Ibrahima Diallo
Kura Labs
Instructor: Tyrone Sanderson
Summer 2021

Documentation

Goal for this deployment:

1. Make a pipeline script with a Build, Test, and Deployment stage.
2. After you create the script; trigger your build for every 10 minutes.
3. Find a way to schedule your ec2 to shutdown by the end of class.

Procedure:

Step 1: Configure an EC2 on AWS

In this case, I used the EC2 I have built for the first deployment.

This is the process I went through in order to build the EC2 for the first deployment.

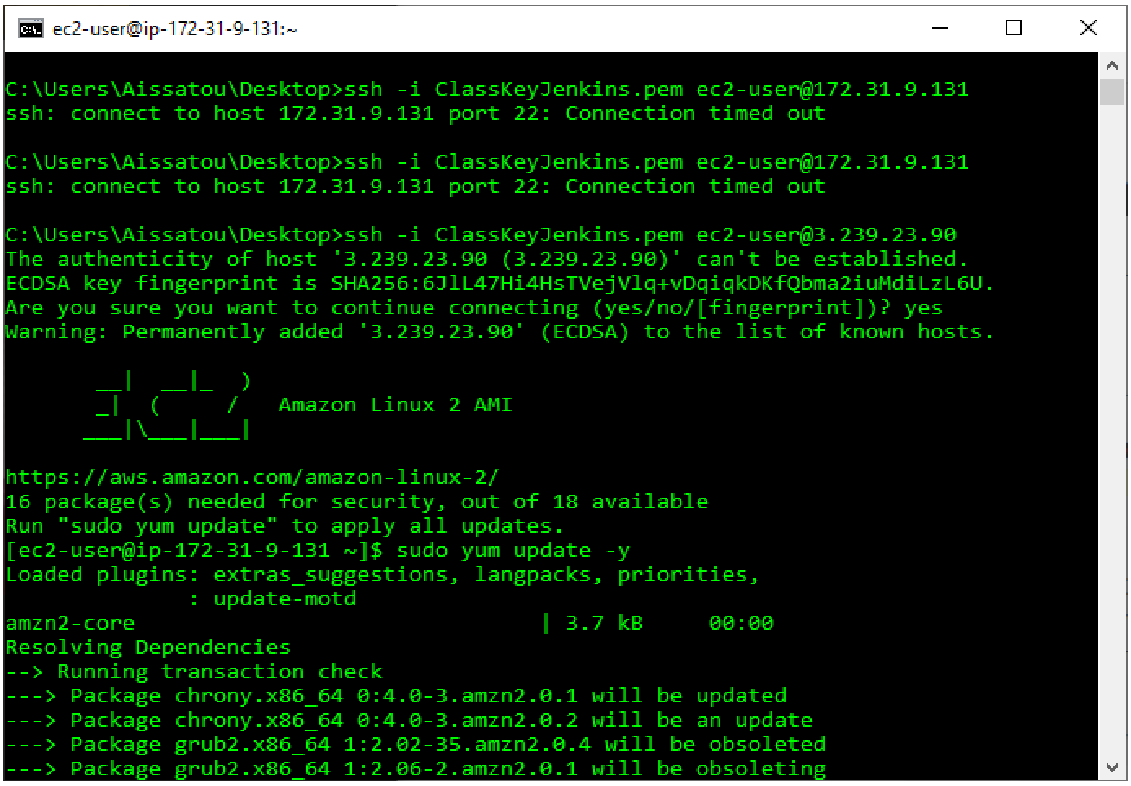
Step 2: Install Jenkins on EC2

The errors you may find are due to mispelling or typing errors for the different commands.

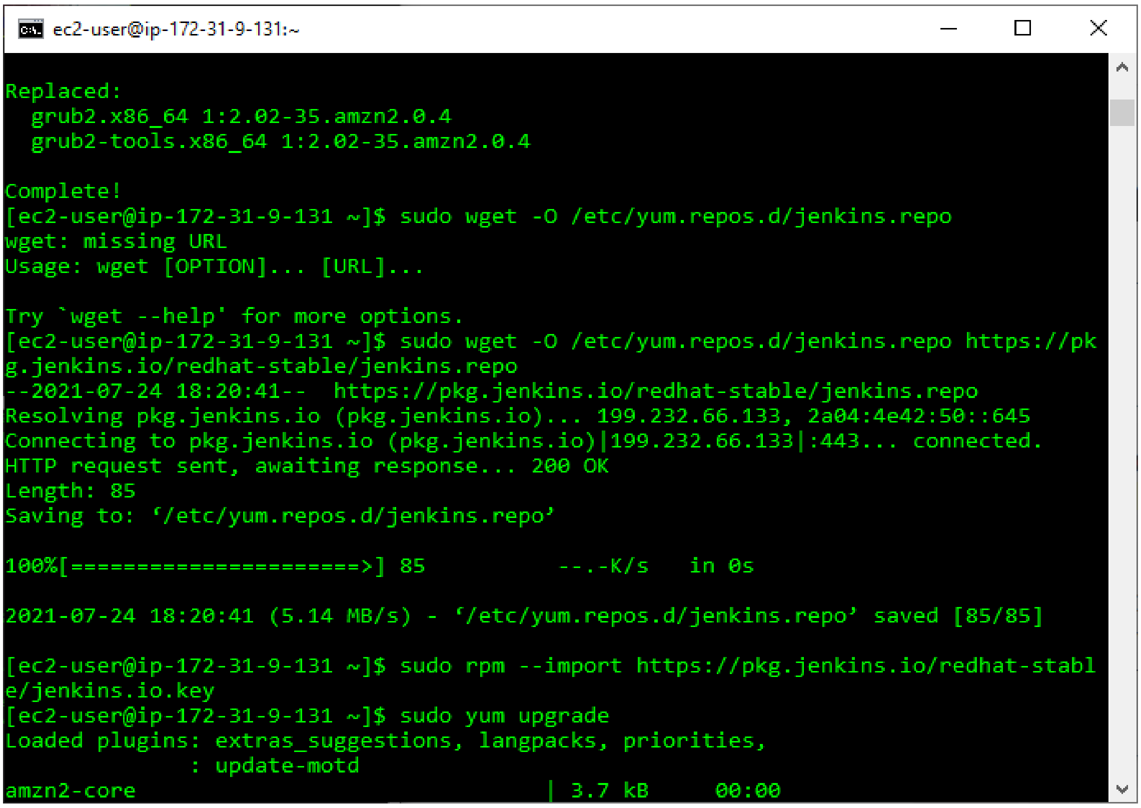
Below are the commands used to set up Jenkins on the EC2.

a. ssh -i ec2_key.pem ec2-user@ec2_public_IP
b. sudo yum update -y
c. sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
d. sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
e. sudo yum upgrade
f. sudo systemctl daemon-reload
g. sudo systemctl start jenkins22
h. sudo systemctl status jenkins
i. sudo cat /var/lib/jenkins/secrets/initialAdminPassword

