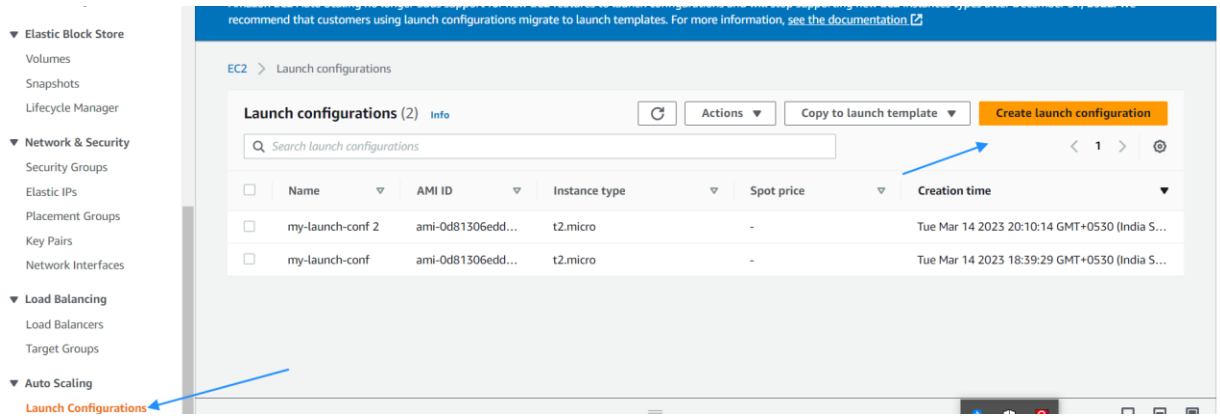
Auto Scalling with load balancer

Steps :-

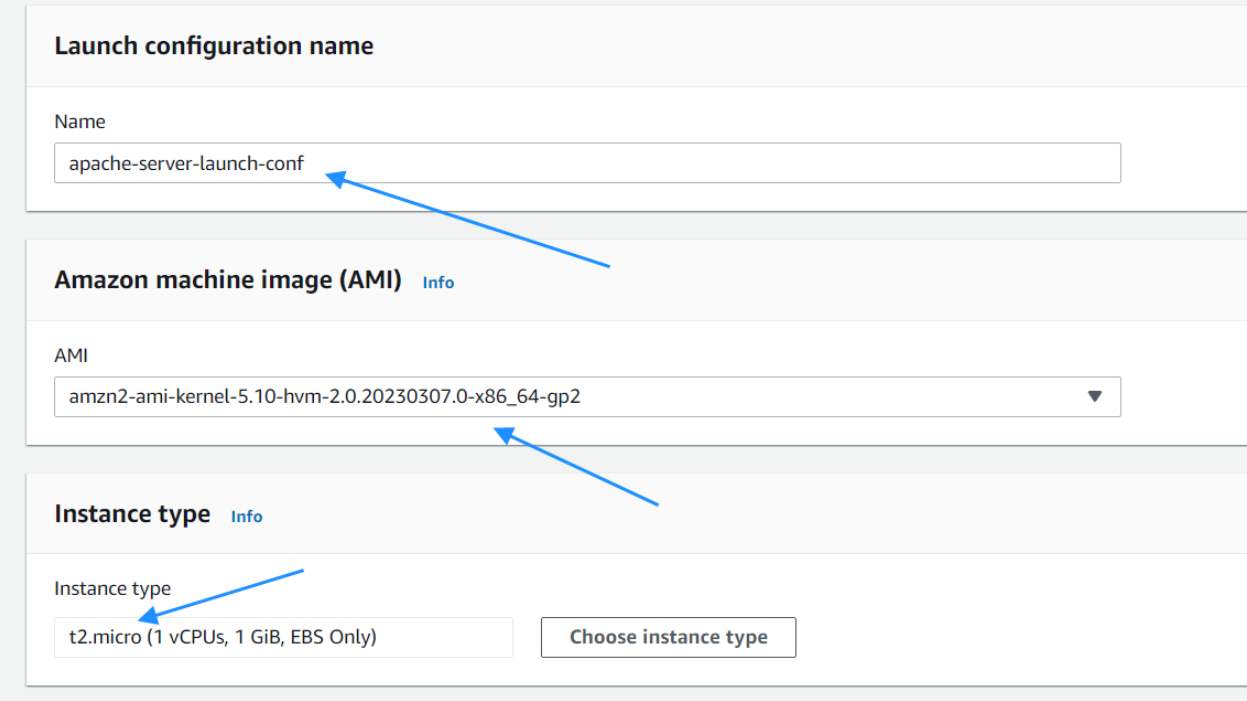
1) Install two instances with ssh and http inbound traffic allow using user data as below

   #!/bin/bash
   sudo -i
   yum install httpd -y
   echo "<h1> this is home page $HOSTNAME</h1>" > /var/www/html/index.html
   systemctl start httpd
   systemctl enable httpd

