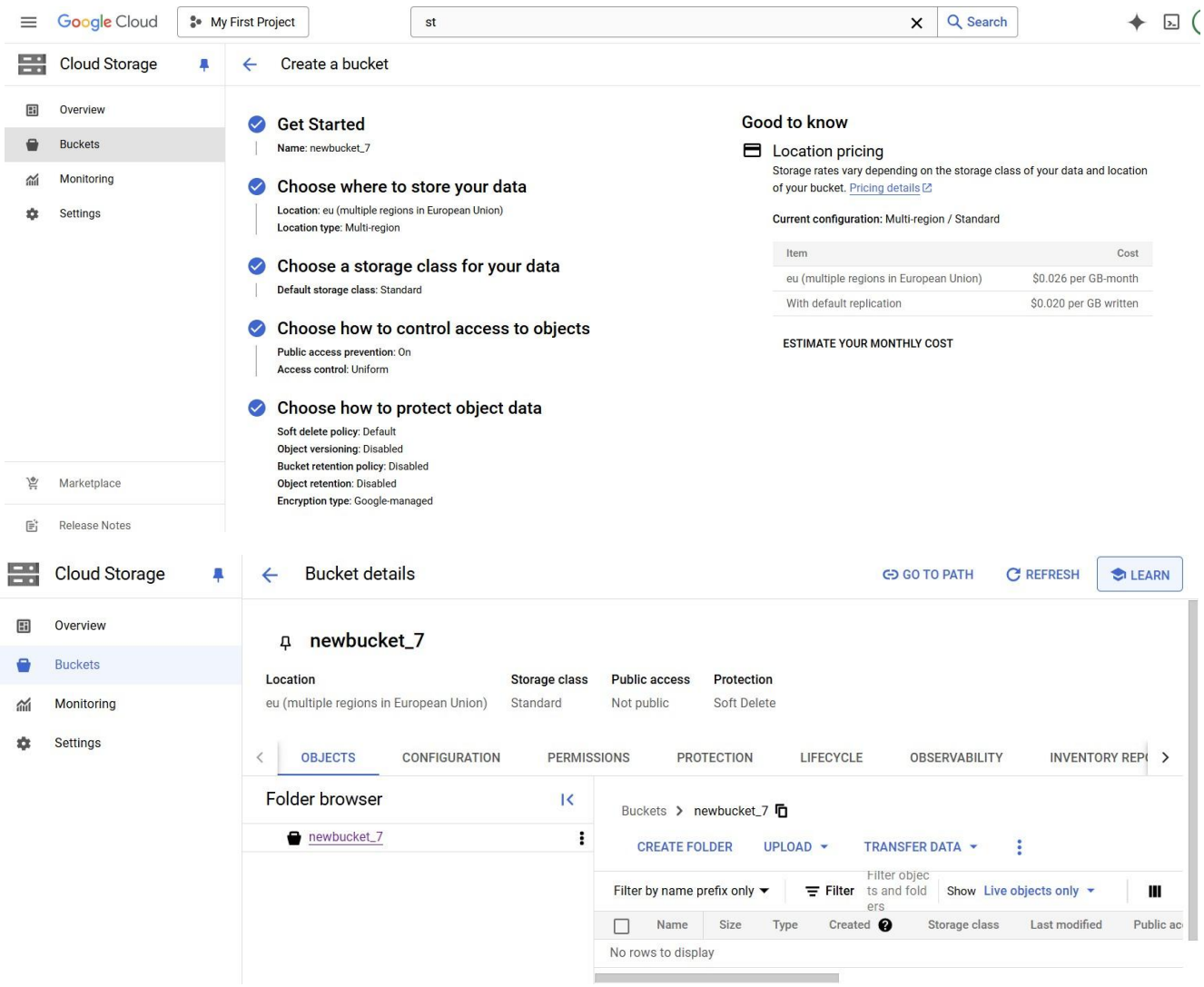
Homework_Lesson_31

1. Создайте новый бакет Amazon S3/GCP Cloud Storage тремя способами: через GUI (консоль в браузере), с помощью CLI, конфиг в terraform.

