AWS: Use of an Elastic Load Balancer

Goal :

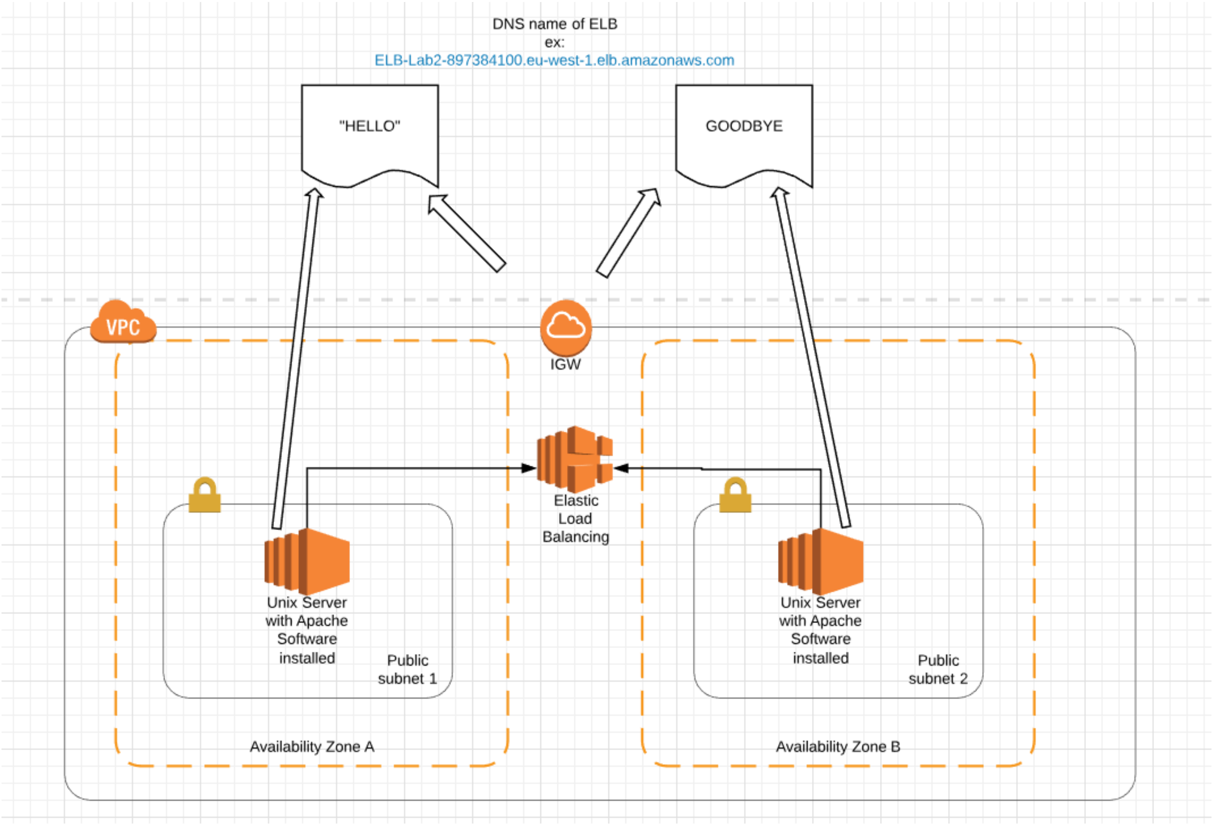
Create an architecture, with 2 instances in 2 different subnets.

Each EC2 is an Apache Web Server, with a different content, managed by an ELB.

On the Web Brower, according to the refresh, we will the content of one Server or the content of the other.

PART 1: THE 2 INSTANCES ARE IN PUBLIC SUBNET

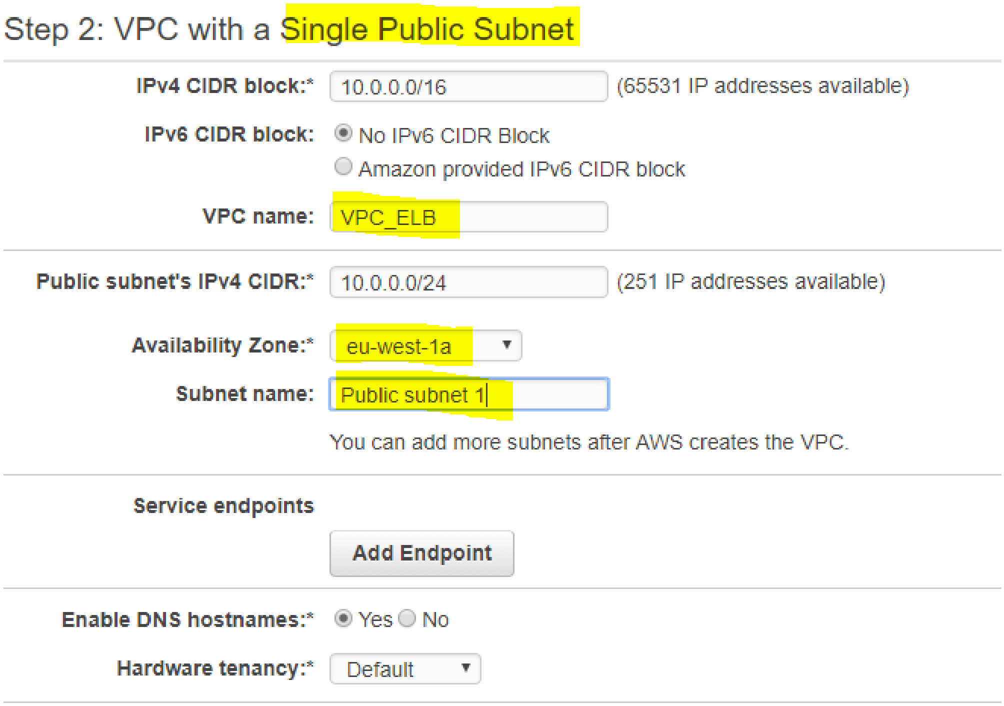
Architecture looks like as below:


DNS name of ELB
ex:
ELB-Lab2-897384100.eu-west-1.elb.amazonaws.com
"HELLO"
GOODBYE
VPC
IGW
Elastic Load Balancing
Unix Server with Apache Software installed
Public subnet 1
Unix Server with Apache Software installed
Public subnet 2
Availability Zone A
Availability Zone B

Step 1: Create the VPC with the 2 Subnets

- **VPC -> Start VPC Wizard**

  Select VPC Configuration: VPC with a Single Public Subnet



Step 2: VPC with a Single Public Subnet

IPv4 CIDR block:* 10.0.0.0/16 (65531 IP addresses available)

IPv6 CIDR block: ● No IPv6 CIDR Block
○ Amazon provided IPv6 CIDR block

VPC name: VPC_ELB

Public subnet's IPv4 CIDR:* 10.0.0.0/24 (251 IP addresses available)

Availability Zone:* eu-west-1a ▼

Subnet name: Public subnet 1

You can add more subnets after AWS creates the VPC.

Service endpoints

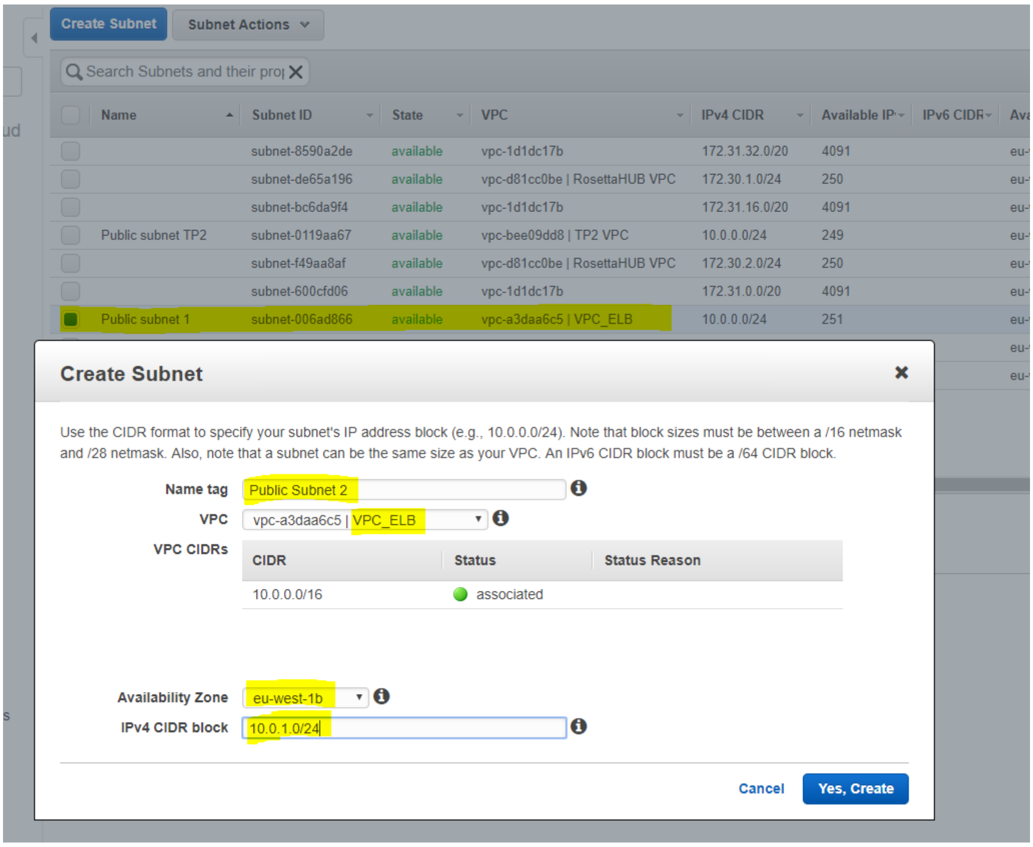
Enable DNS hostnames:* ● Yes ○ No

Hardware tenancy:* Default ▼

Step 2: Add the second Public Subnet

- **Subnet -> Create Subnet**

In the correct VPC, and in a different AZ than the “Public Subnet 1”



	Name	Subnet ID	State	VPC	IPv4 CIDR	Available IP	IPv6 CIDR	Ava
□		subnet-8590a2de	available	vpc-1d1dc17b	172.31.32.0/20	4091		eu-
□		subnet-de65a196	available	vpc-d81cc0be | RosettaHUB VPC	172.30.1.0/24	250		eu-
□		subnet-bc6da9f4	available	vpc-1d1dc17b	172.31.16.0/20	4091		eu-
□	Public subnet TP2	subnet-0119aa67	available	vpc-bee09dd8 | TP2 VPC	10.0.0.0/24	249		eu-
□		subnet-f49aa8af	available	vpc-d81cc0be | RosettaHUB VPC	172.30.2.0/24	250		eu-
□		subnet-600cfd06	available	vpc-1d1dc17b	172.31.0.0/20	4091		eu-
■	Public subnet 1	subnet-006ad866	available	vpc-a3daa6c5 | VPC_ELB	10.0.0.0/24	251		eu-
								eu-
								eu-

Create Subnet ×

Use the CIDR format to specify your subnet's IP address block (e.g., 10.0.0.0/24). Note that block sizes must be between a /16 netmask and /28 netmask. Also, note that a subnet can be the same size as your VPC. An IPv6 CIDR block must be a /64 CIDR block.

