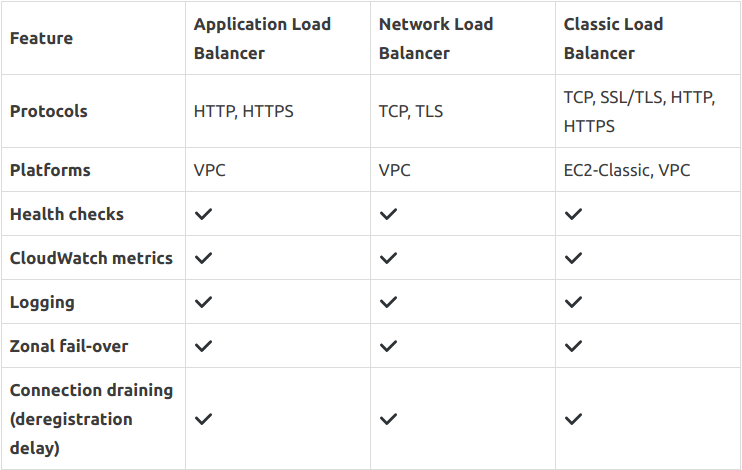
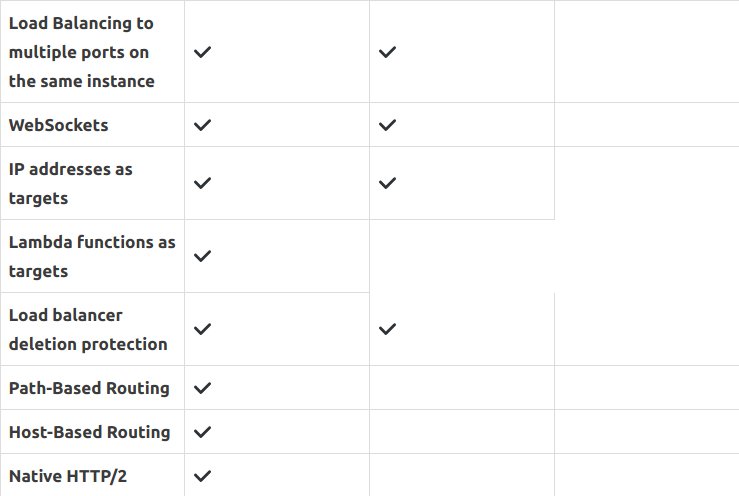
1. ***Differences between ELB, ALB, and NLB. Where will you use which one?***

*Here are some major difference between ELB , ALB and NLB*

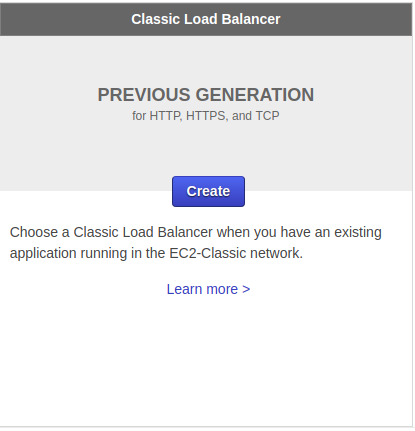
**

**

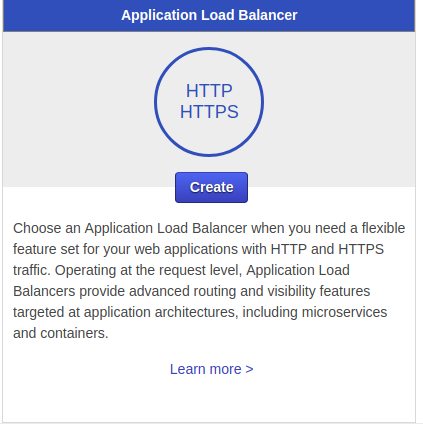
**

*On AWS console the following written differences can be easily seen*

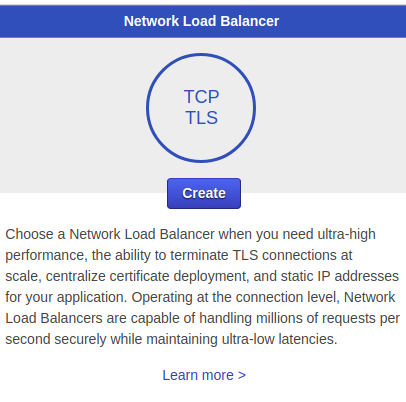
*ELB:- ELB is also known as Classic Load Balancer*

**

*ALB:-*

**

*NLB:-*

**

1. ***Differences between step scaling and target scaling.***

|  |  |
| --- | --- |
| ***STEP SCALING*** | ***TARGET SCALING*** |
| *Step scaling policies increase or decrease the current capacity of your Auto Scaling group based on a set of scaling adjustments, known as step adjustments. The adjustments vary based on the size of the alarm breach.* | *With target tracking scaling policies, you select a scaling metric and set a target value. Amazon EC2 Auto Scaling creates and manages the CloudWatch alarms that trigger the scaling policy and calculates the scaling adjustment based on the metric and the target value. The scaling policy adds or removes capacity as required to keep the metric at, or close to, the specified target value. In addition to keeping the metric close to the target value, a target tracking scaling policy also adjusts to the changes in the metric due to a changing load pattern.* |
| *Granular controlling , manual customization and tuning* | *It’s like a thermostat. Set the target value on your scaling metric and let the target tracking maintain your metric* |