Способ 1: через GUI (веб-консоль)



Способ 2: С помощью CLI (gcloud/gsutil)

Создаем сервисный аккаунт, в нем ключ, ключ импортируем на вм, активируем его и делаем по умолчанию, создаем бакет и проверяем

```
alextmslearn@ubu:~/bucket$ echo $GOOGLE_APPLICATION_CREDENTIALS
/home/alextmslearn/bucket/ordinal-[redacted]f69b8d.json
alextmslearn@ubu:~/bucket$ gcloud auth activate-service-account --key-file=$GOOGLE_APPLICATION_CREDENTIALS
Activated service account credentials for: [bucket@ordinal-[redacted]account.com]
```

```
alextmslearn@ubu:~/bucket$ gcloud config set account bu[redacted].gserviceaccount.com
Updated property [core/account].
alextmslearn@ubu:~/bucket$ gcloud auth list
                 Credentialed Accounts
ACTIVE  ACCOUNT
        584988178322-compute@developer.gserviceaccount.com
*       bucket@ordinal-stone-450713-n3.iam.gserviceaccount.com
```

```
alextmslearn@ubu:~/bucket$ gsutil mb gs://mylapikovbucket/
Creating gs://mylapikovbucket/...
alextmslearn@ubu:~/bucket$
```

Name	Created	Location type	Location	Default storage class	Last modified
mylapikovbucket	Mar 11, 2025, 11:11:54 AM	Multi-region	us	Standard	Mar 11, 2025, 11:11:54 AM
newbucket_7	Mar 10, 2025, 6:01:04 PM	Multi-region	eu	Standard	Mar 10, 2025, 6:01:04 PM

Способ 3: С помощью конфига в terraform.

Ставим terraform и создаем файл конфигурации, инициализируем, проверяем и создаем

```
alextmslearn@ubu:~/terraform$ cat  main.tf
provider "google" {
  credentials = file("/home/alextmslearn/bucket/ordinal-[redacted]8d.json")
  project     = "ordinal-stone-[redacted]"
  region      = "us-central1"
}

resource "google_storage_bucket" "my_bucket" {
  name     = "mylapikovterrabucket"
  location = "US"
}
```

```
alextmslearn@ubu:~/terraform$ terraform apply

Terraform used the selected providers to generate the following execution plan. Resource actions a
  + create

Terraform will perform the following actions:

  # google_storage_bucket.my_bucket will be created
  + resource "google_storage_bucket" "my_bucket" {
      + effective_labels            = {
          + "goog-terraform-provisioned" = "true"
        }
      + force_destroy               = false
      + id                          = (known after apply)
      + location                    = "US"
      + name                        = "mylapikovterrabucket"
      + project                     = (known after apply)
      + project_number              = (known after apply)
      + public_access_prevention    = (known after apply)
      + rpo                         = (known after apply)
      + self_link                   = (known after apply)
      + storage_class               = "STANDARD"
      + terraform_labels            = {
          + "goog-terraform-provisioned" = "true"
        }
      + uniform_bucket_level_access = (known after apply)
      + url                         = (known after apply)

      + soft_delete_policy (known after apply)

      + versioning (known after apply)

      + website (known after apply)
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes
```

```
google_storage_bucket.my_bucket: Creating...
google_storage_bucket.my_bucket: Creation complete after 3s [id=mylapikovterrabucket]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
alextmslearn@ubu:~/terraform$ cat  main.tf
provider "google" {
  credentials = file("/home/alextmslearn/bucket/[redacted]d.json")
  project     = "ordinal-stone-450713-n3"
  region      = "us-central1"
}

resource "google_storage_bucket" "my_bucket" {
  name     = "mylapikovterrabucket"
  location = "US"
}
```