Name tag: Public Subnet 2

VPC: vpc-a3daa6c5 | VPC_ELB

VPC CIDRs:

CIDR	Status	Status Reason
10.0.0.0/16	● associated	

Availability Zone: eu-west-1b

IPv4 CIDR block: 10.0.1.0/24

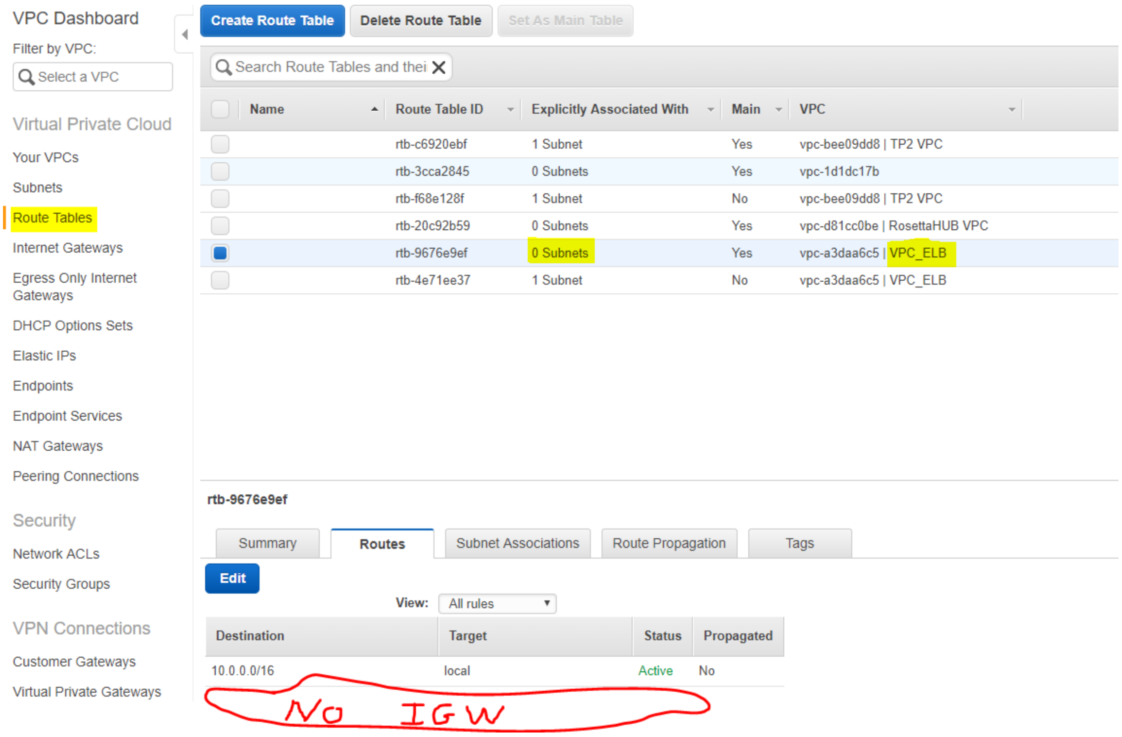
Step 3: Attach the 2nd subnet to the IGW, to be able to communicate with outside

By default, the Public Subnet just created is not connected to an Internet Gateway.

We have to configure it manually.

To do this:

- go to Route Table on the right Dashboard.
- Select the Route of our “VPC_ELB”, which is not associated to a Subnet: “Explicit Associated With 0 Subnets” or Main to “yes”



	Name	Route Table ID	Explicitly Associated With	Main	VPC
□		rtb-c6920ebf	1 Subnet	Yes	vpc-bee09dd8 | TP2 VPC
□		rtb-3cca2845	0 Subnets	Yes	vpc-1d1dc17b
□		rtb-f68e128f	1 Subnet	No	vpc-bee09dd8 | TP2 VPC
□		rtb-20c92b59	0 Subnets	Yes	vpc-d81cc0be | RosettaHUB VPC
☑		rtb-9676e9ef	0 Subnets	Yes	vpc-a3daa6c5 | VPC_ELB
□		rtb-4e71ee37	1 Subnet	No	vpc-a3daa6c5 | VPC_ELB

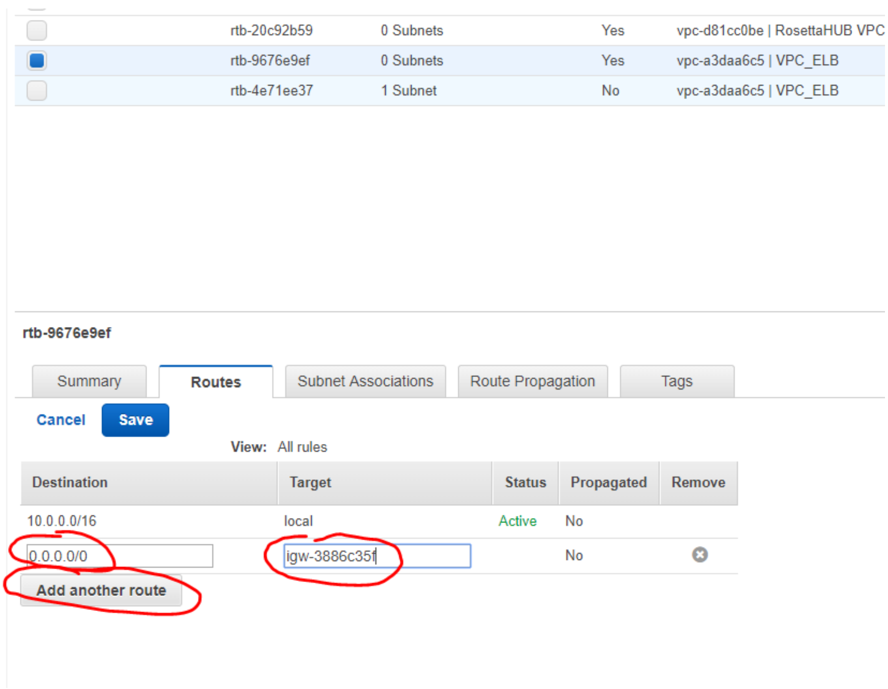
rtb-9676e9ef

View: All rules

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No

NO IGW

- Add another Route



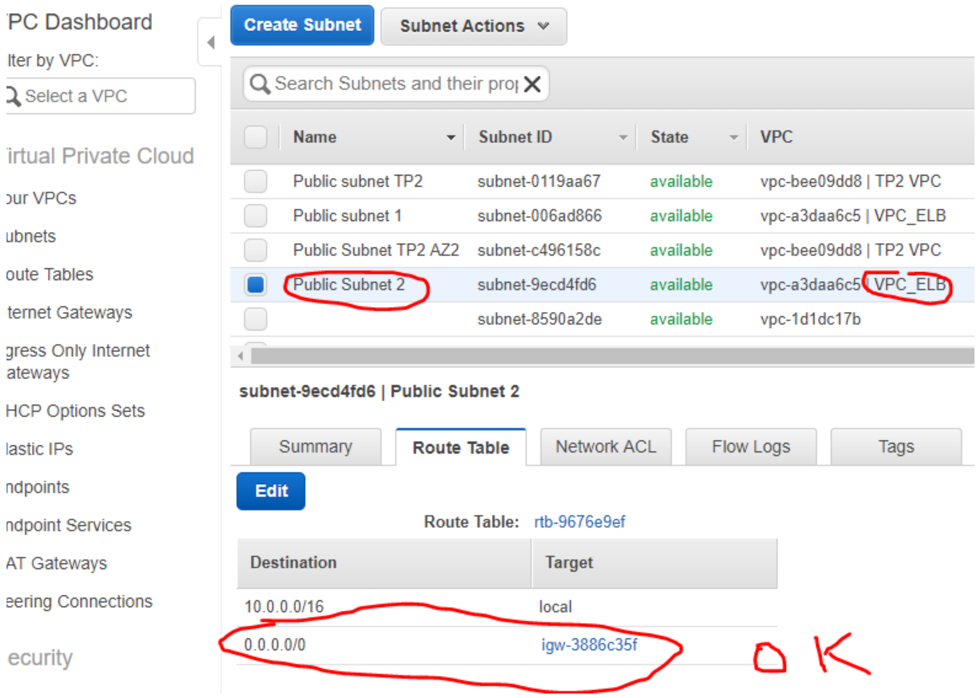
□	rtb-20c92b59	0 Subnets	Yes	vpc-d81cc0be | RosettaHUB VPC
☑	rtb-9676e9ef	0 Subnets	Yes	vpc-a3daa6c5 | VPC_ELB
□	rtb-4e71ee37	1 Subnet	No	vpc-a3daa6c5 | VPC_ELB

rtb-9676e9ef

View: All rules

Destination	Target	Status	Propagated	Remove
10.0.0.0/16	local	Active	No	
0.0.0.0/0	igw-3886c35f		No	⊗

