

1.Launch an EC2 instance and login via ssh furthermore, install httpd(apache).

Launched EC2 instance:

| Instance: i-04292ee2419c181b7 (oshin_ec2) | | |
|---|--|---|
| Details Security Networking Storage Status checks Monitoring Tags | | |
| ▼ Instance summary Info | | |
| Instance ID i-04292ee2419c181b7 (oshin_ec2) | Public IPv4 address 34.229.187.10 open address | Private IPv4 addresses 172.31.1.209 |
| IPv6 address - | Instance state Running | Public IPv4 DNS ec2-34-229-187-10.compute-1.amazonaws.com open address |
| Hostname type IP name: ip-172-31-1-209.ec2.internal | Private IP DNS name (IPv4 only) ip-172-31-1-209.ec2.internal | Elastic IP addresses - |
| Answer private resource DNS name - | Instance type t2.micro | AWS Compute Optimizer finding User: am:aws:sts::949263681218:assumed-role/AWSReservedSSO_training_account_a7c561584130c139/oshingansi@lftecthnology.com is not authorized to perform: compute-optimizer:GetEnrollmentStatus on resource: * because no identity-based policy allows the compute-optimizer:GetEnrollmentStatus action Retry |
| Auto-assigned IP address 34.229.187.10 [Public IP] | VPC ID vpc-00f8984b2df206c70 (Default VPC) open address | |

Login via ssh

```
C:\Users\Oshin>ssh -i oshin_ec2.pem ec2-user@34.229.187.10
The authenticity of host '34.229.187.10 (34.229.187.10)' can't be established.
ECDSA key fingerprint is SHA256:FkbtbH0pUTx0IhPNReYPiBSzn6k/iAxRuicSqnfGqgNI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '34.229.187.10' (ECDSA) to the list of known hosts.

  __|  __|_ )
  _| (    /   Amazon Linux 2 AMI
 ---|\\___|___|

https://aws.amazon.com/amazon-linux-2/
2 package(s) needed for security, out of 10 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-1-209 ~]$ sudo yum update
```

Install httpd

```
Complete!
[ec2-user@ip-172-31-1-209 ~]$ sudo yum install httpd\
^
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.54-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd filesystem = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd filesystem for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
--> Package generic-logos-httpd.noarch 0:18.0.0-4.amzn2 will be installed
--> Package httpd filesystem.noarch 0:2.4.54-1.amzn2 will be installed
--> Package httpd-tools.x86_64 0:2.4.54-1.amzn2 will be installed
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.15.19-1.amzn2.0.1 will be installed
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package Arch Version Repository Size
=====
Installing:
httpd x86_64 2.4.54-1.amzn2 amzn2-core 1.4 M
Installing for dependencies:
apr x86_64 1.7.0-9.amzn2 amzn2-core 122 k
apr-util x86_64 1.6.1-5.amzn2.0.2 amzn2-core 99 k
apr-util-bdb x86_64 1.6.1-5.amzn2.0.2 amzn2-core 19 k
generic-logos-httpd noarch 18.0.0-4.amzn2 amzn2-core 19 k
httpd filesystem.noarch 2.4.54-1.amzn2 amzn2-core 24 k
httpd-tools x86_64 2.4.54-1.amzn2 amzn2-core 88 k
mailcap noarch 2.1.41-2.amzn2 amzn2-core 31 k
mod_http2 x86_64 1.15.19-1.amzn2.0.1 amzn2-core 149 k
Transaction Summary
=====
Install 1 Package (+8 Dependent packages)

Total download size: 1.9 M
Installed size: 5.2 M
```

```
Transaction Summary
=====
Install 1 Package (+8 Dependent packages)

Total download size: 1.9 M
Installed size: 5.2 M
Is this ok [y/d/n/?]: y
Downloading packages:
(1/9): apr-1.7.0-9.amzn2.x86_64.rpm | 122 kB 00:00:00
(2/9): apr-util-1.6.1-5.amzn2.0.2.x86_64.rpm | 99 kB 00:00:00
(3/9): apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64.rpm | 19 kB 00:00:00
(4/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm | 19 kB 00:00:00
(5/9): httpd filesystem-2.4.54-1.amzn2.noarch.rpm | 24 kB 00:00:00
(6/9): httpd-2.4.54-1.amzn2.x86_64.rpm | 1.4 MB 00:00:00
(7/9): httpd-tools-2.4.54-1.amzn2.x86_64.rpm | 88 kB 00:00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00:00
(9/9): mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm | 149 kB 00:00:00
-----
Total 9.6 MB/s | 1.9 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : apr-1.7.0-9.amzn2.x86_64 1/9
Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64 3/9
Installing : httpd-tools-2.4.54-1.amzn2.x86_64 4/9
Installing : httpd filesystem-2.4.54-1.amzn2.noarch 5/9
Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 6/9
Installing : mailcap-2.1.41-2.amzn2.noarch 7/9
Installing : mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
Installing : httpd-2.4.54-1.amzn2.x86_64 9/9
Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9
Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
Verifying : httpd-tools-2.4.54-1.amzn2.x86_64 3/9
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 4/9
Verifying : httpd-2.4.54-1.amzn2.x86_64 5/9
Verifying : mailcap-2.1.41-2.amzn2.noarch 6/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 7/9
Verifying : httpd filesystem-2.4.54-1.amzn2.noarch 8/9
Verifying : apr-1.7.0-9.amzn2.x86_64 9/9

Installed:
httpd.x86_64 0:2.4.54-1.amzn2

Dependency Installed:
apr.x86_64 0:1.7.0-9.amzn2 apr-util.x86_64 0:1.6.1-5.amzn2.0.2 apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 generic-logos-httpd.noarch 0:18.0.0-4.amzn2 httpd filesystem.noarch 0:2.4.54-1.amzn2
httpd-tools.x86_64 0:2.4.54-1.amzn2 mailcap.noarch 0:2.1.41-2.amzn2 mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[ec2-user@ip-172-31-1-209 ~]$
```