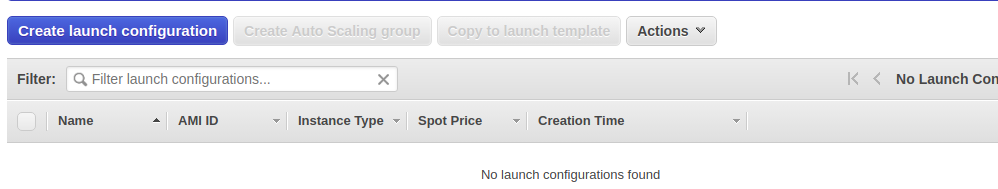
1. ***Differences between Launch configuration and launch template.***

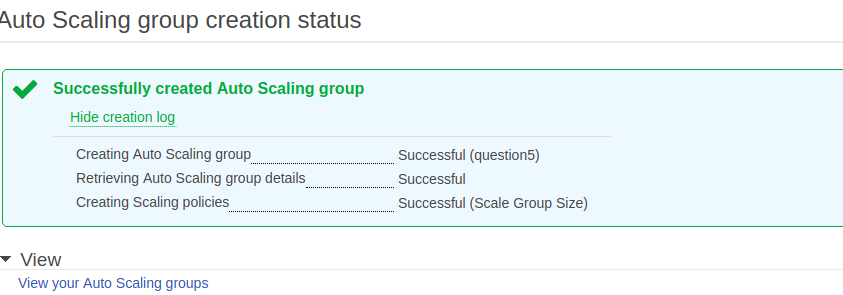
|  |  |
| --- | --- |
| ***LAUNCH CONFIGURATION*** | ***LAUNCH TEMPLATE*** |
| *It is an instance configuration that Auto-Scaling group uses to launch an instance* | *It stores the instance parameters so that when the net time an instance is launched it doesn’t need to be parameterized again* |
| *Can be same as launch template for a particular auto-scaling group* | *Each instance parameter is different according to the need of use and so it differs .* |

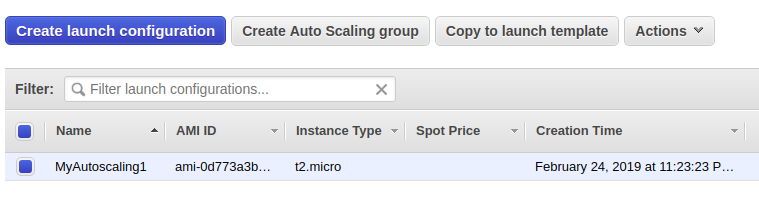
1. ***Differences between EC2 health check and load balancer health check***

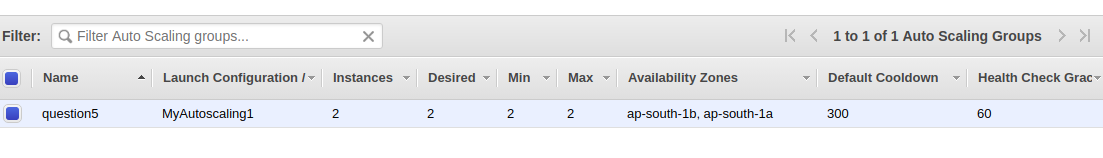
|  |  |
| --- | --- |
| ***EC2 HEALTH CHECK*** | ***LOAD BALANCER HEALTH CHECK*** |
| *Watches for instance availability from hypervisor and networking point of view. For example, in case of a hardware problem, the check will fail. Also, if an instance was misconfigured and doesn't respond to network requests, it will be marked as faulty.* | *Verifies that a specified TCP port on an instance is accepting connections OR a specified web page returns 2xx code. Thus ELB health checks are a little bit smarter and verify that actual app works instead of verifying that just an instance works.* |
|  |  |

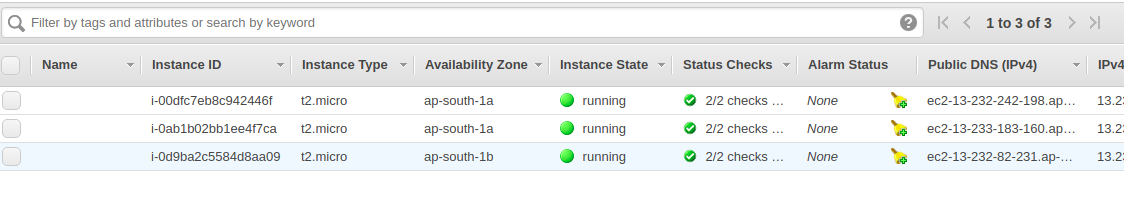
1. ***Create 2 auto-scaling groups with***
   1. ***launch configuration and***