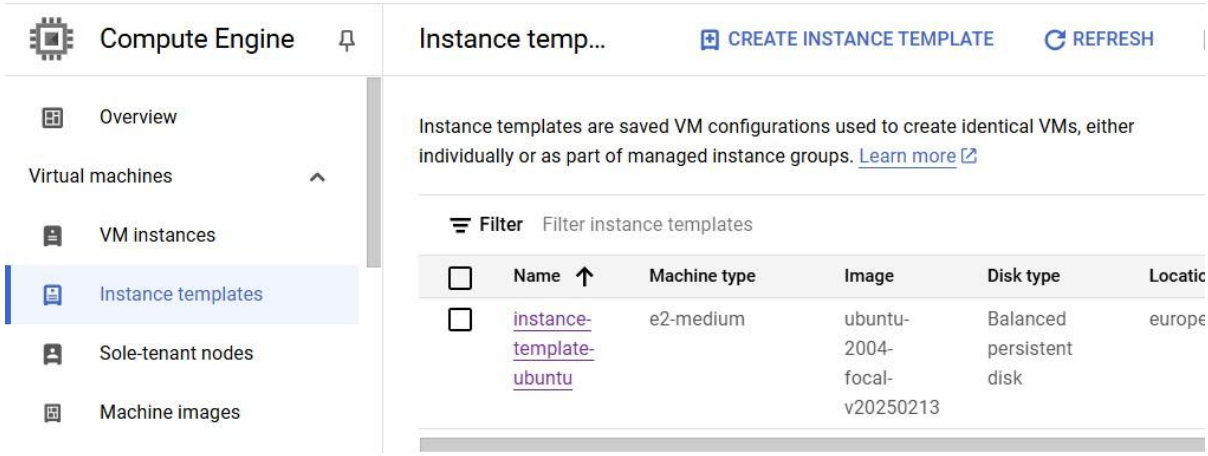
Filter Filter buckets

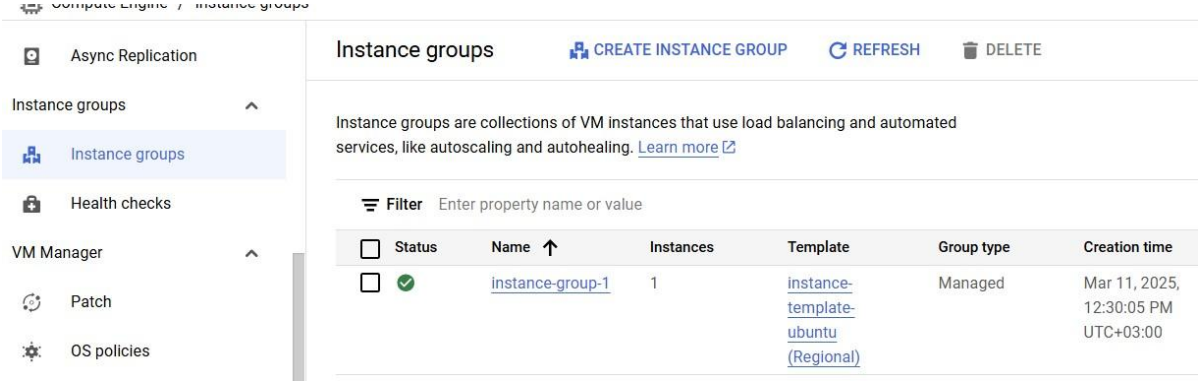
Name	Created	Location type	Location	Default storage class	Last modified
mylapikovbucket	Mar 11, 2025, 11:11:54 AM	Multi-region	us	Standard	Mar 11, 2025, 11:11:54 AM
mylapikovterrabucket	Mar 11, 2025, 11:45:18 AM	Multi-region	us	Standard	Mar 11, 2025, 11:45:18 AM
newbucket_7	Mar 10, 2025, 6:01:04 PM	Multi-region	eu	Standard	Mar 10, 2025, 6:01:04 PM

2. Создайте Amazon EC2/GCP Compute Engine и настройте масштабирование, чтобы автоматически добавлять новые экземпляры в случае увеличения нагрузки. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform.

Способ 1:

Создаем Instance template Ubuntu и Instance group. Включим Autoscaler с параметрами масштабирования