- Check if “Public Subnet 2” is now correctly associated to IGW



	Name	Subnet ID	State	VPC
□	Public subnet TP2	subnet-0119aa67	available	vpc-bee09dd8 | TP2 VPC
□	Public subnet 1	subnet-006ad866	available	vpc-a3daa6c5 | VPC_ELB
□	Public Subnet TP2 AZ2	subnet-c496158c	available	vpc-bee09dd8 | TP2 VPC
☑	Public Subnet 2	subnet-9ecd4fd6	available	vpc-a3daa6c5 | VPC_ELB
□		subnet-8590a2de	available	vpc-1d1dc17b

subnet-9ecd4fd6 | Public Subnet 2

Route Table: rtb-9676e9ef

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	igw-3886c35f

OK

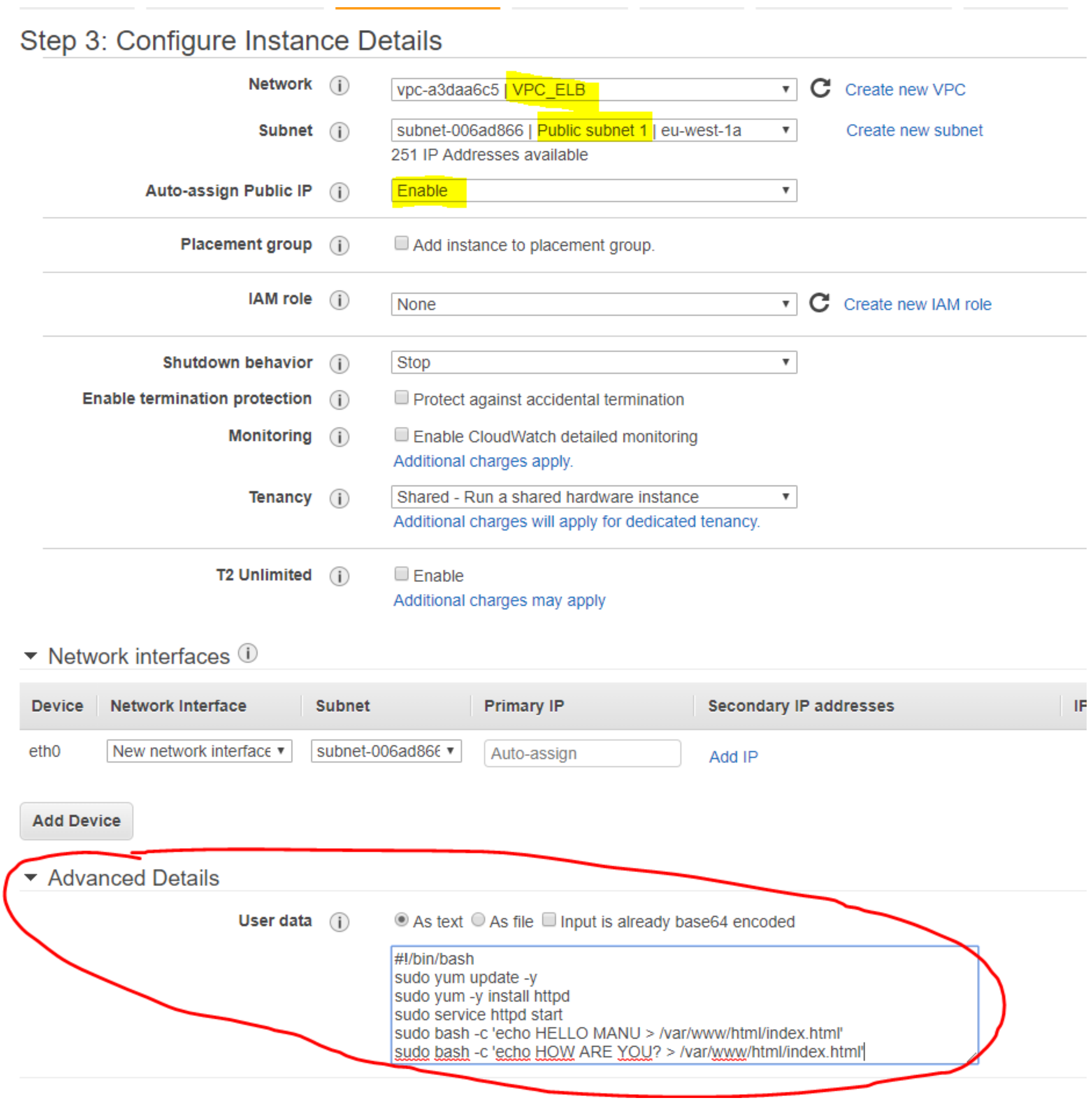
Step 4: Create the 1st Instance (in Public Subnet 1) with Apache Server

In the AWS console:

- Services -> EC2
- Launch Instance
- Step 1: Choose an Amazon Machine Image (AMI) -> Amazon Linux AMI
- Step 2: Choose an Instance Type -> t2.micro
- Step 3: Configure Instance Details : as below
  Do Not Forget to set the “Advanced Details” section, with following text, in order to install Apache:

```
#!/bin/bash
sudo yum update -y
sudo yum -y install httpd
sudo service httpd start
sudo bash -c 'echo HELLO MANU > /var/www/html/index.html'
sudo bash -c 'echo HOW ARE YOU? >> /var/www/html/index.html'
```

This code allows to install Apache.



Step 3: Configure Instance Details

Network	vpc-a3daa6c5 | VPC_ELB ▾ — Create new VPC
Subnet	subnet-006ad866 | Public subnet 1 | eu-west-1a ▾ — Create new subnet 251 IP Addresses available
Auto-assign Public IP	Enable ▾
Placement group	□ Add instance to placement group.
IAM role	None ▾ — Create new IAM role
Shutdown behavior	Stop ▾
Enable termination protection	□ Protect against accidental termination
Monitoring	□ Enable CloudWatch detailed monitoring Additional charges apply.
Tenancy	Shared - Run a shared hardware instance ▾ Additional charges will apply for dedicated tenancy.
T2 Unlimited	□ Enable Additional charges may apply

▾ Network interfaces

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IF
eth0	New network interface ▾	subnet-006ad866 ▾	Auto-assign	Add IP	

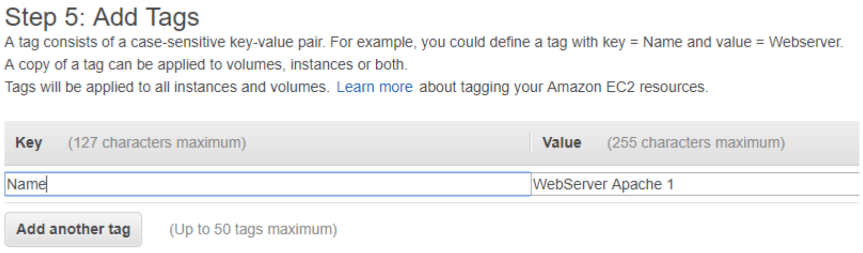
Add Device

▾ Advanced Details

User data ● As text ○ As file □ Input is already base64 encoded

```
#!/bin/bash
sudo yum update -y
sudo yum -y install httpd
sudo service httpd start
sudo bash -c 'echo HELLO MANU > /var/www/html/index.html'
sudo bash -c 'echo HOW ARE YOU? > /var/www/html/index.html'
```

- Step 4: Add Storage -> Next
- Step 5: Add Tags :



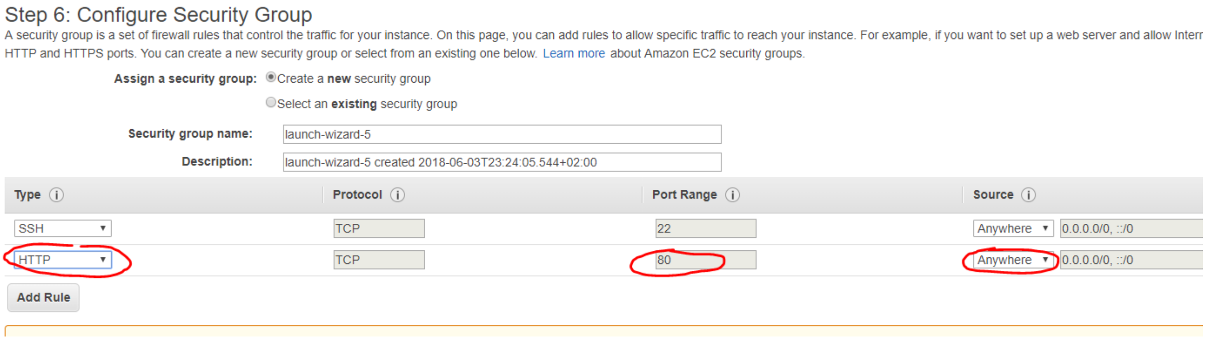
Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (127 characters maximum)	Value (255 characters maximum)
Name	WebServer Apache 1

Add another tag (Up to 50 tags maximum)

- Step 6: Configure Security Group : Add Rule, to allow set up your web server: port HTTP



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Interr HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group ○ Select an **existing** security group

Security group name: launch-wizard-5

Description: launch-wizard-5 created 2018-06-03T23:24:05.544+02:00

Type	Protocol	Port Range	Source	
SSH	TCP	22	Anywhere	0.0.0.0/0, ::/0
HTTP	TCP	80	Anywhere	0.0.0.0/0, ::/0