******

******

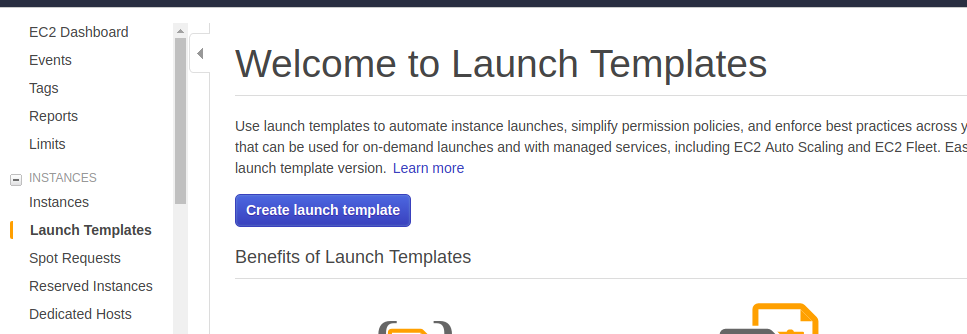
******

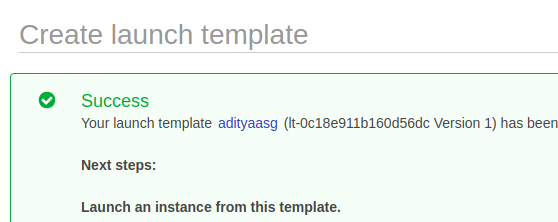
******

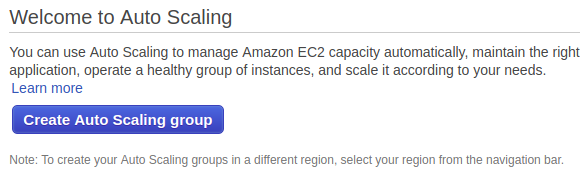
******

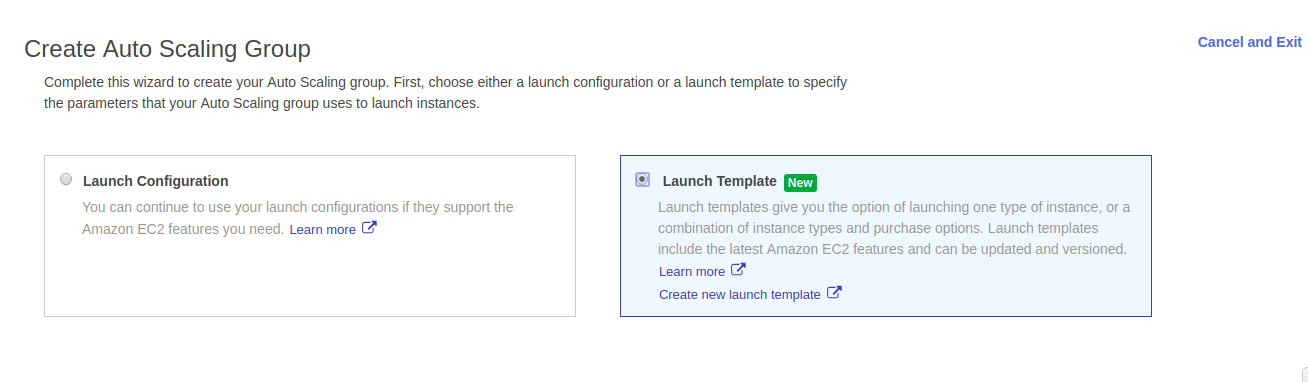
*The auto-scaling is set to increase number of instance as the CPU utilization crosses above 30%*

