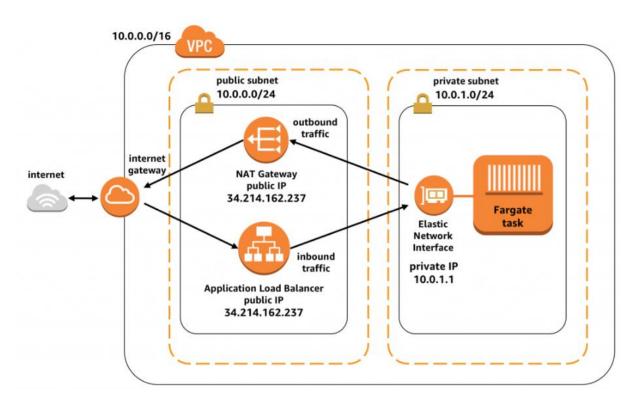
Nama: Primanda Adyatma Hafiz

NIM : 13520022

Source Code

https://github.com/primahafiz/terraform-ecs-cluster

Arsitektur



Arsitektur dari ECS Cluster terdiri atas VPC (Virtual Private Cloud) yang merupakan jaringan virtual yang memuat kumpulan *resources* yang digunakan sehingga dapat saling berinteraksi satu sama lain dan dapat terhubung ke luar VPC. VPC yang dibuat terdiri atas *public subnet* dan *private subnet*. *Public subnet* bisa langsung terhubung ke jaringan eksternal melalui *internet gateway* sedangkan *private subnet* perlu menghubungi *NAT gateway* untuk mengakses jaringan eksternal. Selain itu, seluruh *request* yang masuk akan ditangani oleh *load balancer* untuk selanjutnya diteruskan ke ECS Task pada *private subnet*.

Proses Implementasi

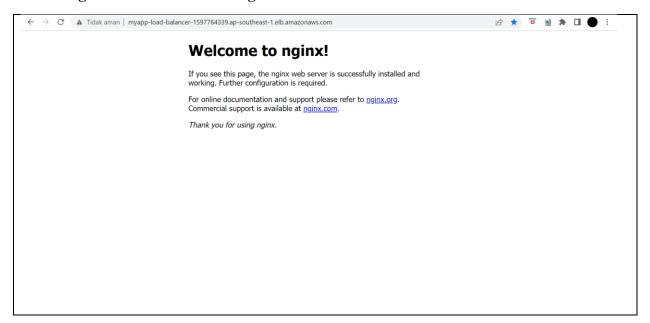
Melihat dokumentasi tentang ECS Cluster AWS
 (https://docs.aws.amazon.com/AmazonECS/latest/developerguide/clusters.html)

- 2) Melihat dokumentasi penggunaan Terraform pada ECS Cluster AWS (https://registry.terraform.io/providers/hashicorp/aws/latest/docs/resources/ecs_cluster)
- 3) Melakukan setup dan instalasi Terraform di lokal
- 4) Mengimplementasikan AWS Component mulai dari ECS Cluster, Load Balancer, kemudian VPC

Kesulitan yang Dialami

- Sebelumnya saya mengimplementasikan *policy* IAM *role* untuk setiap AWS Component tapi ternyata tidak harus diimplementasikan agar komponen dapat berjalan
- Pada awalnya sulit untuk memahami keseluruhan atribut dari setiap komponen tapi setelah membaca dokumentasi pada Terraform dengan lebih detil sekarang sudah lebih memahami

Hasil Pengetesan ECS Cluster dengan Browser



Pelajaran yang Diperoleh

Saya jadi lebih memahami cara untuk mengimplementasikan pembuatan ECS Cluster pada AWS dengan menggunakan Load Balancer. Dalam hal ini, untuk mengimplementasikannya diperlukan VPC sebagai jaringan virtual yang berperan sebagai *container* dari keseluruhan *resource*, selanjutnya diperlukan pula NAT Gateway untuk meneruskan *request* dari *private subnet* ke internet serta *internet gateway* untuk menghubungkan VPC dengan jaringan eksternal. Selain itu, ECS Cluster terdiri atas ECS Task yang menjalankan *job* dari docker *container* dan ECS Service untuk menjalankan beberapa *task* dalam satu waktu.

BONUS

Proses Implementasi

1) Membuat *policy* untuk *autoscaling* dengan metriks berupa ALBRequestCountPerTarget dan target_value bernilai 10 (10 request/target)

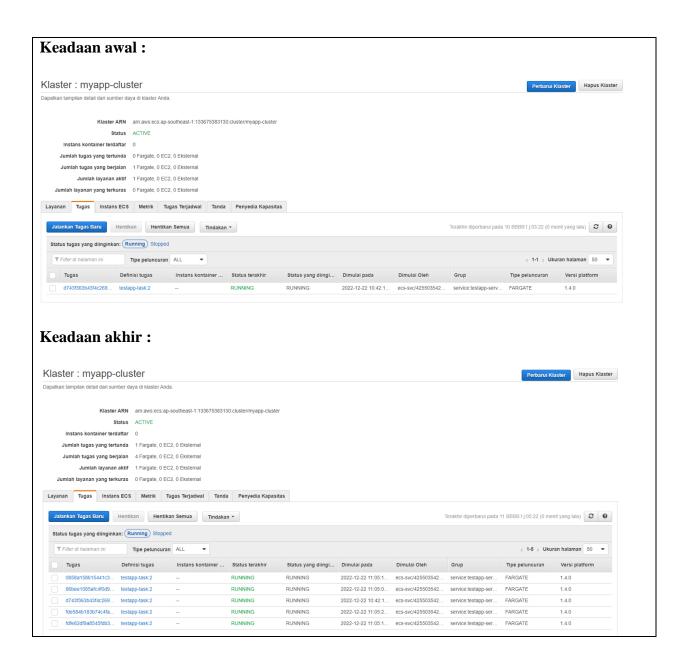
Proses Pengetesan

Testing dilakukan dengan menggunakan tools Apache Benchmark melalui command sebagai berikut:

```
prima@Prima:/mnt/d/Prima/ITB/IF sem 5/PAT/Tugas tambahan/terraform$ ab -n 10000 -c 50 http://m
yapp-load-balancer-1597764339.ap-southeast-1.elb.amazonaws.com/
This is ApacheBench, Version 2.3 <$Revision: 1843412 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/
Benchmarking myapp-load-balancer-1597764339.ap-southeast-1.elb.amazonaws.com (be patient)
apr_pollset_poll: The timeout specified has expired (70007)
Total of 922 requests completed
```

Command di atas akan mengirimkan *request* ke Load Balancer sebanyak 10.000 *request* dengan tiap 50 *request* dikirim secara konkuren.

Hasil yang diperoleh adalah sebagai berikut :



Dari hasil tersebut diperoleh bahwa pada awalnya hanya terdapat 1 *task* karena belum ada *request* sama sekali. Lalu setelah *request* dilakukan, *task* akan ditambah hingga ke kapasitas maksimum yaitu sebanyak 5 *task* karena tiap *task* hanya dapat menangani maksimal sebanyak 10 *request*.