**Follow the below steps for pushing .war (backend and frontend) file to S3 bucket and create folder named as build version:**

1. **Install Java on Linux**
   1. Command line to install java 8

sudo yum install java-1.8.0

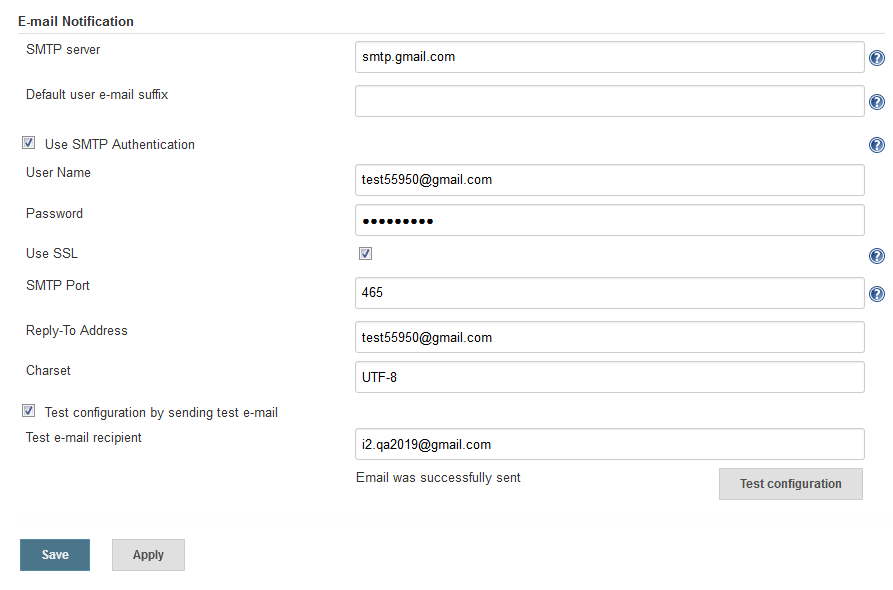
* 1. Command line to remove older version

sudo yum remove java-1.7.0-openjdk

1. **Install Git on Linux**

sudo yum install git

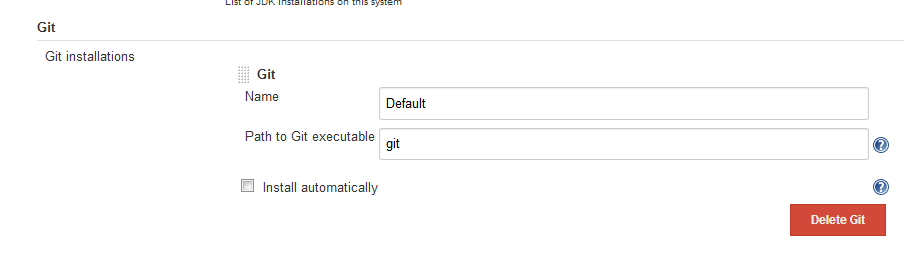
1. **Install Jenkins on Linux**
   1. Refer link → <https://docs.aws.amazon.com/aws-technical-content/latest/jenkins-on-aws/installation.html>
2. **Set up E-mail Notification**
   1. Manage Jenkins >> Configure System >> Email Notification
      1. Set up email server (example - set up Gmail account)



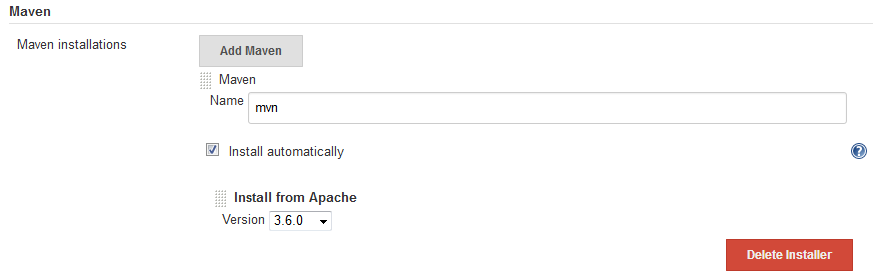
1. **Set up tool configuration in Jenkins (Java, Git, Maven)**
   1. Manage Jenkins >> Global Tool Configuration
      1. JDK → Select Install automatically java 8 and also provide oracle account access.



