Procedure to send email notification to Jenkins when build fails and trigger build through git webhook

Abstract

This document gives step by step procedure to install and configure necessary plugins required to send email notification when build fails and to trigger the build when developer pushes code to github project.
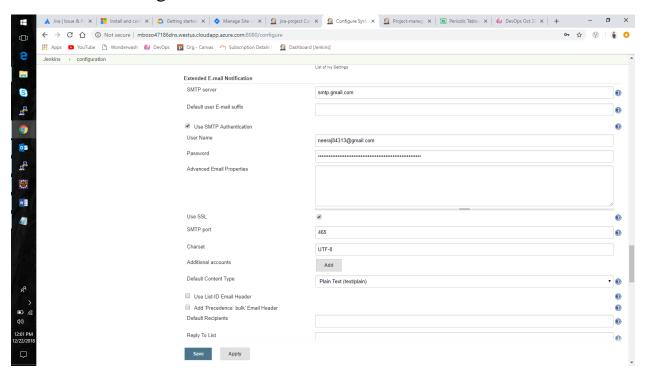
Contents

Plugins required

- Go to **Manage Jenkins → Manage Plugins → Available**
- Select the following plugins and click on install without restart
    - Email Extension Plugin
    - Git plugin
    - Github plugin
    - Mailer plugin

Restart Jenkins after installation if required.

Configure Email Service

- Go to **Manage Jenkins → Configure System**
- Under the **Extended E-mail Notification** section enter all the details and credentials of the sender. After completion it should look something like this :
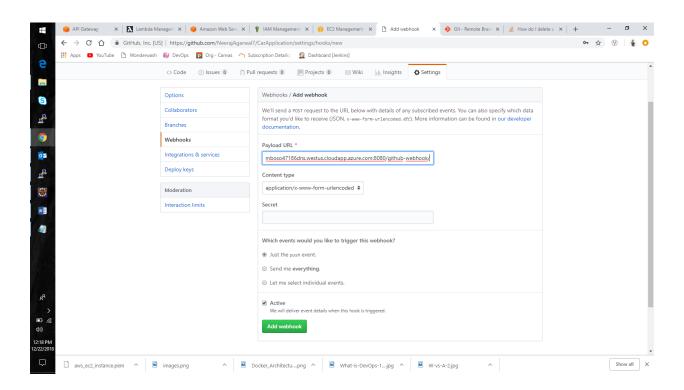

- Enter the same credentials and information provided above in the **E-mail Notification** section
  Note : ensure you have checked the use SSL checkbox
- Go to your gmail account and under account settings → privacy, turn on **Less secure app access**
  Note : for this to work turn off 2-step verification process
- Select the **Test configuration by sending test e-mail** checkbox to send a test email to check if the configurations and correct

Configure Github project for webhook

- Login to your github account and open the project you want to configure
- Go to settings and click on **Webhooks → Add webhook**
- Enter the payload url as

<p style="text-align:center">http://<Jenkins URL>/github-webhook/</p>

- Click on add webhook
- In your Jenkins job go to **configuration** and under build triggers select **GitHub hook trigger for GITScm polling** option.
- Now we have configured our Jenkins job to trigger build when code is pushed to github.

Sample Pipleline Script to run the build steps

Enter the following script in the **Pipeline** section of job configuration

```
node {
  try{
    def mvnHome
    stage('Preparation') {
      git "<Github URl>"
      mvnHome = tool 'maven'
    }
    stage('Build') {
      sh "mvn clean install"
    }
  }catch(err){
    emailext body: "Build failed due to some error", subject: 'Build failure', to: '<Receiver URL>'
    currentBuild.result = 'FAILURE'
  }
}
```

Run the build to check if the script is running.