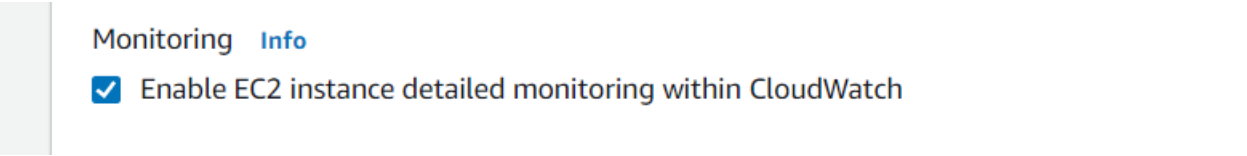
2) Now in load balancer create target group using these two instances
3) Now create load balancer using these home-tg target group
4) Now in auto scaling create launch template same as home instances (ssh and http traffic allow),(same user data like home page)



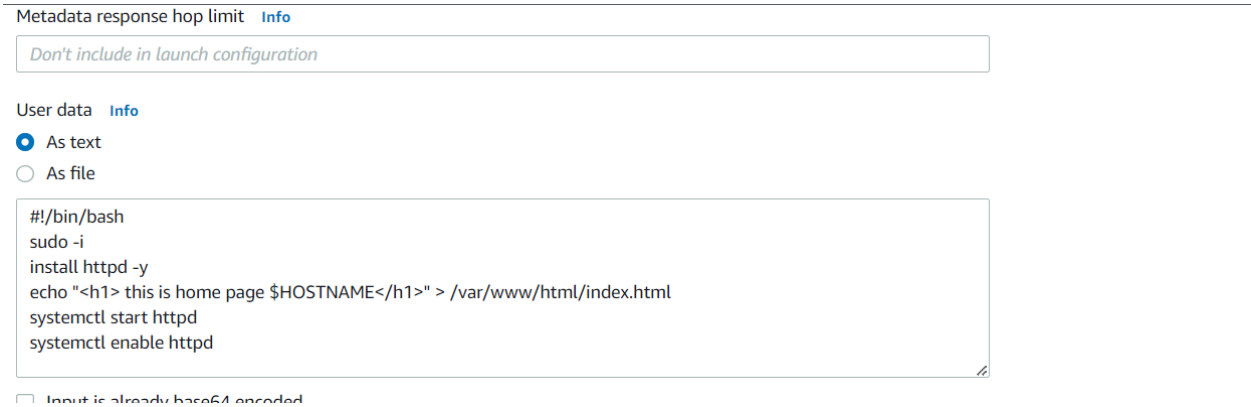
5) Give name to the launch temp → select amazon linux machine image →select instance type t2.micro



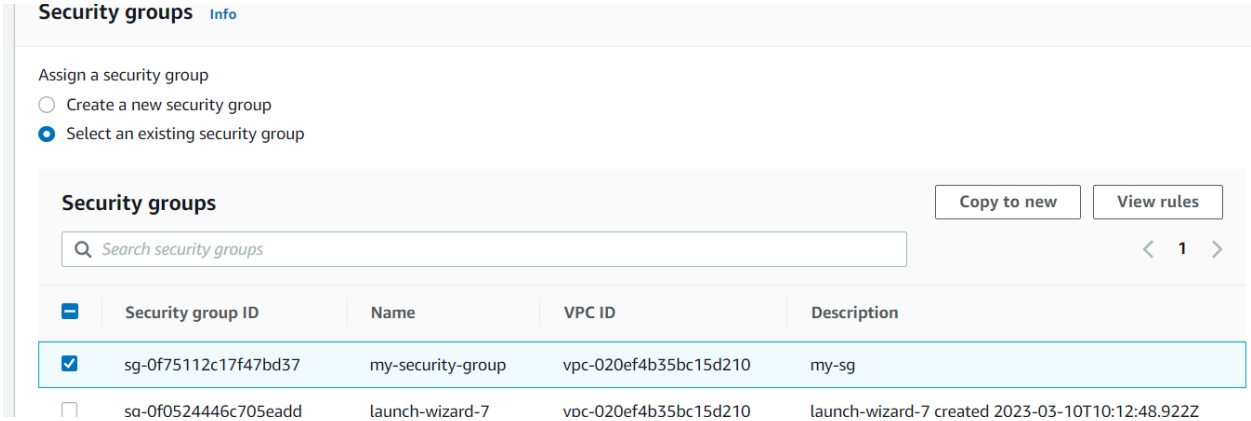
6) Enable instance detailed monitoring



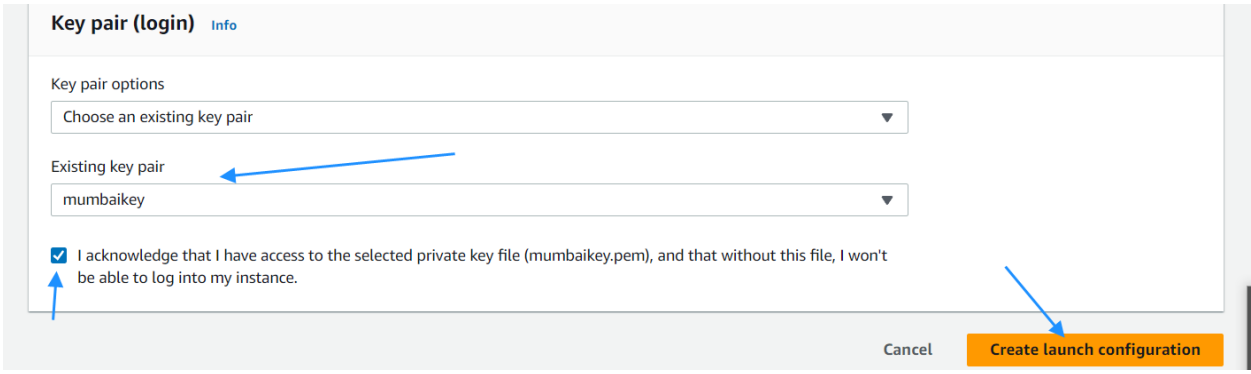
7) In additional user data add user data of home script