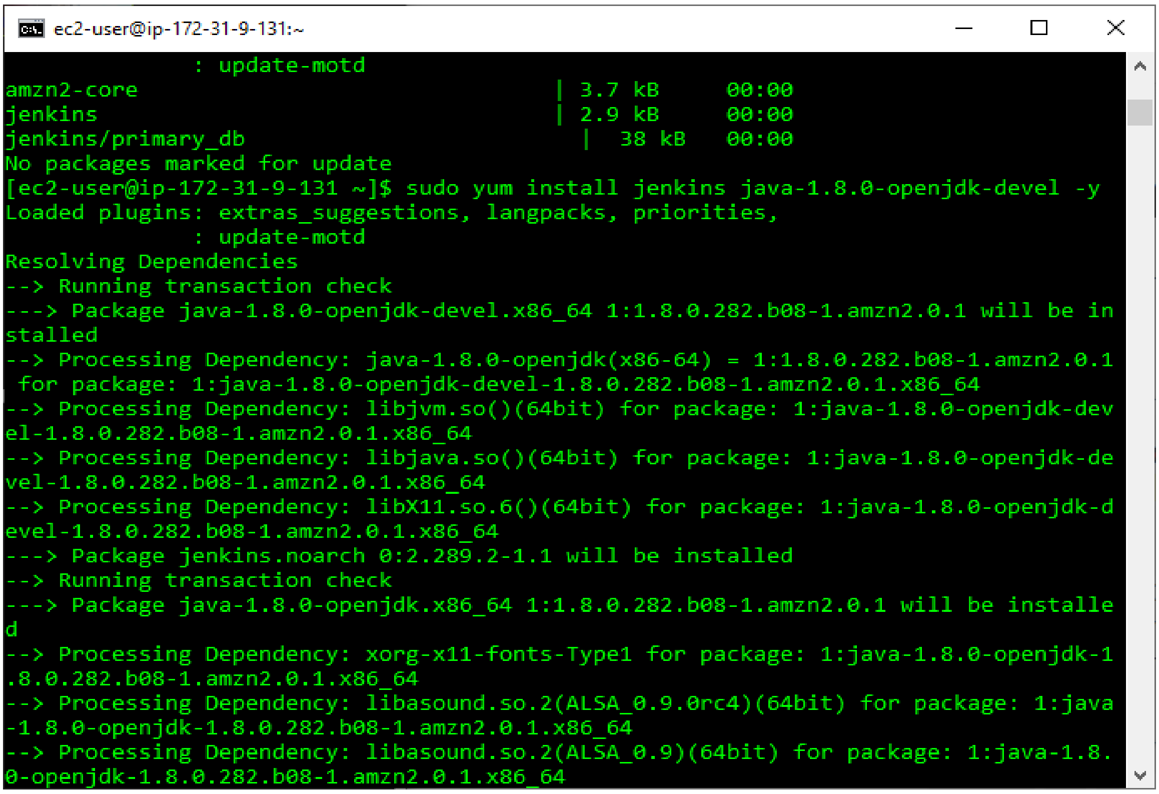
The screenshots of the CLI illustrate the process.



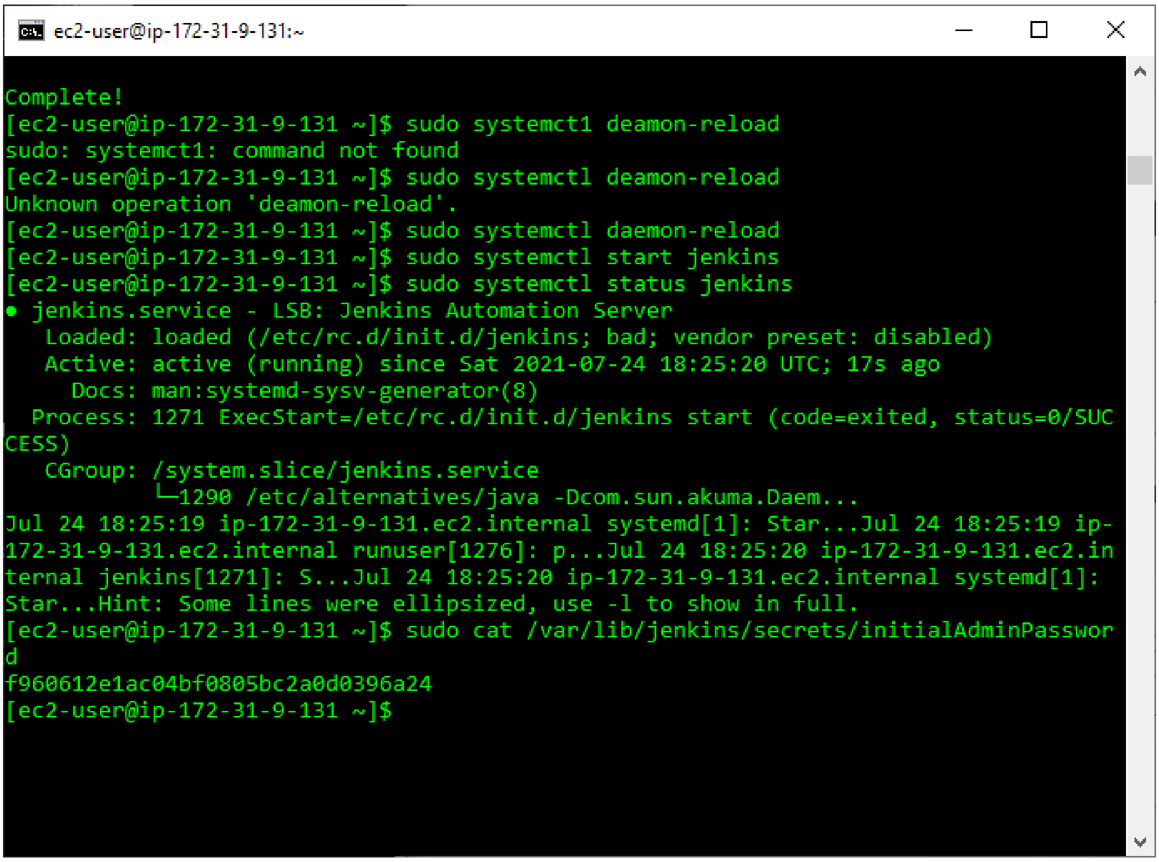
```
C:\Users\Aissatou\Desktop>ssh -i ClassKeyJenkins.pem ec2-user@172.31.9.131
ssh: connect to host 172.31.9.131 port 22: Connection timed out

C:\Users\Aissatou\Desktop>ssh -i ClassKeyJenkins.pem ec2-user@172.31.9.131
ssh: connect to host 172.31.9.131 port 22: Connection timed out

C:\Users\Aissatou\Desktop>ssh -i ClassKeyJenkins.pem ec2-user@3.239.23.90
The authenticity of host '3.239.23.90 (3.239.23.90)' can't be established.
ECDSA key fingerprint is SHA256:6JlL47Hi4HsTVejVlq+vDqiqkDKfQbma2iuMdiLzL6U.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.239.23.90' (ECDSA) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
16 package(s) needed for security, out of 18 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-9-131 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities,
              : update-motd
amzn2-core                                       | 3.7 kB     00:00
Resolving Dependencies
--> Running transaction check
---> Package chrony.x86_64 0:4.0-3.amzn2.0.1 will be updated
---> Package chrony.x86_64 0:4.0-3.amzn2.0.2 will be an update
---> Package grub2.x86_64 1:2.02-35.amzn2.0.4 will be obsoleted
---> Package grub2.x86_64 1:2.06-2.amzn2.0.1 will be obsoleting
```



```
Replaced:
  grub2.x86_64 1:2.02-35.amzn2.0.4
  grub2-tools.x86_64 1:2.02-35.amzn2.0.4

Complete!
[ec2-user@ip-172-31-9-131 ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo
wget: missing URL
Usage: wget [OPTION]... [URL]...

Try `wget --help' for more options.
[ec2-user@ip-172-31-9-131 ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo https://pk
g.jenkins.io/redhat-stable/jenkins.repo
--2021-07-24 18:20:41--  https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 199.232.66.133, 2a04:4e42:50::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|199.232.66.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 85
Saving to: ‘/etc/yum.repos.d/jenkins.repo’

100%[======================>] 85          --.-K/s   in 0s

2021-07-24 18:20:41 (5.14 MB/s) - ‘/etc/yum.repos.d/jenkins.repo’ saved [85/85]

[ec2-user@ip-172-31-9-131 ~]$ sudo rpm --import https://pkg.jenkins.io/redhat-stabl
e/jenkins.io.key
[ec2-user@ip-172-31-9-131 ~]$ sudo yum upgrade
Loaded plugins: extras_suggestions, langpacks, priorities,
              : update-motd
amzn2-core                                       | 3.7 kB     00:00
```