2.Create a Dockerfile for node js and react and push the created docker images to the docker hub repository.

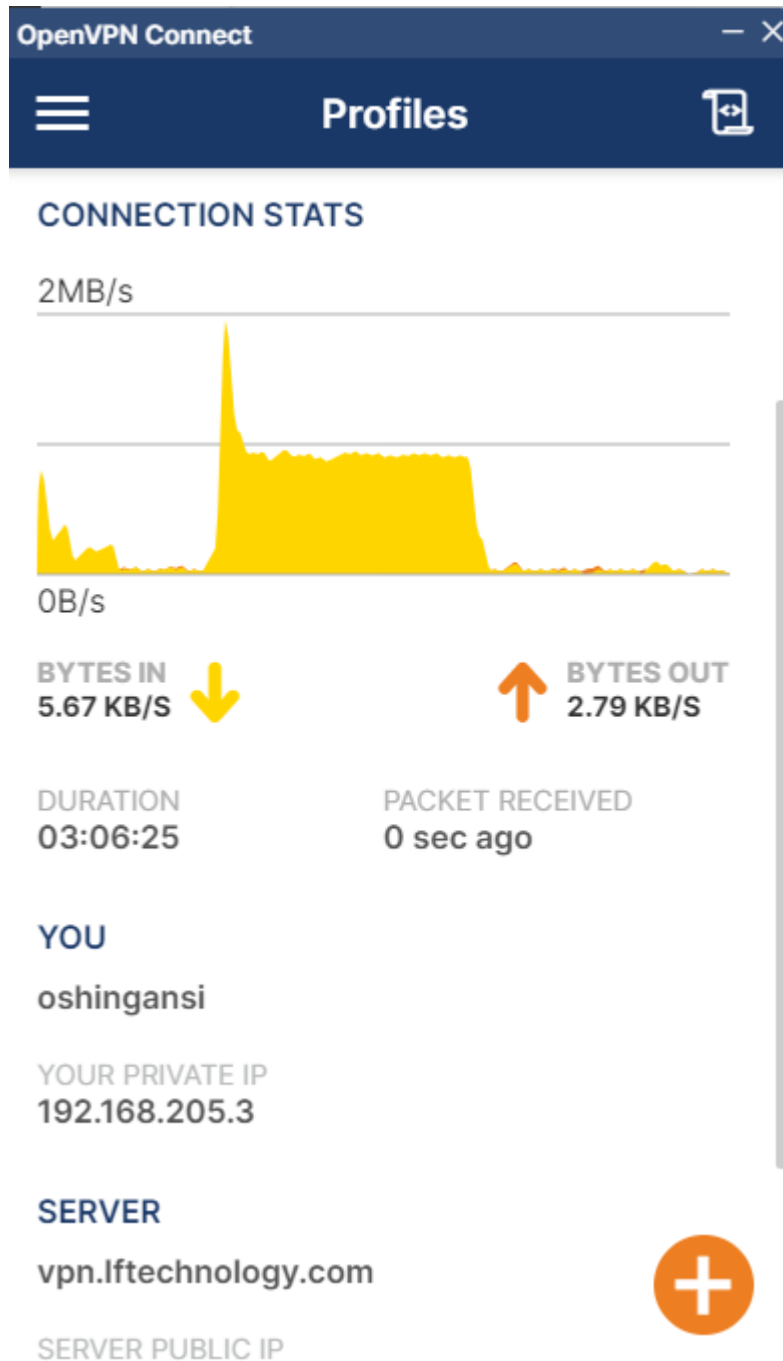
Docker File:

```
Dockerfile
1  # pull official base image
2  FROM node:latest
3
4  # set working directory
5  WORKDIR /app
6
7  # add `/app/node_modules/.bin` to $PATH
8  ENV PATH /app/node_modules/.bin:$PATH
9
10 # install app dependencies
11 COPY package.json ./
12 RUN npm install -g npm@8.16.0
13 RUN npm install
14 RUN npm install react-scripts@3.4.1 -g --silent
15
16 # add app
17 COPY . ./
18
19 # start app
20 CMD ["npm", "start"]
21
```

Docker Commands:

- 1.docker build -t vas:dev .
2. docker run -d -it --rm -p 3001:3000 vas:dev
3. docker tag vas:dev oshin136/vas
4. docker push

3.Setup office vpn and access self learning platforms.



4. List down common HTTP status code (only 10) with means.

- A. Status Code 200 :
This is the standard “OK” status code for a successful HTTP request. The response that is returned is dependent on the request.
- B. Status Code 201:
This is the status code that confirms that the request was successful and, as a result, a new resource was created. Typically, this is the status code that is sent after a POST/PUT request.
- C. Status Code 204 :
This status code confirms that the server has fulfilled the request but does not need to return information.
- D. Status Code 304 :
This status code is used for browser caching. If the response has not been modified, the client/user can continue to use the same response/cached version.
- E. Status Code 400 :
The server cannot understand and process a request due to a client error. Missing data, domain validation, and invalid formatting are some examples that cause the status code 400 to be sent.
- F. Status Code 401 :
This status code request occurs when authentication is required but has failed or not been provided.
- G. Status Code 403 :
Very similar to status code 401, a status code 403 happens when a valid request was sent, but the server refuses to accept it. This happens if a client/user requires the necessary permission or they may need an account to access the resource. Unlike a status code 401, authentication will not apply here.
- H. Status Code 404 :
The most common status code the average user will see. A status code 404 occurs when the request is valid, but the resource cannot be found on the server.
- I. Status Code 409:
A status code 409 is sent when a request conflicts with the current state of the resource. This is usually an issue with simultaneous updates, or versions, that conflict with one another.

J. Status Code 500 :

Another one of the more commonly seen status codes by users, the 500 series codes are similar to the 400 series codes in that they are true error codes. The status code 500 happens when the server cannot fulfil a request due to an unexpected issue. Web developers typically have to comb through the server logs to determine where the exact issue is coming from.

5.Find network id and Broadcast for 150.10.20.30.

Network id: 150.10.20.0

Broadcast Id: 150.10.20.255