* + 1. Git → We need to install Git in our Linux machine, which is already explain above points, then enter name =”Default” and Path to Git executable = “git”

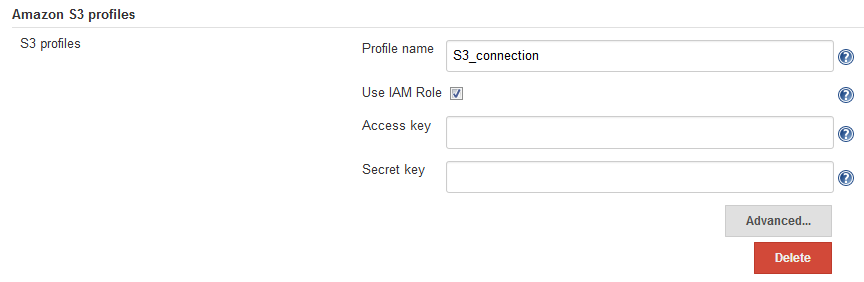


* + 1. Maven → Select install automatically option for maven and provide maven name also.

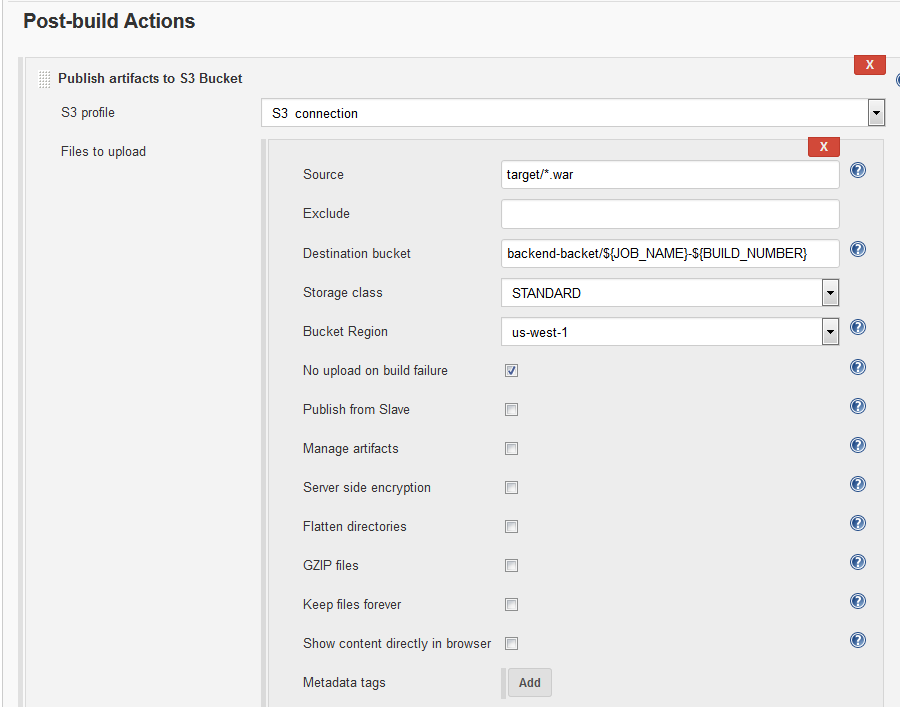


* 1. And Save it

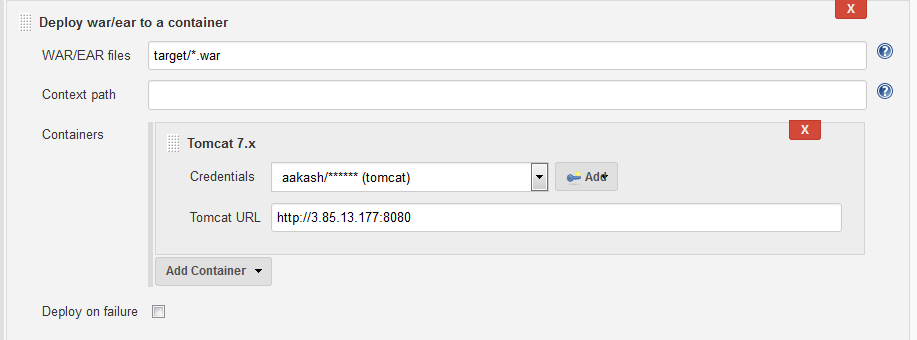
1. **Check and Install following Plug-in**
   1. Git 🡪 {*used to pull code from Git to Jenkins*}
   2. Github 🡪 {*used to pull code from Git to Jenkins*}
   3. Maven Integration 🡪 {*used for build automation  and creating .war files*}
   4. Deploy to container 🡪 {*used to deploy .war file to tomcat server*}
   5. S3 publisher 🡪 {*used to connect Jenkins to AWS*}
   6. Build name setter 🡪 {*modify build name*}
2. **Connect Jenkins TO AWS 🡪 {***For accessing s3 bucket***}**
   1. First create role for IAM user-- S3 full access
   2. Assign that role to Jenkins server
   3. Now move to Jenkins Portal
   4. Manage Jenkins>> Configure System
   5. Go to down of the page -- Amazon S3 profiles and click on Add button
   6. Add S3 profile and select check box



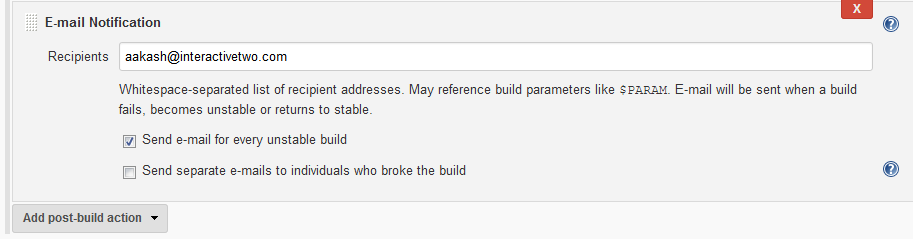
1. **Follow the steps for creating a build (backend) job.**
   1. Click on new item option
   2. Enter an item name (example - Backend\_Build), choose freestyle project and click on “OK” button.
   3. Enter description (this field is optional)
   4. Enter Git details on source course management section, by selecting Git option
