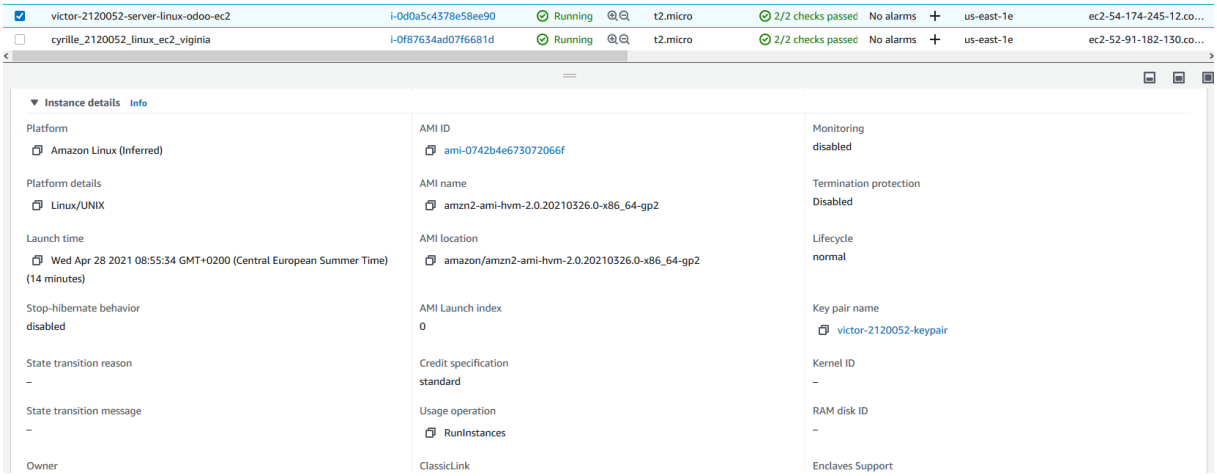
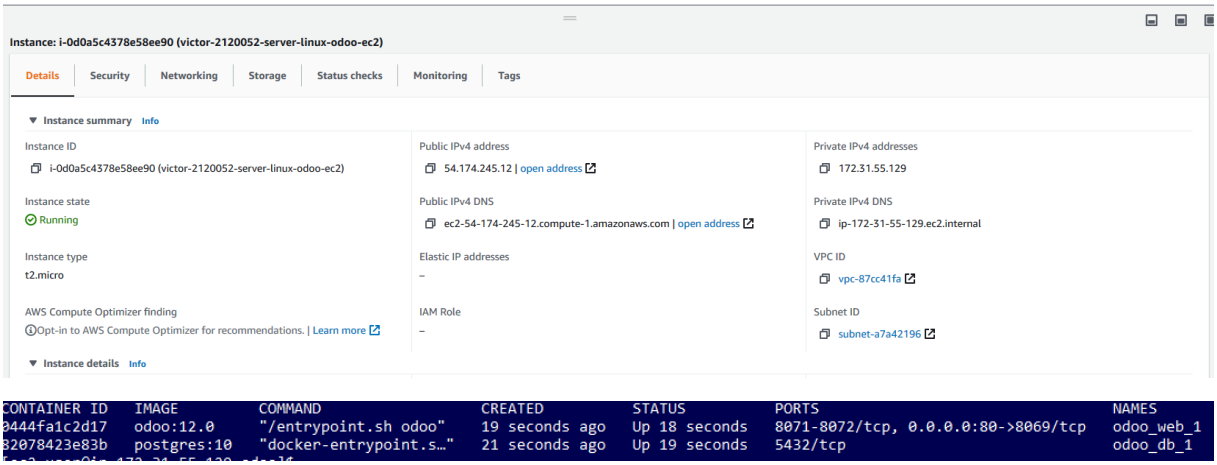
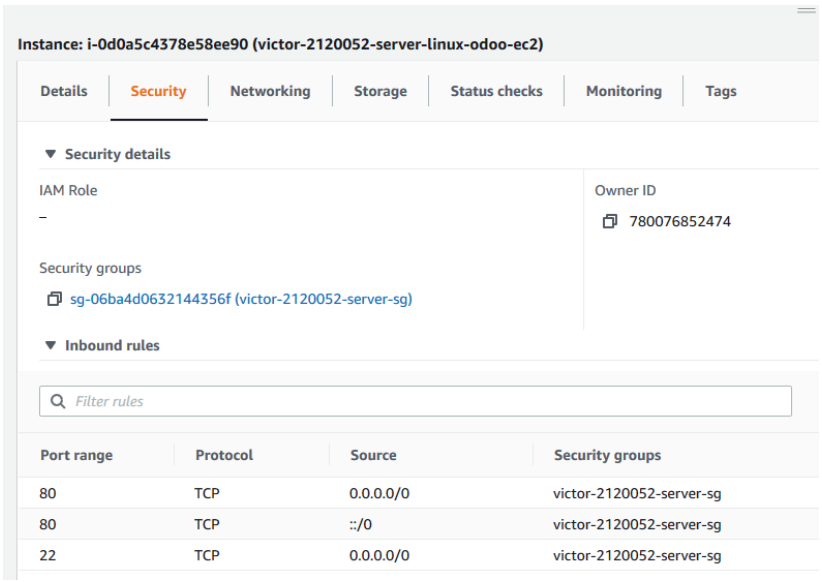
VICTOR