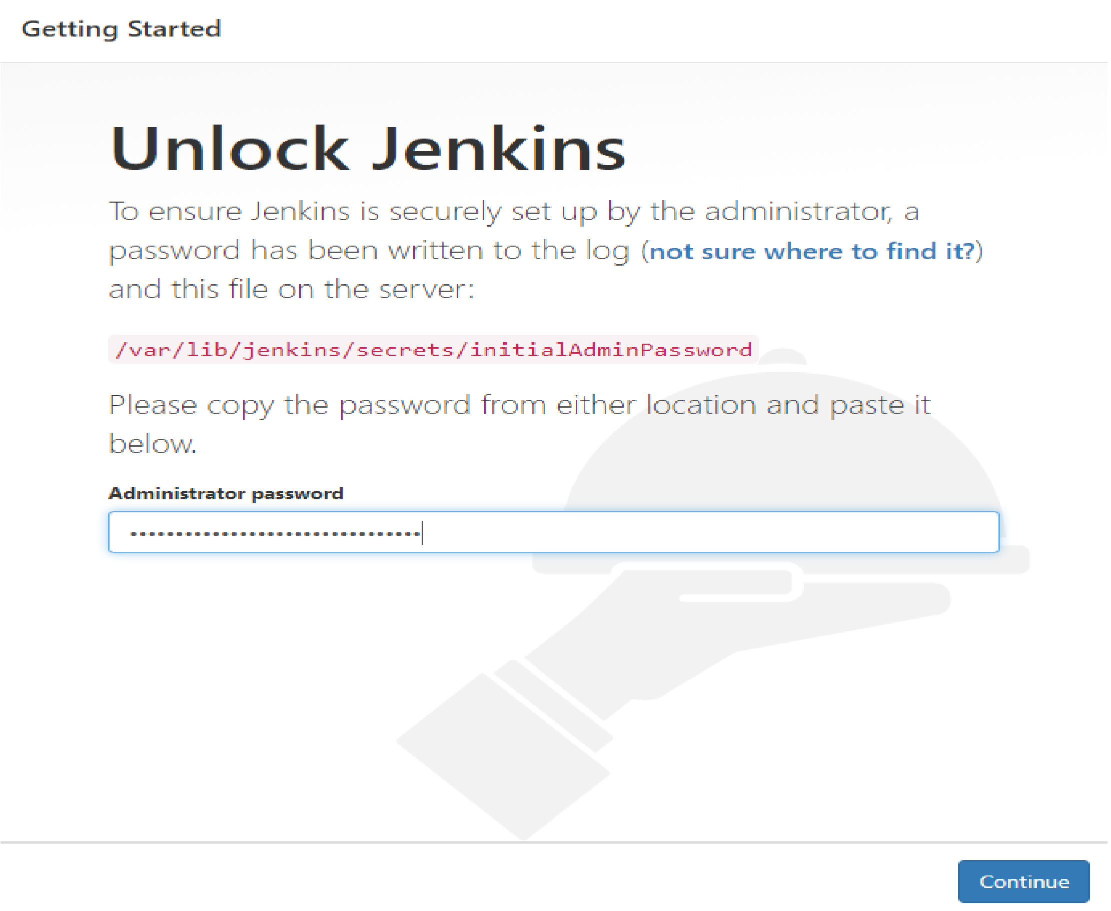


```
              : update-motd
amzn2-core                                       | 3.7 kB     00:00
jenkins                                          | 2.9 kB     00:00
jenkins/primary_db                                 |  38 kB   00:00
No packages marked for update
[ec2-user@ip-172-31-9-131 ~]$ sudo yum install jenkins java-1.8.0-openjdk-devel -y
Loaded plugins: extras_suggestions, langpacks, priorities,
              : update-motd
Resolving Dependencies
--> Running transaction check
---> Package java-1.8.0-openjdk-devel.x86_64 1:1.8.0.282.b08-1.amzn2.0.1 will be in
stalled
--> Processing Dependency: java-1.8.0-openjdk(x86-64) = 1:1.8.0.282.b08-1.amzn2.0.1
 for package: 1:java-1.8.0-openjdk-devel-1.8.0.282.b08-1.amzn2.0.1.x86_64
--> Processing Dependency: libjvm.so()(64bit) for package: 1:java-1.8.0-openjdk-dev
el-1.8.0.282.b08-1.amzn2.0.1.x86_64
--> Processing Dependency: libjava.so()(64bit) for package: 1:java-1.8.0-openjdk-de
vel-1.8.0.282.b08-1.amzn2.0.1.x86_64
--> Processing Dependency: libX11.so.6()(64bit) for package: 1:java-1.8.0-openjdk-d
evel-1.8.0.282.b08-1.amzn2.0.1.x86_64
---> Package jenkins.noarch 0:2.289.2-1.1 will be installed
--> Running transaction check
---> Package java-1.8.0-openjdk.x86_64 1:1.8.0.282.b08-1.amzn2.0.1 will be installe
d
--> Processing Dependency: xorg-x11-fonts-Type1 for package: 1:java-1.8.0-openjdk-1
.8.0.282.b08-1.amzn2.0.1.x86_64
--> Processing Dependency: libasound.so.2(ALSA_0.9.0rc4)(64bit) for package: 1:java
-1.8.0-openjdk-1.8.0.282.b08-1.amzn2.0.1.x86_64
--> Processing Dependency: libasound.so.2(ALSA_0.9)(64bit) for package: 1:java-1.8.
0-openjdk-1.8.0.282.b08-1.amzn2.0.1.x86_64
```



```
Complete!
[ec2-user@ip-172-31-9-131 ~]$ sudo systemct1 deamon-reload
sudo: systemct1: command not found
[ec2-user@ip-172-31-9-131 ~]$ sudo systemctl deamon-reload
Unknown operation 'deamon-reload'.
[ec2-user@ip-172-31-9-131 ~]$ sudo systemctl daemon-reload
[ec2-user@ip-172-31-9-131 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-9-131 ~]$ sudo systemctl status jenkins
● jenkins.service - LSB: Jenkins Automation Server
   Loaded: loaded (/etc/rc.d/init.d/jenkins; bad; vendor preset: disabled)
   Active: active (running) since Sat 2021-07-24 18:25:20 UTC; 17s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 1271 ExecStart=/etc/rc.d/init.d/jenkins start (code=exited, status=0/SUC
CESS)
   CGroup: /system.slice/jenkins.service
           └─1290 /etc/alternatives/java -Dcom.sun.akuma.Daem...
Jul 24 18:25:19 ip-172-31-9-131.ec2.internal systemd[1]: Star...Jul 24 18:25:19 ip-
172-31-9-131.ec2.internal runuser[1276]: p...Jul 24 18:25:20 ip-172-31-9-131.ec2.in
ternal jenkins[1271]: S...Jul 24 18:25:20 ip-172-31-9-131.ec2.internal systemd[1]:
Star...Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-9-131 ~]$ sudo cat /var/lib/jenkins/secrets/initialAdminPasswor
d
f960612e1ac04bf0805bc2a0d0396a24
[ec2-user@ip-172-31-9-131 ~]$
```

Step 3: Configure the Jenkins account using the password obtained after the cat command previously.



Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

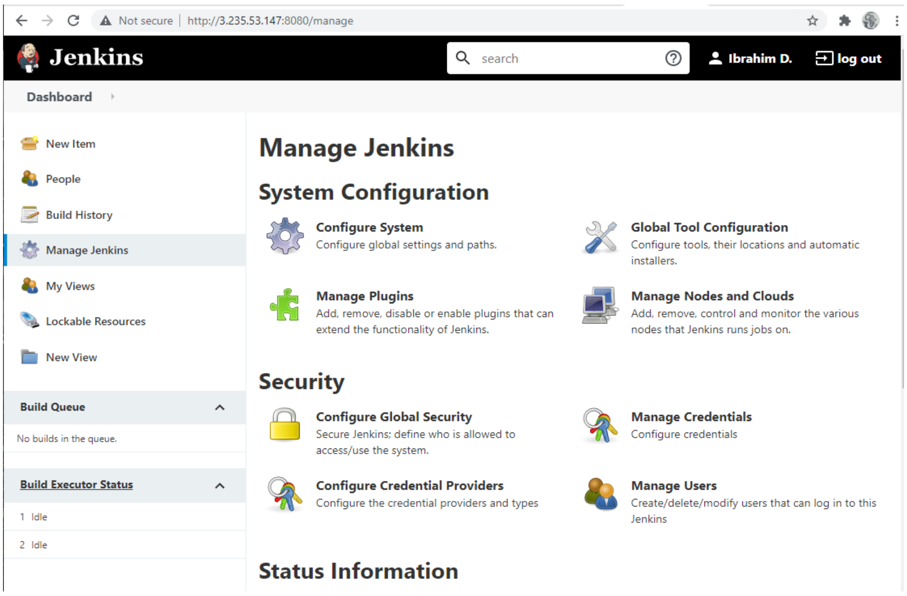
`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

••••••••••••••••••••••••••••••••

Then go to Manage Plugins to verify that Git is installed.



Manage Jenkins

System Configuration

Configure System
Configure global settings and paths.

Global Tool Configuration
Configure tools, their locations and automatic installers.

Manage Plugins
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

Manage Nodes and Clouds
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Security

Configure Global Security
Secure Jenkins; define who is allowed to access/use the system.

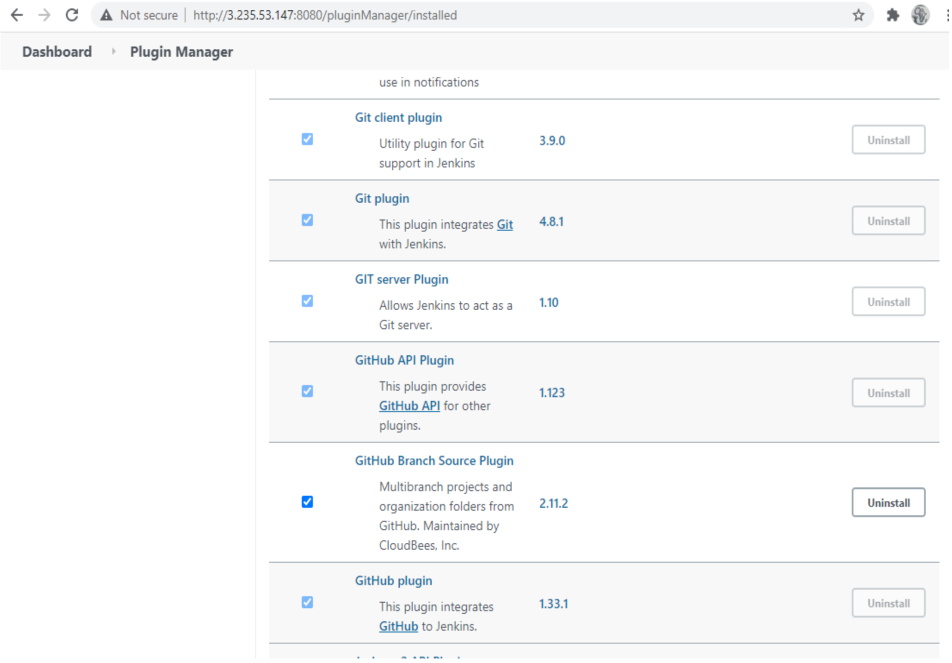
Manage Credentials
Configure credentials

Configure Credential Providers
Configure the credential providers and types

Manage Users
Create/delete/modify users that can log in to this Jenkins

Status Information

We can see below that Git has been installed.