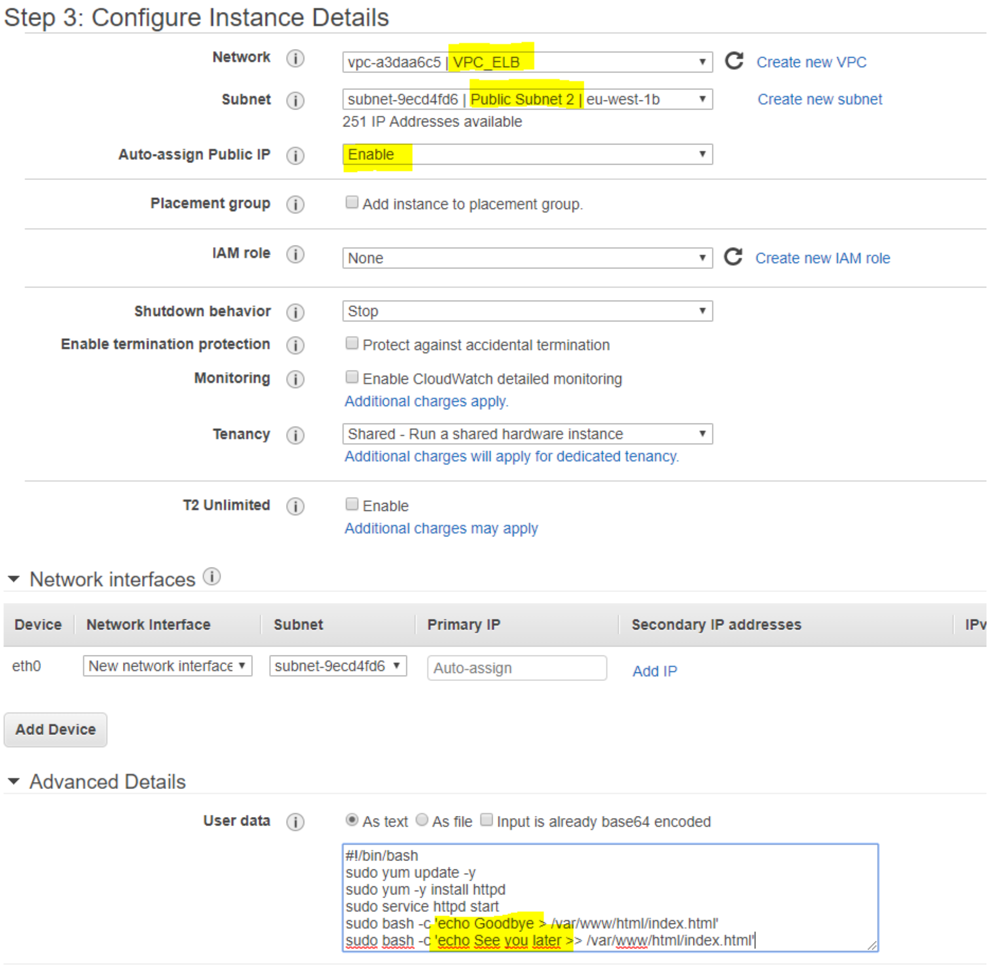
- Step 7: Review Instance Launch

  Select an existing key pair or create a new one

  And Launch Instances

Step 5: Create the 2nd Instance (in Public Subnet 2) with Apache Server

Repeat the same steps than **Step 4,** but for the Public Subnet 2, and specify a different message printed on your web server, to differentiate when we will refresh



Step 3: Configure Instance Details

Network	vpc-a3daa6c5 | VPC_ELB
Subnet	subnet-9ecd4fd6 | Public Subnet 2 | eu-west-1b 251 IP Addresses available
Auto-assign Public IP	Enable
Placement group	□ Add instance to placement group.
IAM role	None
Shutdown behavior	Stop
Enable termination protection	□ Protect against accidental termination
Monitoring	□ Enable CloudWatch detailed monitoring Additional charges apply.
Tenancy	Shared - Run a shared hardware instance Additional charges will apply for dedicated tenancy.
T2 Unlimited	□ Enable Additional charges may apply

Network interfaces

Device	Network Interface	Subnet	Primary IP	Secondary IP addresses	IPv
eth0	New network interface	subnet-9ecd4fd6	Auto-assign	Add IP	

Advanced Details

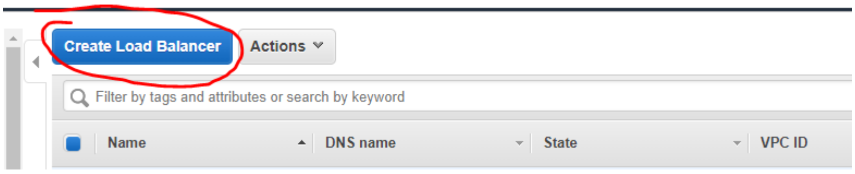
User data ● As text ○ As file □ Input is already base64 encoded

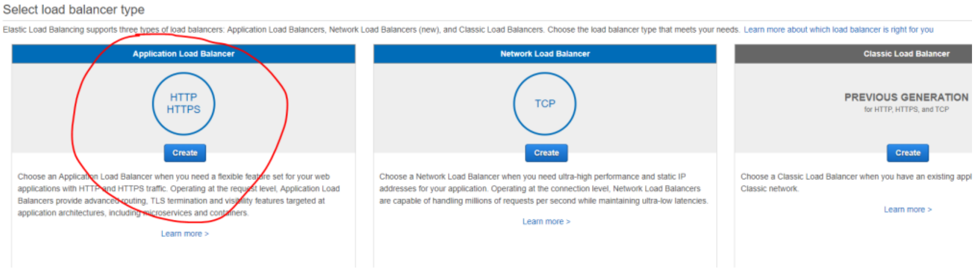
```
#!/bin/bash
sudo yum update -y
sudo yum -y install httpd
sudo service httpd start
sudo bash -c 'echo Goodbye > /var/www/html/index.html'
sudo bash -c 'echo See you later >> /var/www/html/index.html'
```

Step 6: Create and Configure ELB

In the AWS console, on EC2 Dashboards, Select Load Balancers

- And Create Load Balancer




Select load balancer type

Elastic Load Balancing supports three types of load balancers: Application Load Balancers, Network Load Balancers (new), and Classic Load Balancers. Choose the load balancer type that meets your needs. Learn more about which load balancer is right for you

Application Load Balancer

HTTP HTTPS

Create

Choose an Application Load Balancer when you need a flexible feature set for your web applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing, TLS termination and visibility features targeted at application architectures, including microservices and containers.

Learn more >

Network Load Balancer

TCP

Create

Choose a Network Load Balancer when you need ultra-high performance and static IP addresses for your application. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second while maintaining ultra-low latencies.

Learn more >

Classic Load Balancer

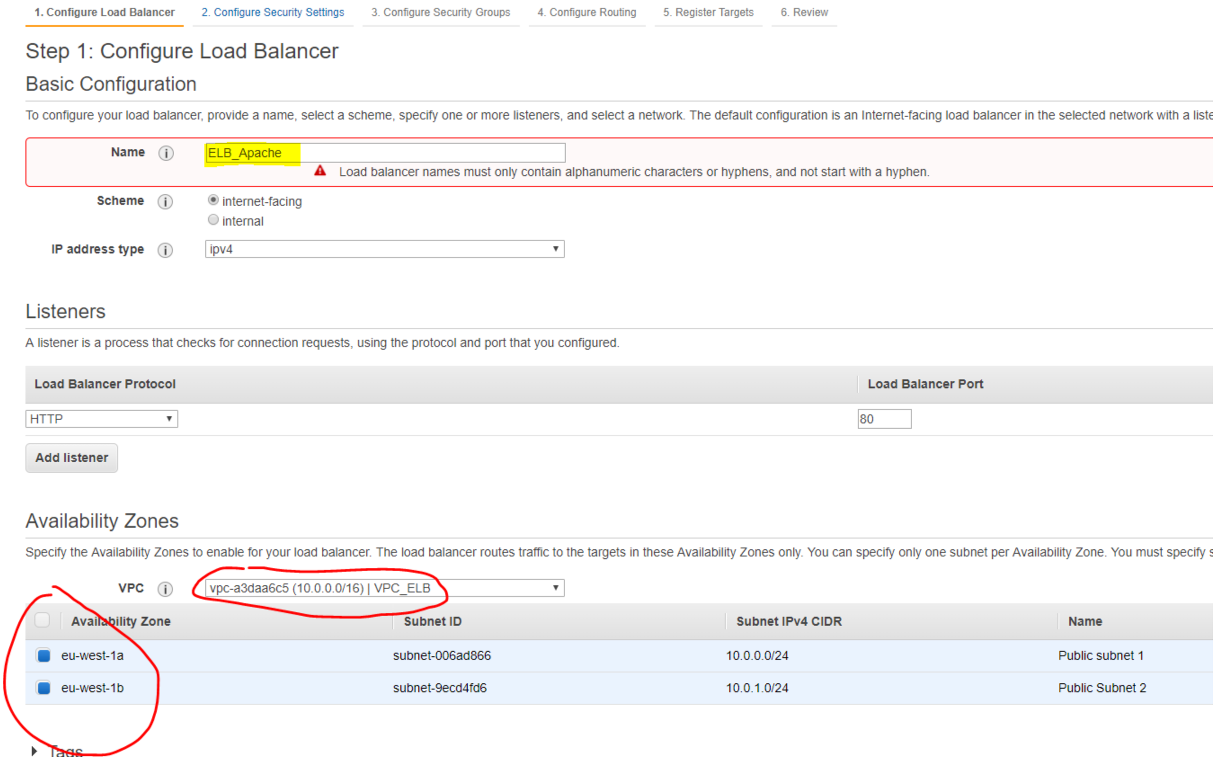
PREVIOUS GENERATION
for HTTP, HTTPS, and TCP

Create

Choose a Classic Load Balancer when you have an existing appl Classic network.

Learn more >

- And Configure as snapshot below, mainly the Availability Zones and the 2 Subnets:


1. Configure Load Balancer 2. Configure Security Settings 3. Configure Security Groups 4. Configure Routing 5. Register Targets 6. Review

Step 1: Configure Load Balancer

Basic Configuration

To configure your load balancer, provide a name, select a scheme, specify one or more listeners, and select a network. The default configuration is an Internet-facing load balancer in the selected network with a liste

Name ⓘ ELB_Apache

⚠ Load balancer names must only contain alphanumeric characters or hyphens, and not start with a hyphen.

Scheme ⓘ ● internet-facing ○ internal

IP address type ⓘ ipv4

Listeners

A listener is a process that checks for connection requests, using the protocol and port that you configured.

