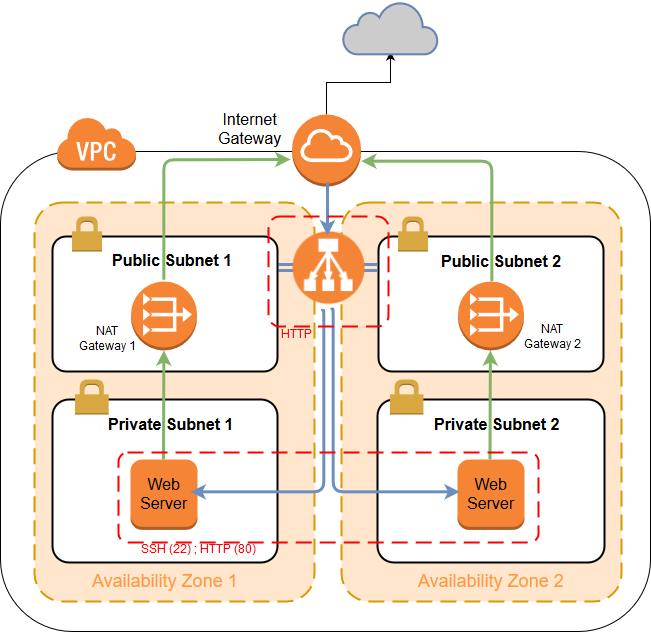
**AWS PROJECT:** **Achieving high availability with multi-AZ architecture and Elastic Load Balancer**

**Objective:** Install a web server and make it highly available by duplicating its image in a second availability zone inside the VPC. Use an elastic load balancer to divide the flow between the two availability zones.

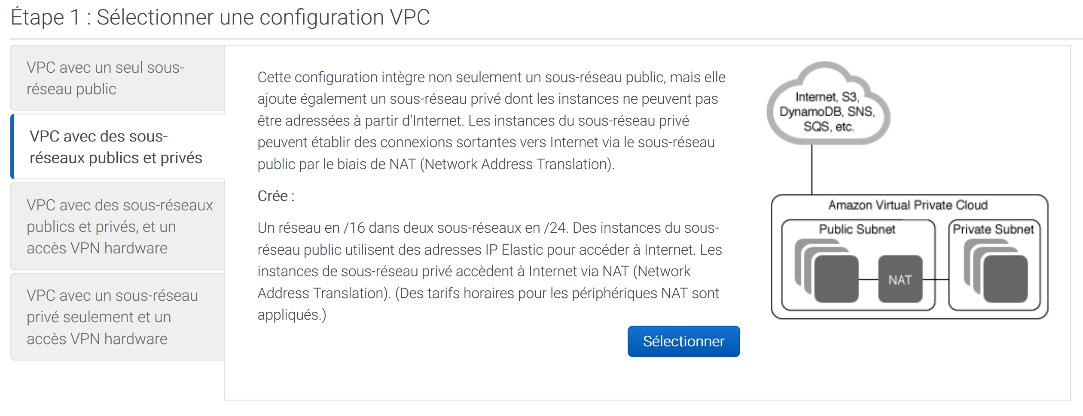


1. Create a VPC with two public and two private subnets

Create two elastic IP addresses that will be used to generate two NAT gateway in your future VPC

