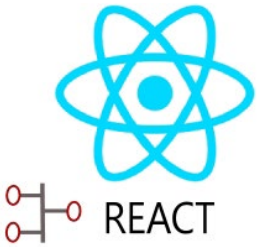

Usage instructions:

- Launch the product via 1-click. **Please wait until** the instance passes **all** status checks and is running. You can connect using your Amazon private key and '**ubuntu**' login via your SSH client.

- To update software, use: **sudo apt-get update**

---

1. After logging into your server, run the following commands:

   sudo su

2. Next, log into the directory:

   cd awesome-project

3. Start React with the following commands:

   npm start

   Be patient while react starts

4. Open your browser to see that the React page is working:
   - http://Public IPv4 address**:3000**

Note:- *If port 3000 is busy with another process, the app will start in port 3001 or any other port available.*

- Next press "CTRL+C" to stop the application

Creating a Test App:

- In the directory "awesome-project" create a test apt:

  npx create-react-app my-app

*(Note: I am using "**my-app**". Replace any name you want)*

- Using an editor. Change text "Learn React" to "Yes it works". **Save and close file.**

  sudo nano src/App.js

- Restart react

npm start

Visit: http://Public IPv4 address:3000 (for changes made to page)

Other useful starter commands:

create-react-app --help

npm start (starts the development server)

npm run build (Bundles the app into static files for production)

npm test (starts the test runner)

npm run eject (removes this tool and copies build dependencies)

Location directories:

nano /lib/systemd/system/react.service

nano /etc/nginx/conf.d/react.conf

Great resources: https://reactjs.org/docs/getting-started.html

https://reactjs.org/tutorial/tutorial.html

AWS Data

- Data Encryption Configuration: This solution does not encrypt data within the running instance.
- User Credentials are stored: /root/.ssh/authorized_keys & /home/ubuntu/.ssh/authorized_keys
- Monitor the health:
    - Navigate to your Amazon EC2 console and verify that you're in the correct region.
    - Choose Instance and select your launched instance.
    - Select the server to display your metadata page and choose the Status checks tab at the bottom of the page to review if your status checks passed or failed.

Extra Information: (Optional)

Allocate Elastic IP

To ensure that your instance **keeps its IP during restarts** that might happen, configure an Elastic IP. From the EC2 console:

1. Select ELASTIC IPs.
2. Click on the ALLOCATE ELASTIC IP ADDRESS.
3. Select the default (Amazon pool of IPv4 addresses) and click on ALLOCATE.
4. From the ACTIONS pull down, select ASSOCIATE ELASTIC IP ADDRESS.
5. In the box that comes up, note down the Elastic IP Address, which will be needed when you configure your DNS.
6. In the search box under INSTANCE, click and find your INSTANCE ID and then click ASSOCIATE.
7. Your instance now has an elastic IP associated with it.
8. For additional help: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html

Using Your Own Domain Name

1. You will need to configure your DNS entry for the new host server you created.
2. Change your domain's "Record Set" value to point to your new instance. Change and copy your "IPv4 Public IP" into the "A" type value.
3. For additional help: https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/rrsets-working-with.html

Deploy a Load Balancer

1. https://docs.aws.amazon.com/elasticloadbalancing/latest/userguide/load-balancer-getting-started.html

Deploy a SSL for a Domain Name

1. Install AWS Certificate: https://docs.aws.amazon.com/elasticloadbalancing/latest/classic/ssl-server-cert.html

   or

2. Installing Cerbot: https://certbot.eff.org/instructions