

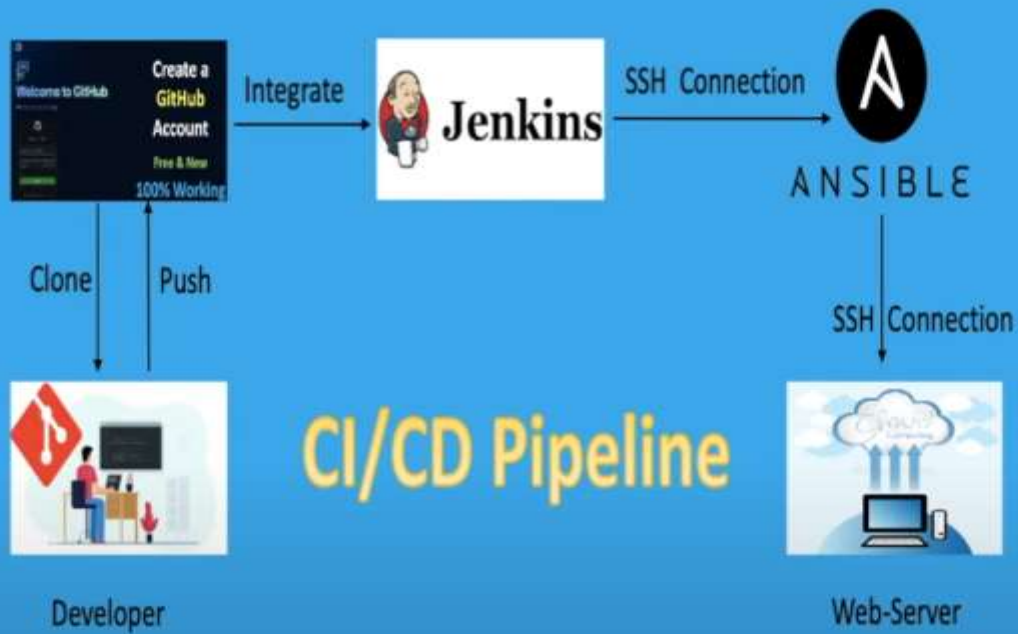
# AWS-DevOps

## CI/CD Pipeline

Project By S.Kripashankar (B.Tech IT)



# CI/CD Pipeline Flowchart



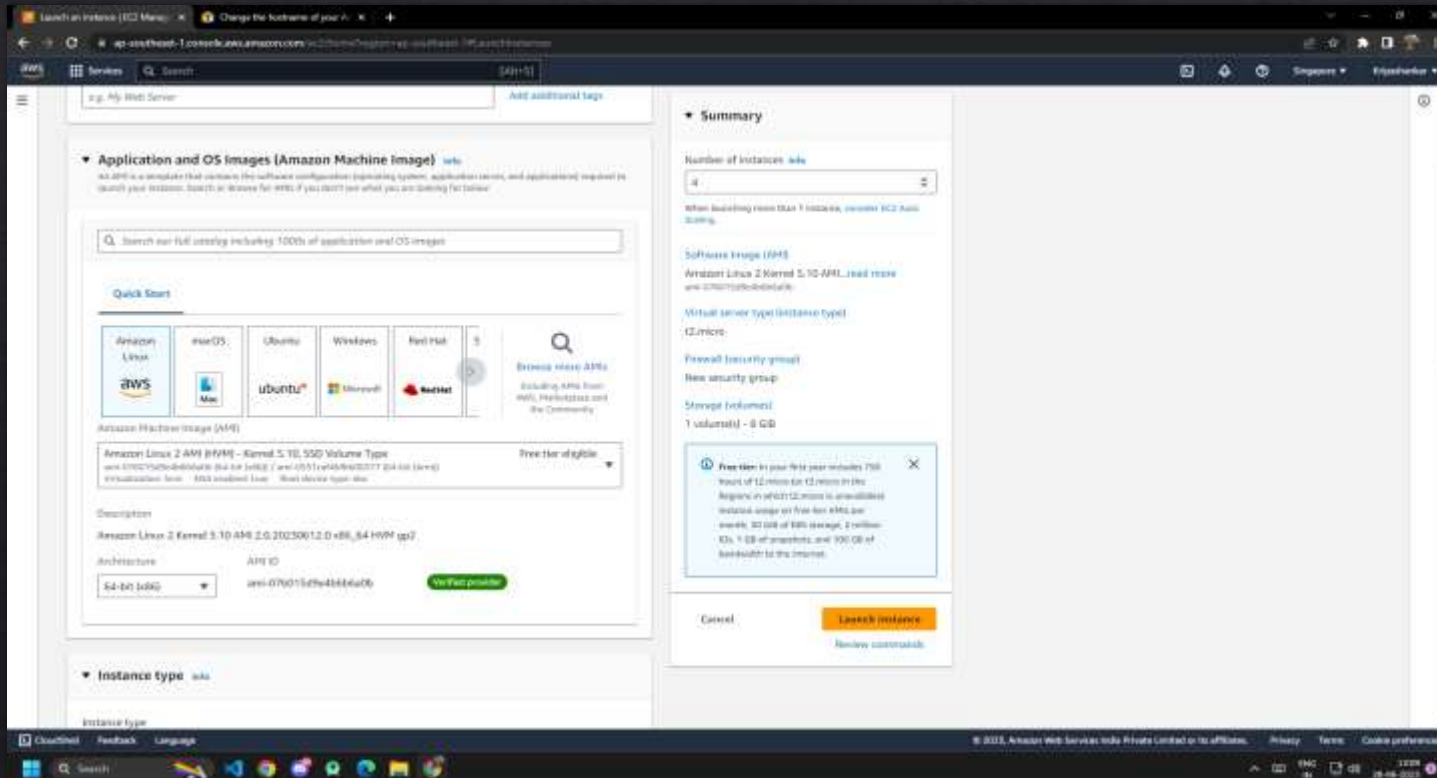
Modules Used in this Project.