Not secure | http://3.235.53.147:8080/pluginManager/installed

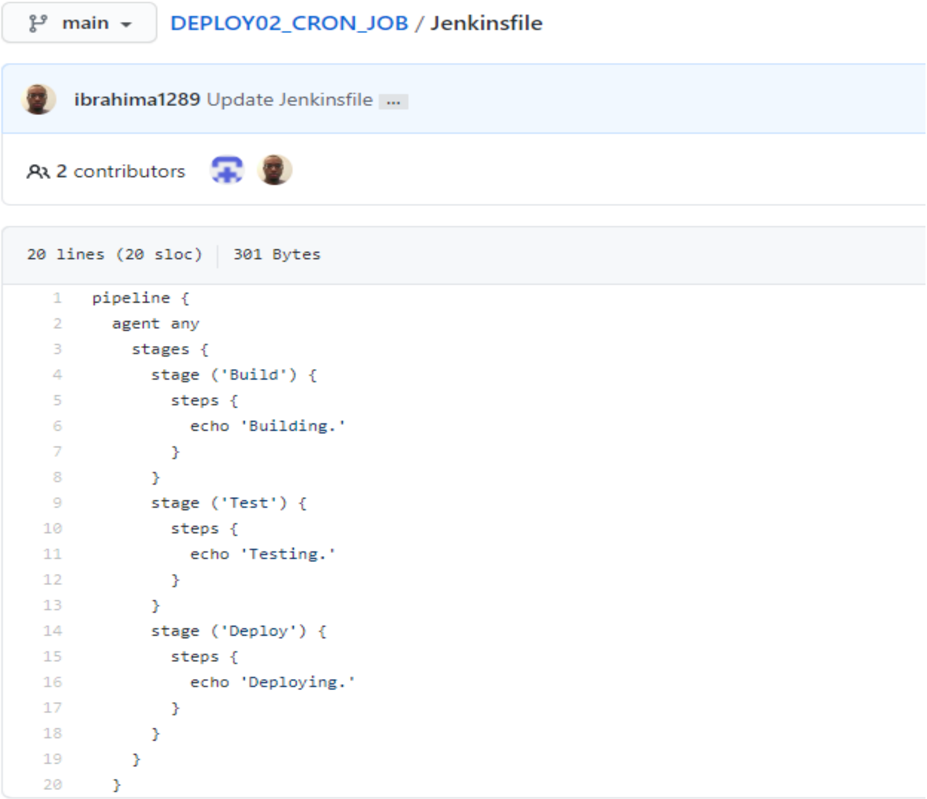
Dashboard › Plugin Manager

use in notifications

☑	**Git client plugin** Utility plugin for Git support in Jenkins	3.9.0		Uninstall
☑	**Git plugin** This plugin integrates Git with Jenkins.	4.8.1		Uninstall
☑	**GIT server Plugin** Allows Jenkins to act as a Git server.	1.10		Uninstall
☑	**GitHub API Plugin** This plugin provides GitHub API for other plugins.	1.123		Uninstall
☑	**GitHub Branch Source Plugin** Multibranch projects and organization folders from GitHub. Maintained by CloudBees, Inc.	2.11.2		Uninstall
☑	**GitHub plugin** This plugin integrates GitHub to Jenkins.	1.33.1		Uninstall

Step 4: Follow the steps provided in this link: https://github.com/kura-labs-org/DEPLOY01_HELLO_WORLD/blob/main/Deployment%231.pdf

Step 5: Make the pipeline script with a Build, Test, and Deployment stage



main ▾ DEPLOY02_CRON_JOB / Jenkinsfile

ibrahima1289 Update Jenkinsfile ...

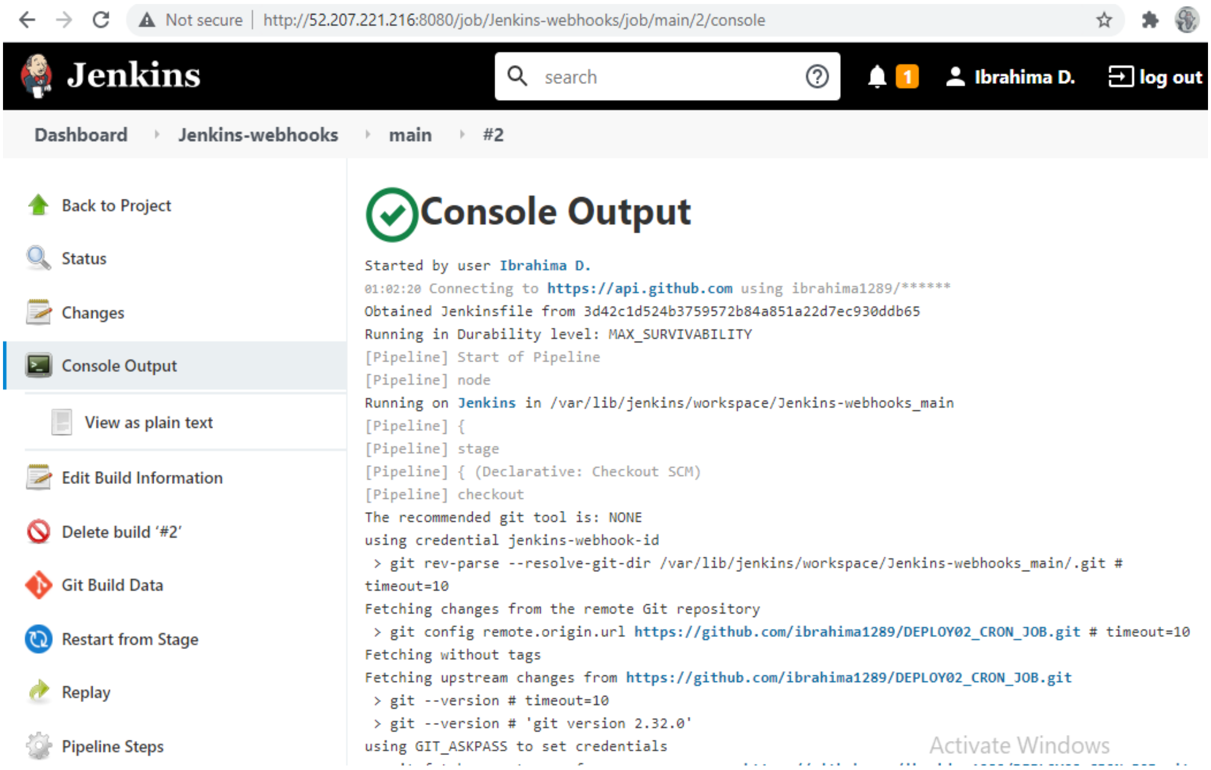
2 contributors

20 lines (20 sloc) | 301 Bytes

```
pipeline {
  agent any
    stages {
      stage ('Build') {
        steps {
          echo 'Building.'
        }
      }
      stage ('Test') {
        steps {
          echo 'Testing.'
        }
      }
      stage ('Deploy') {
        steps {
          echo 'Deploying.'
        }
      }
    }
  }
```

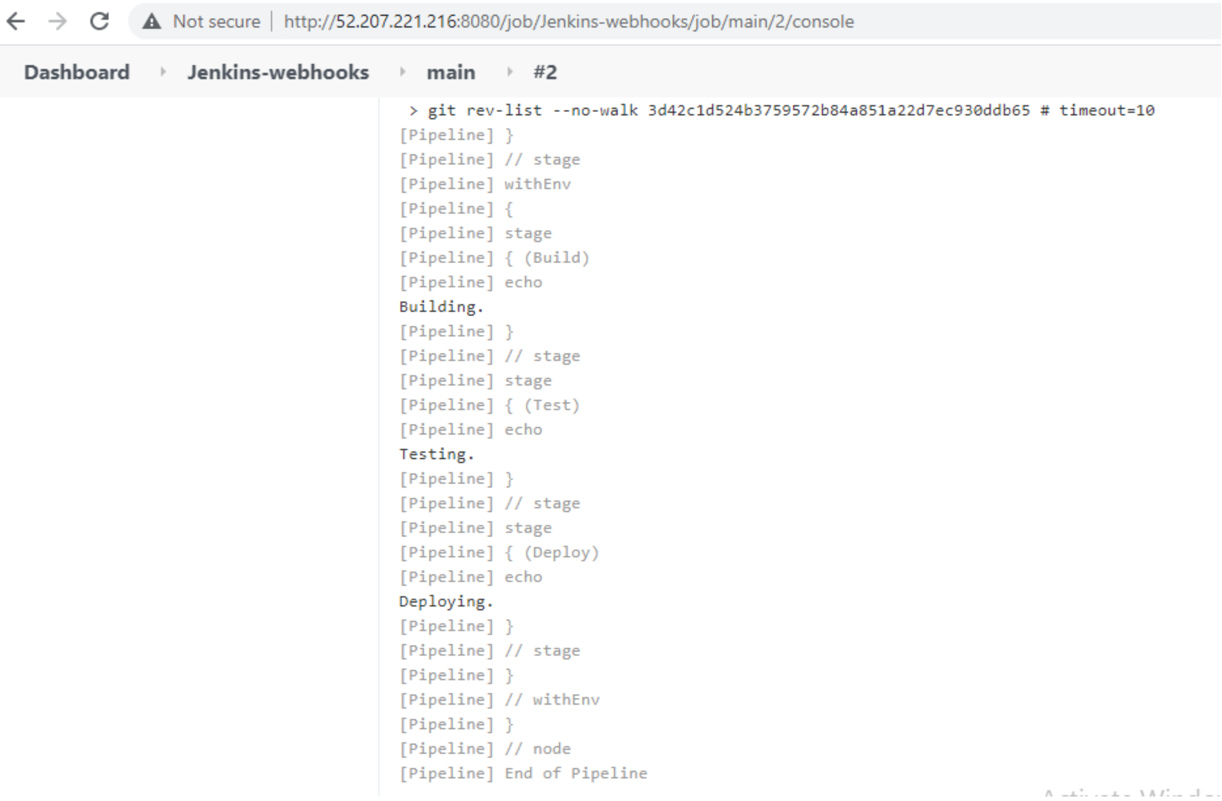
To test if it works, go to Dashboard > Jenkins-webhooks > main , then click Build now.