1-ODOO EC2





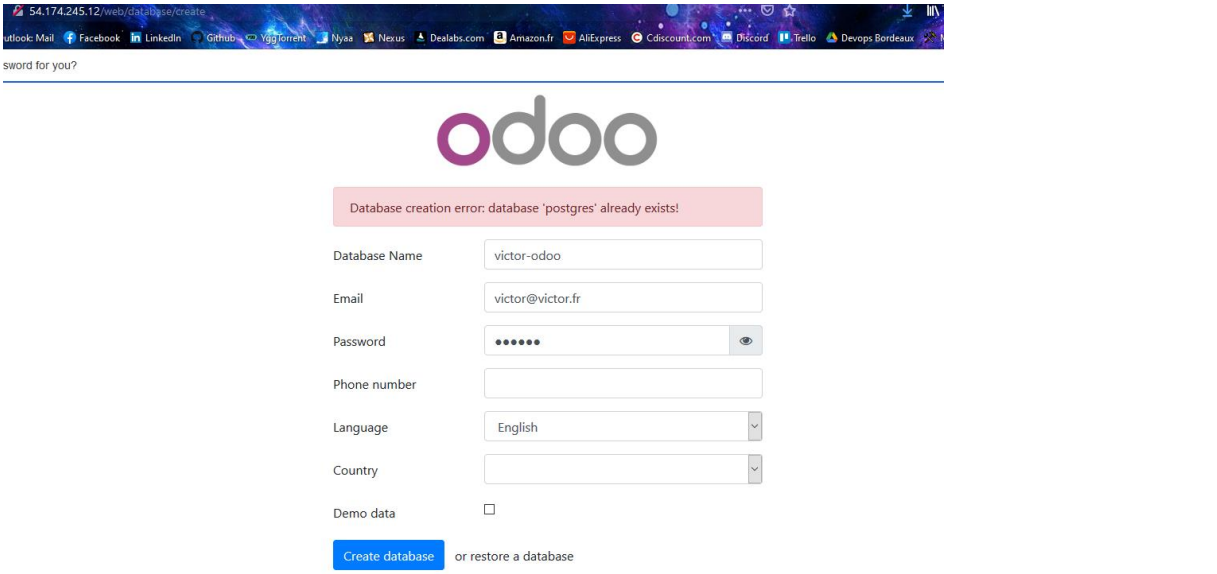

```
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                                   NAMES
0444fa1c2d17   odoo:12.0      "/entrypoint.sh odoo"    19 seconds ago   Up 18 seconds   8071-8072/tcp, 0.0.0.0:80->8069/tcp     odoo_web_1
32078423e83b   postgres:10    "docker-entrypoint.s…"   21 seconds ago   Up 19 seconds   5432/tcp                                odoo_db_1
```

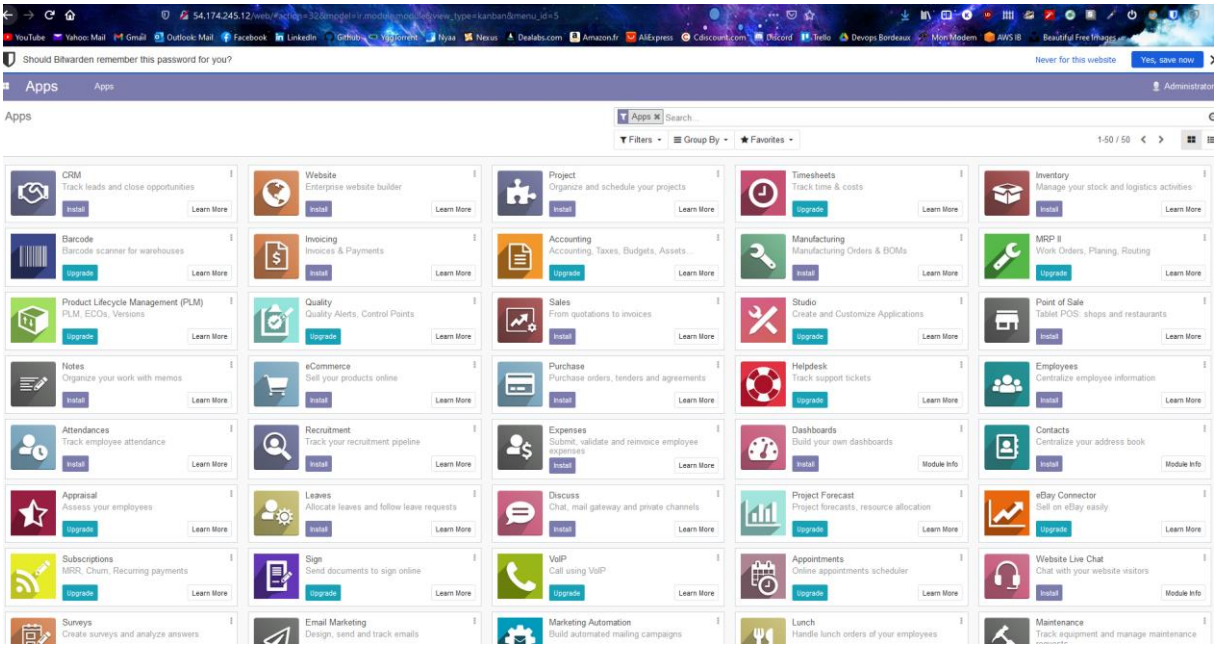


Docker-compose :

```
  GNU nano 2.9.8                                                  docker-compose.yml

version: '2'
services:
  web:
    image: odoo:12.0
    depends_on:
      - db
    ports:
      - "80:8069"
    volumes:
      - odoo-web-data:/var/lib/odoo
  db:
    image: postgres:10
    environment:
      - POSTGRES_DB=postgres
      - POSTGRES_PASSWORD=odoo
      - POSTGRES_USER=odoo
      - PGDATA=/var/lib/postgresql/data/pgdata
    volumes:
      - odoo-db-data:/var/lib/postgresql/data/pgdata
volumes:
  odoo-web-data:
  odoo-db-data:
```