Load Balancer Protocol	Load Balancer Port
HTTP	80

Add listener

Availability Zones

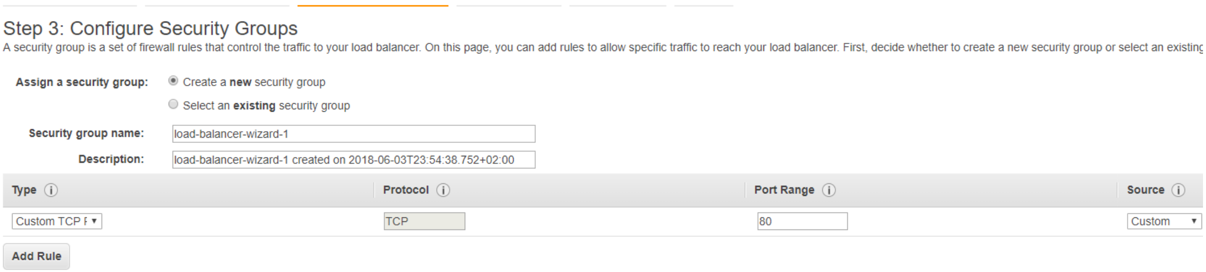
Specify the Availability Zones to enable for your load balancer. The load balancer routes traffic to the targets in these Availability Zones only. You can specify only one subnet per Availability Zone. You must specify s

VPC ⓘ vpc-a3daa6c5 (10.0.0.0/16) | VPC_ELB

Availability Zone	Subnet ID	Subnet IPv4 CIDR	Name
■ eu-west-1a	subnet-006ad866	10.0.0.0/24	Public subnet 1
■ eu-west-1b	subnet-9ecd4fd6	10.0.1.0/24	Public Subnet 2

▸ Tags

- Configure Security Groups


Step 3: Configure Security Groups

A security group is a set of firewall rules that control the traffic to your load balancer. On this page, you can add rules to allow specific traffic to reach your load balancer. First, decide whether to create a new security group or select an existing

Assign a security group: ● Create a **new** security group ○ Select an **existing** security group

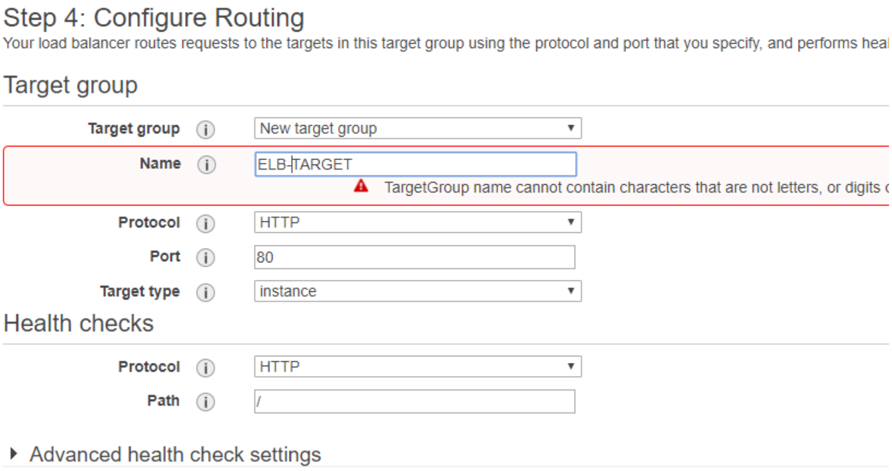
Security group name: load-balancer-wizard-1

Description: load-balancer-wizard-1 created on 2018-06-03T23:54:38.752+02:00

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ
Custom TCP F	TCP	80	Custom

Add Rule

- Configure Routing, as default:



Step 4: Configure Routing

Your load balancer routes requests to the targets in this target group using the protocol and port that you specify, and performs heal

Target group

Target group: New target group

Name: ELB-TARGET

TargetGroup name cannot contain characters that are not letters, or digits c

Protocol: HTTP

Port: 80

Target type: instance

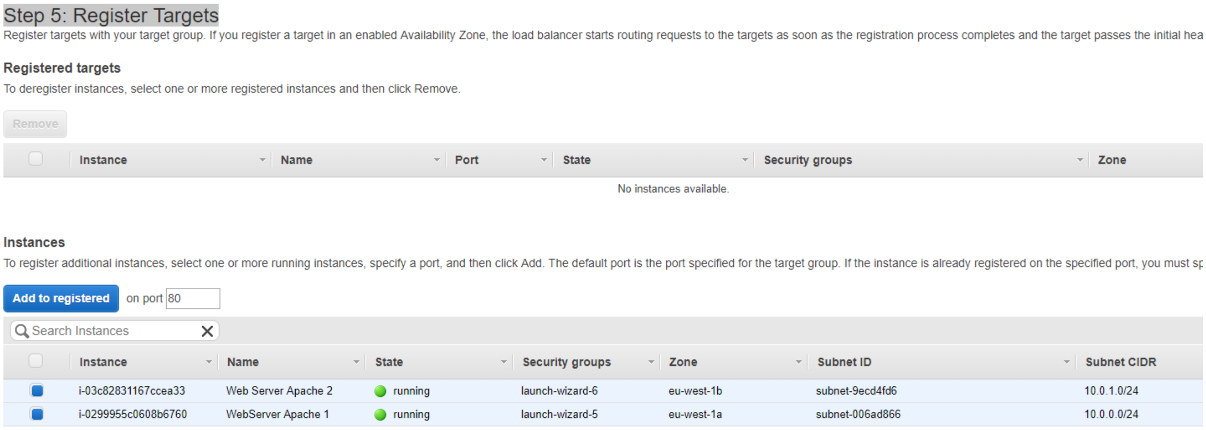
Health checks

Protocol: HTTP

Path: /

Advanced health check settings

- Register Targets: Add our 2 instances in the target group



Step 5: Register Targets

Register targets with your target group. If you register a target in an enabled Availability Zone, the load balancer starts routing requests to the targets as soon as the registration process completes and the target passes the initial hea

Registered targets

To deregister instances, select one or more registered instances and then click Remove.

	Instance	Name	Port	State	Security groups	Zone
	No instances available.					

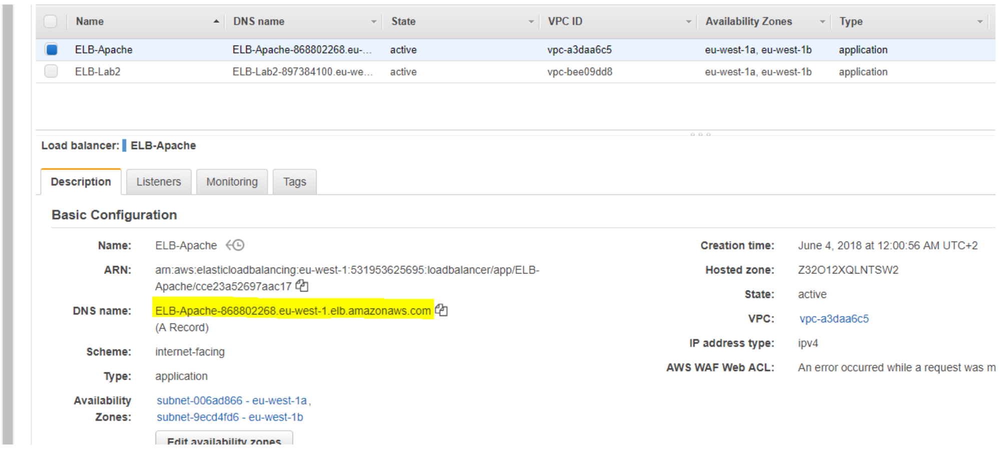
Instances

To register additional instances, select one or more running instances, specify a port, and then click Add. The default port is the port specified for the target group. If the instance is already registered on the specified port, you must sp

Add to registered on port 80

	Instance	Name	State	Security groups	Zone	Subnet ID	Subnet CIDR
☑	i-03c82831167ccea33	Web Server Apache 2	running	launch-wizard-6	eu-west-1b	subnet-9ecd4fd6	10.0.1.0/24
☑	i-0299955c0608b6760	WebServer Apache 1	running	launch-wizard-5	eu-west-1a	subnet-006ad866	10.0.0.0/24

Step 8: Open in a new web Brower, the DNS of the ELB just created:



	Name	DNS name	State	VPC ID	Availability Zones	Type
☑	ELB-Apache	ELB-Apache-868802268.eu-...	active	vpc-a3daa6c5	eu-west-1a, eu-west-1b	application
□	ELB-Lab2	ELB-Lab2-897384100.eu-we...	active	vpc-bee09dd8	eu-west-1a, eu-west-1b	application

Load balancer: ELB-Apache

Description | Listeners | Monitoring | Tags

Basic Configuration

Name: ELB-Apache

ARN: arn:aws:elasticloadbalancing:eu-west-1:531953625695:loadbalancer/app/ELB-Apache/cce23a52697aac17

DNS name: ELB-Apache-868802268.eu-west-1.elb.amazonaws.com (A Record)

Scheme: internet-facing

Type: application

Availability Zones: subnet-006ad866 - eu-west-1a, subnet-9ecd4fd6 - eu-west-1b

Edit availability zones

Creation time: June 4, 2018 at 12:00:56 AM UTC+2

Hosted zone: Z32O12XQLNTSW2

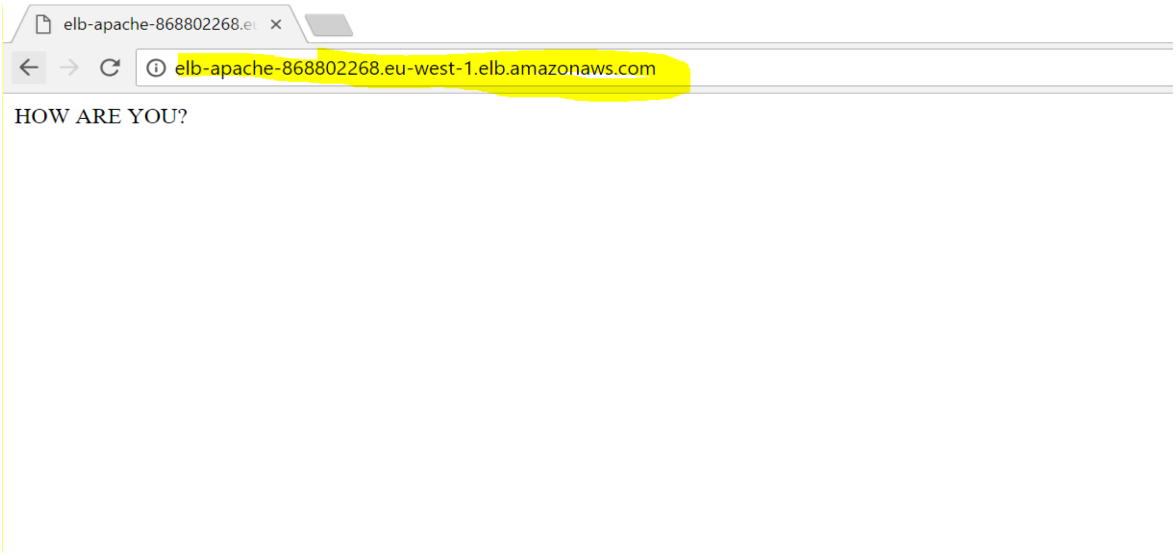
State: active

VPC: vpc-a3daa6c5

IP address type: ipv4

AWS WAF Web ACL: An error occurred while a request was m

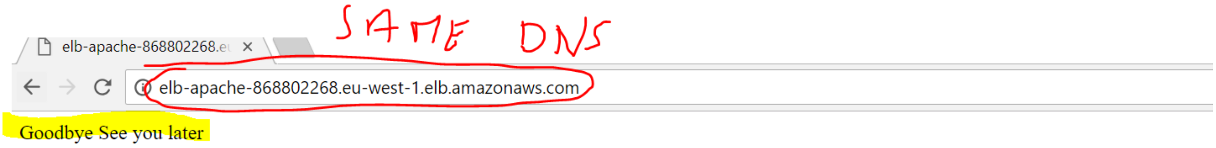
Here we go:



elb-apache-868802268.eu-west-1.elb.amazonaws.com

HOW ARE YOU?