Services -> VPC -> Elastic IPs -> Allocate new address

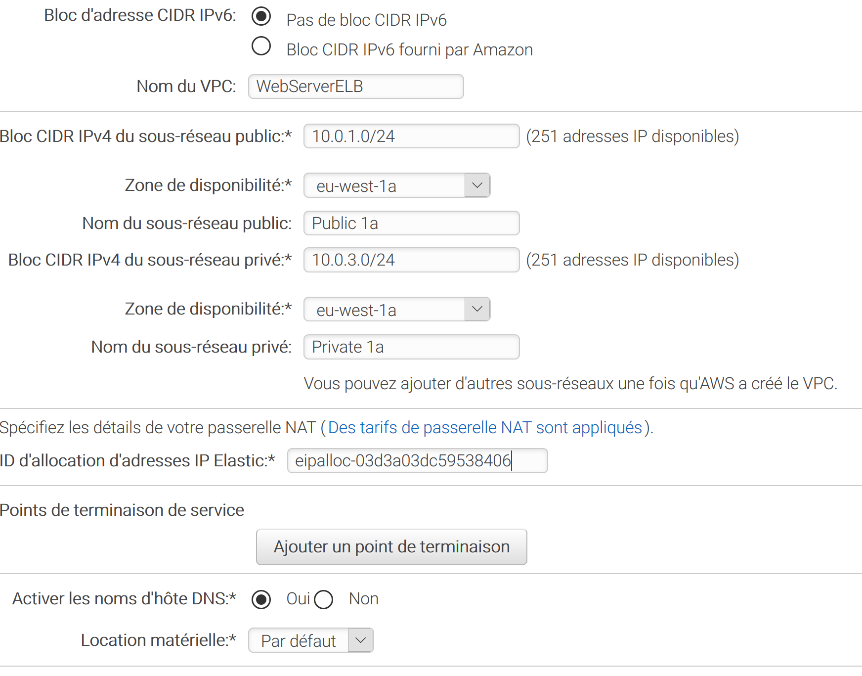
First, create a VPC with one public and one private subnet (using VPC wizard)



Name your VPC and restrict the number of available IP addresses for each subnet

Choose the same availability zone for the public and the private subnet

Create a NAT gateway with one of the elastic IP address you created to allow internet outbound connection of the private subnet



Public Subnet

Private Subnet

NAT Gateway

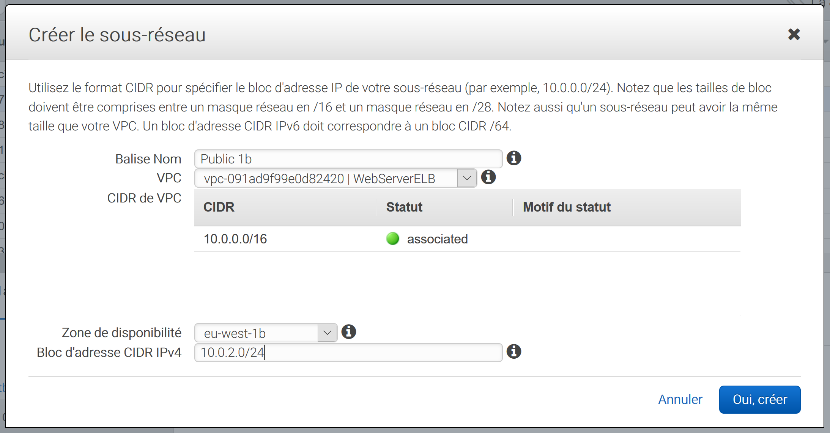
Create two other subnets, public and private, in another availability zone

VPC -> Subnets -> Create subnet

Choose the VPC you created at the previous step

Choose a different availability zone from the one you chose at the creation of the VPC

Repeat the operation once to get two new subnets



New AZ!

In order to make the new subnets public or private, you have to specify their route tables:

* the public subnet has to be associated to the internet gateway
* the private subnet has to be associated with a second NAT gateway

Create a NAT gateway

Services -> VPC -> NAT Gateway

Choose the subnet id that corresponds to the public subnet of the second AZ (the NAT gateway stands in the public subnet)