http://54.174.245.12

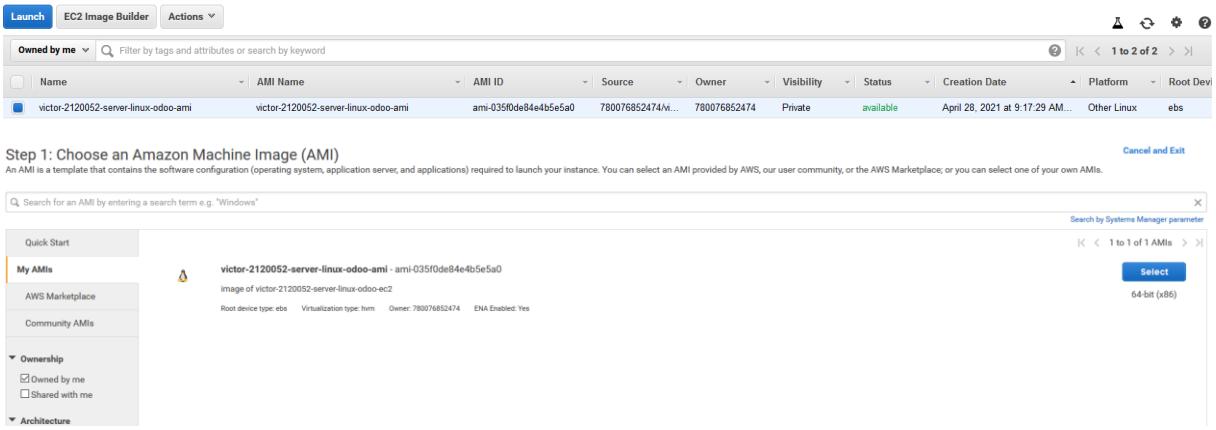
2-ODOO AMI

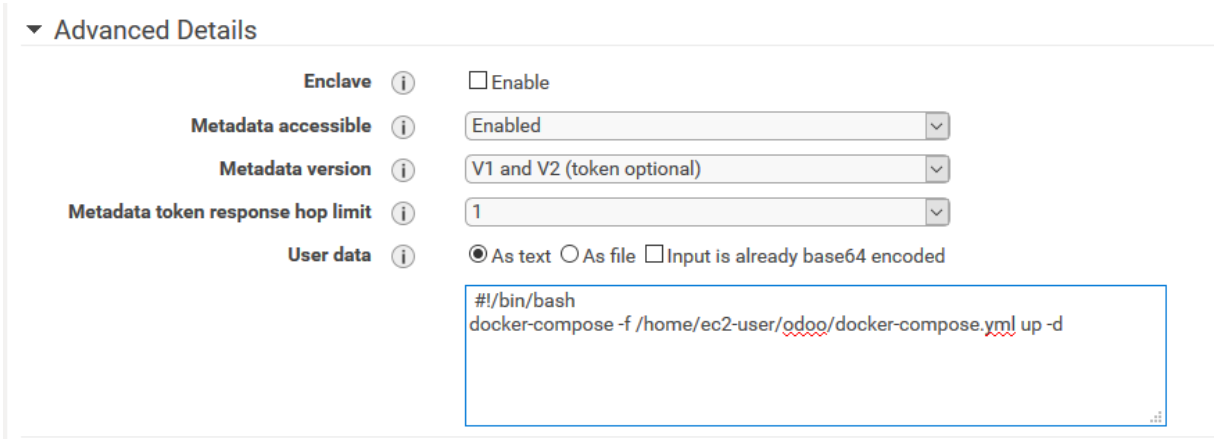
Pour démarrer docker au démarrage :

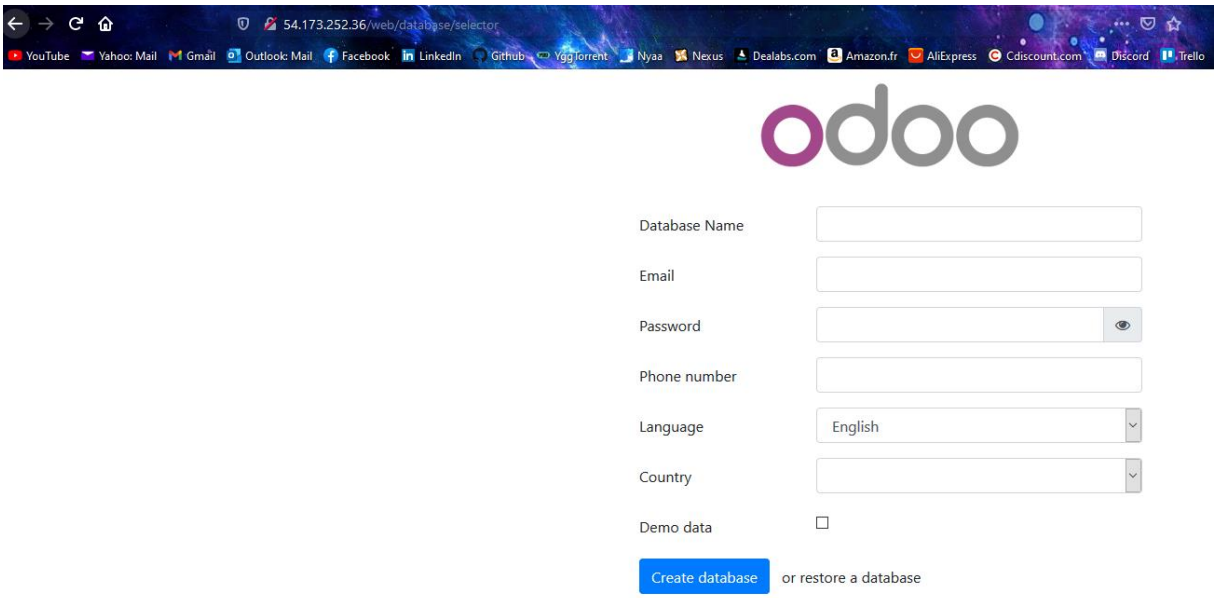
sudo systemctl enable docker

Et on enleve les volumes :

```
[ec2-user@ip-172-31-55-129 odoo]$ docker-compose down -v
Stopping odoo_web_1 ... done
Stopping odoo_db_1  ... done
Removing odoo_web_1 ... done
Removing odoo_db_1  ... done
Removing network odoo_default
Removing volume odoo_odoo-web-data
Removing volume odoo_odoo-db-data
[ec2-user@ip-172-31-55-129 odoo]$
```





