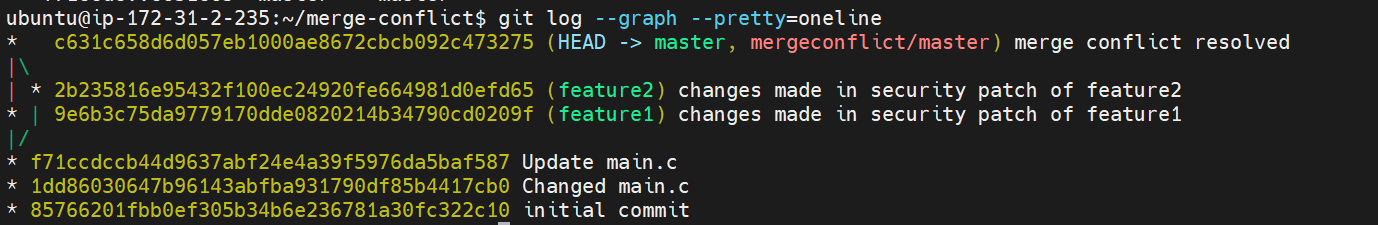
# Assignment

## GIT

### Case study merge-conflict



Solution:

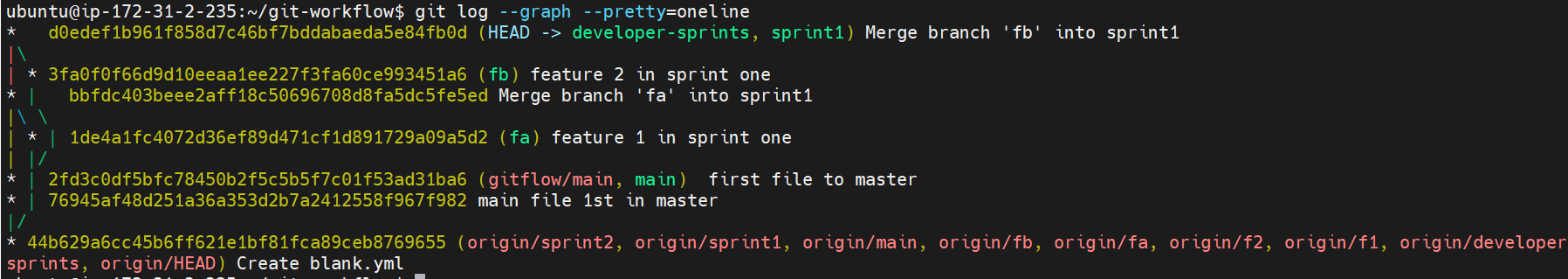


<https://github.com/saaritssinha/merge-conflict>

### Case study git-workflow



Solution:



<https://github.com/saaritssinha/git-workflow>

### Module git assignment 1



Solution:

<https://github.com/saaritssinha/assignment>

## Docker

### Self assignment1

1. Install tomcat using Docker

8080

1. Install Jenkins using Docker in another server

9080

1. Use Jenkins to deploy war file from GitHub
2. Use webserver (nginx/apache) on another server for reverse proxy

80

1. ubuntu@ip-172-31-85-185:~/Docker$ vi Dockerfile //no need

*From tomcat*

*ENV PORT = 8080*

*RUN echo ${PORT}*

*CMD ["catalina.sh" "run"]*

1. ubuntu@ip-172-31-85-185:~/Docker$ docker run -it --rm -p 8080:8080 -d tomcat //run directly
2. docker login
3. docker commit b012c9635c51 448555/tomcatapp:v1
4. docker push 448555/tomcatapp:v1
5. docker pull 448555/tomcatapp:v1
6. docker run -itd -p 8080:8080 421947278b4d(imageid)
7. ubuntu@ip-172-31-85-185:/tmp/move$ docker stop 317c8af065cc(containerid)
8. mkdir /home/ubuntu/Docker/code
9. cd /home/ubuntu/Docker/code/move
10. docker volume ls
11. docker run -it -v /home/ubuntu/Docker/move:/usr/local/tomcat/code -w /usr/local/tomcat/code/ -p 8080:8080 -d --name app\_war 448555/tomcatapp:v3 //check for attached volume
12. docker exec –it [containerid] bash //check for attached volume
13. docker commit containerid 448555/tomcatapp:v2
14. docker push 448555/tomcatapp:v2
15. cd /home/ubuntu/Docker/scripts
16. vi move\_script.sh

*cd /usr/local/tomcat/webapps/*

*rm -rf \**

*mv /usr/local/tomcat/code/boxfuse-sample-java-war-hello/target/hello-1.0.war /usr/local/tomcat/webapps/*

1. chmod 755 move\_script.sh
2. /////clone repo of war //// git clone [https://github.com/boxfuse/boxfuse-sample-java-war-hello.git at /tmp/move/code/](https://github.com/boxfuse/boxfuse-sample-java-war-hello.git%20at%20/tmp/move/code/)
3. docker exec app\_war bash -c "./move\_script.sh"
4. docker restart app\_war
5. <http://18.207.212.110:8080/hello-1.0/>
6. Docker ps
7. Cd /home/ubuntu/Docker/scripts
8. vi tomcat\_deploy.sh