Choose the second elastic IP address you have created



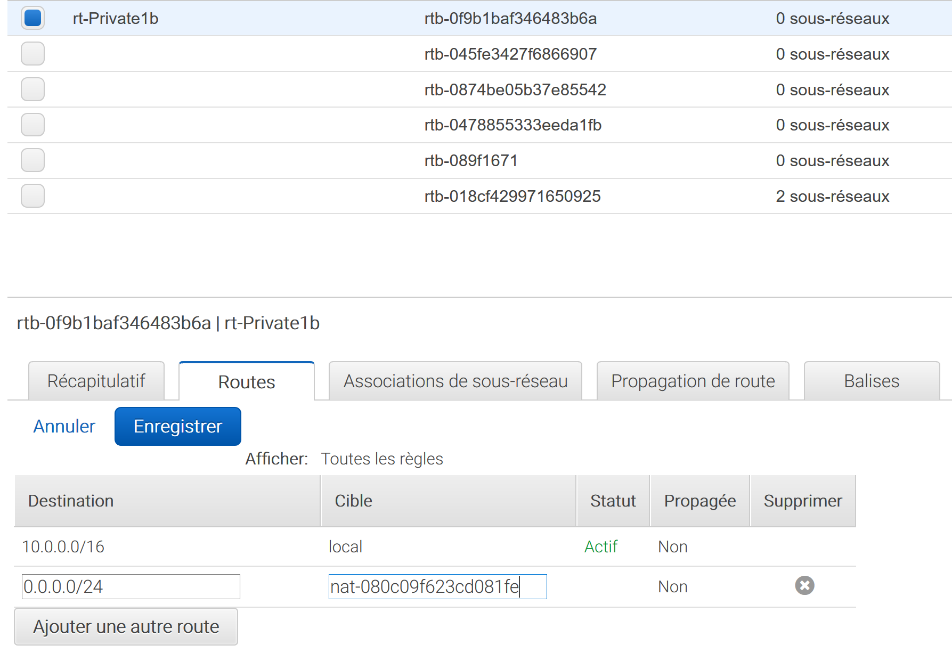
Create a new route table that targets the NAT gateway you created

VPC -> Route table -> create route table

Give it a name and choose your VPC -> Create

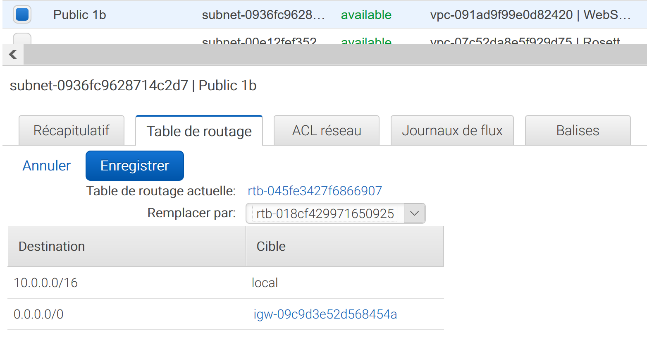
On the route tab, add a new route that target the second NAT Gateway

In subnet association, associate the private subnet of the second AZ.



In Subnets, click on the public subnet of the second AZ and go to the route table tab.

If it points to the NAT, modify the route table and put the one that target the internet gateway (this route table has been automatically created at the creation of the VPC)

You now have a VPC with two public and two private subnets in two distinct availability zones

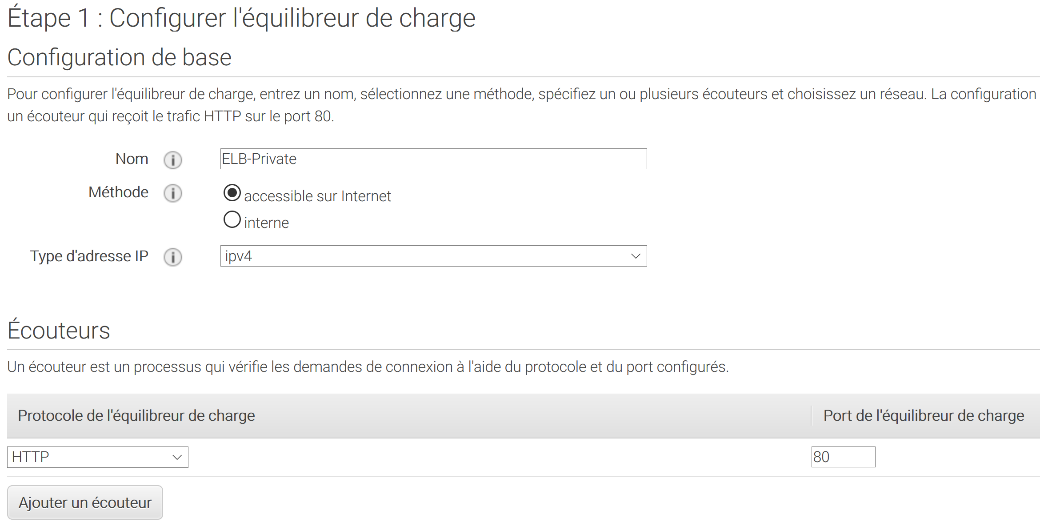
1. Create an Elastic load Balancer to distribute the inbound flow between the two AZ