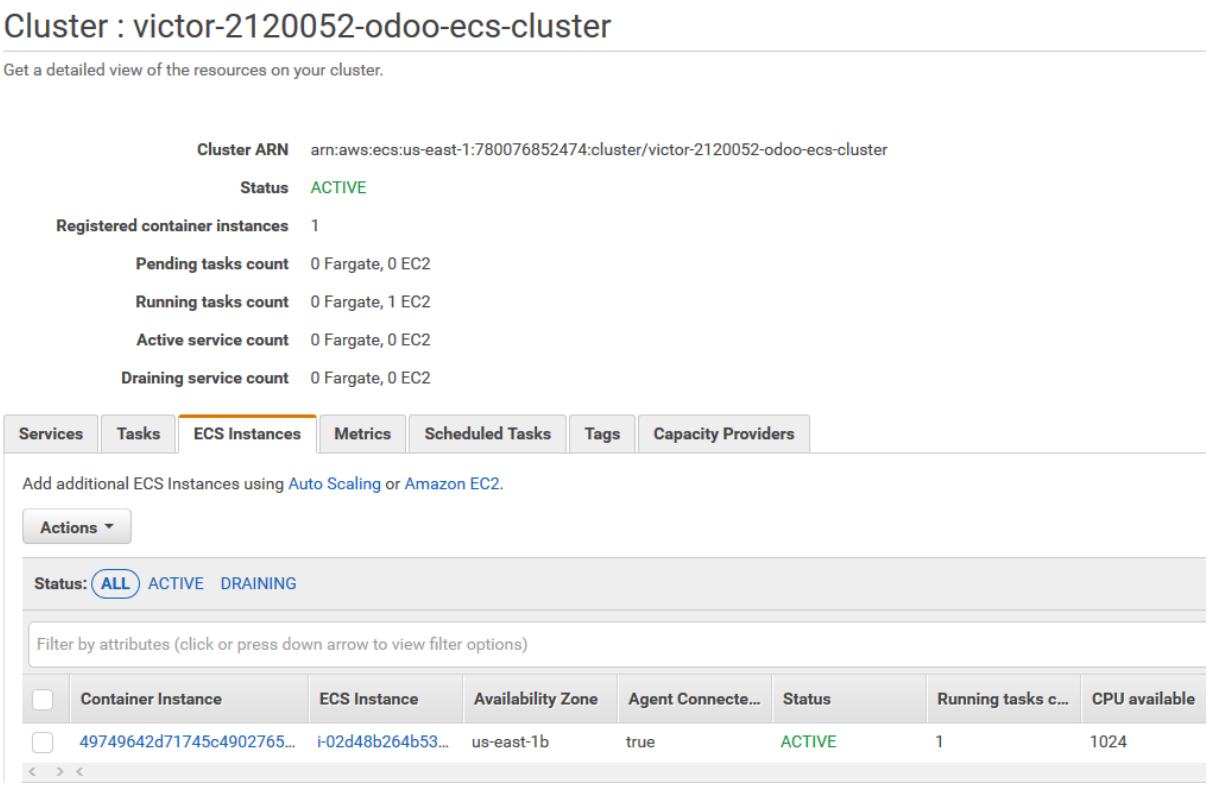


http://54.173.252.36/

3-ODOO ECS

Pour la task definition, launch type doit être EC2 pour pouvoir passer le network mode en bridge (qui est necessaire pour mapper un port du container different du port host).

Besoin de créer un cluster EC2.



Configure task and container definitions

A task definition specifies which containers are included in your task and how they interact with each other. You can also specif volumes for your containers to use. Learn more

Task Definition Name* victor-2120052-odoo-task

Requires Compatibilities* EC2

Task Role Select a role...

Optional IAM role that tasks can use to make API requests to authorized AWS services. Create an Amazon Elastic Container Service Task Role in the IAM Console

Network Mode Bridge

If you choose <default>, ECS will start your container using Docker's default networking mode, which is Bridge on Linux and NAT on Windows. <default> is the only supported mode on Windows.

Container Instance : 49749642d71745c4902765a0cfff02cc

Details | Tags

Cluster	victor-2120052-odoo-ecs-cluster
EC2 Instance	i-02d48b264b535a656
Operating system	Linux
Availability Zone	us-east-1b
Public DNS	ec2-18-206-90-67.compute-1.amazonaws.com
Private DNS	ip-10-0-1-221.ec2.internal
Public IP	18.206.90.67
Private IP	10.0.1.221
Status	ACTIVE
Running tasks count	1
Agent version	1.51.0
Docker version	19.03.13-ce

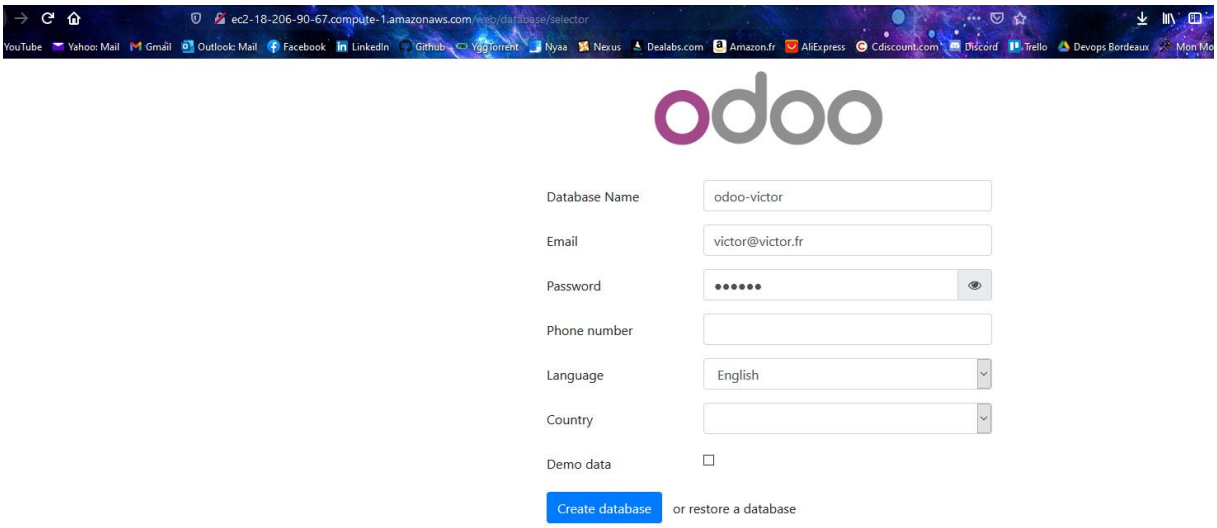
Tasks

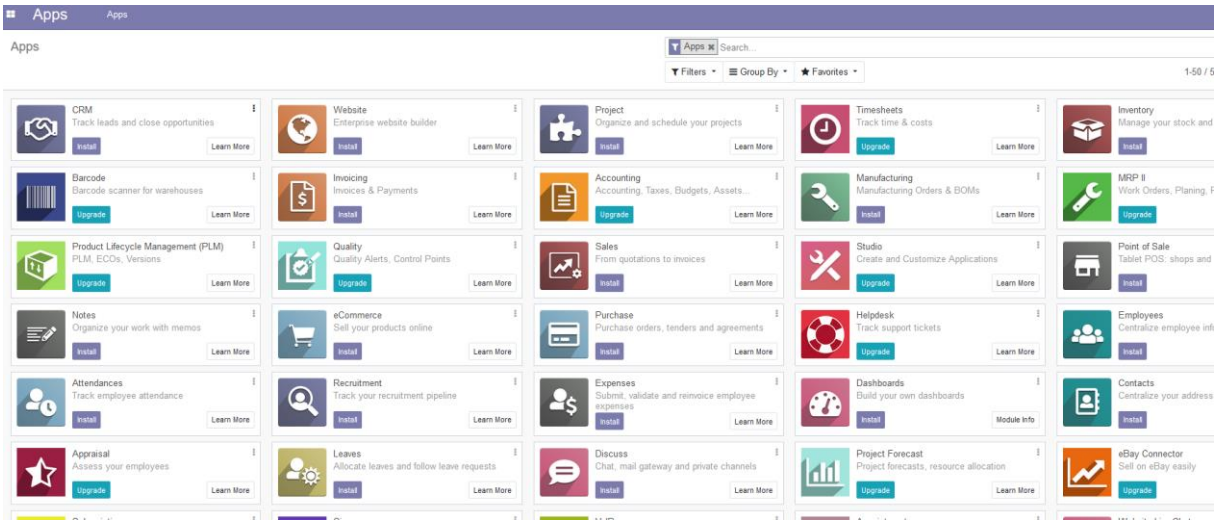
Stop | Stop All

Task status: Running Stopped

Filter in this page

Task	Task definition	Last status
1585fbdaf8be4406951d1c1e5ac6c17d	victor-2120052-odoo-task:1	RUNNING





