

Deploy a React application on EC2 through Jenkins And Send Email Notification

STEP 1 :

Install jenkins and nodejs on your system.

Configure Credentials under manage jenkins to add github login details and also create a credential for EC2 SSH connection

- Under Credential,click on Add credential and select username with password.
- Username should be the username of your github.
- Add the password for the github .
- Give an ID name and description .

Update credentials

Scope ?
Global (Jenkins, nodes, items, all child items, etc) ▼

Username ?
vikaskumarp1061

☐ Treat username as secret ?

Password ?
Concealed Change Password

ID ?
git_login

Description ?
this is a git login credential with username and token password

Save

- Below image is the credential for ec2 ssh connection.
- Add the IP-Address of instance under ID (not necessary)
- Provide the username as Ubuntu
- Paste the pem file contents inside the “**Private key**” option.

Update credentials

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

ID ?

54.237.247.207

Description ?

react ec2

Username

ubuntu

☐ Treat username as secret ?

Private Key

☒ Enter directly

Key



Concealed for Confidentiality

Replace

STEP 2 :

Configure Systems under manage Jenkins

- In Dashboard, click on Manage Jenkins > System.
- Add the hostname as IP address and port number as 22 and also give the credential which was created and test connection.

SSH sites

SSH sites that projects will want to connect

Hostname ?

54.237.247.207

Port ?

22

Credentials

ubuntu (react ec2)

Add

☐ Pty ?

serverAliveInterval ?

0

timeout ?

0

Successful connection

Check connection

- Scroll down and add the following details under Publish over SSH.
- Hostname should be the ip-address of the instance to which we want to deploy the application.
- Username is the name of the user in the EC2.
- Remote directory is the path to which we want to store the files related to the application.
- Click on advanced to add the authentication type as password.
- Copy the contents of the pem file and add it inside the KEY box.
- Set port to 22 and test the configuration, then click on apply and save.

NOTE: In SSH server, when testing the configuration we get error because we have not changed the permission in EC2, since by default, html file permission is under root user, so this error can be neglected.

SSH Servers

SSH Server

Name ?

react

Hostname ?

54.85.63.108

Username ?

ubuntu

Remote Directory ?

/var/www/html

☐ Avoid sending files that have not changed ?

Advanced ▾

✎ Edited

Test Configuration

STEP 3 :

Create a project and build it.

- Now go back to the dashboard, select new item and give a name and select freestyle project.

- Under Source Code management, add the git project url and set the branch name which is in github for that project.

Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

Credentials ?

Branches to build ?

Branch Specifier (blank for 'any') ?

- Under “Build Environment”, select “Delete workspace before build starts” .
- Under “Build Steps” ,click on “execute shell script on remote host using ssh” and all the following commands.
- Do the same by adding one more build step called “Execute shell”.

Build Steps

Execute shell script on remote host using ssh

SSH site

ubuntu@54.237.247.207:22

Command

sudo apt install nginx -y
sudo chown ubuntu:ubuntu /var/www/html

☐ Execute each line ?

☐ Hide command from console output

Execute shell ?

Command

See [the list of available environment variables](#)

pwd
npm install
npm run build

STEP 4 :

- Select “Post Build Actions” and use “Send build artifacts over SSH”
- Select the instance name and under source files,select the build folder
- Remove prefix is used to just copy the contents of build folder instead of the folder itself,so the folder will not be copied,but only files and directories inside it will be copied.
- Under Exec command,make sure to reload nginx to save the changes done.

Send build artifacts over SSH ?

SSH Publishers

SSH Server

Name ?

test react

Advanced

Transfers

Transfer Set

Source files ?

build/**

Remove prefix ?

build

Remote directory ?

Exec command ?

sudo service nginx reload

- Save and build the project, go into console output and wait for the project to be built.

```
> react-scripts build
```

```
Creating an optimized production build...
```

```
Compiled successfully.
```

```
File sizes after gzip:
```

```
59.94 KB  build/static/js/main.4c33c55e.js
353 B     build/static/css/main.2d17c385.css
```

```
The project was built assuming it is hosted at the server root.
You can control this with the homepage field in your package.json.
For example, add this to build it for GitHub Pages:
```

```
"homepage" : "http://myname.github.io/myapp",
```

```
The build folder is ready to be deployed.
You may serve it with a static server:
```