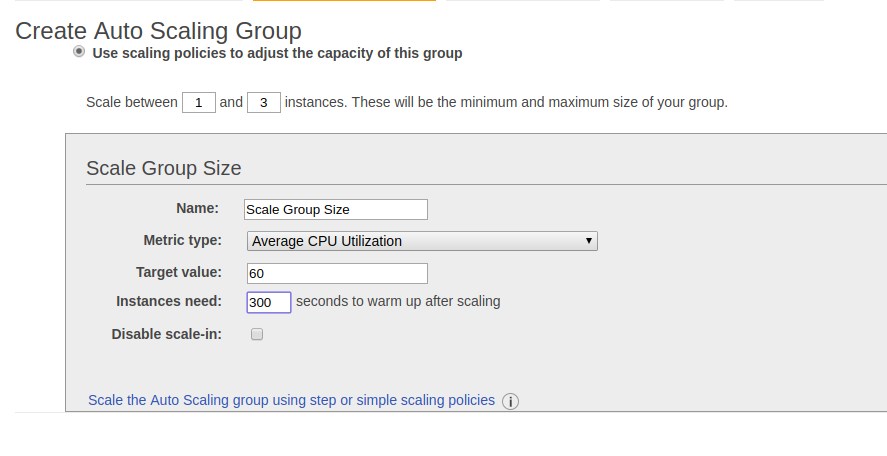
* 1. ***launch template***

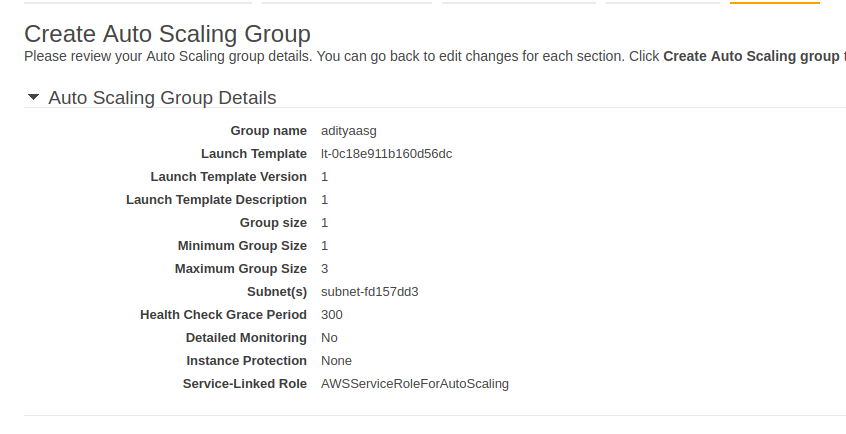
******

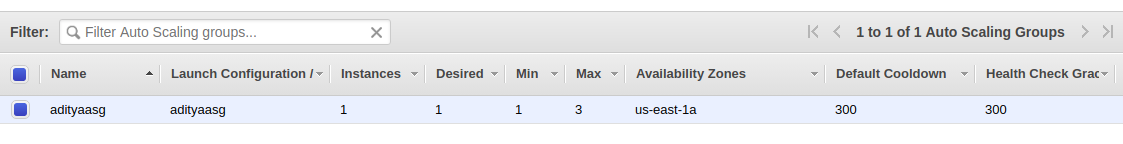
******

******

******

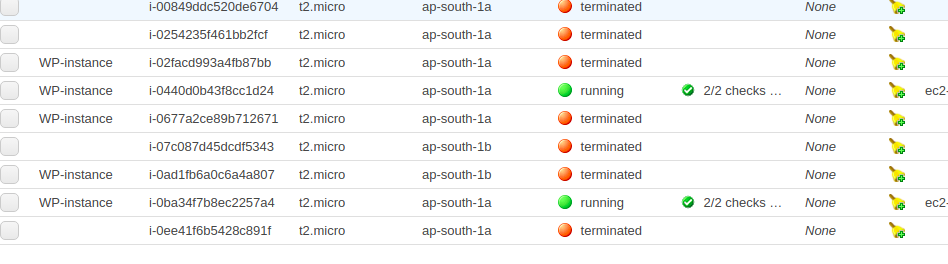
******

******

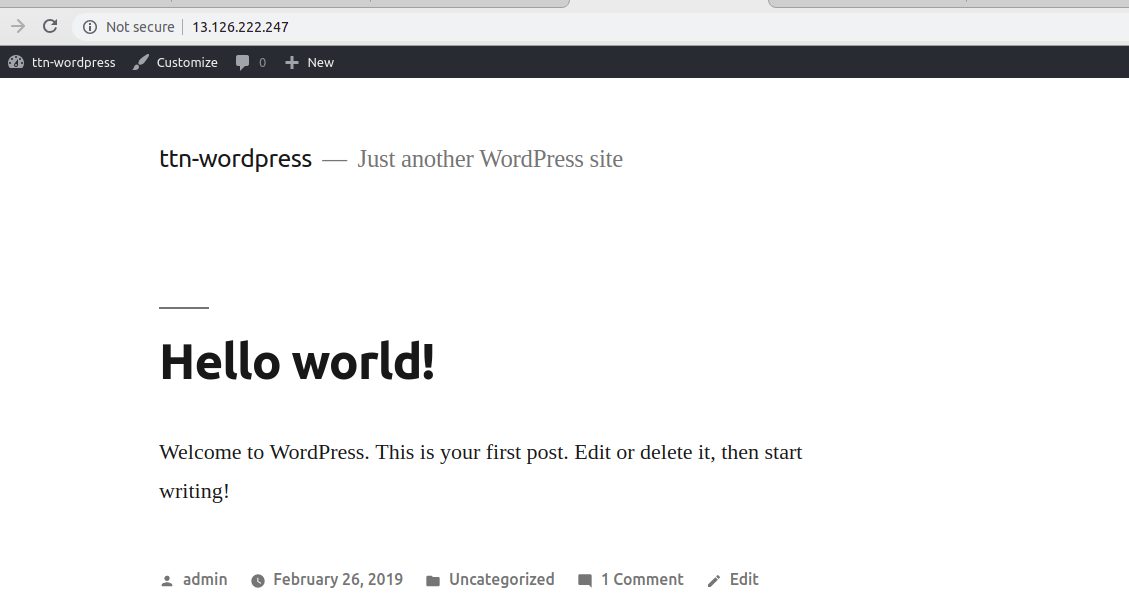
******

1. ***Setup auto -scaling Wordpress application with the Application load balancer. Auto-scaling should be triggered based on CPU usage of EC2 instances.***