4-ODOO EBS

```
version: '2'
services:
  web:
    image: odoo:12.0
    depends_on:
      - db
    ports:
      - "80:8069"
  db:
    image: postgres:10
    environment:
      - POSTGRES_DB=postgres
      - POSTGRES_PASSWORD=odoo
      - POSTGRES_USER=odoo
      - PGDATA=/var/lib/postgresql/data/pgdata
    volumes:
      - /data/db:/var/lib/postgresql/data/pgdata
```

Attach Volume

Volume (i) vol-04b43db0686a7b4d3 (victor-2120052-server-linux-odoo-ebs) in us-east-1e

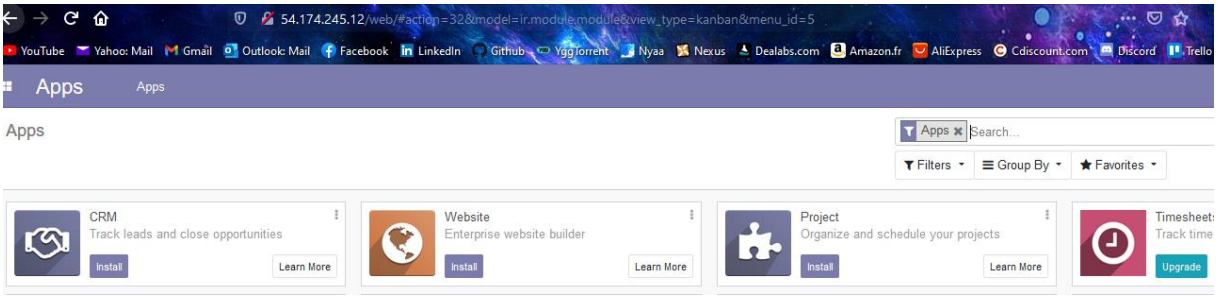
Instance (i) i-0d0a5c4378e58ee90 in us-east-1e

Device (i) /dev/sdf

Linux Devices: /dev/sdf through /dev/sdp

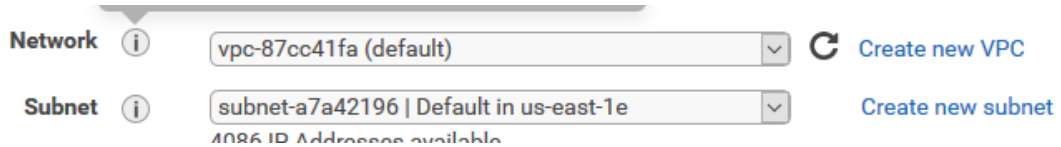
```
[ec2-user@ip-172-31-55-129 ~]$ lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0   8G  0 disk
└─xvda1 202:1    0   8G  0 part /
xvdf    202:80   0   1G  0 disk
```

```
[ec2-user@ip-172-31-55-129 odoo]$ lsblk
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0   8G  0 disk
└─xvda1 202:1    0   8G  0 part /
xvdf    202:80   0   1G  0 disk /data
```



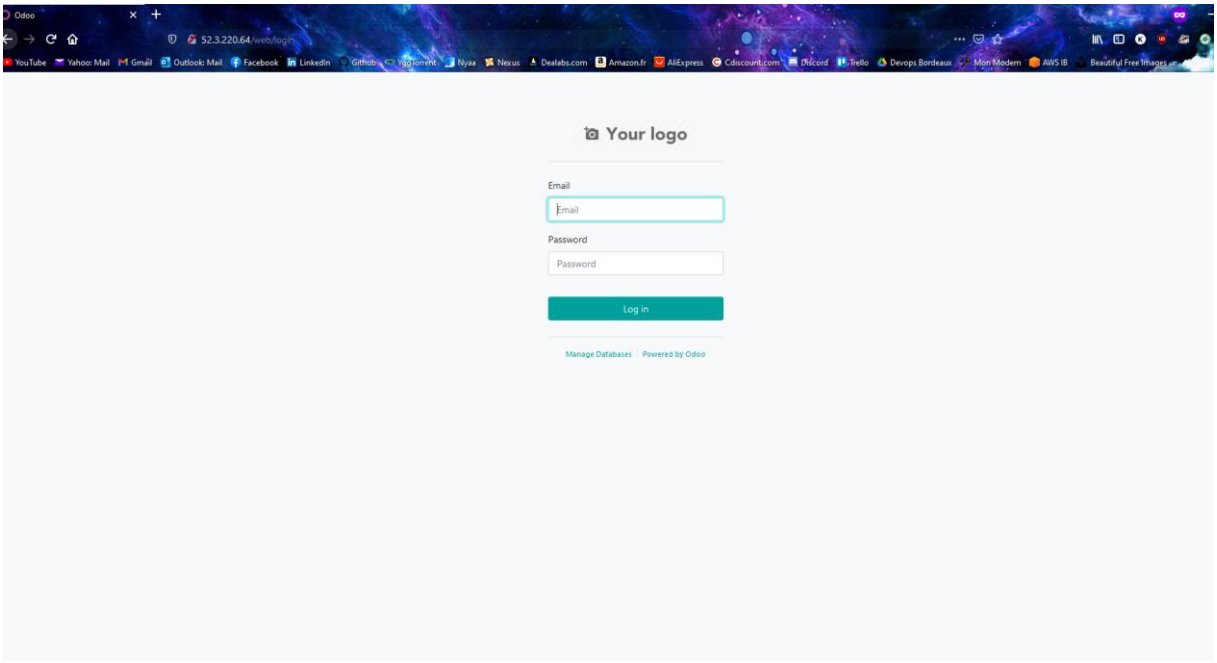

```
[root@ip-172-31-55-129 odoo]# ll /data/db/
total 52
drwx------ 6 libstoragemgmt ec2-user    54 Apr 28 08:29 base
drwx------ 2 libstoragemgmt ec2-user  4096 Apr 28 08:31 global
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_commit_ts
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_dynshmem
-rw------- 1 libstoragemgmt ec2-user  4535 Apr 28 08:25 pg_hba.conf
-rw------- 1 libstoragemgmt ec2-user  1636 Apr 28 08:25 pg_ident.conf
drwx------ 4 libstoragemgmt ec2-user    68 Apr 28 08:31 pg_logical
drwx------ 4 libstoragemgmt ec2-user    36 Apr 28 08:25 pg_multixact
drwx------ 2 libstoragemgmt ec2-user    18 Apr 28 08:28 pg_notify
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_replslot
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_serial
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_snapshots
drwx------ 2 libstoragemgmt ec2-user    84 Apr 28 08:31 pg_stat
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:31 pg_stat_tmp
drwx------ 2 libstoragemgmt ec2-user    18 Apr 28 08:25 pg_subtrans
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_tblspc
drwx------ 2 libstoragemgmt ec2-user     6 Apr 28 08:25 pg_twophase
-rw------- 1 libstoragemgmt ec2-user     3 Apr 28 08:25 PG_VERSION
drwx------ 3 libstoragemgmt ec2-user    92 Apr 28 08:29 pg_wal
drwx------ 2 libstoragemgmt ec2-user    18 Apr 28 08:25 pg_xact
-rw------- 1 libstoragemgmt ec2-user    88 Apr 28 08:25 postgresql.auto.conf
-rw------- 1 libstoragemgmt ec2-user 23051 Apr 28 08:25 postgresql.conf
-rw------- 1 libstoragemgmt ec2-user    36 Apr 28 08:28 postmaster.opts
```