- Ansible
- Jenkins
- Git

# To Do List

- # Keep Ready to all four servers (Developer, Jenkins, Ansible and Web server)*
- # Install required software*
- # Login to the GitHub account and Create new Repository*
- # Integrate Jenkins with GitHub*
- # Generate SSH key for password less connectivity and share it with respective servers to connect with it*
- # Set the root password for connectivity.*
- # Write a playbook in the Ansible server under new folder name sourcecode.*
- # Open inventory file and enter the host (Private Ip) of Web server in Ansible.*
- # Install new plugin named "public over SSH".*
- # Create Job in Jenkins.*
- # Configure Jenkins and Ansible Servers in the manage configuration section.*
- # Install the Apache (httpd) and start service in web server.*

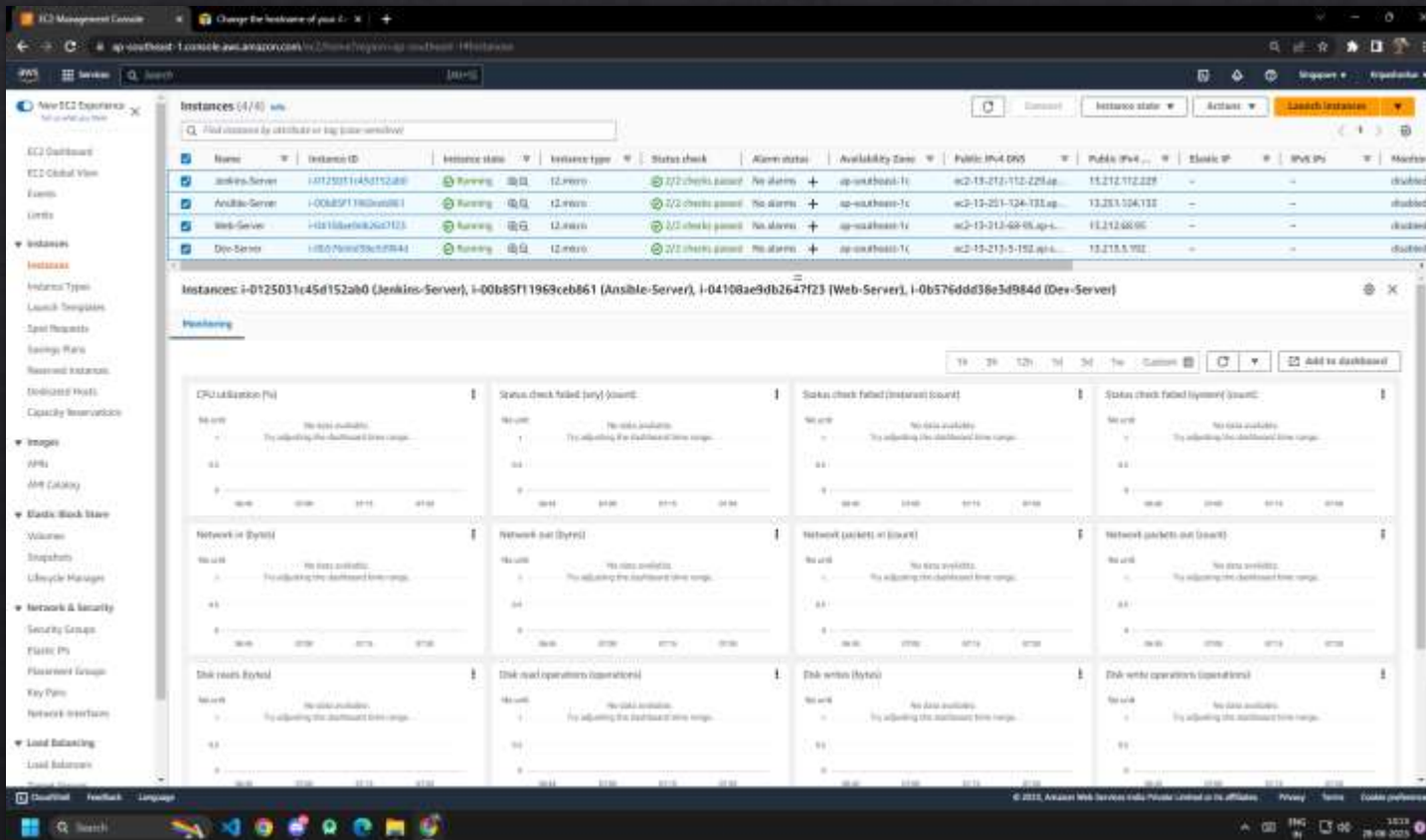
# Creating Instances in EC2



1. Go to EC2 create 4 instances accordingly.

2. At Application and OS images(AMI) Select AWS Linux 2 AMI (HVM).

# Renaming The Instances



Rename the instances accordingly as we need four servers to connect with the CI/CD Pipeline.

The List of the Servers

- Ansible-Server
- Jenkins-Server
- web-Server
- Dev-Server



Instance details | EC2 Management Console

Change the hostname of your instance

ap-southeast-1.console.aws.amazon.com/ec2/home?region=ap-southeast-1#InstanceDetails:instanceId=i-0125031c45d152ab0

Services Search [Alt+S]

New EC2 Experience

EC2 Dashboard  