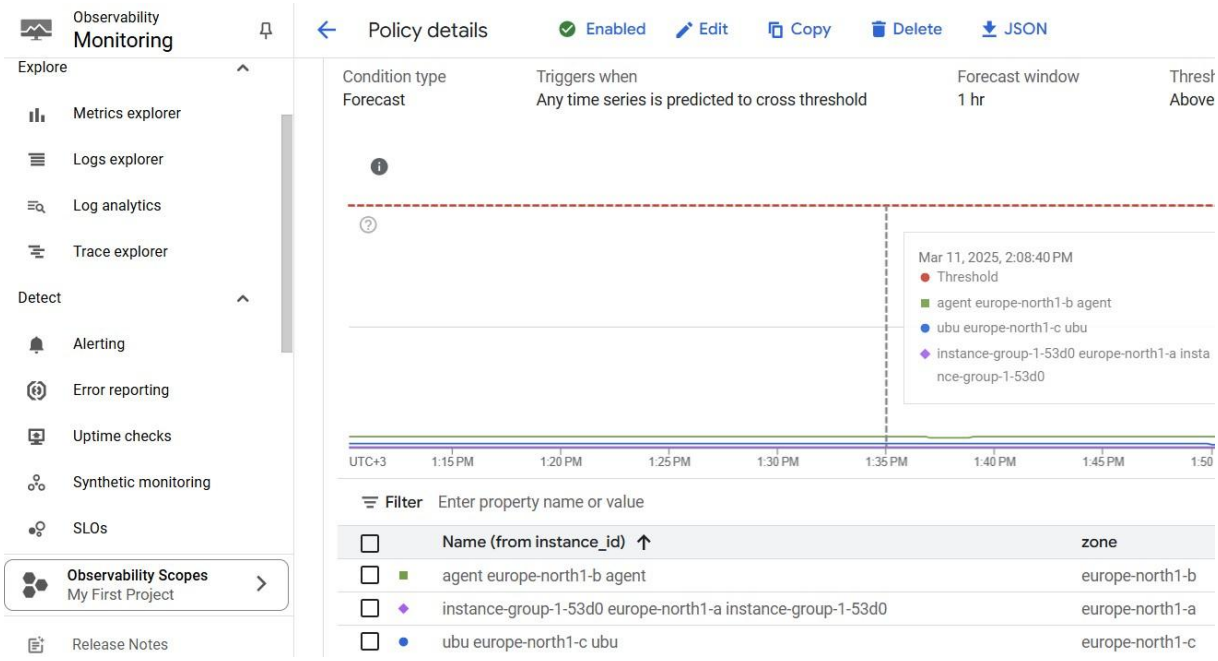




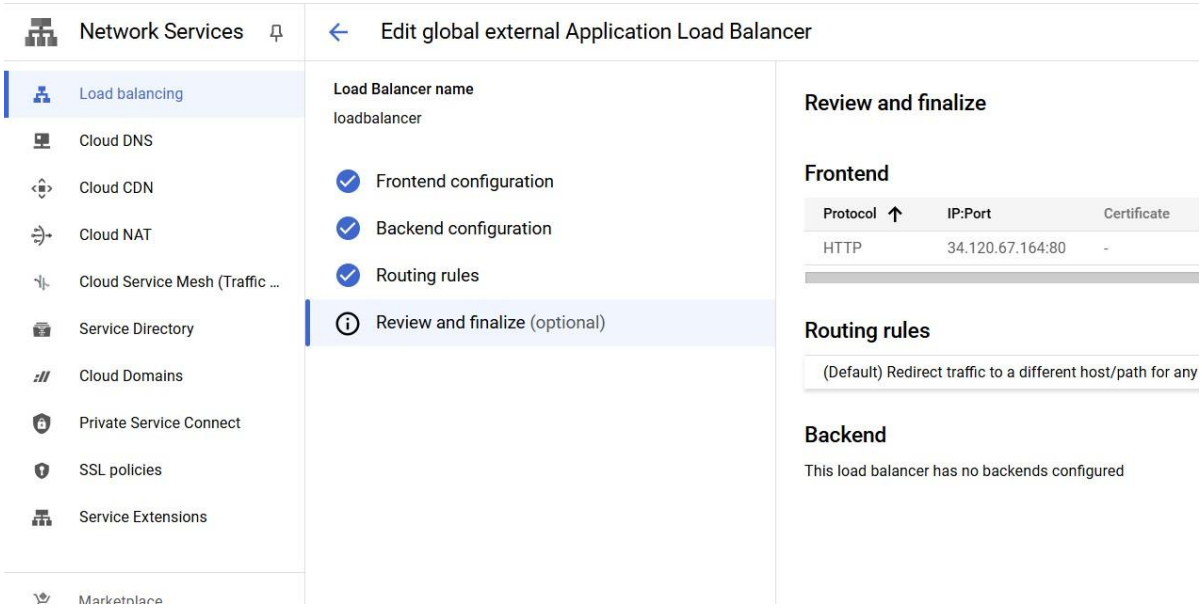
Autoscaling

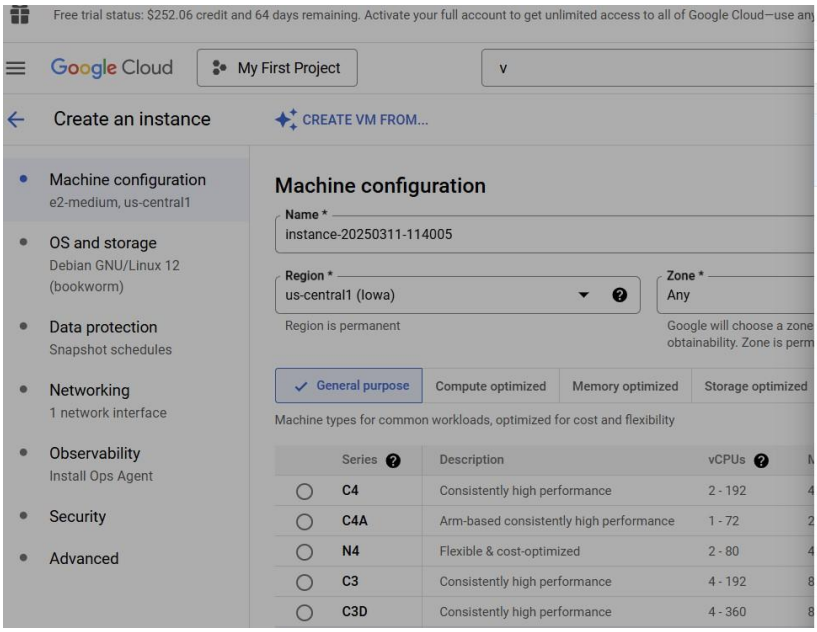
Autoscaling mode	On
Minimum # of instances	1
Maximum # of instances	10
Initialization period	60 seconds
Autoscaling signal	
CPU utilization	60%
Predictive autoscaling	Off
Scale in controls	On
Limit reduction to	10 %
Time frame	10 minutes
Scaling schedules	MANAGE SCHEDULES