Volume dans meme AZ que l'instance ou l'on va restaurer les données :




```
NAME    MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
xvda    202:0    0   8G  0 disk
└─xvda1 202:1    0   8G  0 part /
xvdf    202:80   0   1G  0 disk /data
[ec2-user@ip-172-31-58-168 odoo]$
```

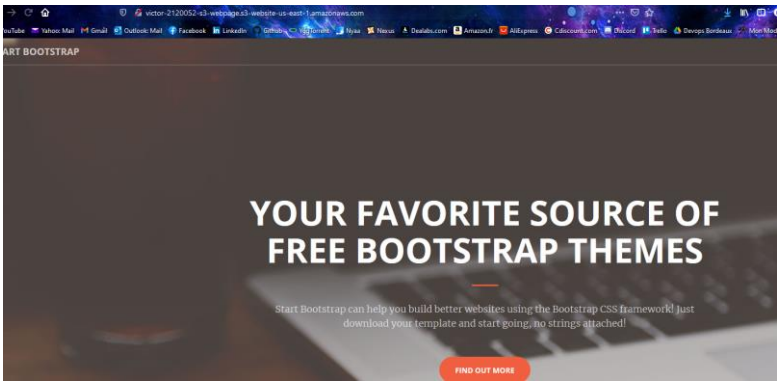
Odoo, reprends là ou on l'a laissé avant, sur une autre instance. Les données sont restaurées

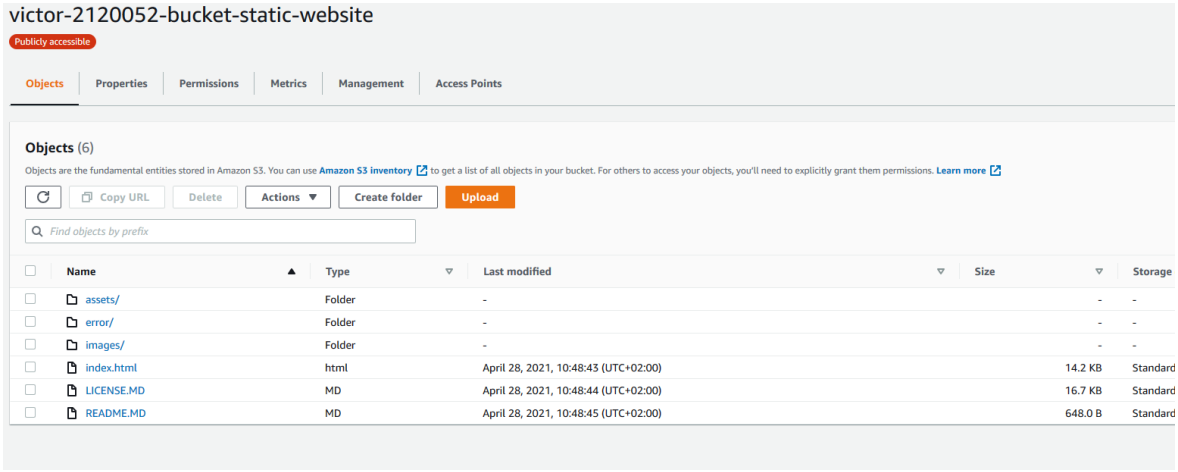


http://52.3.220.64/

5 – S3 static website

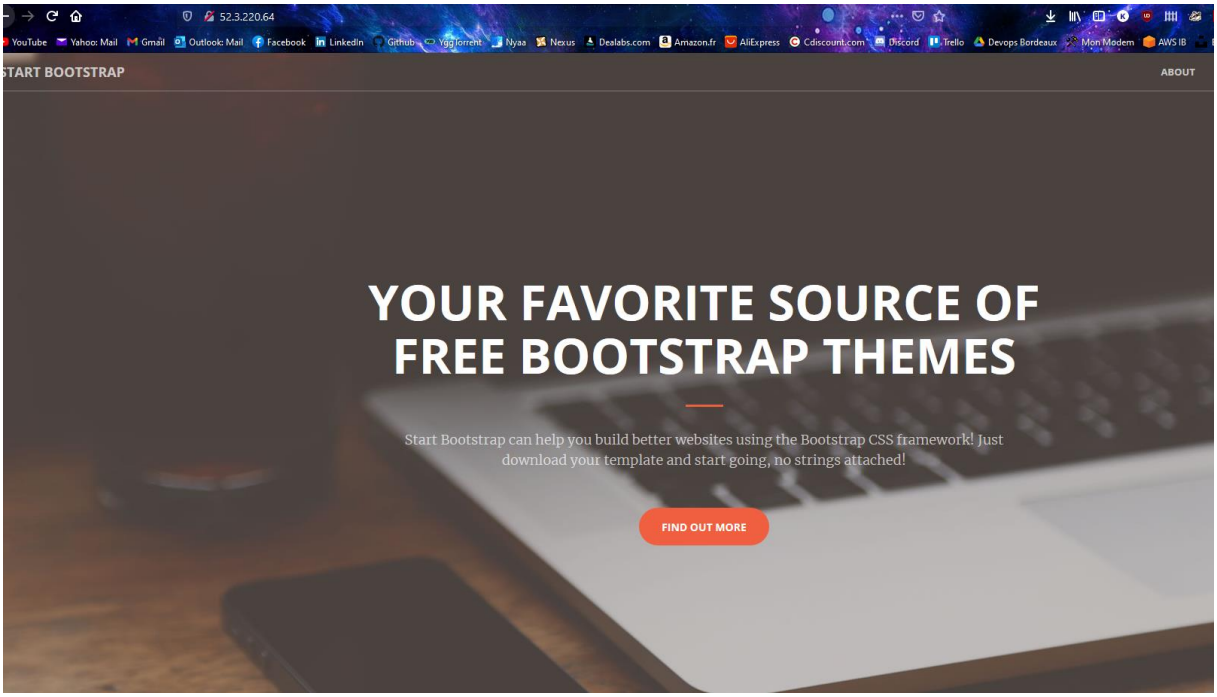
http://victor-2120052-s3-webpage.s3-website-us-east-1.amazonaws.com/