In services -> EC2 -> Load Balancing -> Load Balancer -> Create Load Balancer

Create a HTTP/HTTPS load balancer

Configure the load balancer:

Give it a name (e.g. ELB1) ; internet-facing ; listener HTTP (port 80)

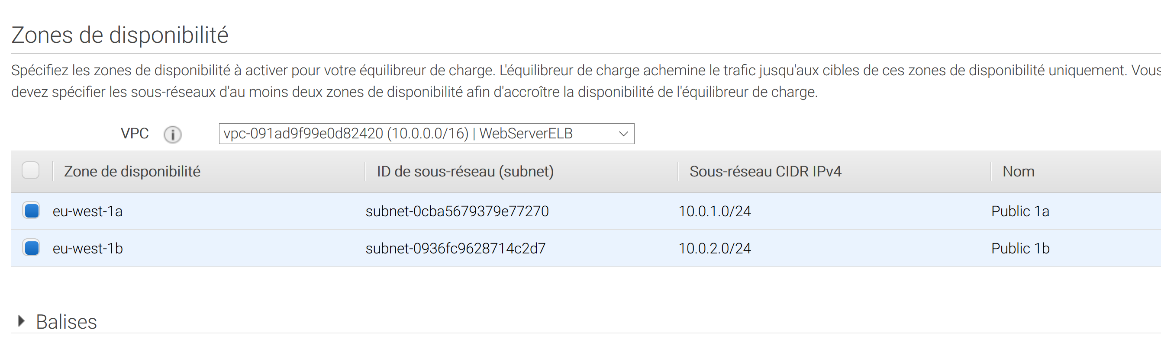


Availability zones:

Choose your VPC

Then select the first AZ and pick the public subnet

Select the second AZ and pick the public subnet



Configure security settings: ignore security warning (because we are using HTTP and not HTTPS protocol)

Configure security groups: create a new security group:

Protocol: HTTP (port80), source anywhere

Configure routing:



Register target: For the moment, you do not have any instance so no need to register the target, we will do it later.

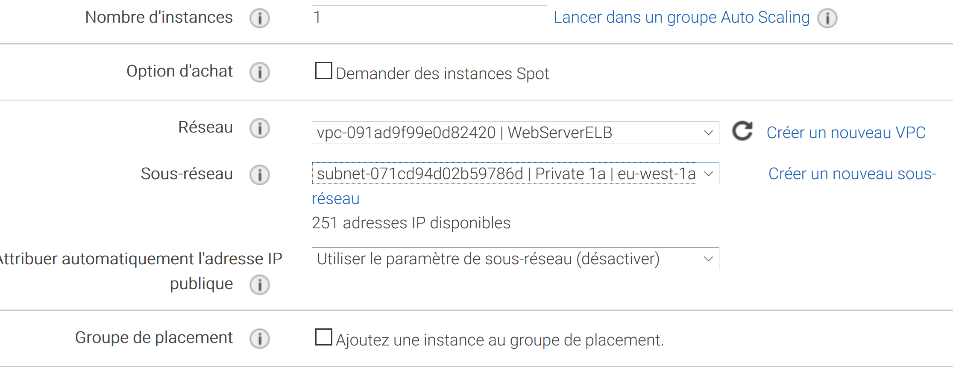
1. Create instances in private subnets

EC2 -> Launch an instance

Choose a Linux AMI

Configuration: choose your VPC and the private subnet of the first AZ

Disable public IP address attribution



We want to create a web server, so we are going to use user data to install it at launch and create an html page. To test the function of the load balancer we will create two different html pages in the two different AZ, one displaying “HELLO”, the other one “GOOD BYE”.