EC2 Global View  
Events  
Limits  
Instances  
Images  
Elastic Block Store  
Network & Security  
Load Balancing

Instance summary for i-0125031c45d152ab0 (Jenkins-Server)

Updated less than a minute ago

Public IPv4 address copied

Instance ID: i-0125031c45d152ab0 (Jenkins-Server)

IPv6 address: -

Hostname type: IP name: ip-172-31-8-206.ap-southeast-1.compute.internal

Answer private resource DNS name: IPv4 (A)

Auto-assigned IP address: 15.212.112.229 [Public IP]

IAM Role: -

IMDSv2: Optional

Instance state: Running

Private IP DNS name (IPv4 only): ip-172-31-8-206.ap-southeast-1.compute.internal

Instance type: t2.micro

VPC ID: vpc-07b596a8732e7c3dc

Subnet ID: subnet-01524825d0a3320b3

Private IPv4 addresses: 172.31.8.206

Public IPv4 DNS: ec2-15-212-112-229.ap-southeast-1.compute.amazonaws.com [open address]

Elastic IP addresses: -

AWS Compute Optimizer finding: Opt-in to AWS Compute Optimizer for recommendations. [Learn more]

Auto Scaling Group name: -

Details Security Networking Storage Status checks Monitoring Tags

Instance details info

Platform: Amazon Linux [Inferred]

Platform details: Linux/UNIX

Stop protection: Disabled

Instance auto-recovery: Default

AMI Launch index: 3

Credit specification: standard

AMI ID: ami-07b015d9e4b6b6a0b

AMI name: amazon2-ami-kernel-5.10-hvm-2.0.20230612.0-x86\_64-gp2

Launch time: Wed Jun 28 2023 15:11:00 GMT+0530 (India Standard Time) (3 minutes)

