

# Steps for Jenkins GitHub Webhook integration

In review, the steps to configure a Jenkins GitHub webhook trigger is:

- # Create a Jenkins build job that uses a GitHub URL
- # Click the GitHub hook trigger for GITScm polling checkbox on the build job
- # Create and copy a Jenkins API token for the Jenkins user who will run the build job
- # Create a trigger in your GitHub repository's settings page
- # Set the GitHub payload URL to be your Jenkins' IP address with /github-webhook/ appended to it
- # Set the Jenkins API token as the GitHub webhook secret token
- # Save the GitHub Webhook and then Jenkins builds will occur when a commit is pushed to the repo



# JENKINS

# Jenkins Email Notification

- # Click the 'Manage Jenkins' menu option displayed at the right side of the screen. You will be redirected to the 'Manage Jenkins' page, where you need to select the 'Manage Plugin' option.
- # Click the 'Available' tab present at the top of the 'Manage Plugin' page.
- # Type 'Notification' in the 'Filter' field displayed at the top-right side of the 'Manage Plugin' page. Click the checkbox next to the 'Email-ext plugin' option and 'Email-ext Template Plugin' option. Click the 'Install without restart' button.
- # Go to the Jenkins home page and click the 'Manage Jenkins' menu option. Then, select the 'Configure System' option.
- # Enter the SMTP server name under 'Email Notification'. Click the 'Advanced' button and then click the checkbox next to the 'Use SMTP Authentication' option. Now, set the following fields  
Check the email notification functionality by clicking the checkbox next to the 'Test configuration by sending Test e-mail recipient' option. Enter a valid email id and click the 'Test configuration' button to check whether the email id is valid or not.
- # Similarly enter the SMTP server name under 'Extended Email Notification'. Click the 'Advanced' button and then click the checkbox next to the 'Use SMTP Authentication' option. Now, set the following fields
- # Go to the home page and click on a New Item to create a Freestyle Jenkins Project.
- # Then, Click the 'Add post-build action' drop-down and select the 'E-mail Notification' value. Enter the recipient email id in the 'E-mail Notification' box and select the checkbox next to the 'Send e-mail for every unstable build' option.
- # Click the 'Add post-build action' drop-down and select the 'Editable Email Notification' value. Then fill the  
# Project Recipient List : email\_id@gmail.com  
Click the 'Advance Settings...' button in the 'Editable Email Notification' box. Click the 'Add Trigger' drop-down and select the 'Always' option and Click the 'Save' button.
- # Go to Email-Notification Jenkins job and click on Build now.
- # Check the Recipient's mail box.



# JENKINS

## Important Settings SMTP

SMTP server name : smtp.gmail.com  
User name:  
user\_email\_id@gmail.com  
Password: 123456  
Use SSL : Checked  
SMTP Port: 465

# Output

gokulsm1197 / Learning

Q Type [?] to search

>\_

+ ▾

<> Code ⌚ Issues 🔗 Pull requests ⌚ Actions 🗑 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

⚙ General

Access

🔍 Collaborators

🔧 Moderation options ▾

Code and automation

🌿 Branches

🔖 Tags

📁 Rules ▾

⌚ Actions ▾

**🔗 Webhooks**

📁 Environments

📁 Settings

## Webhooks

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

✓ http://65.2.121.100:8080/github-we... (push)

Edit

Delete

Jenkins

Q Search (CTRL+K)

Dashboard > gmail-jenkins > #12 > Console Output

📄 Status

</> Changes

**📄 Console Output**

📄 View as plain text

🔍 Edit Build Information

🗑 Delete build '#12'