   5. Select Poll SCM on Build Triggers Section, by this Jenkins is checking code of Git in provide time, so whenever some code is changed in Git, then automatically build will run.
   6. Select “Invoke top-level Maven targets” and enter “clean package” for creating a war file of success build
   7. Move to Post build Action
      1. Add “Publish artifacts to S3 Bucket” and enter all necessary details.
      2. For adding folder on S3 bucket enter “*<bucket name>/<folder name>${BUILD\_NUMBER}*” on destination bucket field.



* + 1. Add “Deploy war/ear to a container” and enter all necessary details.



* + 1. Add Email Notification details (for getting notification for build failure with console output).
    2. And click on save



* 1. Run a Build
  2. Check on S3 bucket, war file successfully publish with folder
  3. Check tomcat server, war file successfully deployed

*>>* ***Refer link for making angular build to .war file -*** [***http://csetutorials.com/deploy-angular-2-maven-webapp-on-tomcat.html***](http://csetutorials.com/deploy-angular-2-maven-webapp-on-tomcat.html)

1. **Follow the steps for creating a build (frontend) job.**
   1. Click on new item option
   2. Enter an item name (example - Backend\_Build), choose freestyle project and click on “OK” button.
   3. Follow the steps from 7(c ) to 7 (j)

***Note****: We can run Jenkins from our local system and connect to AWS s3 bucket using IAM user (Role: S3\_Full\_Access) credential [explained in Step#6- Connect Jenkins server with Amazon S3****]***

**Following code we used:**

1. **Backend code →** <https://github.com/aakash0726/Backend_Code.git>
2. **Frontend code** → <https://github.com/aakash0726/FrontEnd_Code.git>