**PROJECT SCOPE:**

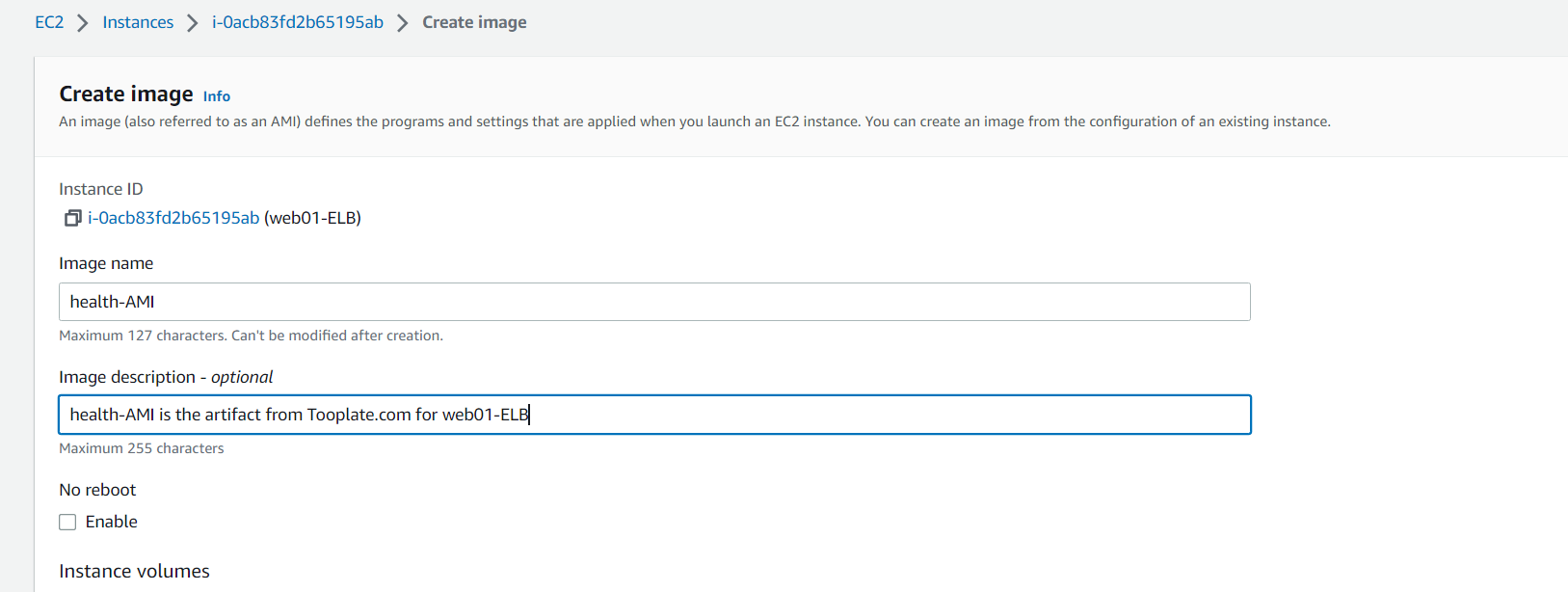
**Setup a Website using an artifact (tooplate.com) on an EC2 instance and create an ELB for the service**

STEPS: (Remember AMI can be used to change region of your Instance)

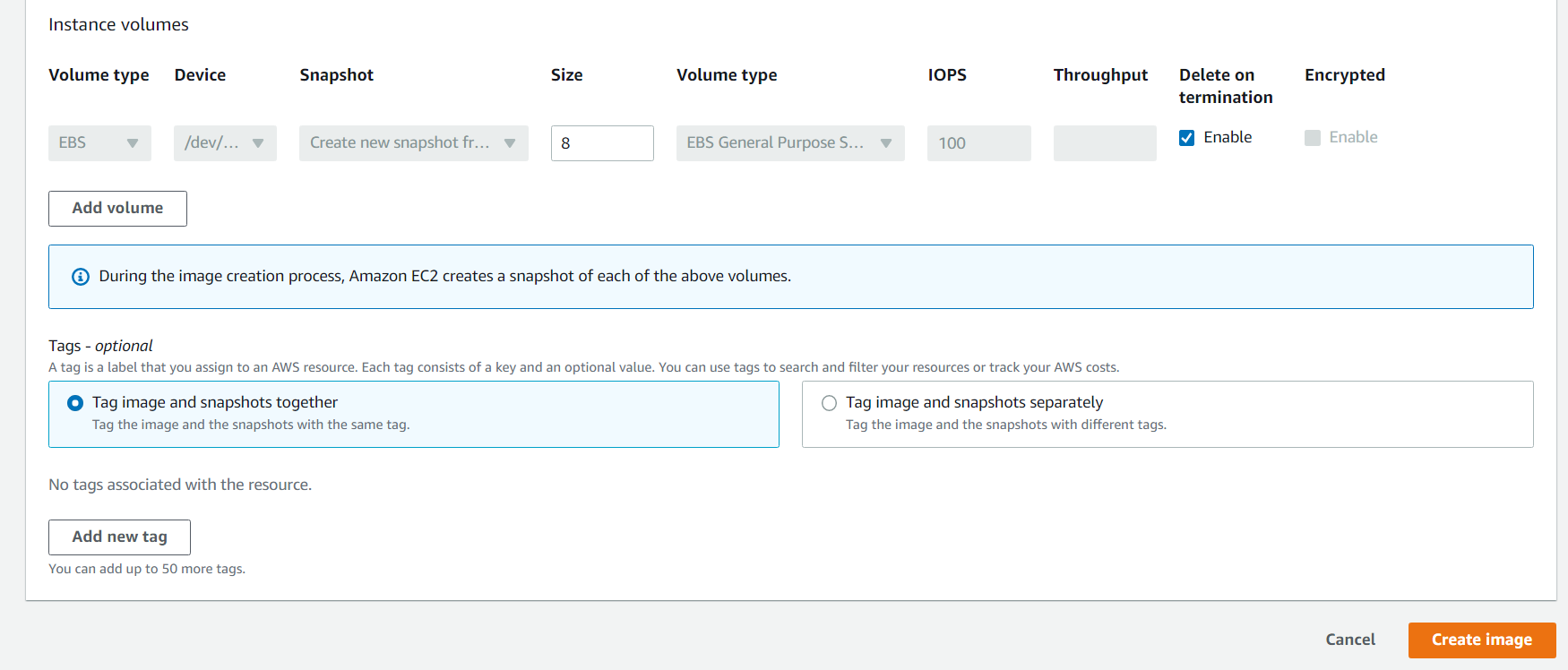
* Login to your AWS console
* Create an EC2 instance, provision it using the script below:

|  |
| --- |
| #!/bin/bash  # Variable Declaration  #PACKAGE="httpd wget unzip"  #SVC="httpd"  URL='https://www.tooplate.com/zip-templates/2098\_health.zip'  ART\_NAME='2098\_health'  TEMPDIR="/tmp/webfiles"  yum --help &> /dev/null  if [ $? -eq 0 ]  then  # Set Variables for CentOS  PACKAGE="httpd wget unzip"  SVC="httpd"  echo "Running Setup on CentOS"  # Installing Dependencies  echo "########################################"  echo "Installing packages."  echo "########################################"  sudo yum install $PACKAGE -y > /dev/null  echo  # Start & Enable Service  echo "########################################"  echo "Start & Enable HTTPD Service"  echo "########################################"  sudo systemctl start $SVC  sudo systemctl enable $SVC  echo  # Creating Temp Directory  echo "########################################"  echo "Starting Artifact Deployment"  echo "########################################"  mkdir -p $TEMPDIR  cd $TEMPDIR  echo  wget $URL > /dev/null  unzip $ART\_NAME.zip > /dev/null  sudo cp -r $ART\_NAME/\* /var/www/html/  echo  # Bounce Service  echo "########################################"  echo "Restarting HTTPD service"  echo "########################################"  systemctl restart $SVC  echo  # Clean Up  echo "########################################"  echo "Removing Temporary Files"  echo "########################################"  rm -rf $TEMPDIR  echo  sudo systemctl status $SVC  ls /var/www/html/  else  # Set Variables for Ubuntu  PACKAGE="apache2 wget unzip"  SVC="apache2"  echo "Running Setup on CentOS"  # Installing Dependencies  echo "########################################"  echo "Installing packages."  echo "########################################"  sudo apt update  sudo apt install $PACKAGE -y > /dev/null  echo  # Start & Enable Service  echo "########################################"  echo "Start & Enable HTTPD Service"  echo "########################################"  sudo systemctl start $SVC  sudo systemctl enable $SVC  echo  # Creating Temp Directory  echo "########################################"  echo "Starting Artifact Deployment"  echo "########################################"  mkdir -p $TEMPDIR  cd $TEMPDIR  echo  wget $URL > /dev/null  unzip $ART\_NAME.zip > /dev/null  sudo cp -r $ART\_NAME/\* /var/www/html/  echo  # Bounce Service  echo "########################################"  echo "Restarting HTTPD service"  echo "########################################"  systemctl restart $SVC  echo  # Clean Up  echo "########################################"  echo "Removing Temporary Files"  echo "########################################"  rm -rf $TEMPDIR  echo  sudo systemctl status $SVC  ls /var/www/html/  fi |

* Create an AMI by selecting the instance, click Actions drop down, click create image
* Remember Snapshot is used to create volumes while AMI is used to create an Instance



* continue the settings professionally and click create AMI



* Go to Images (AMI) section on the left, select your AMI and you can launch it same way you launch instances (Remember to create & use Launch Template using the AMI)
* Create your endpoints, go to Load Balancing on the left pane, click on Target groups
* Click on Target groups (Group of running instances)
* Provide the settings professionally and Health checks part
* Register Targets, pick the instances to link up and click create target
* Go to Load Balancer on the left pane, and click on create
* Select Application Load Balancers, provide settings professionally and click view load balancers
* copy the DNS name from the load balancer created and paste in your url
* If it doesn’t work, check your target group health check or security groups or both.
* For both instance, in the security group allow port 80 from the security group of the load balancer.