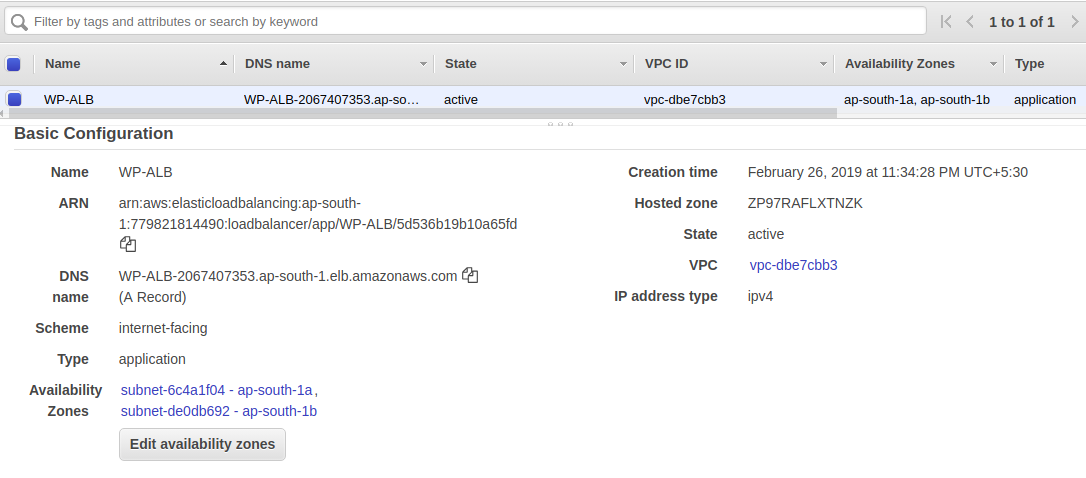
*The load balancer and autoscaling doing their job to handle traffic and CPU usage over instance*

**

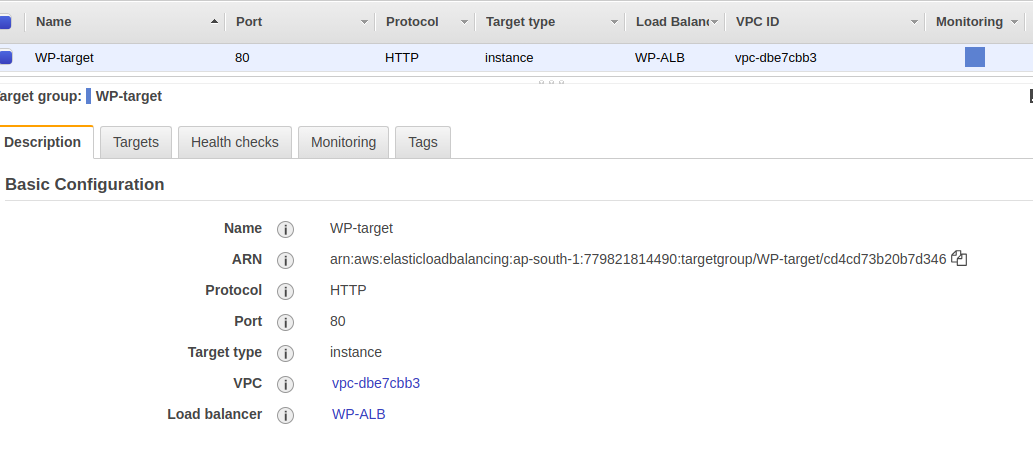
*The home page of the wordpress application*

