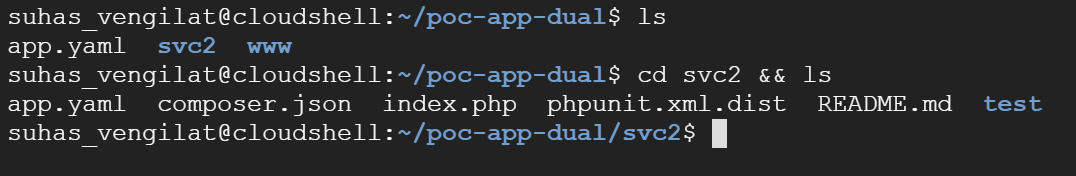
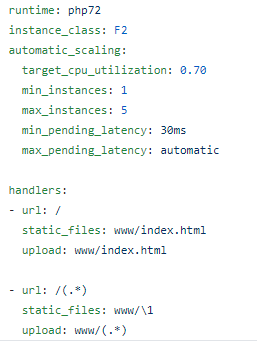
Domain-Mapping with Google Domains for multiple app engine services

1. Create a project for setting up your application
2. Open the Cloud shell session for your project
3. Create a directory with the same name as your project id
4. Upload the files related to the application to the directory just created



Here, www directory contains the files which would build the default service and the svc2 directory contains the files for the second service which we would create

1. Create app.yaml for the first service



For more information on creating app.yaml refer to the link below

<https://cloud.google.com/appengine/docs/standard/python/config/appref>

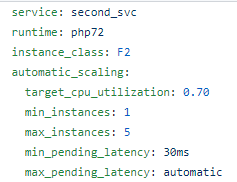
1. Execute the command gcloud app deploy

This will create the default service for the app engine application, we can create a service different from the default as well but it is mandatory to create a default service for app engine.

To stream logs from command line execute:

gcloud app logs tail –s default

1. Deploy the second service



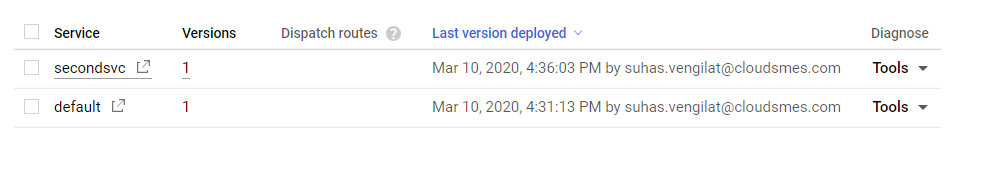
If you have a yaml file with different name, then run

gcloud app deploy filename.yaml

or else move to the directory with app.yaml (svc2) and run the gcloud command

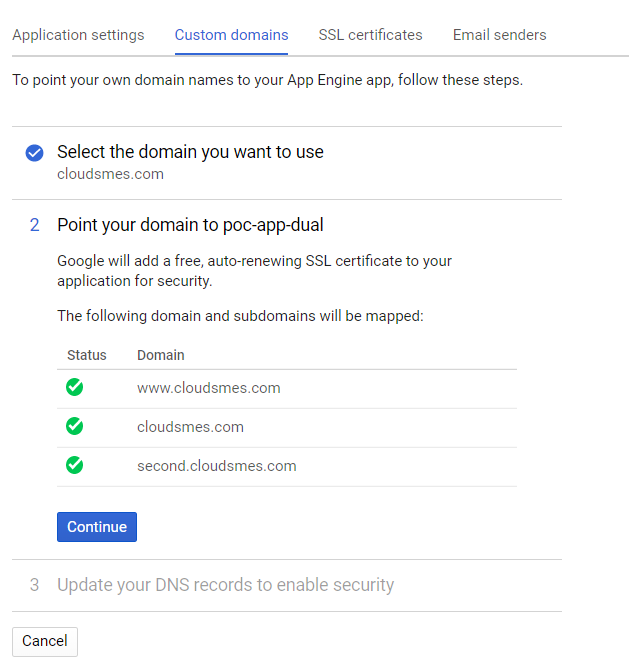
‘gcloud app deploy’