After few refresh



SAME DNS

elb-apache-868802268.eu-west-1.elb.amazonaws.com

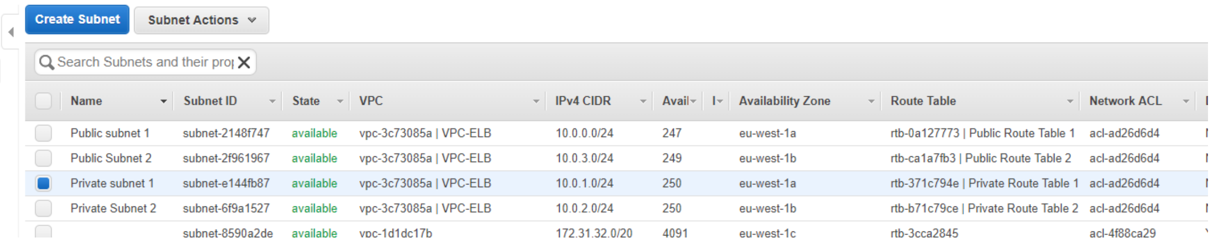
Goodbye See you later

PART 2: THE 2 INSTANCES ARE IN PRIVATE SUBNET

Now the instances have to be in the private subnets, and no more instances in the public.

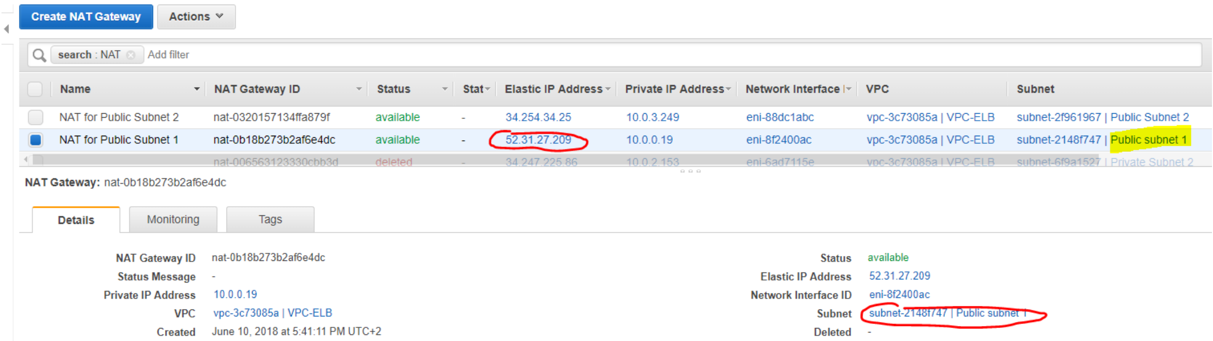
So ,we need:

- The VPC and Internet Gateway
- 2 Availability Zones: eu-west-1a and eu-west-1b
- 4 subnets: 2 in each AZ : 1 public, 1 private:



	Name	Subnet ID	State	VPC	IPv4 CIDR	Avail	I	Availability Zone	Route Table	Network ACL
□	Public subnet 1	subnet-2148f747	available	vpc-3c73085a | VPC-ELB	10.0.0.0/24	247		eu-west-1a	rtb-0a127773 | Public Route Table 1	acl-ad26d6d4
□	Public Subnet 2	subnet-2f961967	available	vpc-3c73085a | VPC-ELB	10.0.3.0/24	249		eu-west-1b	rtb-ca1a7fb3 | Public Route Table 2	acl-ad26d6d4
■	Private subnet 1	subnet-e144fb87	available	vpc-3c73085a | VPC-ELB	10.0.1.0/24	250		eu-west-1a	rtb-371c794e | Private Route Table 1	acl-ad26d6d4
□	Private Subnet 2	subnet-6f9a1527	available	vpc-3c73085a | VPC-ELB	10.0.2.0/24	250		eu-west-1b	rtb-b71c79ce | Private Route Table 2	acl-ad26d6d4
□		subnet-8590a2de	available	vpc-1d1dc17b	172.31.32.0/20	4091		eu-west-1c	rtb-3cca2845	acl-4f88ca29

- 2 NAT Gateways: one in each AZ and **public** subnet.
  Each NAT Gateway associated to 1 Elastic IP

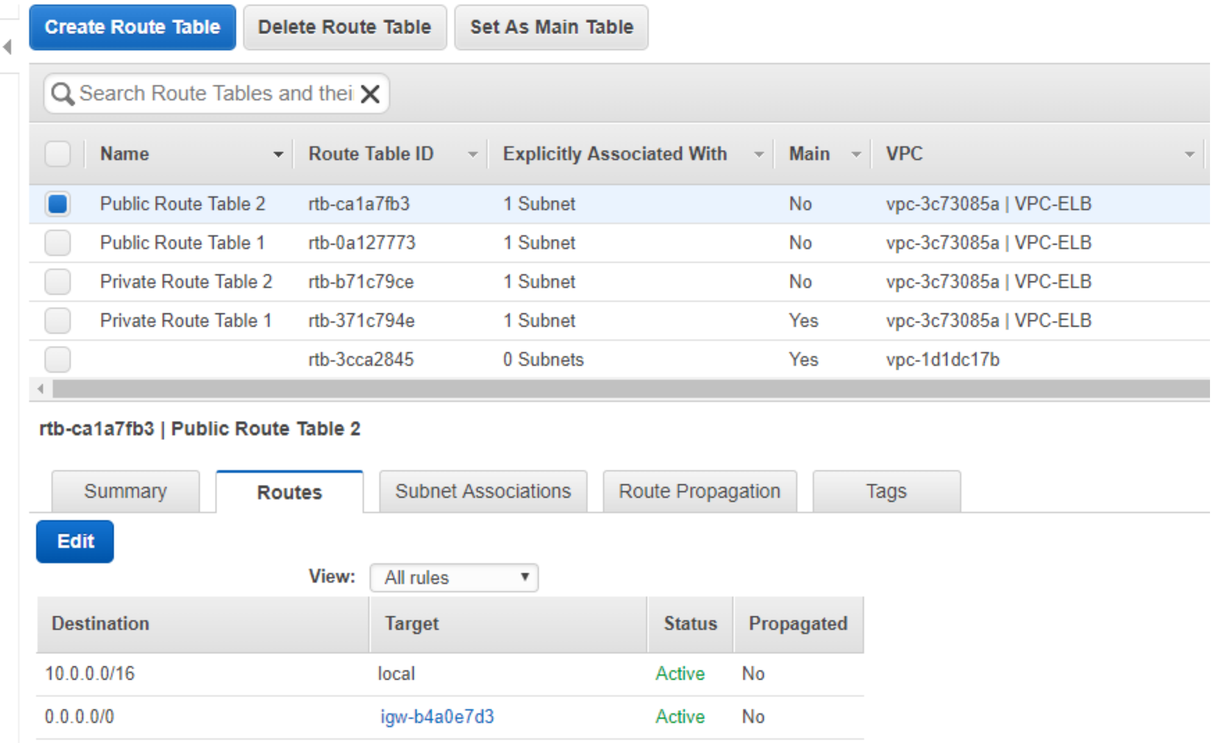


	Name	NAT Gateway ID	Status	Stat	Elastic IP Address	Private IP Address	Network Interface	VPC	Subnet
□	NAT for Public Subnet 2	nat-0320157134ffa879f	available	-	34.254.34.25	10.0.3.249	eni-88dc1abc	vpc-3c73085a | VPC-ELB	subnet-2f961967 | Public Subnet 2
■	NAT for Public Subnet 1	nat-0b18b273b2af6e4dc	available	-	52.31.27.209	10.0.0.19	eni-8f2400ac	vpc-3c73085a | VPC-ELB	subnet-2148f747 | Public subnet 1
□		nat-006563123330cbb3d	deleted	-	34.247.225.86	10.0.2.153	eni-6ad7115e	vpc-3c73085a | VPC-ELB	subnet-6f9a1527 | Private Subnet 2

NAT Gateway: nat-0b18b273b2af6e4dc

NAT Gateway ID	nat-0b18b273b2af6e4dc	Status	available
Status Message	-	Elastic IP Address	52.31.27.209
Private IP Address	10.0.0.19	Network Interface ID	eni-8f2400ac
VPC	vpc-3c73085a | VPC-ELB	Subnet	subnet-2148f747 | Public subnet 1
Created	June 10, 2018 at 5:41:11 PM UTC+2	Deleted	-

