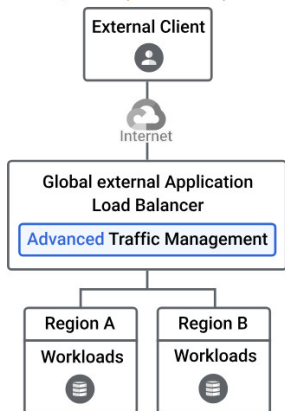
✓ Global or single region deployment

Global workloads

4 Load balancer generation

Global external Application Load Balancer
Load balancer with EXTERNAL_MANAGED load balancing scheme (Recommended)

Classic Application Load Balancer
Previous generation load balancer with EXTERNAL load balancing scheme



NEXT

5 Create load balancer

← Create a load balancer

✓ **Type of load balancer**

Application Load Balancer

✓ **Public facing or internal**

Public facing (external)

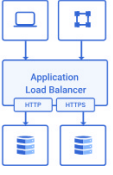
✓ **Global or single region deployment**

Global workloads

✓ **Load balancer generation**

Global external Application Load Balancer

5 Create load balancer



The diagram shows an 'Application Load Balancer' box connected to two server boxes above it and two database boxes below it. The database boxes are labeled 'MySQL' and 'HTTPS'. To the right of the diagram, text reads: 'You are about to create an Application Load Balancer with following features:'. Below this text is a bulleted list: '• Public facing (external)', '• Global'.

CONFIGURE

CANCEL

6.3. Configure the "load balancer"

6.3.1. Do the frontend configuration

- Give the frontend a name
- Choose HTTPS
- Pick the static IP address previously created
- Pick the certificate previously created

Click "done".

Frontend configuration

Configure the load balancer's frontend IP address, port, and protocol. Configure an SSL ce

New Frontend IP and port

Name Lowercase, no spaces.

Description

Protocol

HTTPS (includes HTTP/2 and HTTP/3)
▼

Select HTTPS to support clients that support HTTP/2. The load balancer automatically offers HTTP/2 as part of the TLS handshake.

Network Service Tier
Premium

Global HTTP(S) load balancing only supports the Premium Network Service tier. [Learn more](#)

IP version

IPv4
▼

IP address

ip-for-machinelearning-000-pe
▼

Port *

443
▼

Application load balancing supports all TCP ports. [Learn more](#)

Certificate *

cert-ml-000-pe
▼
?

▼ ADDITIONAL CERTIFICATES

SSL policy *

GCP default
▼

HTTP/3 (QUIC) negotiation

Automatic (default)
▼

Enable HTTP to HTTPS redirect

Requires a reserved external IP address. Enabling HTTP to HTTPS redirect automatically generates a separate URL map with the HTTP to HTTPS redirection configuration.

HTTP keepalive timeout seconds

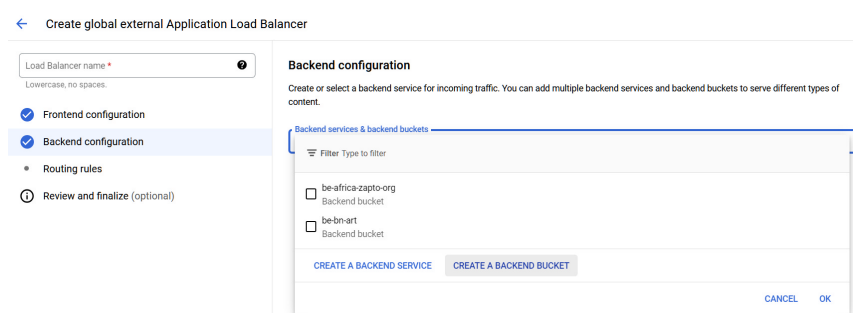
The time an idle client connection is kept open by the load balancer. [Learn more](#)

^ SHOW LESS

DONE

6.3.2 Do the backend configuration

Make sure to create a new "backend bucket"



Browse and select the correct "go.*" bucket for the project.

Name the backend bucket, then press create.

Confirm with "OK".

6.3.3 Complete the load balancer creation

Choose a name for the load balancer (e.g. "lb-ml-000-pe"), press create.