**

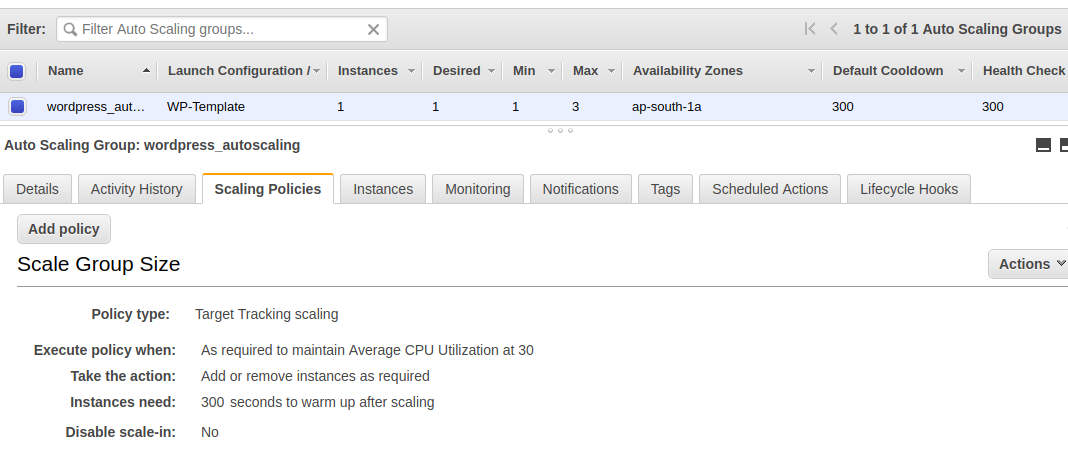
*Load Balancers basic configuration*

**

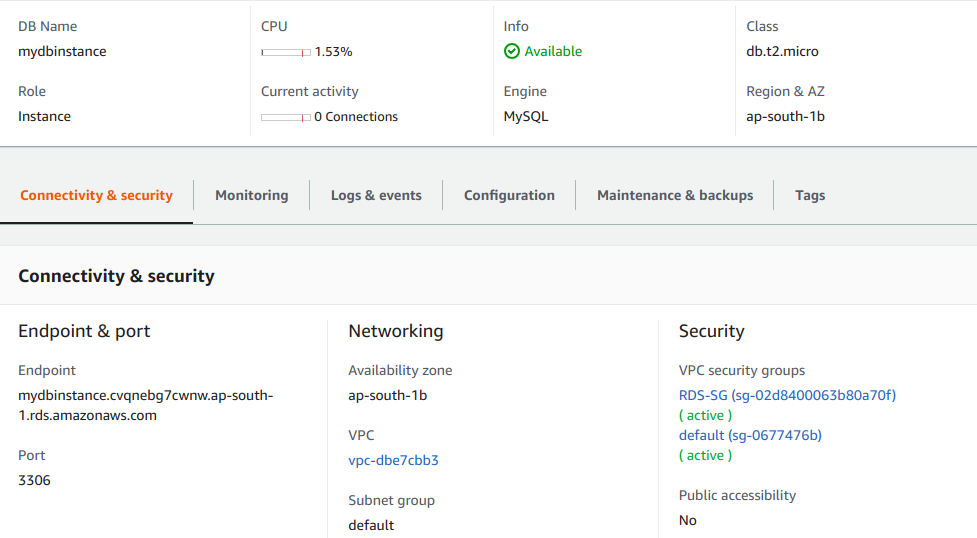
*Target group of load balancer*

**

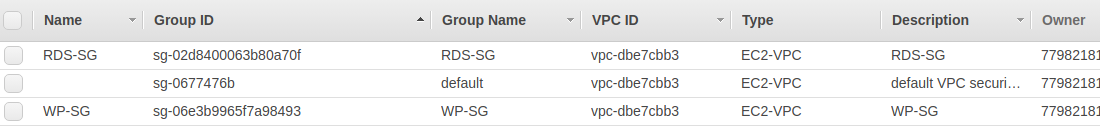
*Group scaling policies*

**

*The RDS instance*

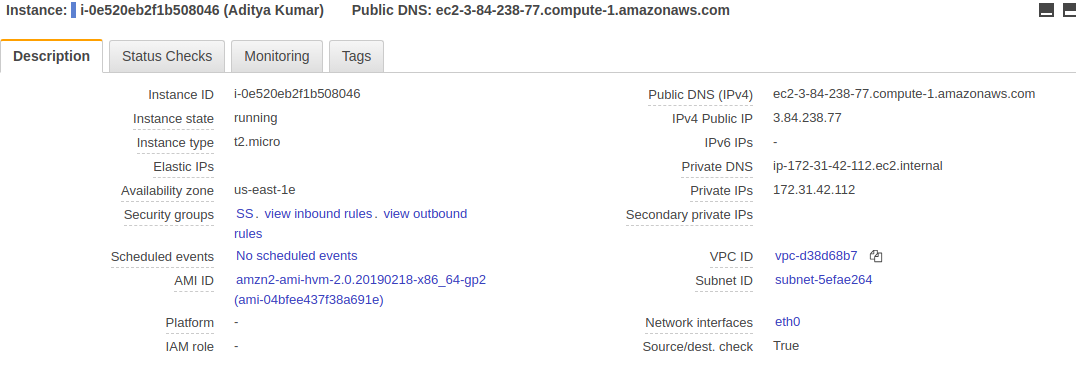
**

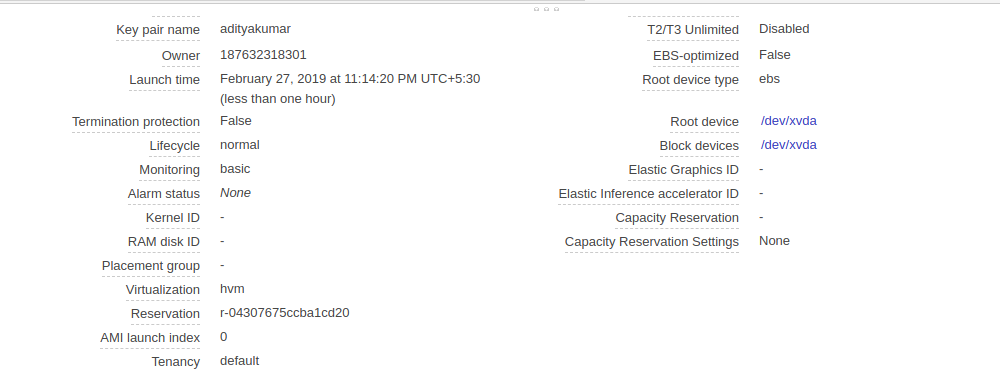
*Security group in VPC*

**

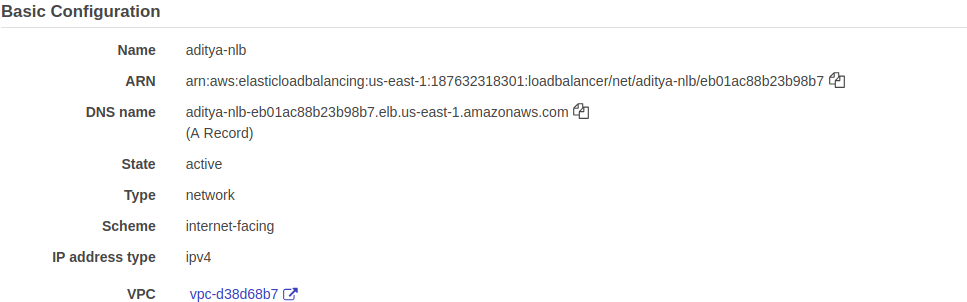
*The whole setup of the wordpress is done on Amazon Linux in EC2 and Database on RDS.This whole setup is done because if we use MySQL database in EC2 instance then it would be difficult for us to put the instance in AUTO SCALING*