- 4 route tables: one route table for each subnet



	Name	Route Table ID	Explicitly Associated With	Main	VPC
■	Public Route Table 2	rtb-ca1a7fb3	1 Subnet	No	vpc-3c73085a | VPC-ELB
□	Public Route Table 1	rtb-0a127773	1 Subnet	No	vpc-3c73085a | VPC-ELB
□	Private Route Table 2	rtb-b71c79ce	1 Subnet	No	vpc-3c73085a | VPC-ELB
□	Private Route Table 1	rtb-371c794e	1 Subnet	Yes	vpc-3c73085a | VPC-ELB
□		rtb-3cca2845	0 Subnets	Yes	vpc-1d1dc17b

rtb-ca1a7fb3 | Public Route Table 2

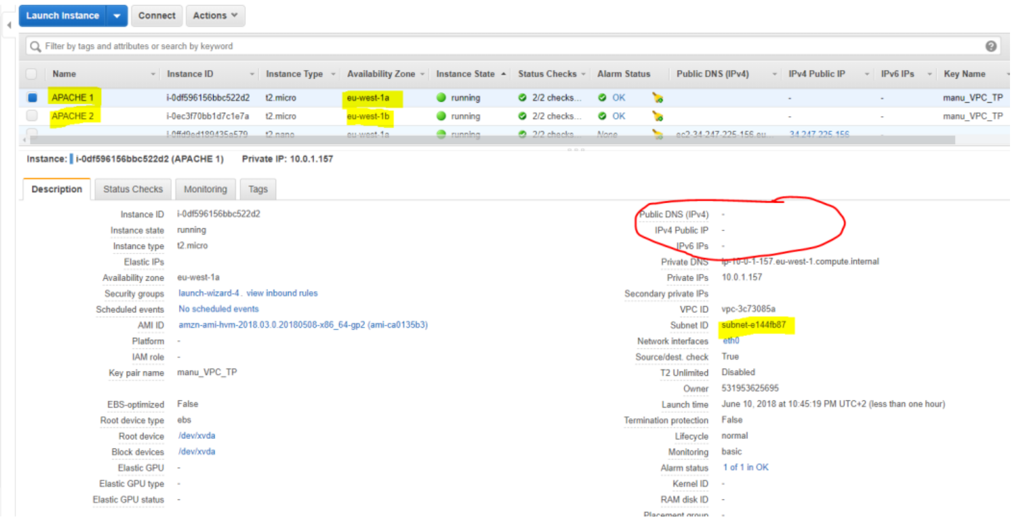
View: All rules

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	No
0.0.0.0/0	igw-b4a0e7d3	Active	No

  - Edit the tab "Routes" and :
    - Add IGW for public subnet

        - Add NAT for the private subnet
    - Edit the tab “Subnet Associations” and associate the corresponding subnet

- 1 instance in each Private Subnet, with the script/commands to install Apache Server in User Data section:



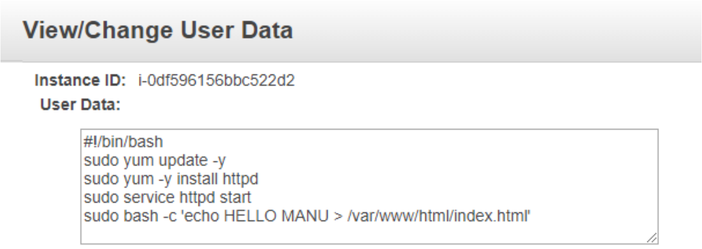
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4 Public IP	IPv6 IPs	Key Name
APACHE 1	i-0df596156bbc522d2	t2.micro	eu-west-1a	running	2/2 checks...	OK		-	-	manu_VPC_TP
APACHE 2	i-0ec3f70bb1d7c1e7a	t2.micro	eu-west-1b	running	2/2 checks...	OK		-	-	manu_VPC_TP
	i-0ffd9ed189435a579	t2.nano	eu-west-1a	running	2/2 checks...	None	ec2-34-247-225-156.eu...	34.247.225.156	-	

Instance: i-0df596156bbc522d2 (APACHE 1) Private IP: 10.0.1.157

Instance ID i-0df596156bbc522d2
Instance state running
Instance type t2.micro
Elastic IPs
Availability zone eu-west-1a
Security groups launch-wizard-4 . view inbound rules
Scheduled events No scheduled events
AMI ID amzn-ami-hvm-2018.03.0.20180508-x86_64-gp2 (ami-ca0135b3)
Platform -
IAM role -
Key pair name manu_VPC_TP

EBS-optimized False
Root device type ebs
Root device /dev/xvda
Block devices /dev/xvda
Elastic GPU -
Elastic GPU type -
Elastic GPU status -

Public DNS (IPv4) -
IPv4 Public IP -
IPv6 IPs -
Private DNS ip-10-0-1-157.eu-west-1.compute.internal
Private IPs 10.0.1.157
Secondary private IPs
VPC ID vpc-3c73085a
Subnet ID subnet-e144fb87
Network interfaces eth0
Source/dest. check True
T2 Unlimited Disabled
Owner 531953625695
Launch time June 10, 2018 at 10:45:19 PM UTC+2 (less than one hour)
Termination protection False
Lifecycle normal
Monitoring basic
Alarm status 1 of 1 in OK
Kernel ID -
RAM disk ID -
Placement group -

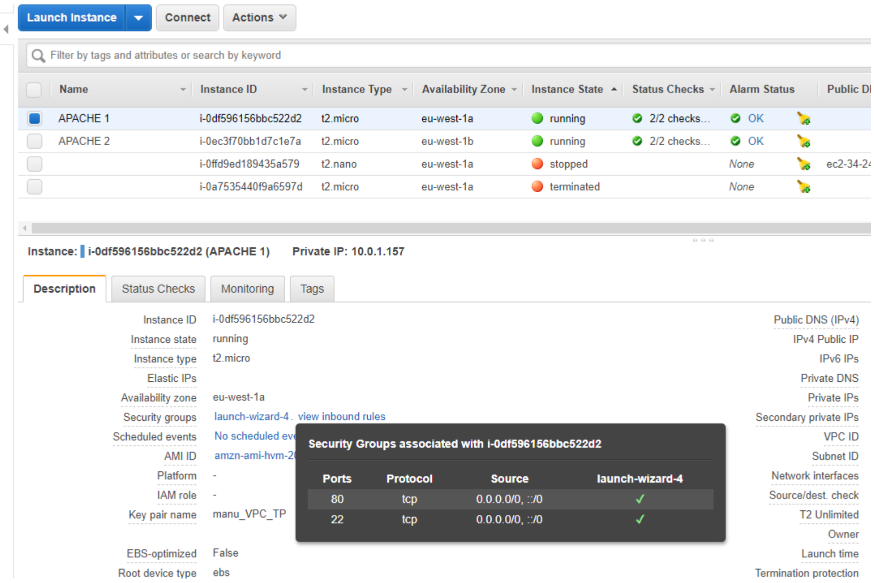


View/Change User Data

Instance ID: i-0df596156bbc522d2
User Data:

```
#!/bin/bash
sudo yum update -y
sudo yum -y install httpd
sudo service httpd start
sudo bash -c 'echo HELLO MANU > /var/www/html/index.html'
```

<u>And following Security Group:</u>



Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DI
APACHE 1	i-0df596156bbc522d2	t2.micro	eu-west-1a	running	2/2 checks...	OK	
APACHE 2	i-0ec3f70bb1d7c1e7a	t2.micro	eu-west-1b	running	2/2 checks...	OK	
	i-0ffd9ed189435a579	t2.nano	eu-west-1a	stopped		None	ec2-34-24
	i-0a7535440f9a6597d	t2.micro	eu-west-1a	terminated		None	

Instance: i-0df596156bbc522d2 (APACHE 1) Private IP: 10.0.1.157

Instance ID i-0df596156bbc522d2
Instance state running
Instance type t2.micro
Elastic IPs
Availability zone eu-west-1a
Security groups launch-wizard-4 . view inbound rules
Scheduled events No scheduled eve
AMI ID amzn-ami-hvm-2
Platform -
IAM role -
Key pair name manu_VPC_TP

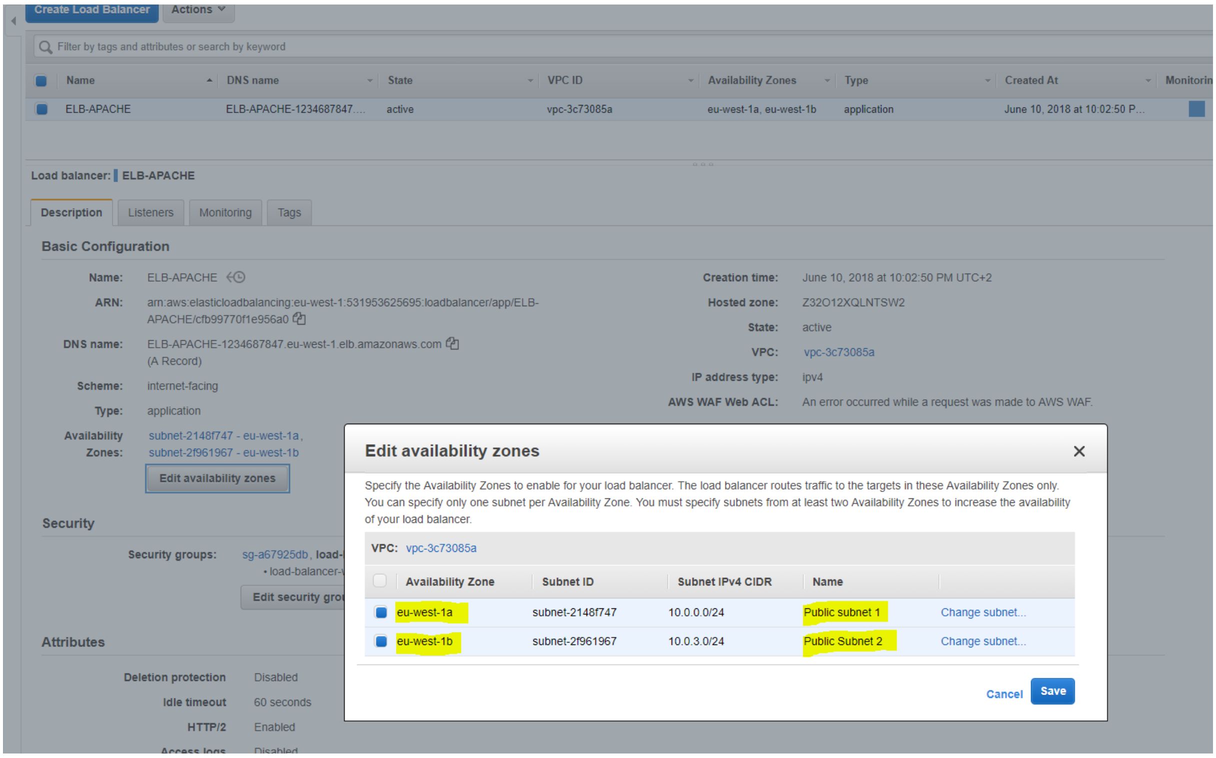
EBS-optimized False
Root device type ebs

Public DNS (IPv4)
IPv4 Public IP
IPv6 IPs
Private DNS
Private IPs
Secondary private IPs
VPC ID
Subnet ID
Network interfaces
Source/dest. check
T2 Unlimited
Owner
Launch time
Termination protection

Security Groups associated with i-0df596156bbc522d2

Ports	Protocol	Source	launch-wizard-4
80	tcp	0.0.0.0/0, ::/0	✓
22	tcp	0.0.0.0/0, ::/0	✓

- The ELB, between our 2 AZs and selecting the **Public Subnet**



Filter by tags and attributes or search by keyword

Name	DNS name	State	VPC ID	Availability Zones	Type	Created At	Monitorin
ELB-APACHE	ELB-APACHE-1234687847....	active	vpc-3c73085a	eu-west-1a, eu-west-1b	application	June 10, 2018 at 10:02:50 P...	