8) Select existing security group which has http and ssh inbound traffic allowed

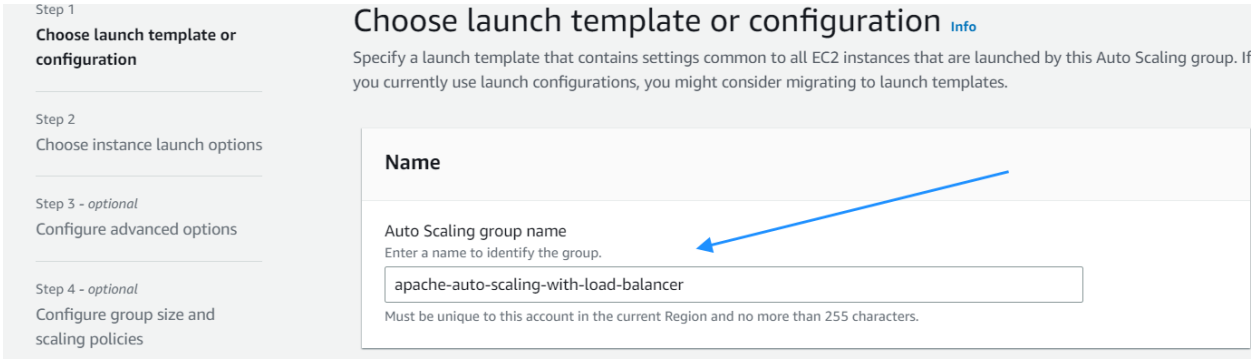


9) Select existing key-pair and create launch configuration