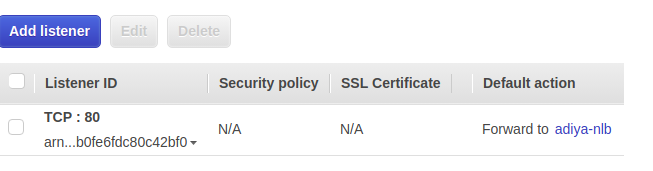
1. ***Use NLB that replaces the ALB in the above setup.***

******

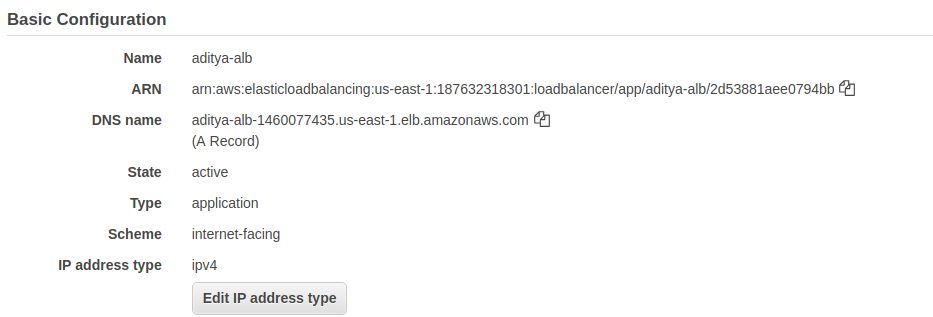
******

***NLB Configuration***

******

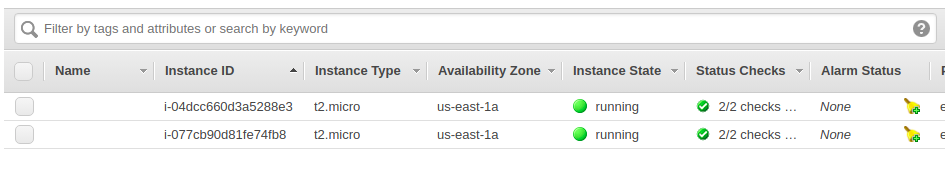
******

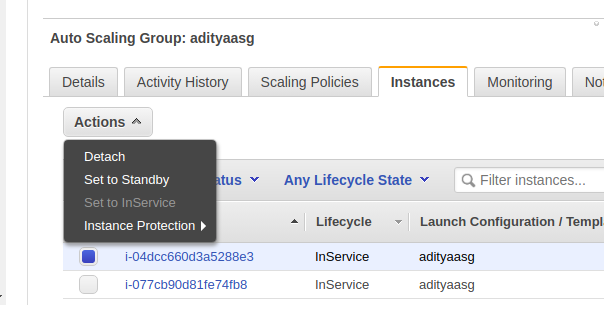
***ALB Configuration***

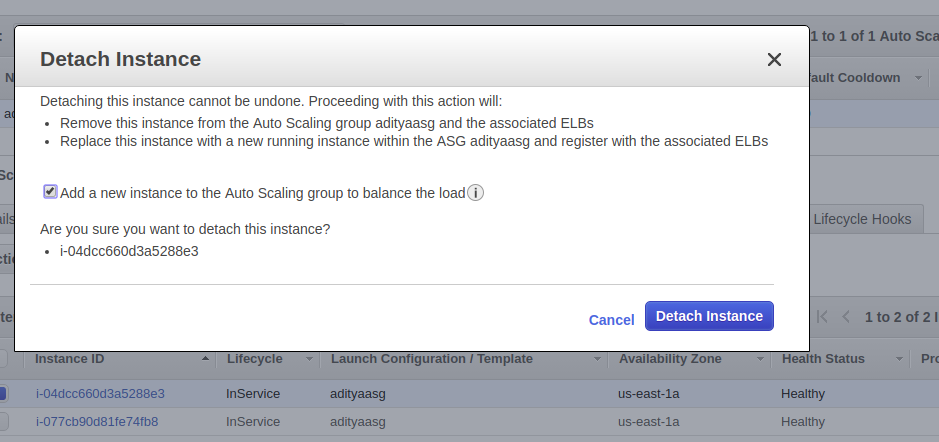
******

*Adding a NLB over the instance as NLB does not requires any security group.What will happen is that NLB will replace the place of ALB and then the Layer 4 of OSI model will be taken care of and then ALB of Layer 7 will be taken care .*

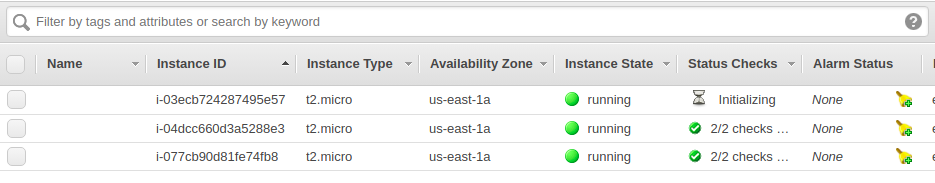
1. ***Take an instance out of the ASG.***