📄 Polling Log

🔗 Git Build Data

⬅ Previous Build

✅ Console Output

```
Started by GitHub push by gokulsm1197
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/gmail-jenkins
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/gmail-jenkins/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/gokulsm1197/Learning # timeout=10
Fetching upstream changes from https://github.com/gokulsm1197/Learning
> git --version # timeout=10
> git --version # 'git version 2.34.1'
> git fetch --tags --force --progress -- https://github.com/gokulsm1197/Learning +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/main:{commit} # timeout=10
Checking out Revision fd995ceas5d0d1de69602ab6065eda92aa734 (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f fd995ceas5d0d1de69602ab6065eda92aa734 # timeout=10
Commit message: "Update Jenkins.sh"
> git rev-list --no-walk a528c252b016083afeFeb112ae705009eeae13e32 # timeout=10
No emails were triggered.
[gmail-jenkins] $ /bin/sh -xe /tmp/jenkinsS45741035743067727.sh
+ bash jenkins.sh
hi (.....()).....()
good
vvv good.....<12222>.....()
vvv good.....<112222>.....()
vvv good.....<111222>.....()
vvv good.....<111122>.....()
vvv good.....<111112>.....()
vvv good.....<55555555555555>.....()
vvv good.....<66666666666666>.....()
Email was triggered for: Always
Sending email for trigger: Always
Sending email to: gokulsm1197@gmail.com
Finished: SUCCESS
```

gokulsm1197 / Learning

Q Type [?] to search

>\_

+ ▾

<> Code ⌚ Issues 🔗 Pull requests ⌚ Actions 🗑 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

📁 Files

🌿 main ▾ + 🔍

Q Go to file t

📄 DOCKER & AWS.pdf

📄 Docker Push Output.pdf

📄 Dockerfile

📄 Ifstatement.sh

📄 Sample bash scripting.sh

📄 System and Networkdetails.sh

📄 docker-compose.yml

📄 filecreation.sh

**📄 jenkins.sh**

📄 siva.sh

Learning / jenkins.sh 📄

🌿 gokulsm1197 Update jenkins.sh


Code



Blame

10 lines (10 loc) · 399 Bytes

```
1  #!/bin/bash
2  echo "hi (.....()).....()"
3  echo "good"
4  echo "vvv good.....<12222>.....()"
5  echo "vvv good.....<112222>.....()"
6  echo "vvv good.....<111222>.....()"
7  echo "vvv good.....<111122>.....()"
8  echo "vvv good.....<111112>.....()"
9  echo "vvv good.....<55555555555555>.....()"
10 echo "vvv good.....<66666666666666>.....()"
```


output


Jenkins


gokulsm log out


Dashboard >

+ New Item

 People

 Build History

 Manage Jenkins

 My Views

Build Queue





No builds in the queue.

Build Executor Status

1 Idle


2 Idle

All +

| S   | W   | Name ↓        | Last Success                     |
|---|---|---------------|----------------------------------|
|  |  | gmail-jenkins | 8 min 8 sec <a href="#">#12</a>  |
|  |  | test-jenkins  | 8 min 13 sec <a href="#">#31</a> |

Icon: S M L

Icon legend

 Gmail

Compose

Inbox 1

Starred










Snoozed

Sent


Drafts 3

More

Labels +





gmail-jenkins - Build # 12 - Successful! Inbox x

 address not configured yet <gokulsm1197@gmail.com>  
to me ▾  
gmail-jenkins - Build # 12 - Successful:  
Check console output at <http://65.2.121.100:8080/job/gmail-jenkins/12/> to view the results.

Reply

Forward





 Jenkins


Dashboard > gmail-jenkins > #12


Status


</> Changes


 Console Output


 Edit Build Information

 Delete build '#12'

 Polling Log


 Git Build Data


 Previous Build

 **#12 (Jan 18, 2024, 12:43:26 PM)**

</> Changes


1. Update jenkins.sh ([commit: fd6995c](#)) ([details](#) / [githubweb](#))

 Started by GitHub push by gokulsm1197

 git

Revision: fd6995cea55d0d18e66960d2ab6065eda92aa734  
Repository: <https://github.com/gokulsm1197/Learning>

- refs/remotes/origin/main



+