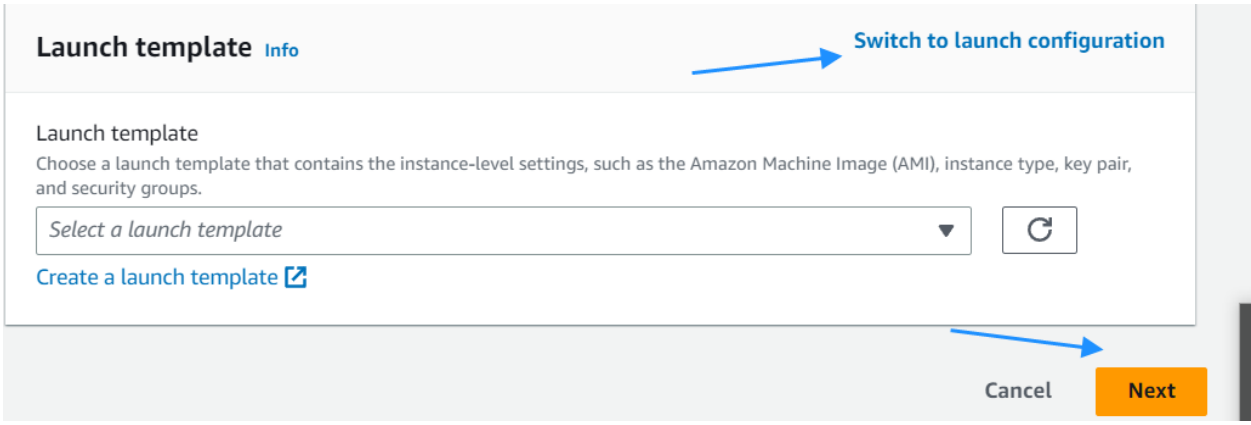


10) Now go to auto scaling → create an auto scaling group

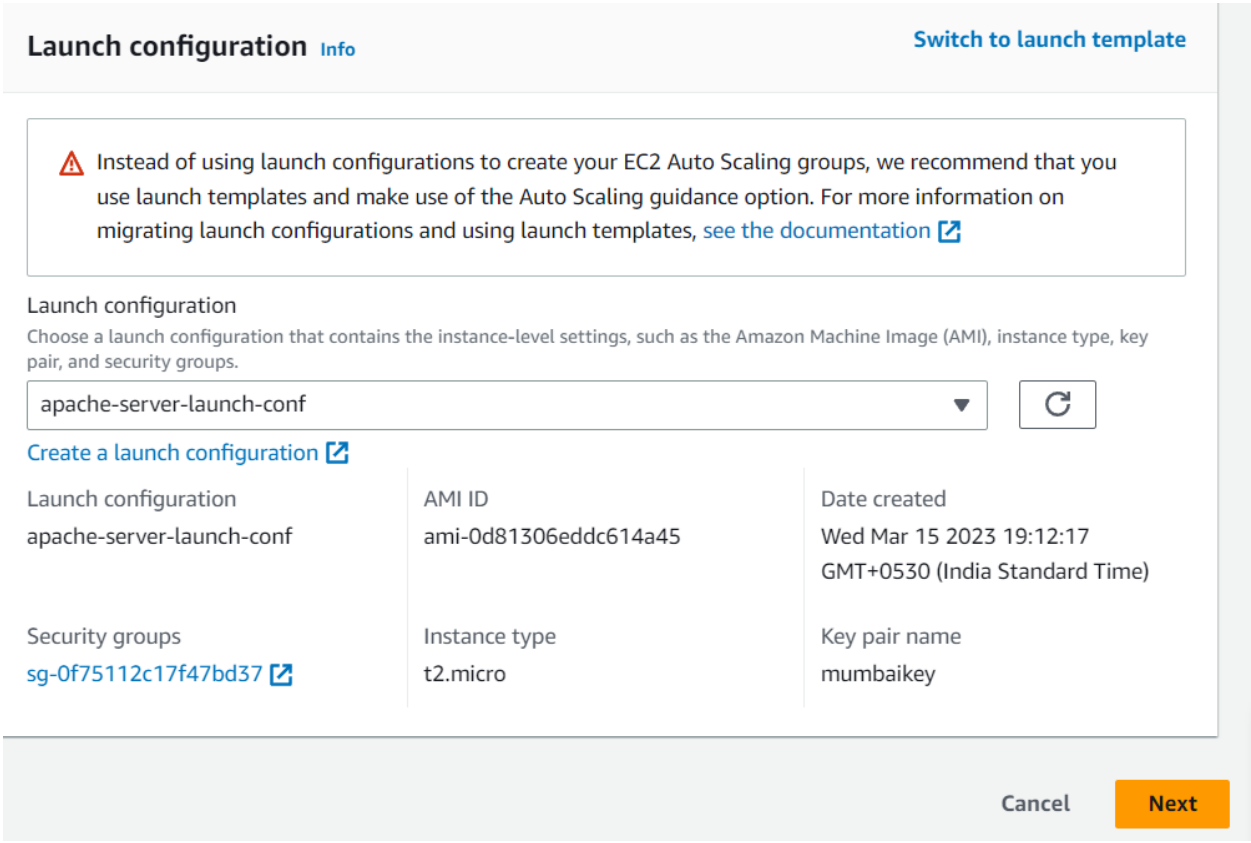
11) Give auto scalling group name