6 – EC2 static website

docker run -d --name httpd -p 80:80 -v /home/ec2-user/static-webpage-example/src:/usr/local/apache2/htdocs httpd



7 – VPC static website

Your VPCs (3) Info

Filter VPCs

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR (Network border group)
victor-2120052-vpc	vpc-01b7a201a1ef5dce7	Available	11.0.0.0/16	–

Subnets (1/10) Info

Filter subnets

Name	Subnet ID	State	VPC	IPv4 CIDR
–	subnet-096d895e16b4754c9	Available	vpc-0467aade53bcccd58	10.0.0.0/24
–	subnet-a7a42196	Available	vpc-87cc41fa	172.31.48.0/20
–	subnet-8e2351d1	Available	vpc-87cc41fa	172.31.32.0/20
–	subnet-983a1196	Available	vpc-87cc41fa	172.31.64.0/20
–	subnet-028b1224052a2d448	Available	vpc-0467aade53bcccd58	10.0.1.0/24
victor-2120052-vp...	subnet-06a0d7987c577dcef	Available	vpc-01b7a201a1ef5dce7 | vict...	11.0.1.0/24
victor-2120052-vp...	subnet-0977a6c12305807a7	Available	vpc-01b7a201a1ef5dce7 | vict...	11.0.0.0/24
–	subnet-75c9bf13	Available	vpc-87cc41fa	172.31.0.0/20

On s'assure que les subnet ne sont pas dans la meme zone (pas d'intérêt sinon)

Pour le 2e subnet publique, on ajoute une route vers l'internet gateway

On pense a activer l'auto ipv4 assign

Route Tables > Edit routes

Edit routes

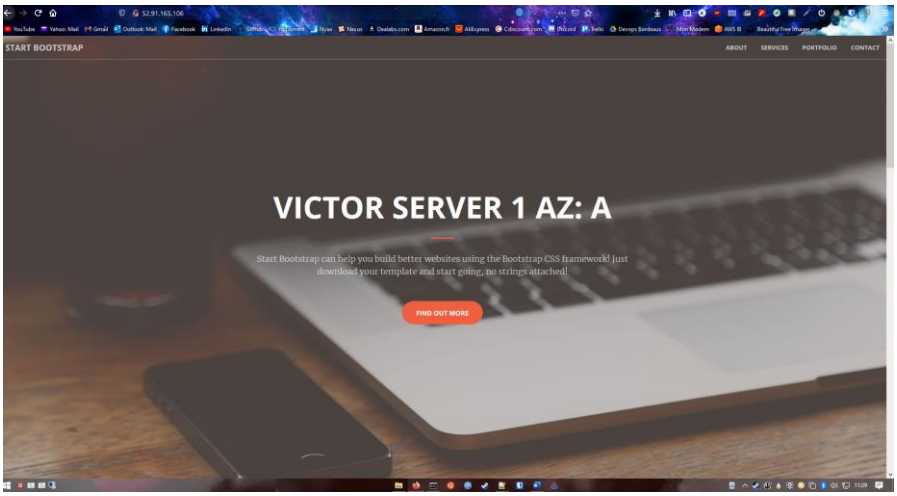
Destination	Target	Status	Propagated
11.0.0.0/16	local	active	No
0.0.0.0/0	igw-0daddf36812213b35		No

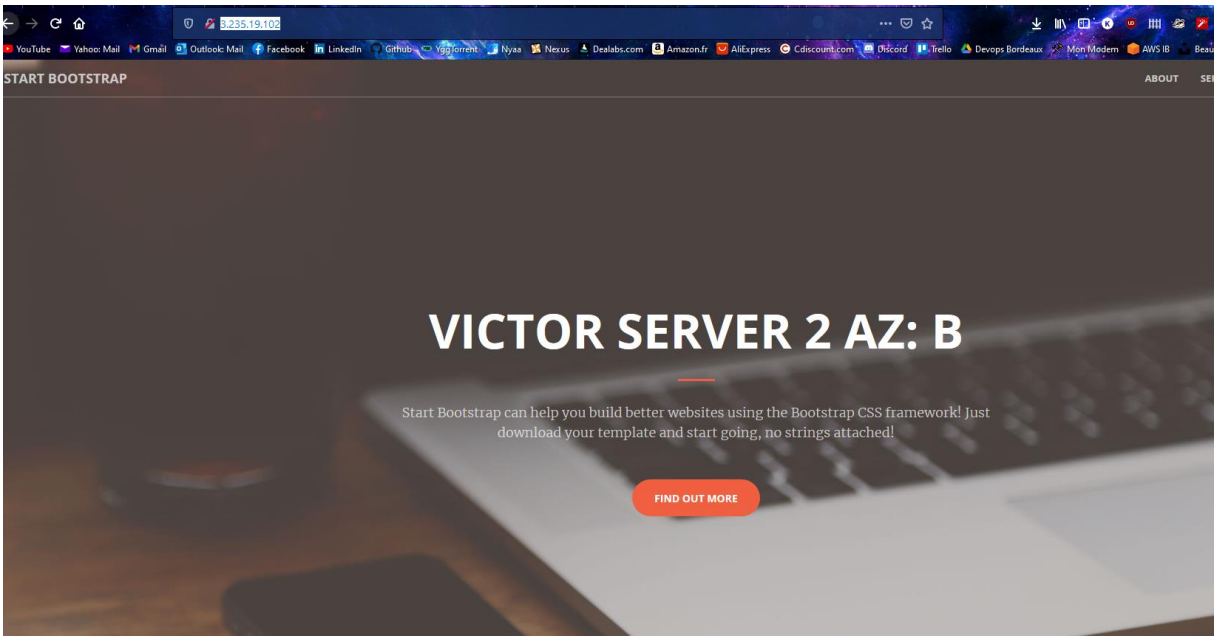
Add route

* Required

Cancel Save routes

On selectionne le vpc, puis un subnet different pour chacune des deux instances.