Load balancer: ELB-APACHE

Basic Configuration

Name: ELB-APACHE

ARN: arn:aws:elasticloadbalancing:eu-west-1:531953625695:loadbalancer/app/ELB-APACHE/cfb99770f1e956a0

DNS name: ELB-APACHE-1234687847.eu-west-1.elb.amazonaws.com (A Record)

Scheme: internet-facing

Type: application

Availability Zones: subnet-2148f747 - eu-west-1a, subnet-2f961967 - eu-west-1b

Creation time: June 10, 2018 at 10:02:50 PM UTC+2

Hosted zone: Z32O12XQLNTSW2

State: active

VPC: vpc-3c73085a

IP address type: ipv4

AWS WAF Web ACL: An error occurred while a request was made to AWS WAF.

Security

Security groups: sg-a67925db, load-
• load-balancer-

Attributes

Deletion protection: Disabled

Idle timeout: 60 seconds

HTTP/2: Enabled

Access logs: Disabled

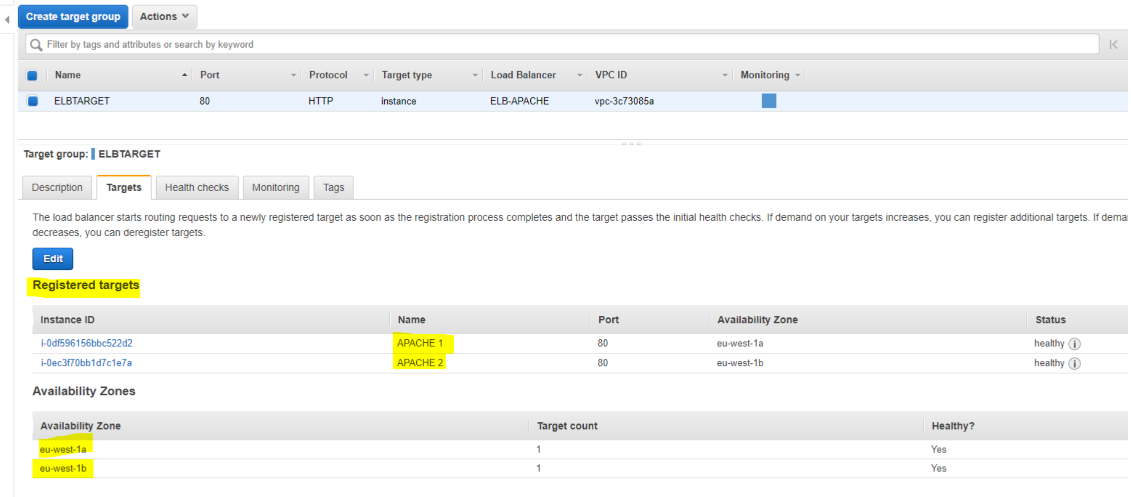
Edit availability zones

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 subnet per Availability Zone. You must specify subnets from at least two Availability Zones to increase the availability of your load balancer.

VPC: vpc-3c73085a

	Availability Zone	Subnet ID	Subnet IPv4 CIDR	Name	
☑	eu-west-1a	subnet-2148f747	10.0.0.0/24	Public subnet 1	Change subnet...
☑	eu-west-1b	subnet-2f961967	10.0.3.0/24	Public Subnet 2	Change subnet...

And register the 2 instances, which are in the Private Subnet



Filter by tags and attributes or search by keyword

Name	Port	Protocol	Target type	Load Balancer	VPC ID	Monitoring
ELBTARGET	80	HTTP	instance	ELB-APACHE	vpc-3c73085a	

Target group: ELBTARGET

The load balancer starts routing requests to a newly registered target as soon as the registration process completes and the target passes the initial health checks. If demand on your targets increases, you can register additional targets. If dema decreases, you can deregister targets.

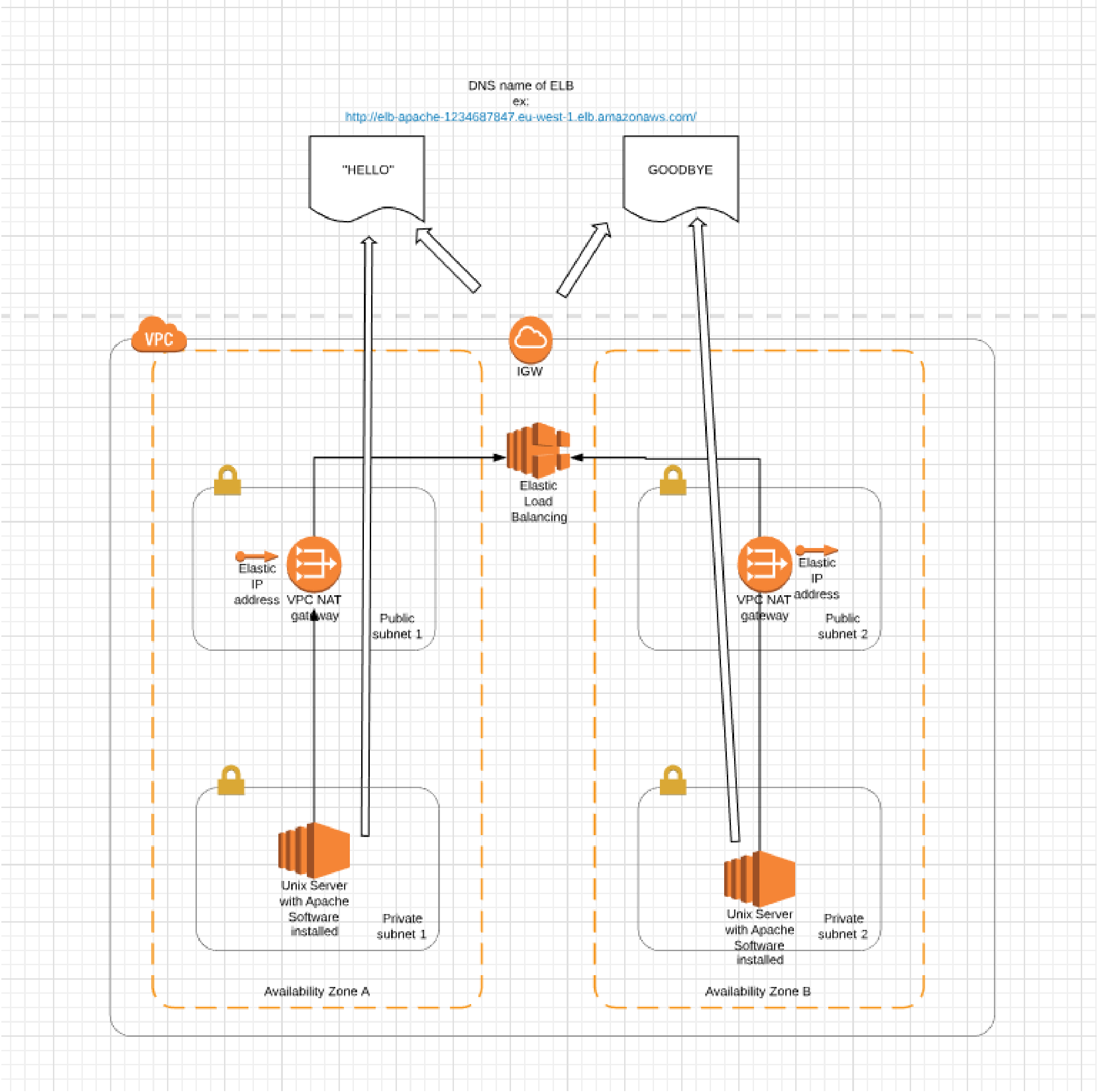
Registered targets

Instance ID	Name	Port	Availability Zone	Status
i-0df596156bbc522d2	APACHE 1	80	eu-west-1a	healthy ⓘ
i-0ec3f70bb1d7c1e7a	APACHE 2	80	eu-west-1b	healthy ⓘ

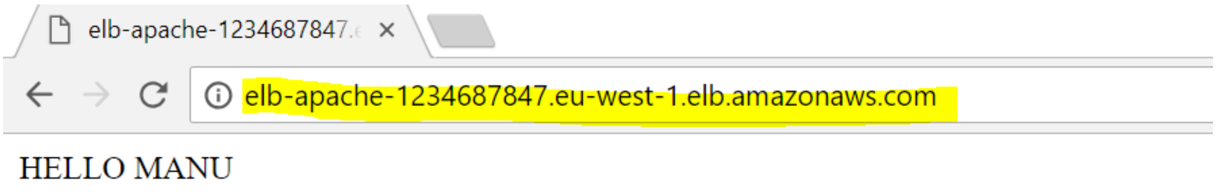
Availability Zones

Availability Zone	Target count	Healthy?
eu-west-1a	1	Yes
eu-west-1b	1	Yes

Here the final architecture:



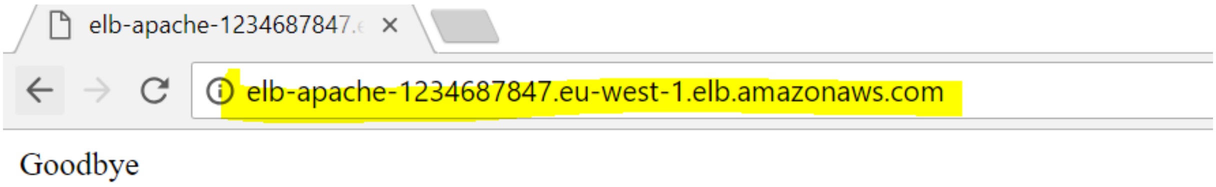
Copy paste the DNS name of the Load Balancer in a brower:



elb-apache-1234687847.eu-west-1.elb.amazonaws.com

HELLO MANU

And Refresh:



elb-apache-1234687847.eu-west-1.elb.amazonaws.com

Goodbye