After that, the result I had was successful. We can see the results below.



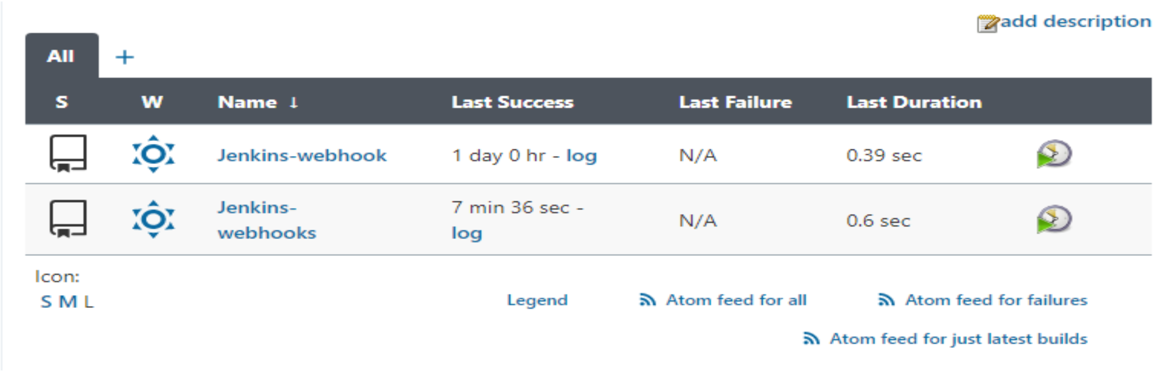
Console Output

```
Started by user Ibrahima D.
01:02:20 Connecting to https://api.github.com using ibrahima1289/******
Obtained Jenkinsfile from 3d42c1d524b3759572b84a851a22d7ec930ddb65
Running in Durability level: MAX_SURVIVABILITY
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/Jenkins-webhooks_main
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
The recommended git tool is: NONE
using credential jenkins-webhook-id
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/Jenkins-webhooks_main/.git #
timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/ibrahima1289/DEPLOY02_CRON_JOB.git # timeout=10
Fetching without tags
Fetching upstream changes from https://github.com/ibrahima1289/DEPLOY02_CRON_JOB.git
 > git --version # timeout=10
 > git --version # 'git version 2.32.0'
using GIT_ASKPASS to set credentials
```


Not secure | http://52.207.221.216:8080/job/Jenkins-webhooks/job/main/2/console
Dashboard
Jenkins-webhooks
main
#2

```
 > git rev-list --no-walk 3d42c1d524b3759572b84a851a22d7ec930ddb65 # timeout=10
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Build)
[Pipeline] echo
Building.
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Test)
[Pipeline] echo
Testing.
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy)
[Pipeline] echo
Deploying.
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
```

Note: **Jenkins-webhooks** is the set up for the second deployment, and **Jenkins-webhook** without the '**s**' at the end, the first deployment. They both are running on the same EC2.



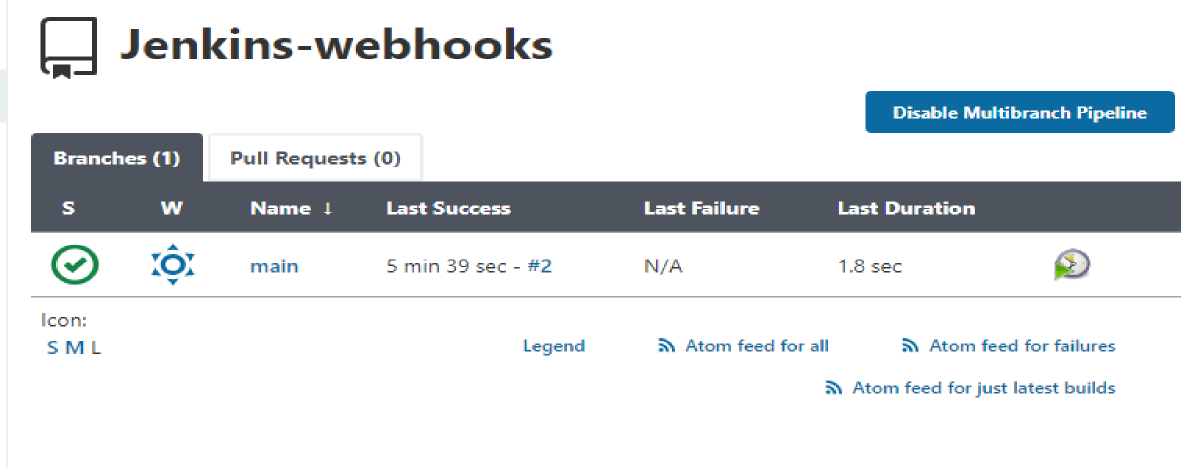
add description

All +

S	W	Name ↓	Last Success	Last Failure	Last Duration	
	☀	Jenkins-webhook	1 day 0 hr - log	N/A	0.39 sec	
	☀	Jenkins-webhooks	7 min 36 sec - log	N/A	0.6 sec	

Icon: S M L

Legend Atom feed for all Atom feed for failures Atom feed for just latest builds



Jenkins-webhooks

Disable Multibranch Pipeline

Branches (1) Pull Requests (0)

S	W	Name ↓	Last Success	Last Failure	Last Duration	
✓	☀	main	5 min 39 sec - #2	N/A	1.8 sec	

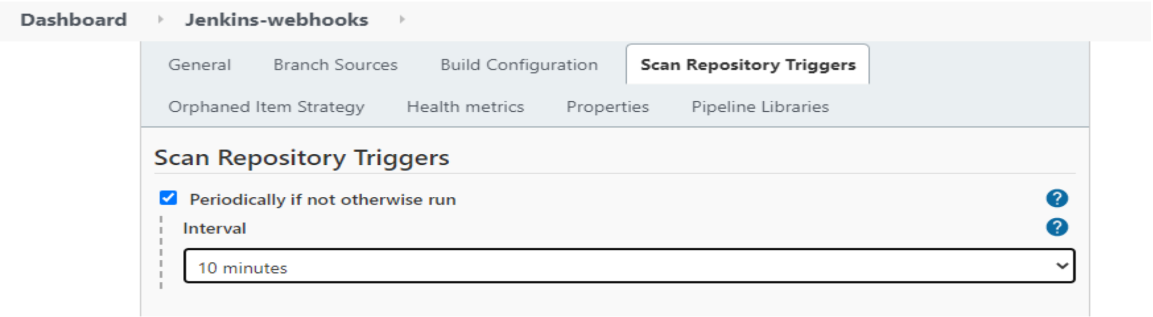
Icon: S M L

Legend Atom feed for all Atom feed for failures Atom feed for just latest builds

Step 6. To trigger my build for every 10 minutes, I tried using the Jenkins UI as following:

Dashboard > Jenkins-webhooks > Configure > Scan Repository Triggers >

Then, check the box scan and select the time interval (10 minutes in this case). And finally save.