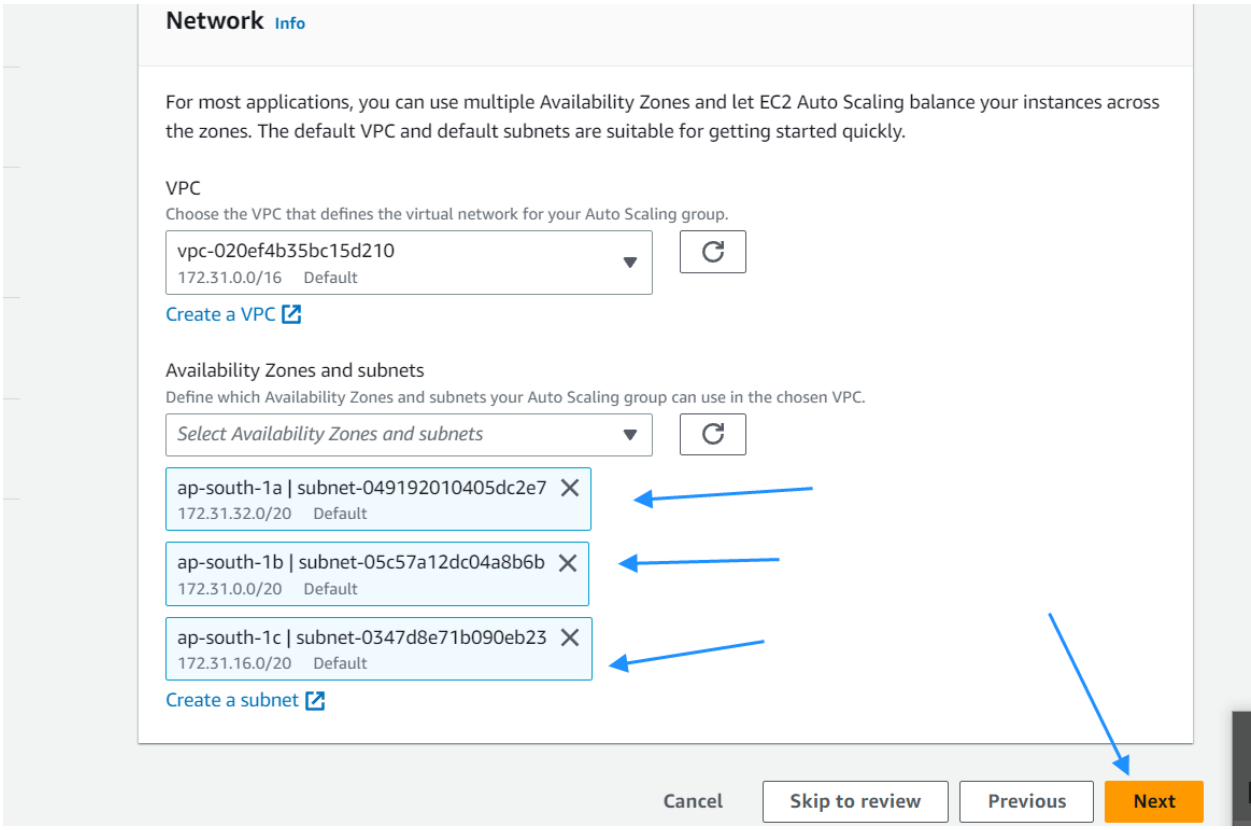
12) Switch to launch configuration



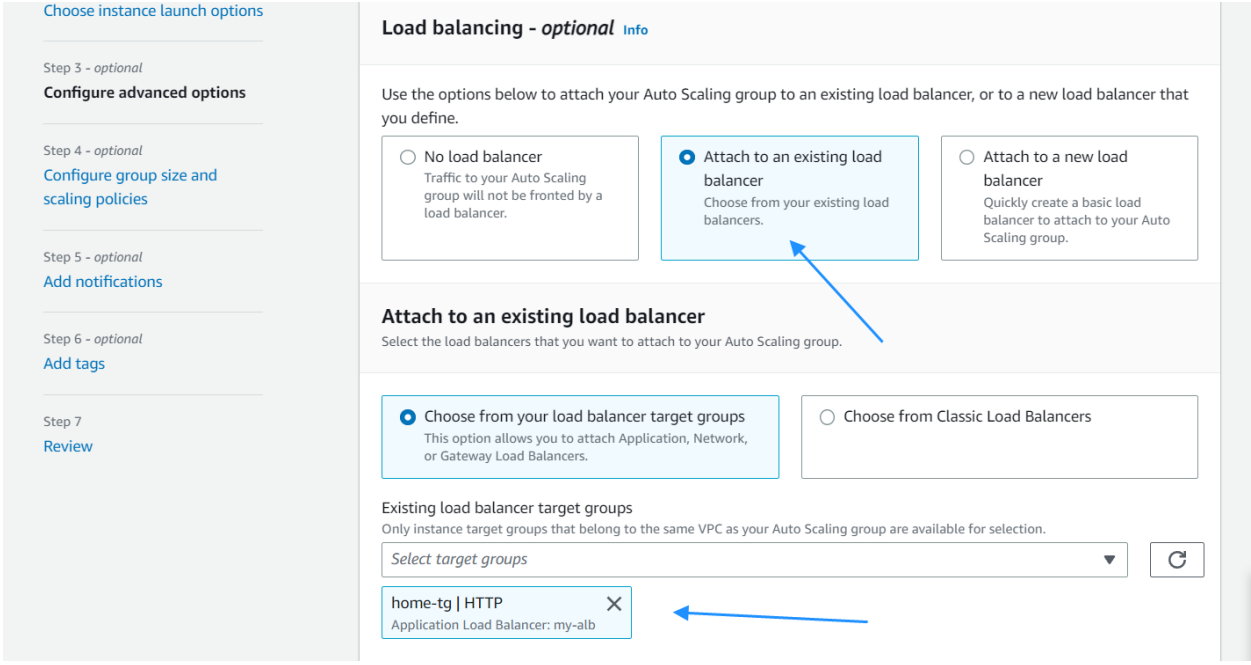
13) Select existing launch template which has home user data in it and click on next