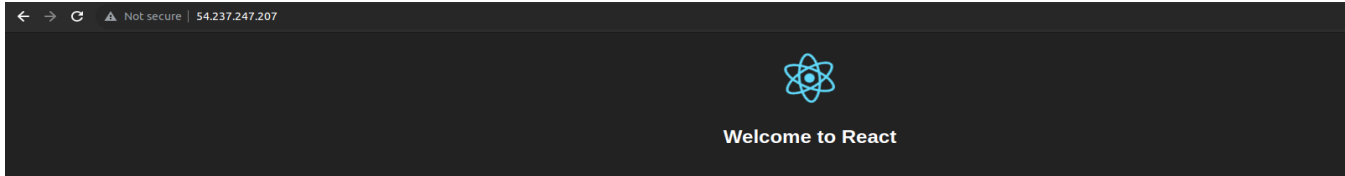
```
npm install -g serve
serve -s build
```

```
Find out more about deployment here:
```

```
http://bit.ly/2vY88Kr
```

```
SSH: Connecting from host [vikas-laptop]
SSH: Connecting with configuration [test react] ...
SSH: EXEC: completed after 401 ms
SSH: Disconnecting configuration [test react] ...
SSH: Transferred 9 file(s)
Finished: SUCCESS
```

- Copy paste the ip address of the instance and paste it in browser, we will get the application which is automated and deployed on EC2 server.



[HOME](#)

[ABOUT](#)

[CONTACTS](#)

This is Home page.

built using React.js + React Router v4 + Express.js

Now, It is Server-side rendering !

STEP 5 :

Configure Email notification upon the build success or failure.

- Move into Manage Jenkins > System
- Under “Extended Email Notification”, add the details as shown.
- Create a Credential and add the username as your email and password should be the app password.
 - To generate the app password ,first turn on 2 step authentication.
 - After that,click on Security > 2 step authentication > app passwords
 - Select the app as mail and the device which you are using and click on generate.

Extended E-mail Notification

SMTP server

smtp.gmail.com

SMTP Port

465

Advanced ^

 Edited

Credentials

vikaskumar.sglc@gmail.com/***** (this is a 2step auth password)

Add ▾



Use SSL



Use TLS



Use OAuth 2.0

- Under “E-mail Notification”, add the following details, under username, add the email id you generated password for , and add the password and SMTP port.

E-mail Notification

SMTP server

smtp.gmail.com

Default user e-mail suffix ?

Advanced ^

 Edited



Use SMTP Authentication ?

User Name

vikaskumar.sglc@gmail.com

Password



Concealed



Use SSL ?



Use TLS

SMTP Port ?

465

STEP 6 :

Update the project with email notification turned on.

- Select the project to which you want to add the email notification.
- Go to the post build action and select “**Editable Email Notification**”.
- Under “**Project Recipient List**” add the email address
- Click on advanced settings and scroll down to trigger options.
- Click on Add trigger and select “**Success**” option to send mail only when build is successful.
- Make sure to add “**Recipient List**” under “**Success**”.

The screenshot shows the 'Success' trigger configuration. Under the 'Send To' section, there are two dashed boxes. The first box is labeled 'Developers' and the second box is labeled 'Recipient List'. Both boxes have a red 'X' icon in the top right corner. Below these boxes is an 'Add' button with a dropdown arrow.

- Under “**Success**” ,click on advanced option and edit the options.
- Add email in “**Recipient List**”
- Change the “**subject**” , “**Content**” and “**Content type**” to your desired text.

The screenshot shows the 'Advanced' settings for the 'Success' trigger. The 'Recipient List' field contains the email address 'vikaskumar.sglc@gmail.com'. The 'Reply-To List' field contains the text '\$PROJECT_DEFAULT_REPLYTO'. The 'Content Type' dropdown menu is set to 'Plain Text (text/plain)'. The 'Subject' field contains the text 'Success'. The 'Content' field is empty.

- Apply the changes and save the project, then build it
- The console output will show like this at the end if build is successful.

```
/var/lib/jenkins/workspace/sample react
Email was triggered for: Success
Email was triggered for: Success
Sending email for trigger: Success
Sending email to: vikaskumar.sglc@gmail.com
Sending email for trigger: Success
Sending email to: vikaskumar.sglc@gmail.com
Finished: SUCCESS
```

- Open the Email and check for the mail received.

sample react - Build # 6 - Successful! Inbox x



address not configured yet <vikaskumar.sglc@gmail.com>

to me ▼

sample react - Build # 6 - Successful:

Check console output at <http://10.170.3.103:8080/job/sample%20react/6/> to view the results.

↩ Reply

➦ Forward