1. Check the services

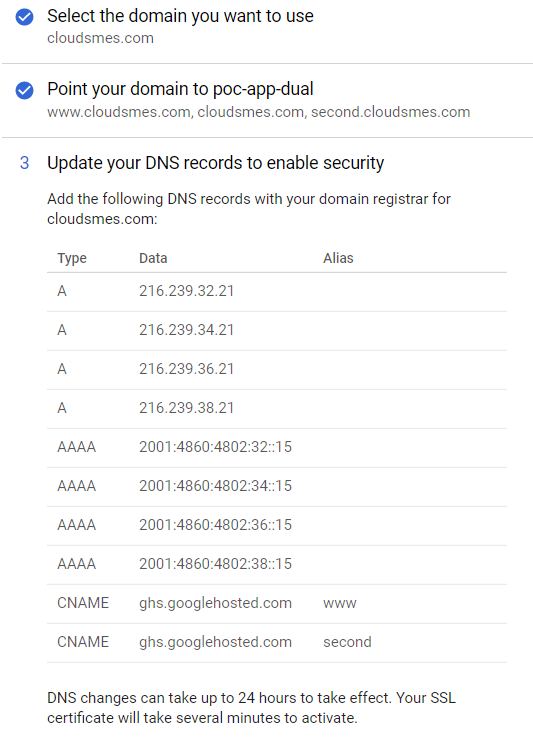


1. Go to settings -> custom domains -> add a custom domain

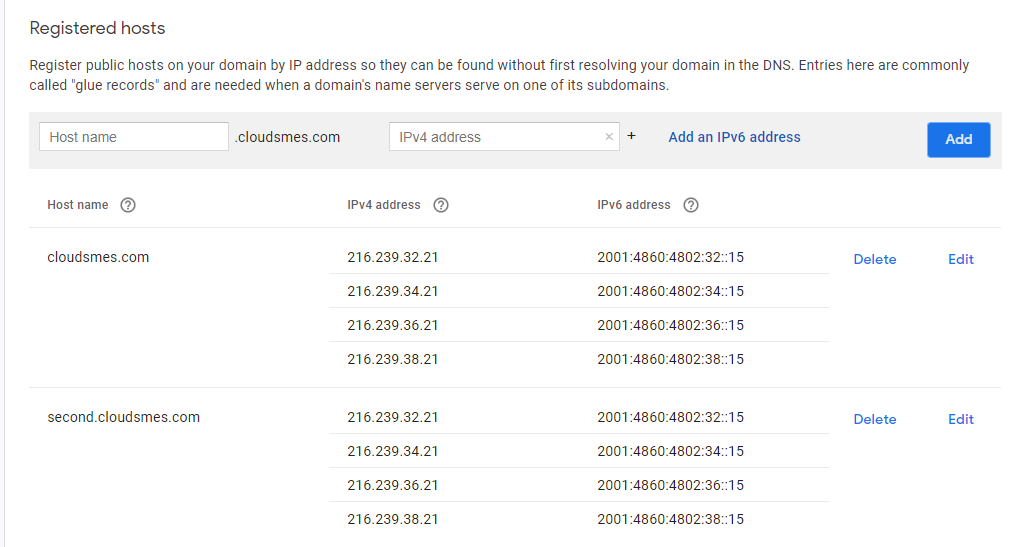
Add your domain names and click on save mappings



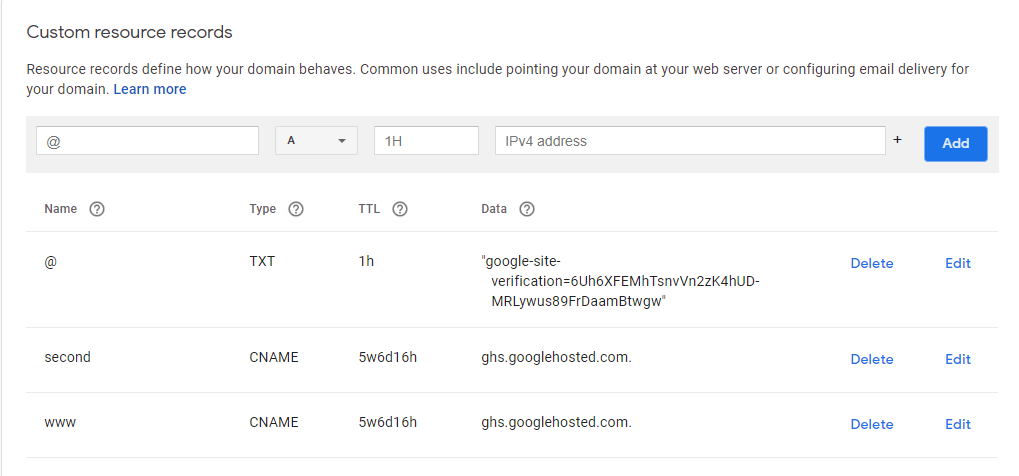
1. Click on continue
2. You will see something similar to this



1. Make a note of all the 8 ip addresses visible here
2. Open your Google Domains console
3. Click on DNS
4. Under Registered Hosts section we need to add these 8 ips for both of the domain names



1. Under custom resource records



The TXT resource record is used to verify that the domain name has been verified by google. This can be achieved through Webmaster central

Refer to the link below for more information

<https://cloud.google.com/appengine/docs/standard/python/mapping-custom-domains>

1. Create a file called dispatch.yaml



1. Run gcloud app deploy dispatch.yaml
2. You should see the default service on [www.cloudsmes.com](http://www.cloudsmes.com) & the second service on second.cloudsmes.com

Errors encountered:

1. Make sure that app.yaml and dispatch.yaml are syntactically correct
2. The domain might already be mapped to another project or app

In that case you need delete the existing mapping

If mapping doesn’t exist and the error prevails

Go to: <https://cloud.google.com/appengine/docs/admin-api/reference/rest/v1/apps.domainMappings/create?apix=true&apix_params=%7B%22parent%22%3A%22apps%2Fdual-svc-deployment%22%2C%22overrideStrategy%22%3A%22OVERRIDE%22%2C%22resource%22%3A%7B%22id%22%3A%22cloudsmes.com%22%7D%7D>

Add parent as apps/project\_id

Add the domain name in request body as

{

“id”: “domain\_name”

}

If everything works properly you should get 200 OK HTTP response

You can then map the domain to app engine

The same can be achieved through command line by executing the command

gcloud app domain-mappings create domain\_name

Repeat the same for second domain and add the IP values to the Google Domain console as shown

An important point to make note of here is that we use managed security for our domains. If you aren’t using managed security, mapping subdomains to services is quite simple. Google’s documentation suggests that you just add a wildcard (\*) CNAME entry to the DNS for your domain.

All subdomains will be served by google. App Engine will try to map a subdomain to a service of the same name.

However, if you are using managed security, you’ll need to tell App Engine about each individual subdomain so that each subdomain can have its own SSL certificate ( the reason why we need to create dispatch.yaml).