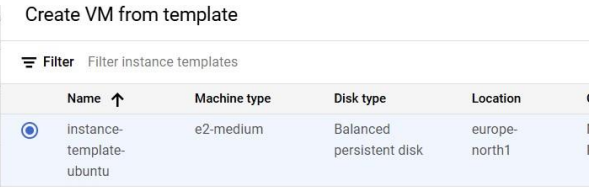
Включим Monitoring по CPU и алерт на почту

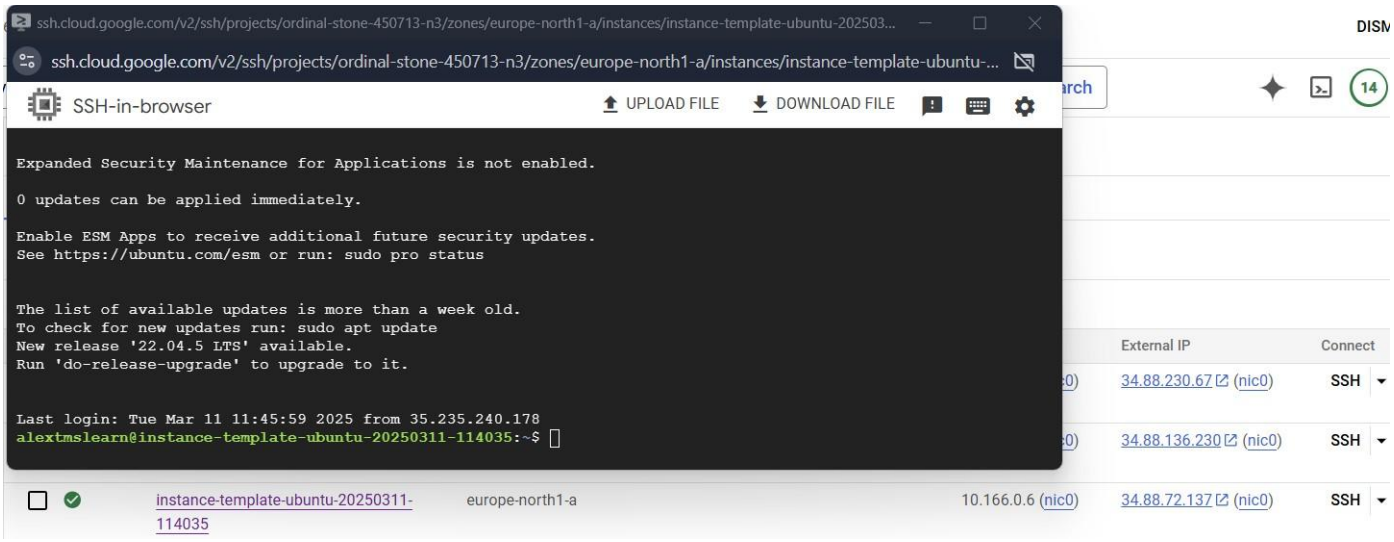


Настроим балансировщика

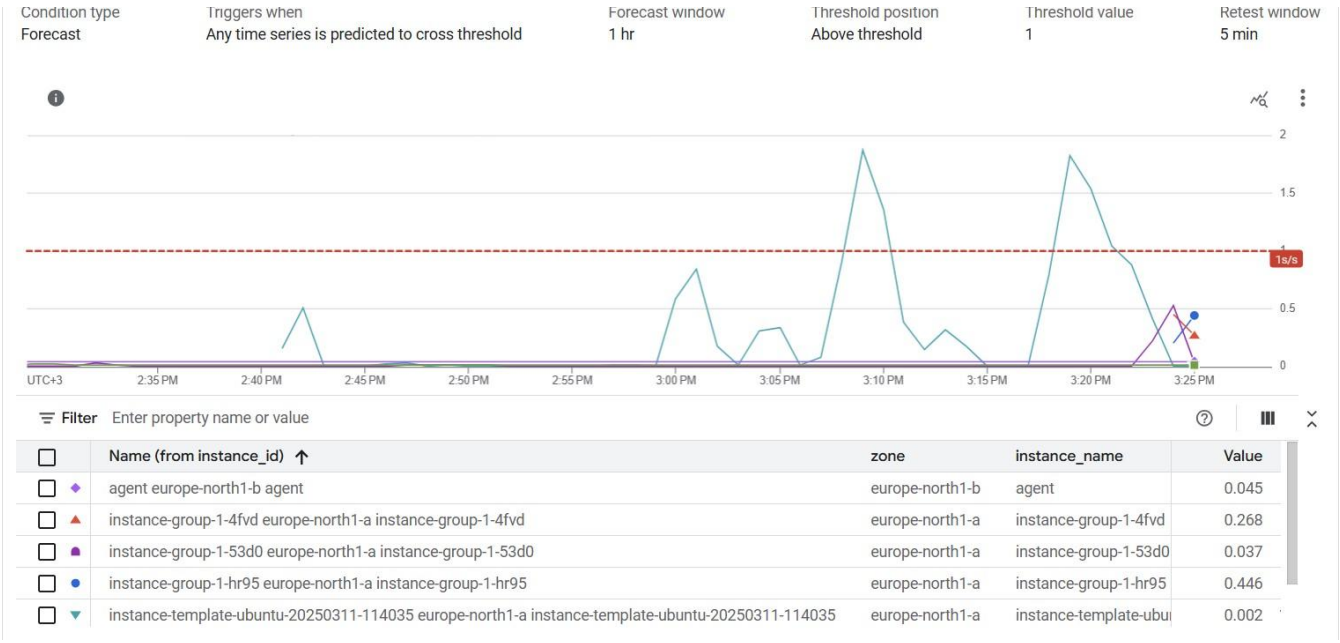








Протестируем балансировку нагрузки с помощью Apache Benchmark



VM instances

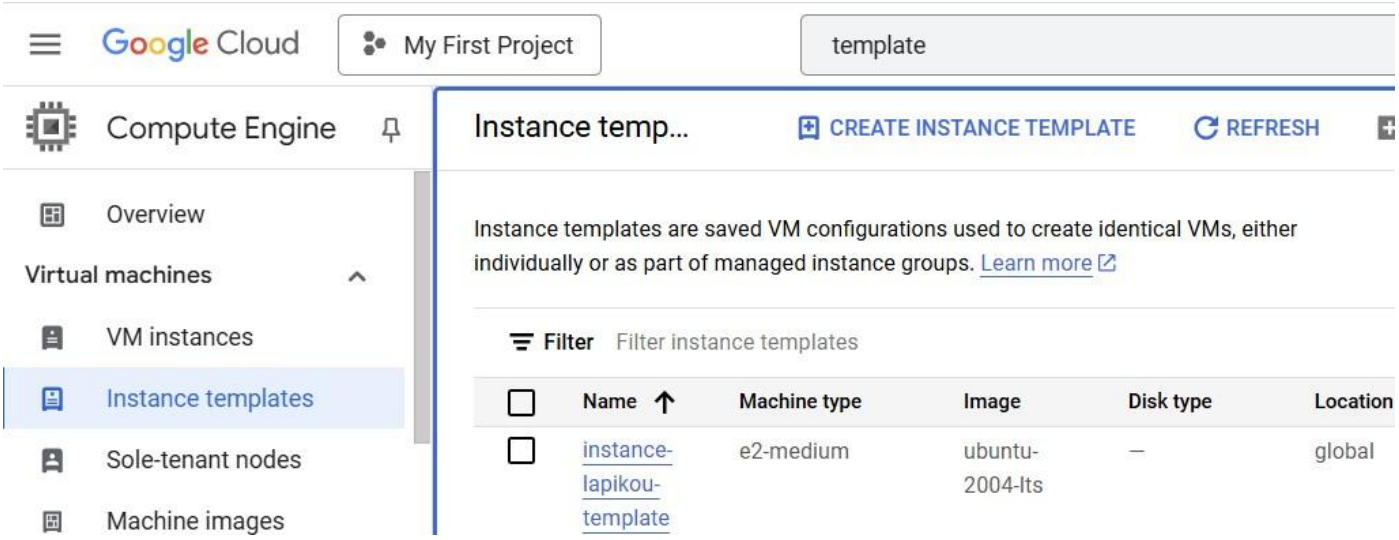
Filter Enter property name or value

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP
✓	agent	europe-north1-b	Save $26 / mo		10.166.0.4 (nic0)	34.88.230.67 (nic0)
✓	instance-group-1-4fvd	europe-north1-a		instance-	10.166.0.7 (nic0)	34.88.161.186 (nic0)
✓	instance-group-1-53d0	europe-north1-a		instance-	10.166.0.5 (nic0)	34.88.136.230 (nic0)
✓	instance-group-1-759l	europe-north1-a		instance-	10.166.0.10 (nic0)	35.228.221.104 (nic0)
✓	instance-group-1-gr5q	europe-north1-a		instance-	10.166.0.9 (nic0)	34.88.240.129 (nic0)
✓	instance-group-1-hr95	europe-north1-a		instance-	10.166.0.8 (nic0)	35.228.73.78 (nic0)
✓	instance-template-ubuntu-20250311-114035	europe-north1-a			10.166.0.6 (nic0)	34.88.72.137 (nic0)
✓	ubu	europe-north1-c			10.166.0.3 (nic0)	34.88.149.200 (nic0)