*docker exec app\_war bash -c "./move\_script.sh"*

*docker restart app\_war*

1. chmod 755 tomcat\_deploy.sh

SSH btw tomcat server n Jenkins server

1. tomcat to Jenkins (<https://myadventuresincoding.wordpress.com/2011/12/22/linux-how-to-ssh-between-two-linux-computers-without-needing-a-password/> )
2. ssh-keygen -t rsa in tomcat
3. copy public\_key to jenkins server’s .ssh/authorised\_keys

Set alias

1. vim ~/.bashrc

alias sshtojenkins=”ssh ubuntu@3.84.157.195"

1. source ~/.bashrc
2. sshtojenkins

Jenkins server

1. docker pull jenkins/Jenkins
2. docker run --name app\_jenkins -d -v /home/ubuntu/Docker/jenkins:/var/jenkins\_home -w /var/jenkins\_home -p 9080:8080 -p 50000:50000 jenkins/jenkins
3. docker ps
4. docker exec -it app\_jenkins bash
5. jenkins@c04d34586fe7:~$ cat secrets/initialAdminPassword
6. Setup Jenkins
7. <https://www.linkedin.com/pulse/how-deploy-war-file-from-jenkins-tomcat-using-docker-de-avila-julio/?trk=public_profile_article_view>

### Assignment-docker



Solution:

Steps:

1. Cd /home/ubuntu/Docker/
2. Git clone <https://github.com/hshar94/helloworld>
3. Cd helloworld
4. Vi Dockerfile

FROM ubuntu:18.04

RUN apt-get update

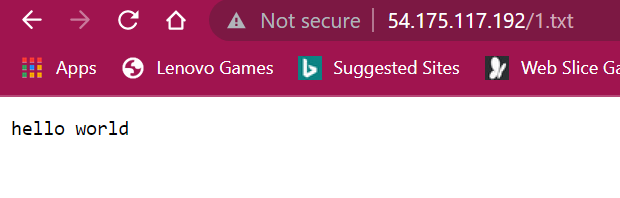
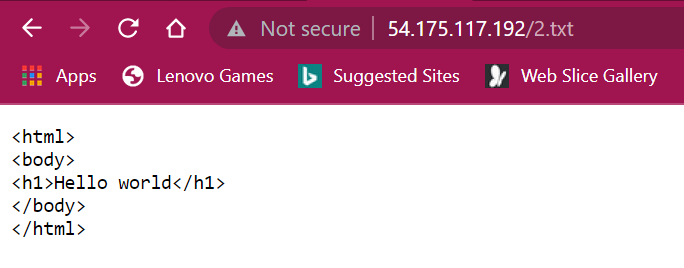
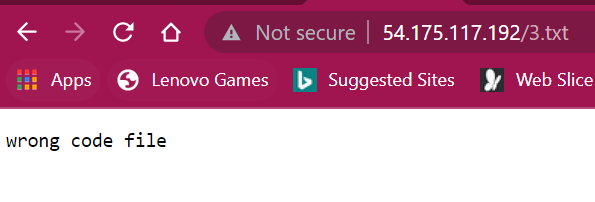
RUN apt-get upgrade -y

RUN apt install apache2 -y

COPY . /var/www/html/

ENTRYPOINT apachectl -D FOREGROUND

ENV name SAARIT

1. docker build . -t apache\_docker
2. docker run -it -p 80:80 -d apache\_docker
3. curl localhost:80
4. 
5. 
6. 
7. 
8. docker commit 37a38ce25db0 448555/dockerassignment
9. docker push 448555/dockerassignment
10. <https://hub.docker.com/repository/docker/448555/dockerassignment>
11. docker pull 448555/dockerassignment:latest

## Puppet

### Assignment:

* ubuntu@ip-172-31-85-185:~/new$ sudo puppet module generate devops-apache

{

"name": "devops-apache",

"version": "1.0.0",

"author": "devops",

"summary": "apache",

"license": "Apache-2.0",

"source": "apache",

"project\_page": "apache",

"issues\_url": "apache",

"dependencies": [

{

"name": "puppetlabs-stdlib",

"version\_requirement": ">= 1.0.0"

}

],

"data\_provider": null

}

* ubuntu@ip-172-31-85-185:~/new/apache/manifests$ sudo vi init.pp

class apache {

package {'apache2':

ensure => installed,

}

file { '/var/www/html/index.html':

  ensure => absent,

}

file { '/var/www/html/':

          ensure => present,

          source => "/home/ubuntu/new/helloworld/\*",

}

}

* ubuntu@ip-172-31-85-185:~/new$ git clone <https://github.com/hshar94/helloworld>
* ubuntu@ip-172-31-85-185:~/new$ cd helloworld/
* On slaves-🡪 sudo puppet agent –test

## Ansible