6.3.4 Wait

The configuration might take a few minutes to complete.

Above all, the SSL certificate might ~24 hours to propagate.

But, with a free DNS service where you probably can NOT edit DNS records of type A (you would link the domain name to the static IP adress created earlier), the "provisioning" of the certificate will probably fail and HTTPS will never work.

If you could edit DNS records of type A, the provisioning would complete and https would work.

Sub-guide: verify Domain Ownership with Google

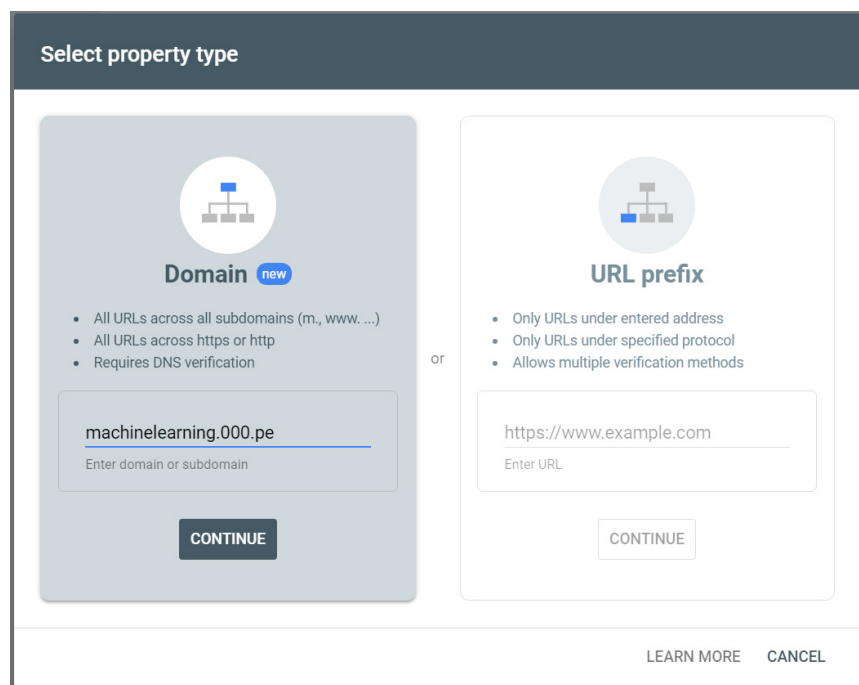
This page and snapshots illustrate the adding of a domain "property" to a Google Cloud account of yours. The idea is to prove that you own a domain name, so you can do things with it in the Google Cloud. For example, so you can host a static website on a bucket named after your domain name.

Start by visiting:

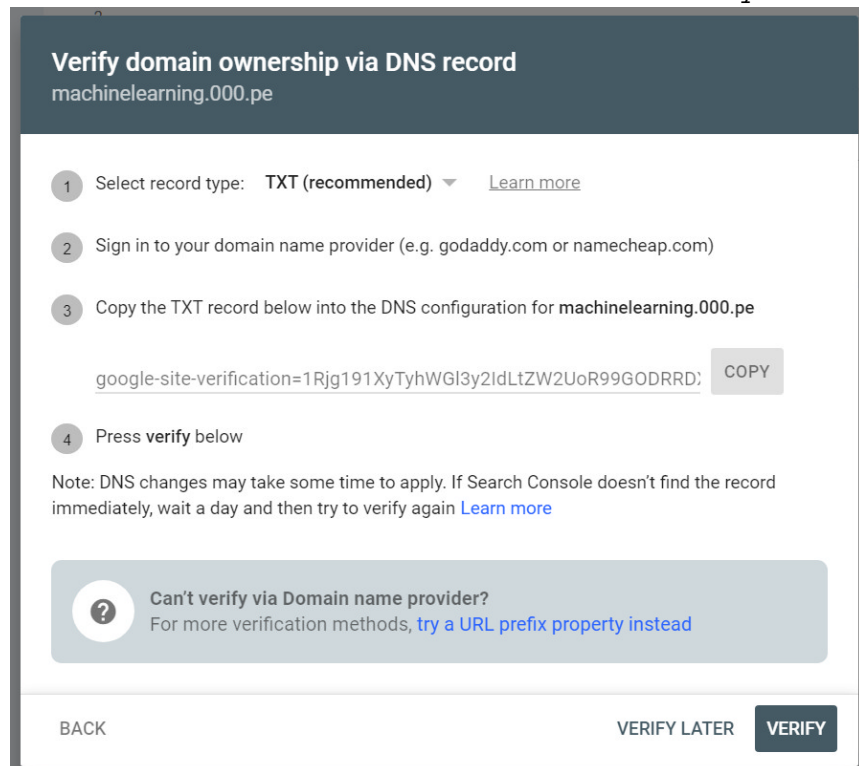
<https://search.google.com/search-console/>