User data to install the web server:

**#!/bin/bash**

**sudo yum update -y**

**sudo yum -y install httpd**

**sudo service httpd start**

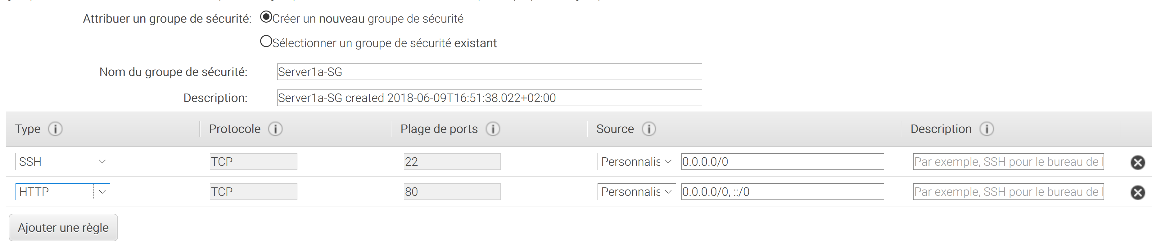
**sudo bash -c 'echo HELLO > /var/www/html/index.html'**

Next, configure security group. Create a new security group

SSH (if you want to connect to your instance)

HTTP

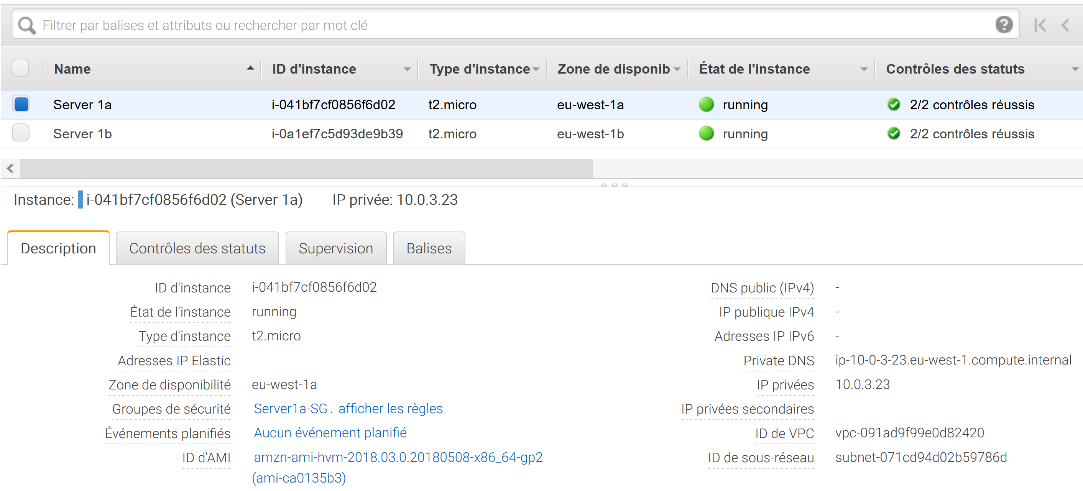
Source: Anywhere or your IP



Review and create instance.

Create a second instance in the private subnet of the second AZ. Proceed exactly the same way as previously but change the location of the instance (private subnet 2) and modify the user data to display “GOOD BYE” instead of “HELLO” on the html page.

You now have two web servers with no public IP in two distinct AZ.