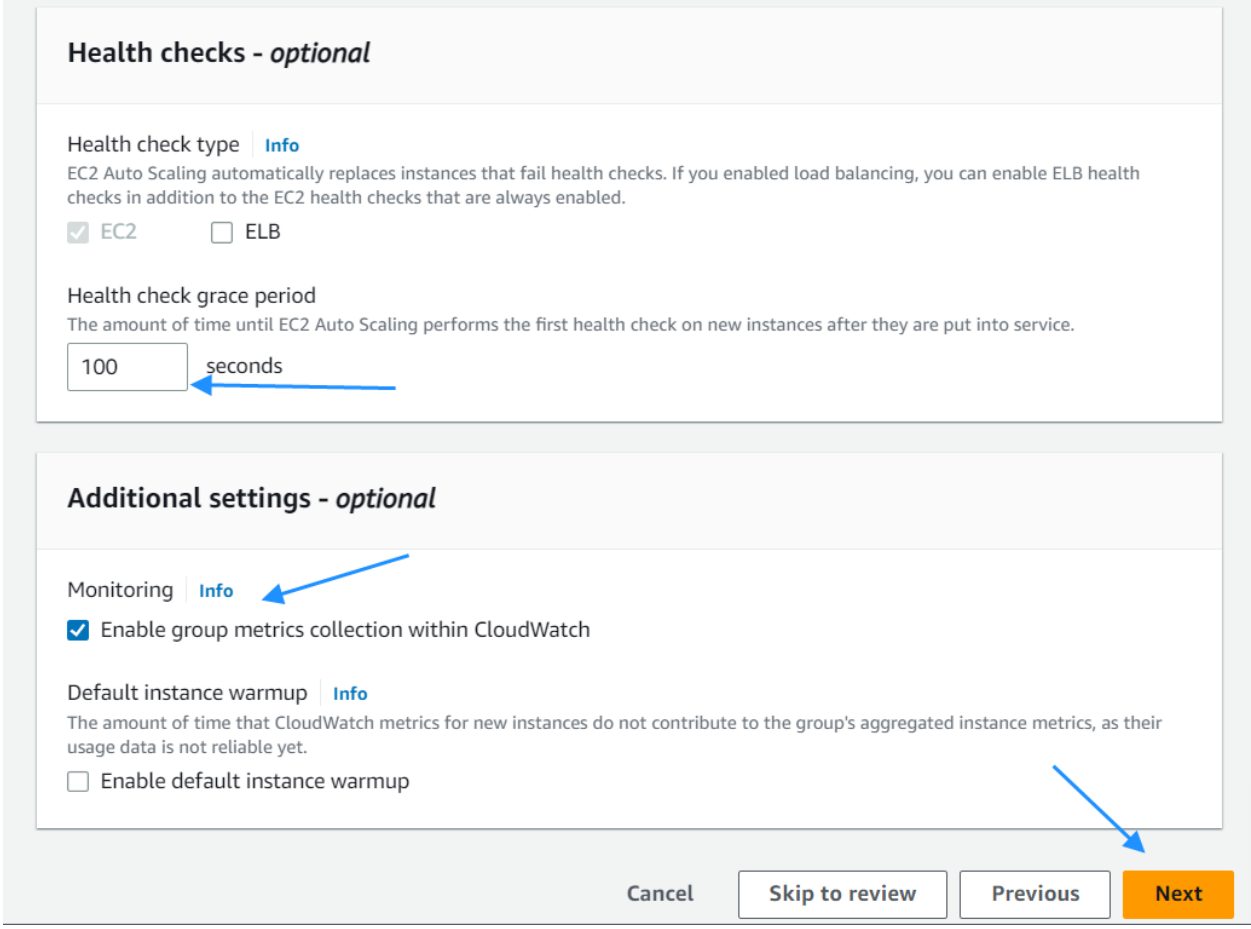
14) Select default VPC and select all subnet and click on next



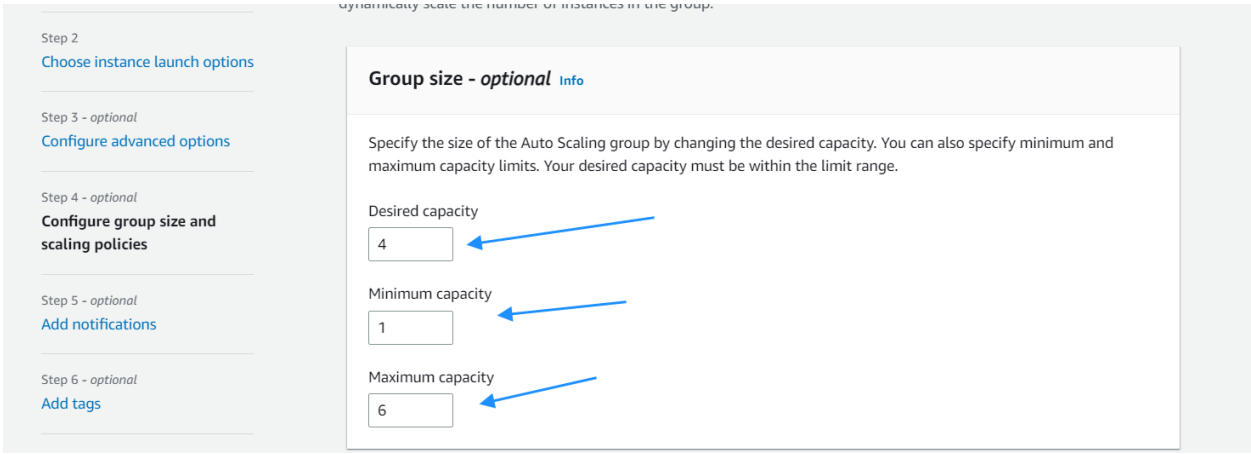
15) Now in step 3 select existing load balancer → select existing target group which is home-TG as shown in fig