Scan Repository Triggers

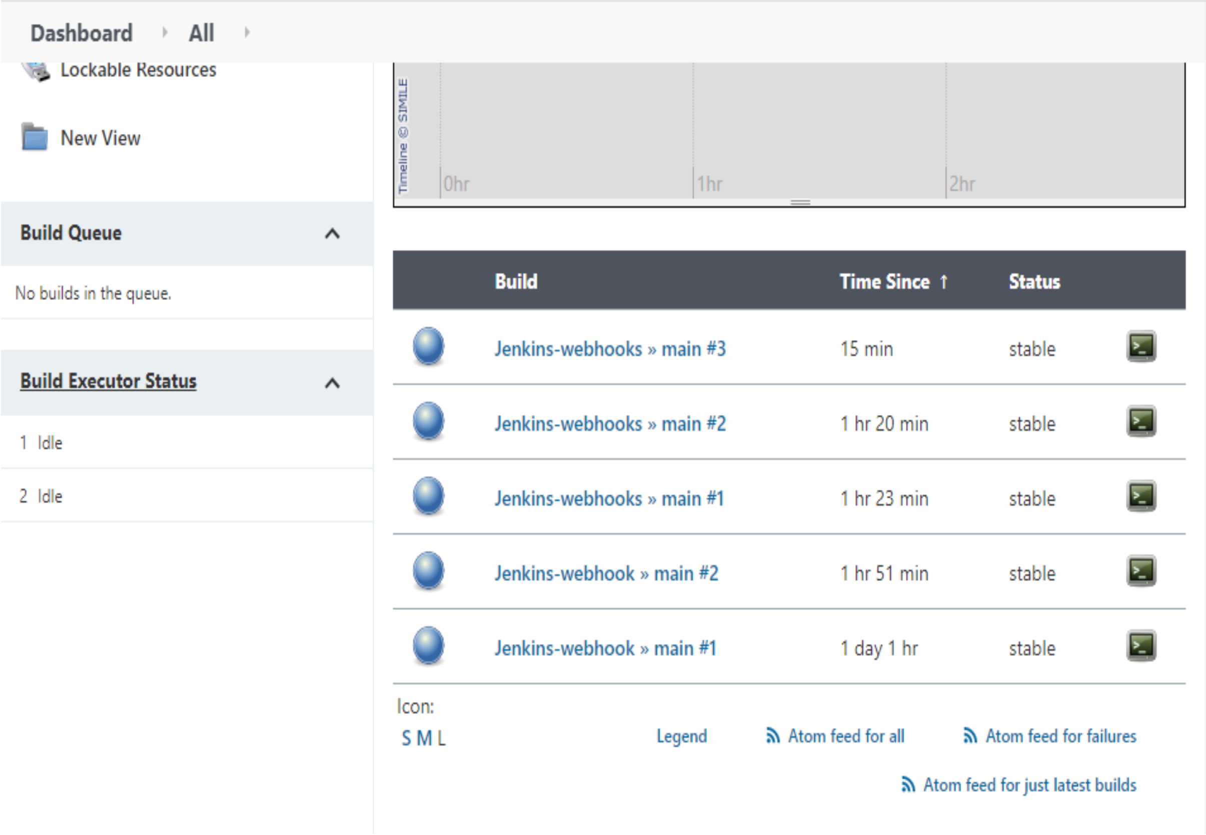
☑ Periodically if not otherwise run

Interval

10 minutes

Unfortunately, this method did not work. Here is the result after 10 minutes.

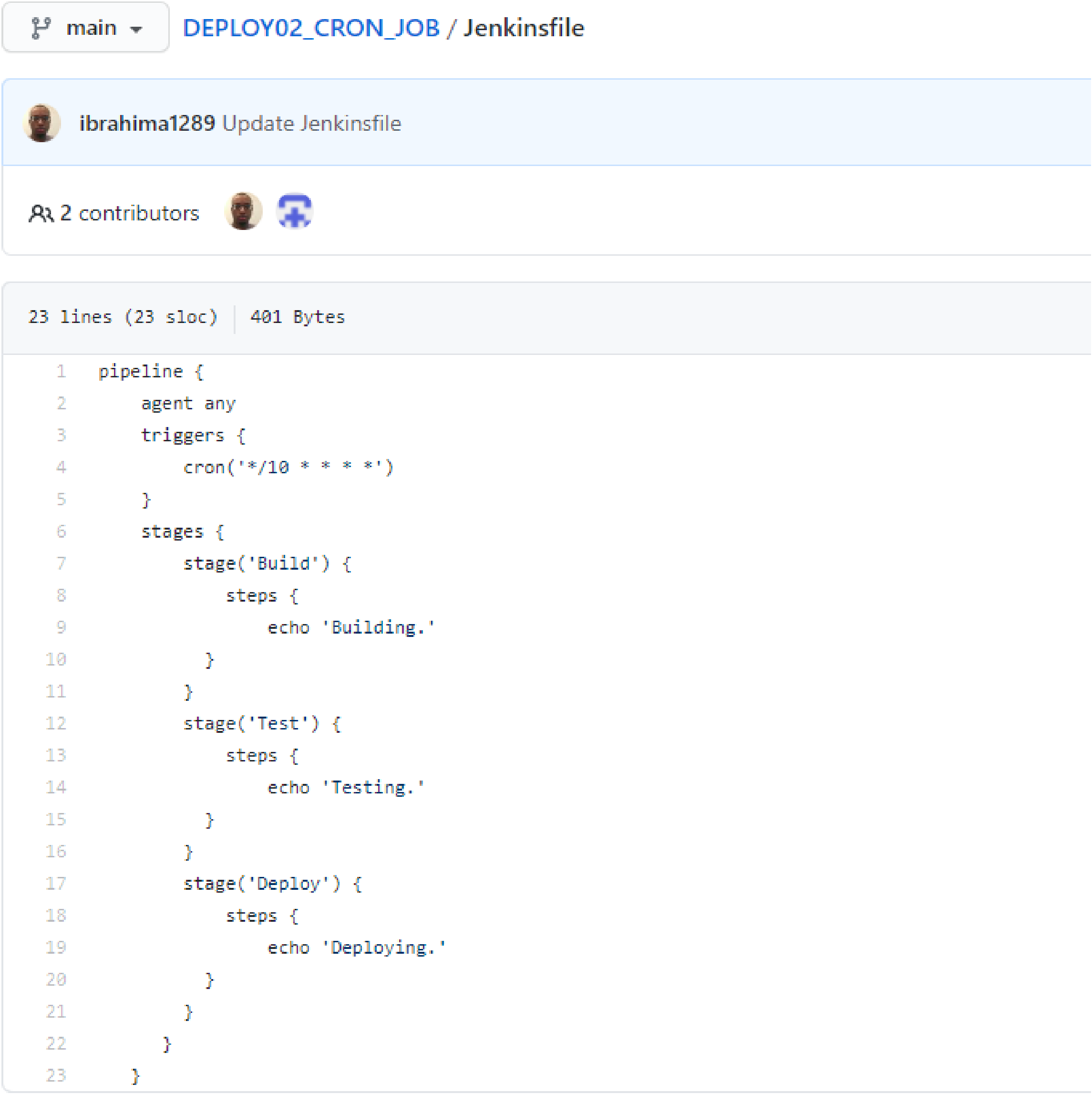
We can see that 15 minutes after the previous build, there is no update.