Способ 3: С помощью terraform

Создаем шаблон экземпляра

```
alextmslearn@ubu:~/terraform$ gcloud compute instance-templates create instance-lapikou-template \
    --image-family ubuntu-2004-lts \
    --image-project ubuntu-os-cloud \
    --machine-type e2-medium \
    --region europe-north1
Created [https://www.googleapis.com/compute/v1/projects/[redacted]nceTemplates/instance-lapikou-template].
NAME                        MACHINE_TYPE  PREEMPTIBLE  CREATION_TIMESTAMP
instance-lapikou-template   e2-medium                  2025-03-11T06:59:43.863-07:00
alextmslearn@ubu:~/terraform$
```



Создаем группу управляемых экземпляров

```
alextmslearn@ubu:~/terraform$ gcloud compute instance-groups managed create instance-lapikou-group \
    --base-instance-name base-lapikou-instance \
    --template instance-lapikou-template \
    --size 1 \
    --zone europe-north1-a
Created [https://www.googleapis.com/compute/v1/projects/o[redacted]713-n3/zones/europe-north1-a/instanceGroupManagers/instance-lapikou-group].
NAME                  LOCATION         SCOPE  BASE_INSTANCE_NAME     SIZE  TARGET_SIZE  INSTANCE_TEMPLATE          AUTOSCALED
instance-lapikou-group  europe-north1-a  zone   base-lapikou-instance  0     1            instance-lapikou-template  no
alextmslearn@ubu:~/terraform$
```

Filter Enter property name or value

Status	Name	Instances	Template	Group type	Creation time	Reco
	instance-lapikou-group	0 → 1	instance-lapikou-template	Managed	Mar 11, 2025, 5:02:11 PM UTC+03:00	

Настроим авто масштабирование для группы экземпляров

```
alextmslearn@ubu:~/terraform$ gcloud compute instance-groups managed set-autoscaling instance-lapikou-group \
    --zone europe-north1-a \
    --target-cpu-utilization 0.6 \
    --min-num-replicas 1 \
    --max-num-replicas 5 \
    --cool-down-period 60
Created [https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/autoscalers/instance-lapikou-group-11nh].
---
autoscalingPolicy:
  coolDownPeriodSec: 60
  cpuUtilization:
    utilizationTarget: 0.6
  maxNumReplicas: 5
  minNumReplicas: 1
  mode: ON
creationTimestamp: '2025-03-11T07:09:00.424-07:00'
id: '2827364437059304723'
kind: compute#autoscaler
name: instance-lapikou-group-11nh
selfLink: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/autoscalers/instance-lapikou-group-11nh
status: ACTIVE
target: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/instanceGroupManagers/instance-lapikou-group
zone: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a
alextmslearn@ubu:~/terraform$
```

Compute Engine / Instance groups / Instance group: instance-lapikou-group

- Disks
- Storage Pools
- Snapshots
- Images
- Async Replication

nstance groups

- Instance groups
- Health checks

VM Manager

- Patch
- OS policies

Bare Metal Solution

instance-lapikou-group EDIT UPDATE VMS RESTART/REPLACE VMS DELETE GROUP

Stopped	0	0
Suspended	0	0

Autoscaling

Autoscaling mode	On
Minimum # of instances	1
Maximum # of instances	5
Initialization period	60 seconds
Autoscaling signal	
CPU utilization	60%
Predictive autoscaling	Off
Scale in controls	Off
Scaling schedules	MANAGE SCHEDULES

Настроим проверку здоровья и балансировщик нагрузки

```
alextmslearn@ubu:~/terraform$ gcloud compute health-checks create http http-health-check \
    --port 80 \
    --request-path /health
Created [https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/global/healthChecks/http-health-check].
NAME                PROTOCOL
http-health-check   HTTP
alextmslearn@ubu:~/terraform$
```

```
alextmslearn@ubu:~/terraform$ gcloud compute backend-services create backend-lapikou-service \
    --protocol HTTP \
    --port-name http \
    --health-checks http-health-check \
    --global \
    --project ordinal-stone-450713-n3
Created [https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/global/backendServices/backend-lapikou-service].
NAME                     BACKENDS  PROTOCOL
backend-lapikou-service            HTTP
alextmslearn@ubu:~/terraform$
```