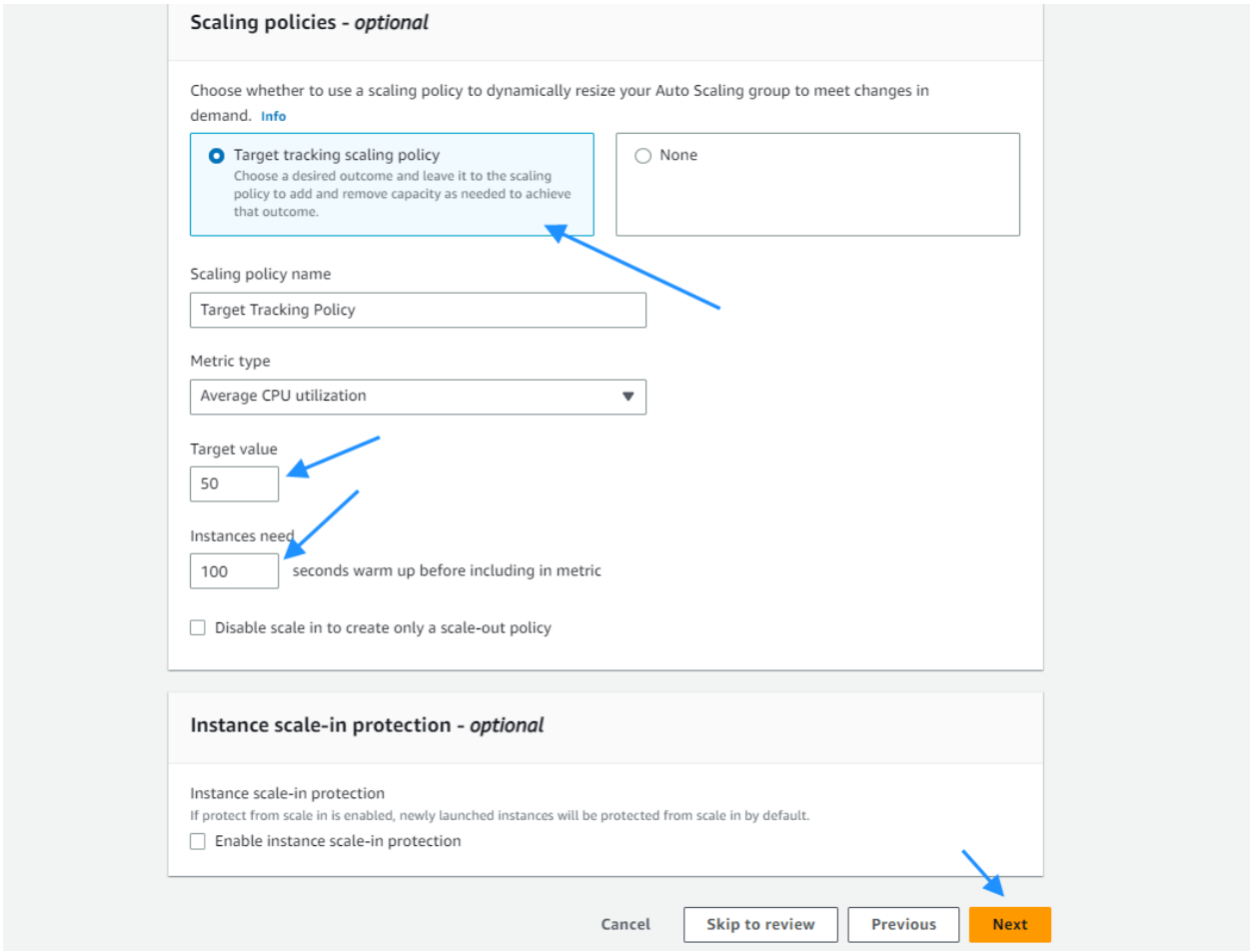
16) Now reduce health check grace period to 100 s → in additional setting enable group matrics collection within cloudwatch and click on next



17) Now in step 4 give desired capacity 4 min capacity 1 max capacity 6



18) Now in scaling policy select target tracking scaling policy→ matrics type **cpu utilization** → target value 50 → instance need 100 and click on next

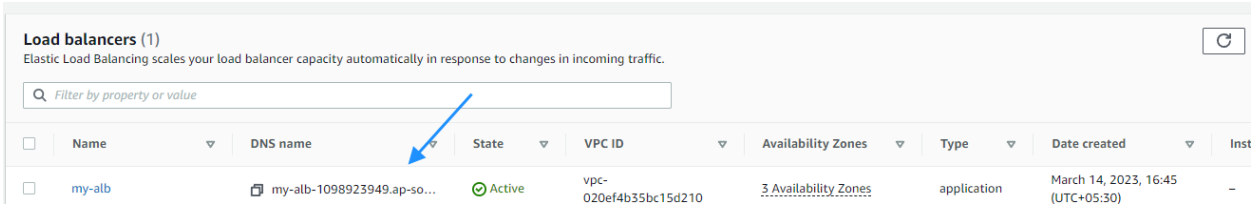


19) Give notification if you want

20) Give tags if needed

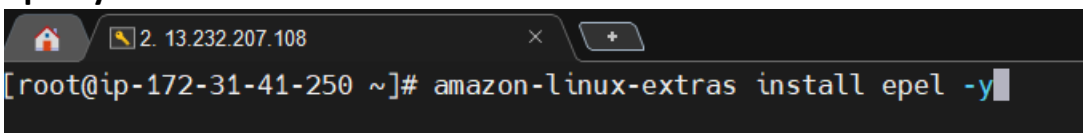
21) Review and create auto scaling group

22) Now go to load balance and copy DNS and hit in new tab. It will distribute traffic across different 4 desired instance

