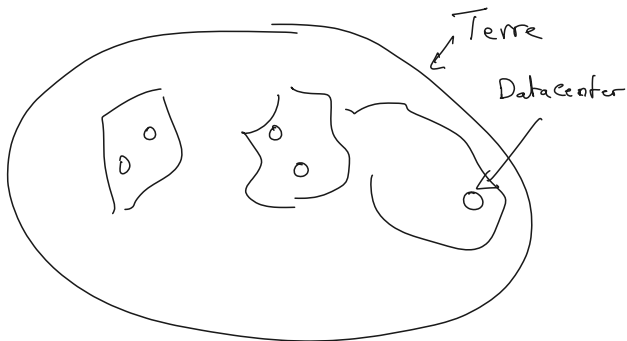
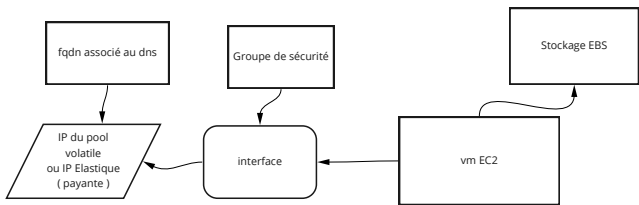


AWS



Limites de l'offre gratuite :

https://docs.aws.amazon.com/fr_fr/awsaccountbilling/latest/aboutv2/free-tier-limits.html



Script de création de VM en impératif:

```
#!/bin/bash
```

```
AMIID=$(aws ec2 describe-images --filters "Name=description, Values=Amazon Linux AMI
2018.03.0.20180412 x86_64 HVM EBS" --query "Images[0].ImageId" --output text )

VPCID=$(aws ec2 describe-vpcs --filter "Name=isDefault, Values=true" --query "Vpcs[0].VpcId" --
output text)
SUBNETID=$(aws ec2 describe-subnets --filters "Name=vpc-id, Values=$VPCID" --query
"Subnets[0].SubnetId" --output text)
SGID=$(aws ec2 create-security-group --group-name mysecuritygroup --description "My security group"
--vpc-id $VPCID --output text)
aws ec2 authorize-security-group-ingress --group-id $SGID --protocol tcp --port 22 --cidr 0.0.0.0/0
INSTANCEID=$(aws ec2 run-instances --image-id $AMIID --key-name vm1 --instance-type t2.micro --
security-group-ids $SGID --subnet-id $SUBNETID --query "Instances[0].InstanceId" --output text)

echo "waiting for $INSTANCEID ..."
aws ec2 wait instance-running --instance-ids $INSTANCEID
PUBLICNAME=$(aws ec2 describe-instances --instance-ids $INSTANCEID --query
"Reservations[0].Instances[0].PublicDnsName" --output text)
echo "$INSTANCEID is accepting SSH connections under $PUBLICNAME"
echo "ssh -i mykey.pem ec2-user@$PUBLICNAME"
read -p "Press [Enter] key to terminate $INSTANCEID ..."
aws ec2 terminate-instances --instance-ids $INSTANCEID
echo "terminating $INSTANCEID ..."
aws ec2 wait instance-terminated --instance-ids $INSTANCEID
aws ec2 delete-security-group --group-id $SGID
```

fichier d'infra
CloudFormation
(+ paramètres)
+
URL vers le script

Script au démarrage
(installation de python-ssh
pour Ansible)

```
Script de lancement de l'infrastructure :
$ SharedSecret=$(openssl rand -base64 30)
$ Password=$(openssl rand -base64 30)
$ aws cloudformation create-stack --stack-name vpn --
template-url \
https://s3.amazonaws.com/awsinaction/chapter5/vpn-
cloudformation.json \
--parameters ParameterKey=KeyName,ParameterValue=mykey \
ParameterKey=VPC,ParameterValue=$VpcId \
ParameterKey=Subnet,ParameterValue=$SubnetId \
ParameterKey=IPSecSharedSecret,ParameterValue=$SharedSec
ret \
ParameterKey=VPNUser,ParameterValue=vpn \
ParameterKey=VPNPassword,ParameterValue=$Password
$ aws cloudformation describe-stacks --stack-name vpn \
--query Stacks[0].Outputs > vphost
$ ansible-playbook -i vphost vpn_playbook.yml
```

Elastic Beanstalk

(via une url) :