Проверим работу под нагрузкой

Создаем глобальный IP адрес

```
alextmslearn@ubu:~/terraform$ gcloud compute addresses create lapikou-global-ip --global --project ordinal-stone-450713-n3
Created [https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/global/addresses/lapikou-global-ip].
alextmslearn@ubu:~/terraform$
```

```
alextmslearn@ubu:~/terraform$ gcloud compute addresses describe lapikou-global-ip --global --project ordinal-stone-450713-n3
address: 34.49.95.24
addressType: EXTERNAL
creationTimestamp: '2025-03-11T07:21:56.848-07:00'
description: ''
id: '3522885032882338283'
ipVersion: IPV4
kind: compute#address
labelFingerprint: 42WmSpB8rSM=
name: lapikou-global-ip
networkTier: PREMIUM
selfLink: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/global/addresses/lapikou-global-ip
status: RESERVED
```

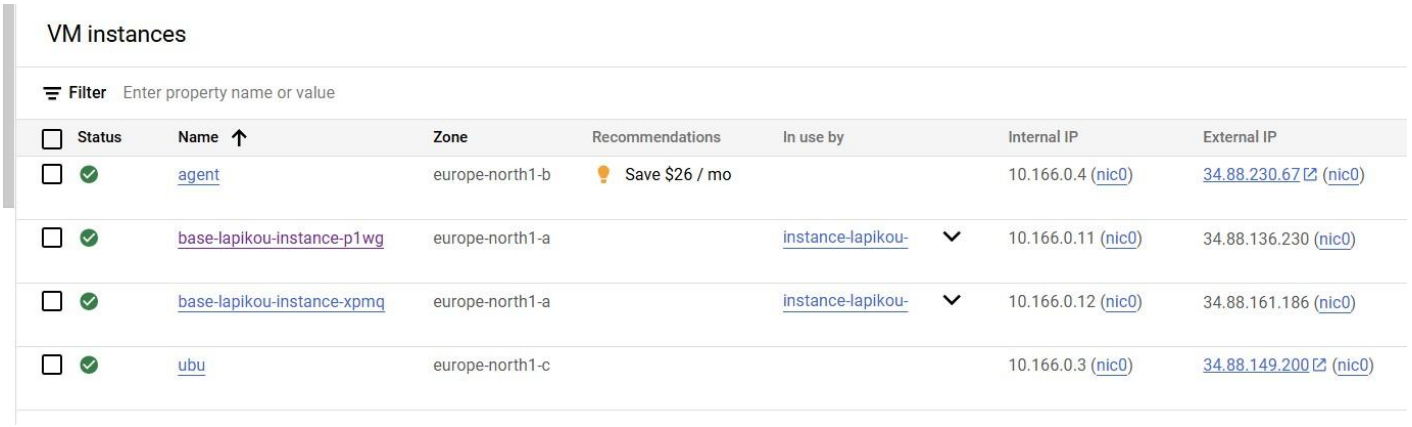
Поставим apache и отправим 1000 запросов с параллельностью в 10 соединений к нашему балансировщику нагрузки:

Нагрузки в 60 % недостаточно для условия масштабирования, укажем принудительно мин 2

```
alextmslearn@ubu:~/terraform$ gcloud compute instance-groups managed set-autoscaling instance-lapikou-group \
    --zone europe-north1-a \
    --target-cpu-utilization 0.6 \
    --min-num-replicas 2 \
    --max-num-replicas 5
Updated [https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/autoscalers/instance-lapikou-group-11nh].
---
autoscalingPolicy:
  coolDownPeriodSec: 60
  cpuUtilization:
    utilizationTarget: 0.6
  maxNumReplicas: 5
  minNumReplicas: 2
  mode: ON
creationTimestamp: '2025-03-11T07:09:00.424-07:00'
id: '2827364437059304723'
kind: compute#autoscaler
name: instance-lapikou-group-11nh
selfLink: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/autoscalers/instance-lapikou-group-11nh
status: ACTIVE
target: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a/instanceGroupManagers/instance-lapikou-group
zone: https://www.googleapis.com/compute/v1/projects/ordinal-stone-450713-n3/zones/europe-north1-a
alextmslearn@ubu:~/terraform$
```

Проверим состояние экземпляров

```
alextmslearn@ubu: /terraform$ gcloud compute instance-groups managed list-instances instance-lapikou-group --zone europe-north1-a
NAME                        ZONE            STATUS   HEALTH_STATE  ACTION  INSTANCE_TEMPLATE          VERSION_NAME  LAST_ERROR
base-lapikou-instance-p1wg  europe-north1-a  RUNNING                NONE    instance-lapikou-template
base-lapikou-instance-xpmq  europe-north1-a  RUNNING                NONE    instance-lapikou-template
alextmslearn@ubu:~/terraform$
```



3.* Создайте базу через GUI (консоль в браузере), с помощью CLI, конфиг в terraform данных Amazon RDS/GCP CloudSQL и подключитесь к ней из виртуальной машины, которую вы создали ранее. Сделайте это тремя способами: через GUI (консоль в браузере) с помощью CLI, конфиг в terraform.