Commands for Lambda Containerization:

```
user@user-HP-Pavilion-14-Notebook-PC:~/Downloads$ chmod 400 sk33.pem; providing read access to the owner of sk33.pem
user@user-HP-Pavilion-14-Notebook-PC:~/Downloads$ ssh -i "sk33.pem" ec2-user@ec2-3-82-206-232.compute-1.amazonaws.com; secure login to EC2
```

Building Docker Image on EC2:

```
[ec2-user@ip-172-31-88-107 ~]$ mkdir lambda-docker
[ec2-user@ip-172-31-88-107 ~]$ cd lambda-docker/
[ec2-user@ip-172-31-88-107 lambda-docker]$ ls

[ec2-user@ip-172-31-88-107 lambda-docker]$ vim app.py
[ec2-user@ip-172-31-88-107 lambda-docker]$ ls
app.py
[ec2-user@ip-172-31-88-107 lambda-docker]$ vim Dockerfile
[ec2-user@ip-172-31-88-107 lambda-docker]$ ls
app.py  Dockerfile
[ec2-user@ip-172-31-88-107 lambda-docker]$ vim requirements.txt
[ec2-user@ip-172-31-88-107 lambda-docker]$ ls
app.py  Dockerfile  requirements.txt

[ec2-user@ip-172-31-88-107 home]$ sudo apt-get update

[ec2-user@ip-172-31-88-107 home]$  sudo amazon-linux-extras install docker

[ec2-user@ip-172-31-88-107 home]$ sudo service docker start

[ec2-user@ip-172-31-88-107 home]$ sudo chkconfig  docker on

[ec2-user@ip-172-31-88-107 home]$ sudo yum install -y git

[ec2-user@ip-172-31-88-107 home]$ sudo curl -i -L https://github.com/docker/compose/release/latest/download/docker-compose-$(uname -s)-$(uname -m) -o /usr/local/bin/docker-compose

[ec2-user@ip-172-31-88-107 ~]$ sudo chmod +x /usr/local/bin/docker-compose
[ec2-user@ip-172-31-88-107 ~]$ docker-compose versionDocker Compose version v2.11.0
[ec2-user@ip-172-31-88-107 ~]$ docker images
```

Commands to push build Docker image to ECR:

```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 204298662877.dkr.ecr.us-east-1.amazonaws.com
```

Commands to run above aws ecr get-login-password command successfuly:

```
[ec2-user@ip-172-31-88-107 ~]$ sudo apt-get install gnupg2 pass
[ec2-user@ip-172-31-88-107 lambda-docker]$ sudo amazon-linux-extras install epel -y
[ec2-user@ip-172-31-88-107 ~]$ sudo apt install awscli -y

[ec2-user@ip-172-31-88-107 ~]$ pwd
/home/ec2-user
[ec2-user@ip-172-31-88-107 ~]$ cd .aws
[ec2-user@ip-172-31-88-107 .aws]$ ls
config  credentials
[ec2-user@ip-172-31-88-107 .aws]$ cd cerdentials
-bash: cd: cerdentials: No such file or directory
[ec2-user@ip-172-31-88-107 .aws]$ cat credentials
[default]
aws_access_key_id =
[[ec2-user@ip-172-31-88-107 .aws]$ vi credentials
[ec2-user@ip-172-31-88-107 .aws]$ cat credentials
[default]
aws_access_key_id = AKIAS7EJGGPOY4PQ63GB
aws_secret_access_key = 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
[ec2-user@ip-172-31-88-107 .aws]$ cd ..
[ec2-user@ip-172-31-88-107 ~]$ pwd
/home/ec2-user
```

Commands to build and tag the docker image:

```
docker build -t docker-lambda .

docker tag docker-lambda:latest 204298662877.dkr.ecr.us-east-1.amazonaws.com/docker-lambda:latest
```

Command to push the built docker image to private repository in ECR:

```
docker push 204298662877.dkr.ecr.us-east-1.amazonaws.com/docker-lambda:latest
```

Successful implementation of Docker image Push commands:

```
[ec2-user@ip-172-31-88-107 lambda-docker]$

[ec2-user@ip-172-31-88-107 ~]$ pwd
/home/ec2-user
[ec2-user@ip-172-31-88-107 ~]$ cd .aws
[ec2-user@ip-172-31-88-107 .aws]$ ls
config  credentials

[ec2-user@ip-172-31-88-107 .aws]$ cat credentials
[default]
aws_access_key_id =
[[ec2-user@ip-172-31-88-107 .aws]$ vi credentials
[ec2-user@ip-172-31-88-107 .aws]$ cat credentials
[default]
aws_access_key_id = AKIAS7EJGGPOY4PQ63GB
aws_secret_access_key = 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
```

```
[ec2-user@ip-172-31-88-107 .aws]$ cd ..
[ec2-user@ip-172-31-88-107 ~]$ pwd
/home/ec2-user
[ec2-user@ip-172-31-88-107 lambda-docker]$ aws configure
AWS Access Key ID [None]: AKIAS7EJGGPOY4PQ63GB
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: JSON
[ec2-user@ip-172-31-88-107 ~]$ aws ecr get-login-password --region us-east-1 | docker
login --username AWS --password-stdin 204298662877.dkr.ecr.us-east-
1.amazonaws.com
WARNING! Your password will be stored unencrypted in /home/ec2-
user/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[ec2-user@ip-172-31-88-107 ~]$ docker build -t docker-lambda .
Sending build context to Docker daemon  5.602MB
Step 1/4 : FROM python:alpine
 ---> 4da4c1dc8c72
Step 2/4 : COPY ./content .
 ---> Using cache
 ---> d98f96cdbeac
Step 3/4 : RUN pip install -r requirements.txt
 ---> Using cache
 ---> 205c0cc68ce1
Step 4/4 : CMD python3 bootstrap.py
 ---> Using cache
 ---> a04ad5caa96a
Successfully built a04ad5caa96a
Successfully tagged docker-lambda:latest
[ec2-user@ip-172-31-88-107 ~]$ docker tag docker-lambda:latest
204298662877.dkr.ecr.us-east-1.amazonaws.com/docker-lambda:latest
[ec2-user@ip-172-31-88-107 ~]$ docker push 204298662877.dkr.ecr.us-east-
1.amazonaws.com/docker-lambda:latest
The push refers to repository [204298662877.dkr.ecr.us-east-1.amazonaws.com/docker-
lambda]
67678fa66cbe: Pushed
e2cd8f647283: Pushed
804e16ced35b: Pushed
71aefba622a6: Pushed
765276be336a: Pushed
bdd2dbc0f630: Pushed
994393dc58e7: Pushed
latest: digest:
sha256:f58270bbcb6a6c97c2c28d6fc399f49e9c38715c7387ea03c0f8afa88c5b7ae6 size:
1787
[ec2-user@ip-172-31-88-107 ~]$
```