In this example I am trying to add "machinelearning.000.pe".

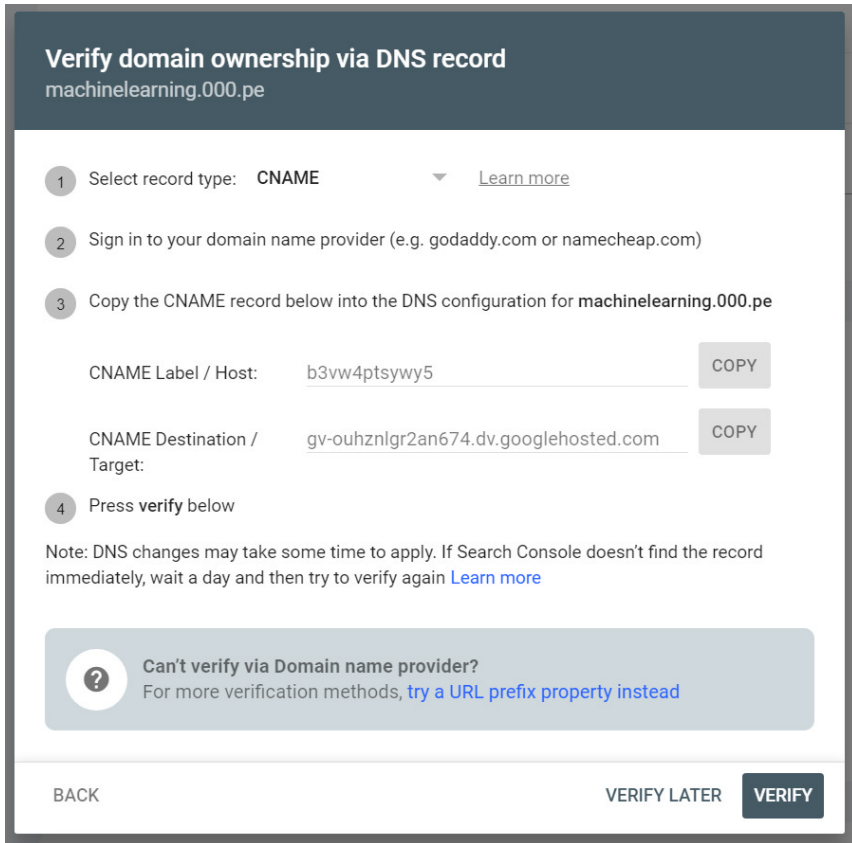
This is free domain name from infinityfree.com, one of the rare free services that will allow you to edit the required DNS records.



TXT records will **NOT** work with infinityfree.com



So, pick validation via **CNAME** records:



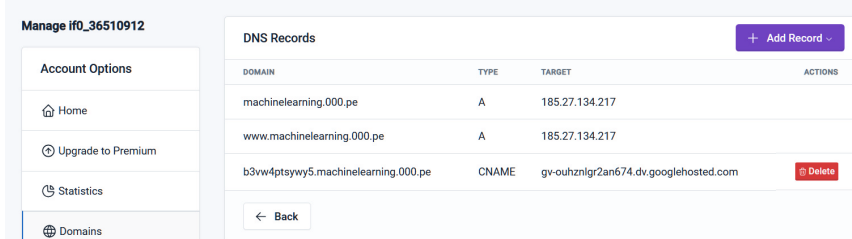
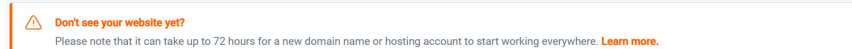
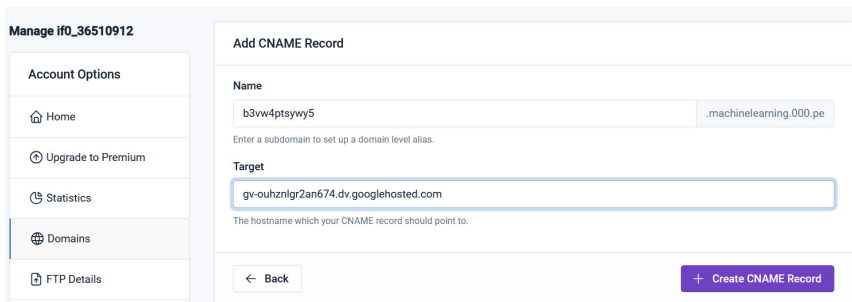
Enter the CNAME label + target at a **NEW CNAME DNS RECORD** at infinityfree.com
 You'll need to know your account ID and your domain name.

URLs will be like:

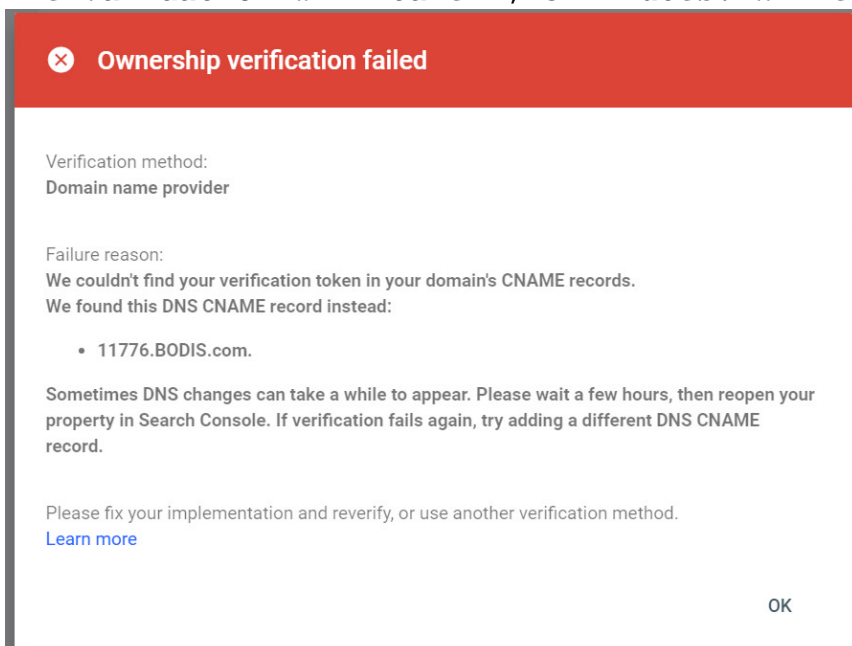
<https://dash.infinityfree.com/accounts/<account id here>/edit>
<https://dash.infinityfree.com/accounts/<account id here>/dnsRecords>

The direct URL for editing the DNS records of some domain name is:


<https://dash.infinityfree.com/accounts/<account id here>/domains/<domain name here>/dnsRecords>



The validation will take 2, 3 minutes. While the DNS propagation happens, you'll get something like:



When the DNS propagation completes, you'll have success:

 **Ownership verified**

Verification method:
Domain name provider

To stay verified, don't remove the DNS-CNAME record. To avoid losing verification, you may want to add multiple verification methods from the **Settings > Ownership verification**.

DONE GO TO PROPERTY

```
# list static IPs
gcloud compute addresses list

gcloud compute addresses delete <ip name> --global

# observe certificates
gcloud compute ssl-certificates describe [CERTIFICATE_NAME] --global

# list certificates
gcloud compute ssl-certificates list

# delete a certificate
gcloud compute ssl-certificates delete [CERTIFICATE_NAME] --global
```