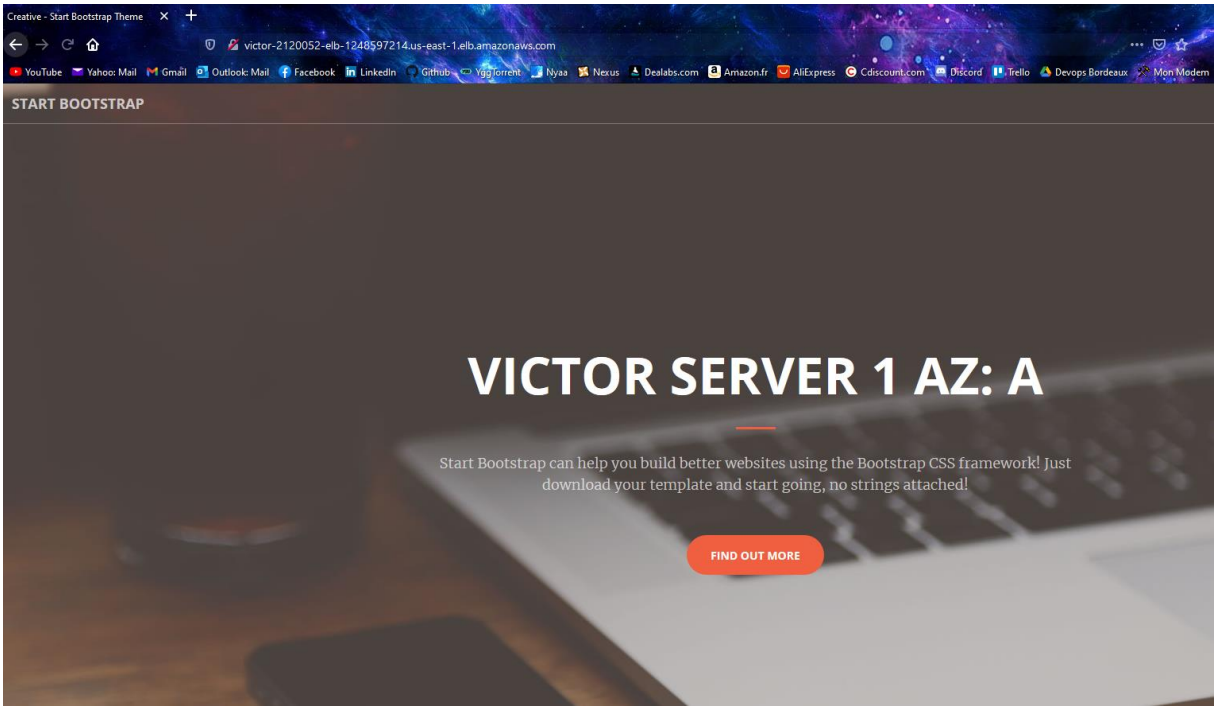


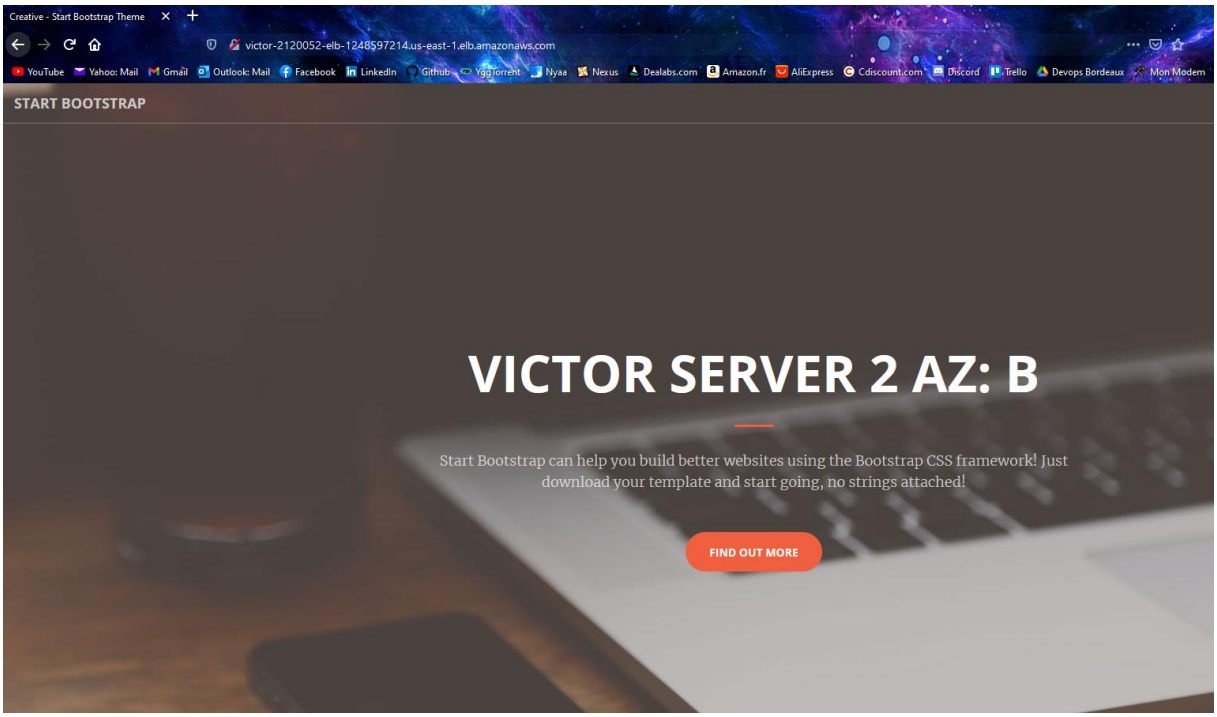
Load balancer

Availability Zones

Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one su
balancer.

VPC	vpc-01b7a201a1ef5dce7 (11.0.0.0/16) | victor-2120052-vpc	
Availability Zones	☑ **us-east-1a**	subnet-0977a6c12305807a7 (victor-2120052-vpc-su
	IPv4 address	Assigned by AWS
	☑ **us-east-1b**	subnet-06a0d7987c577dcef (victor-2120052-vpc-sub
	IPv4 address	Assigned by AWS





http://victor-2120052-elb-1248597214.us-east-1.elb.amazonaws.com/

meme adresse pour les 2, le load balancing fonctionne