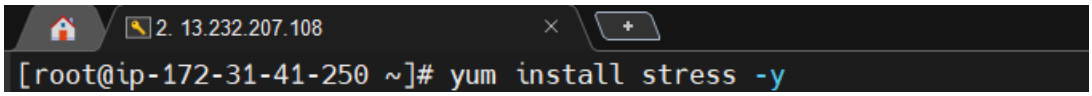


23) Now for auto scaling take ssh of 4 desired instances and increase stress or cpu utilization using following command

24) First install epel repository using **amazon-linux-extras install epel -y**



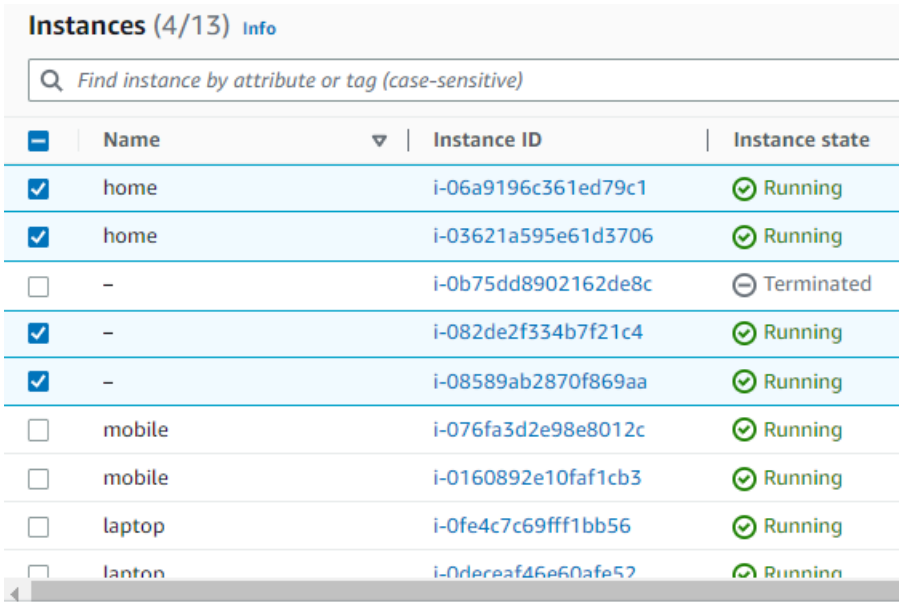
25) Now install stress package using **yum install stress -y**

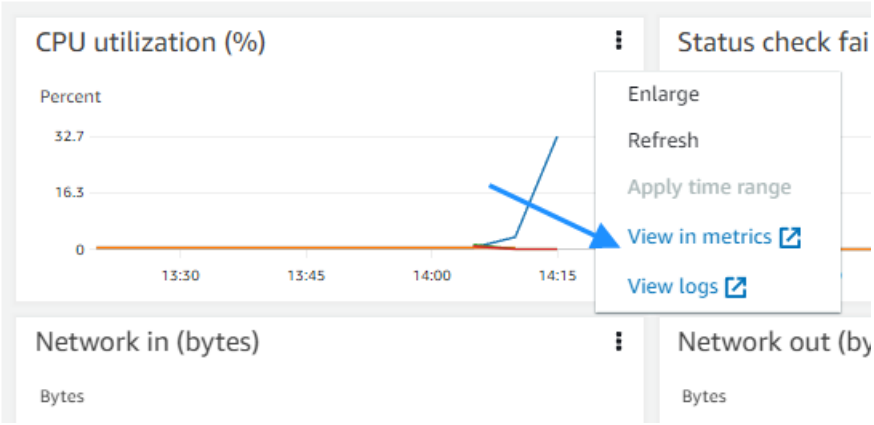


26) Now increase stress or cpu utilization using **stress –cpu 8 –io 4 –vm 2 –vm-bytes 128M –timeout 10M &**

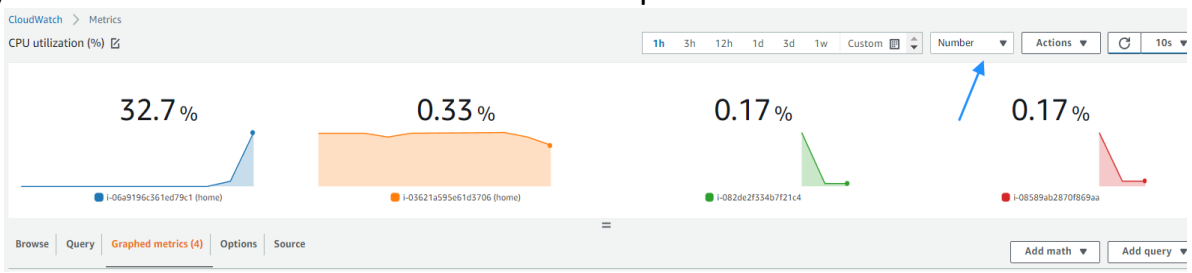
27) Now check stress using top command

28) Now select all desired instances → monitoring click on 3 dots → and click on view in matrics





29) Now click on numbers it will show cpu utilization in %



30) When cpu utilization goes over 50% it will automatically launch instances to maximum capacity and after hitting url of load balancer DNS it will show 6 public ip which is max capacity of instances