	Build	Time Since ↑	Status	
●	Jenkins-webhooks » main #3	15 min	stable	
●	Jenkins-webhooks » main #2	1 hr 20 min	stable	
●	Jenkins-webhooks » main #1	1 hr 23 min	stable	
●	Jenkins-webhook » main #2	1 hr 51 min	stable	
●	Jenkins-webhook » main #1	1 day 1 hr	stable	

To solve this issue, I found a new way of scheduling a new build every 10 minutes by updating the script using the source #3 listed below.

Here is the new script in Jenkinsfile.



main

DEPLOY02_CRON_JOB / Jenkinsfile

ibrahima1289 Update Jenkinsfile

2 contributors

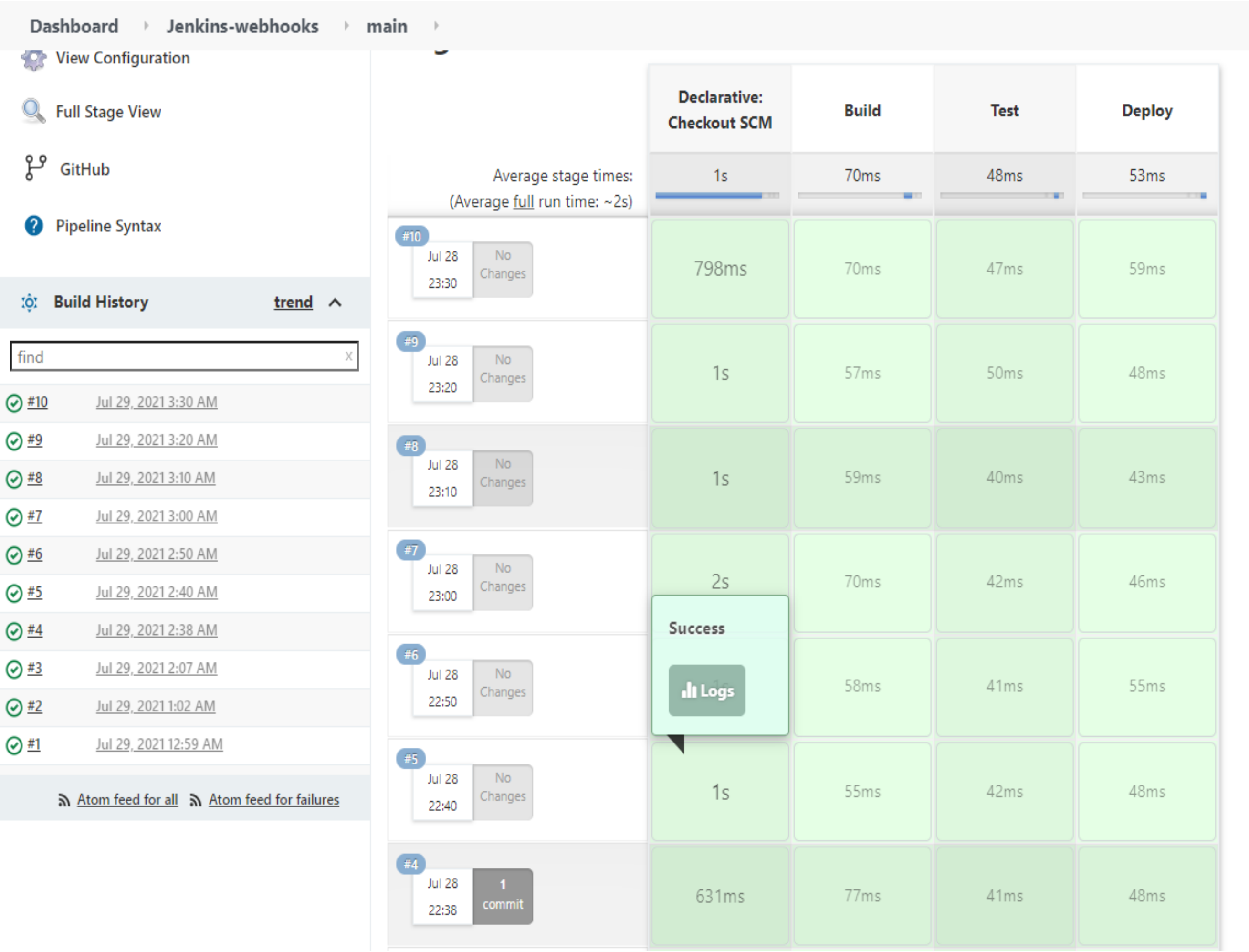
23 lines (23 sloc) | 401 Bytes

```
pipeline {
    agent any
    triggers {
        cron('*/10 * * * *')
    }
    stages {
        stage('Build') {
            steps {
                echo 'Building.'
          }
        }
        stage('Test') {
            steps {
                echo 'Testing.'
          }
        }
        stage('Deploy') {
            steps {
                echo 'Deploying.'
          }
        }
      }
   }
```

The trigger was added by using CRON notation:

triggers { cron ('*/10 * * * *') }

An hour later, The build history can be seen below. This method was successful.



✓	#10	Jul 29, 2021 3:30 AM
✓	#9	Jul 29, 2021 3:20 AM
✓	#8	Jul 29, 2021 3:10 AM
✓	#7	Jul 29, 2021 3:00 AM
✓	#6	Jul 29, 2021 2:50 AM
✓	#5	Jul 29, 2021 2:40 AM
✓	#4	Jul 29, 2021 2:38 AM
✓	#3	Jul 29, 2021 2:07 AM
✓	#2	Jul 29, 2021 1:02 AM
✓	#1	Jul 29, 2021 12:59 AM

	Declarative: Checkout SCM	Build	Test	Deploy
Average stage times: (Average full run time: ~2s)	1s	70ms	48ms	53ms
#10 Jul 28 23:30 No Changes	798ms	70ms	47ms	59ms
#9 Jul 28 23:20 No Changes	1s	57ms	50ms	48ms
#8 Jul 28 23:10 No Changes	1s	59ms	40ms	43ms
#7 Jul 28 23:00 No Changes	2s	70ms	42ms	46ms
#6 Jul 28 22:50 No Changes		58ms	41ms	55ms
#5 Jul 28 22:40 No Changes	1s	55ms	42ms	48ms
#4 Jul 28 22:38 1 commit	631ms	77ms	41ms	48ms

Step 7: Schedule your ec2 to shutdown by the end of class

I have not been able to schedule the EC2 instance to stop.

sources

1. https://docs.cloudbees.com/docs/admin-resources/latest/automating-with-jenkinsfile/creating-jenkinsfile
2. https://www.devopsschool.com/blog/setting-up-the-cron-jobs-in-jenkins-using-build-periodically-scheduling-the-jenins-job/
3. https://www.jenkins.io/doc/book/pipeline/syntax/#triggers