2 different AZ

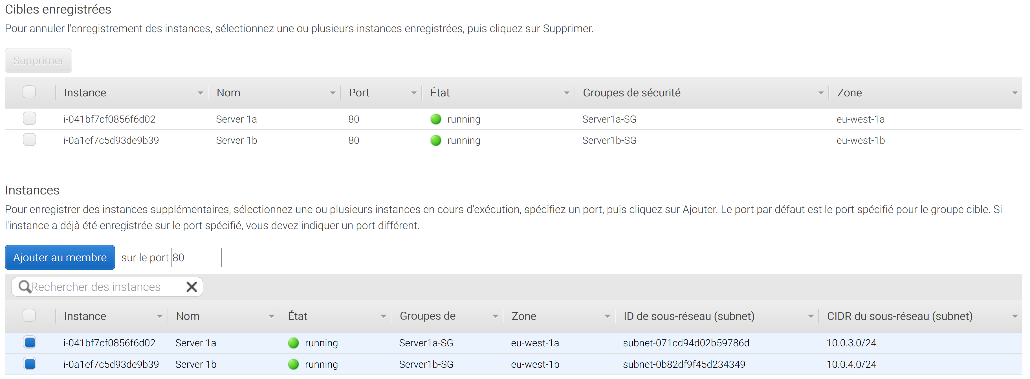
no public IP

1. Connect load balancer to instances

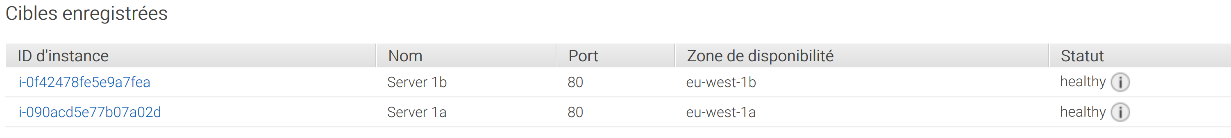
EC2 -> Load balancing -> target group

A target group has been created at the creation of the load balancer. For the moment, no instance is registered.

Register instances: choose the two private web servers, add to the group, save

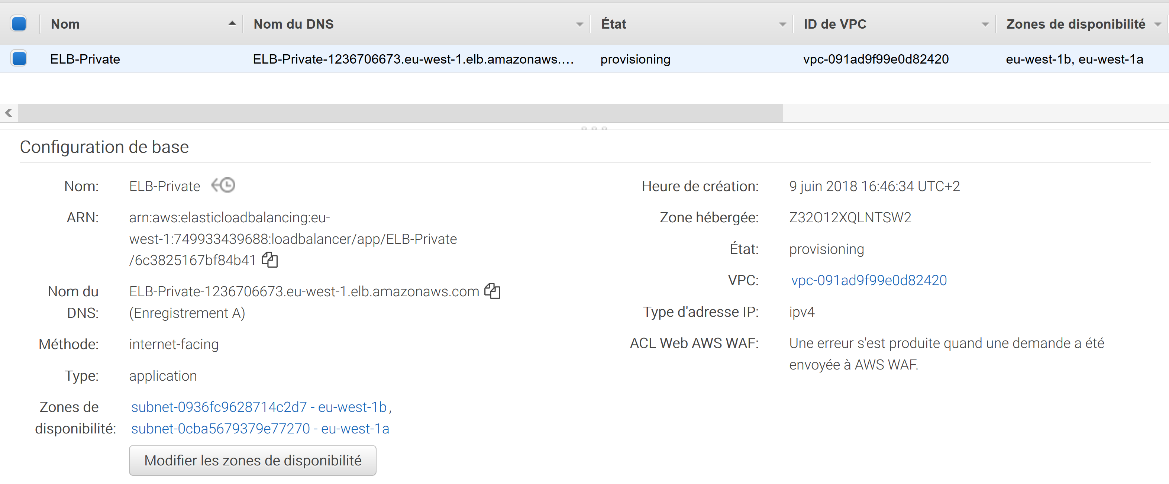


Wait few minutes and you should see your two registered instances with a “healthy” status



Come back on the elastic load balancer tab, copy the DNS name of the load balancer you have created for your two instances.

DNS name



Paste it in a new tab of your internet browser.

You should see a web page displaying “HELLO” or “GOOD BYE”. If you refresh the page several times, you should see an alternation in the page displayed, sometimes “HELLO”, sometimes “GOOD BYE”.