******

******

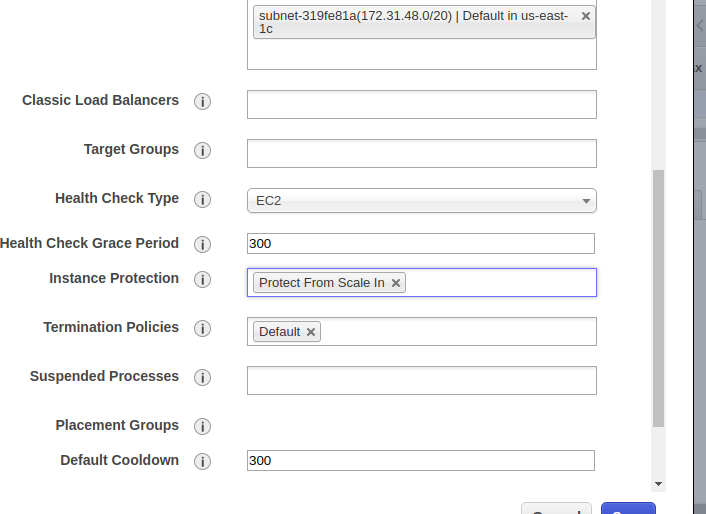
******

***Now after this instance get detached, one more instance is launched because the desired capacity we mentioned =2.***

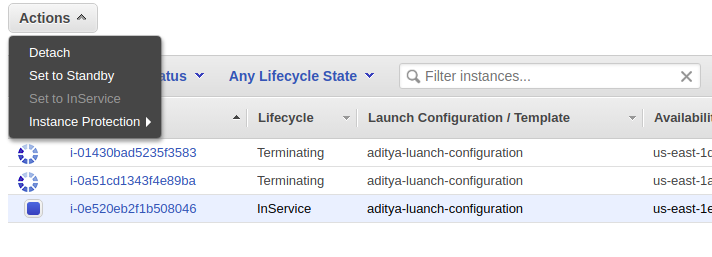
******

1. ***Put scale-in protection on an instance in the ASG.***

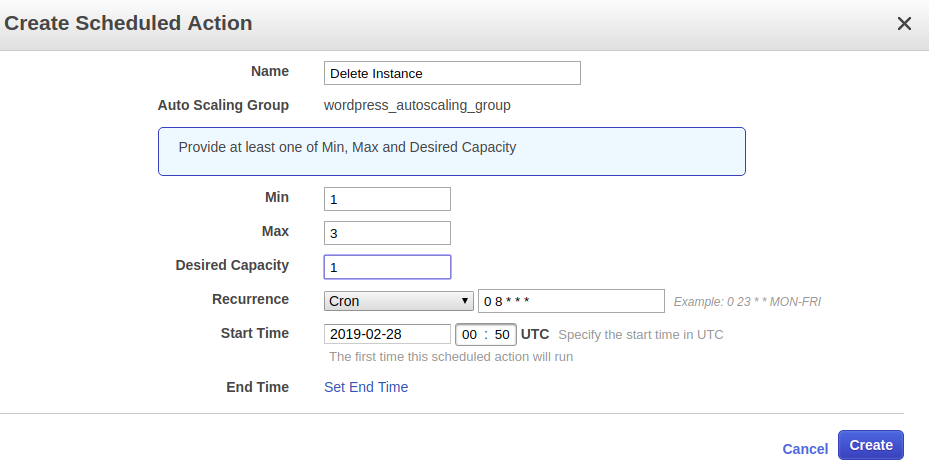
*Scale-in protection can be easily added at the time of creation or after the creation of an Auto Scaling Group*

**

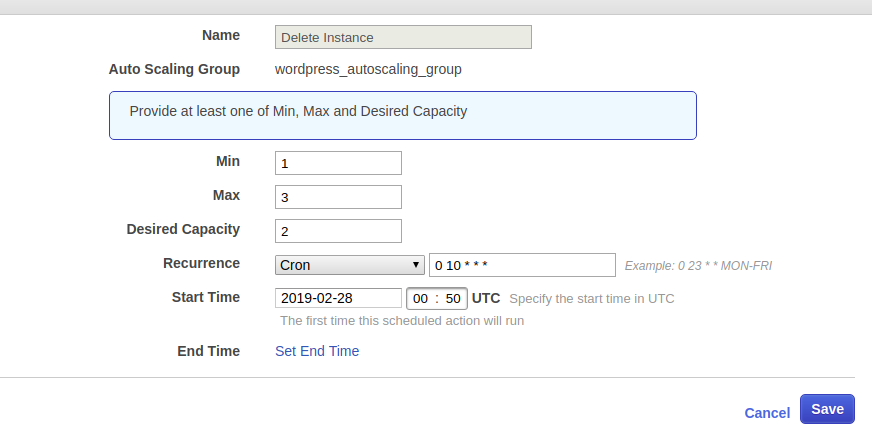
*This screenshot shows the scale-in protection after the creation of the Auto Scaling group .*

**

1. ***Put Schedules in ASG to:***
   1. ***Remove all instances of the ASG at 8 PM***

******

* 1. ***Launch a minimum of 2 instances at 10 AM***

******