Lifecycle: normal

Key pair assigned at launch: Project2023

Kernel ID: -

Monitoring: disabled

Termination protection: Disabled

AMI location: amazon/amzn2-ami-kernel-5.10-hvm-2.0.20230612.0-x86\_64-gp2

Stop-hibernate behavior: disabled

State transition reason: -

State transition message: -

CloudShell Feedback Language

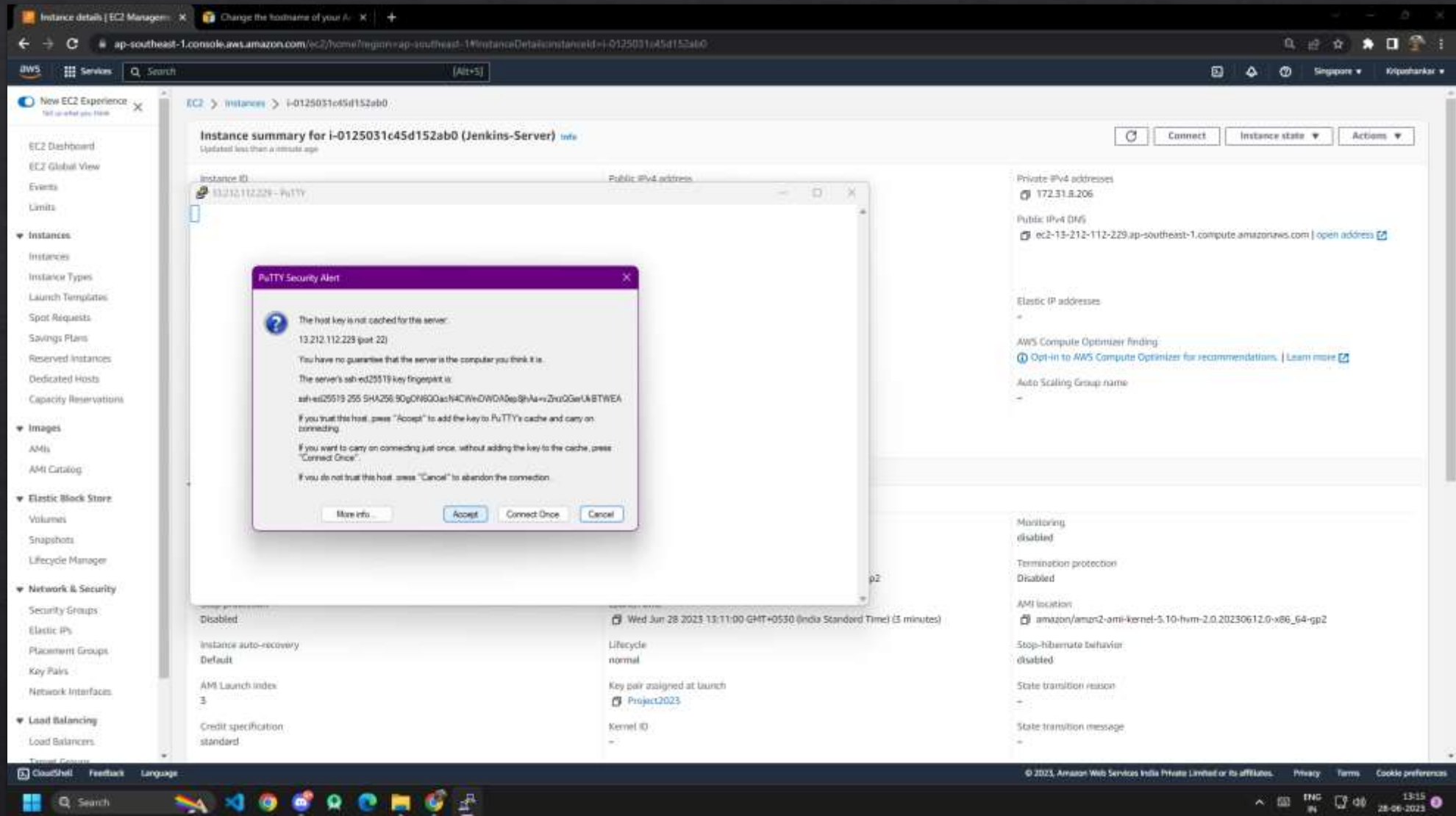
© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

13:13 28-06-2023

Copy the Public IP of the Jenkins Server.

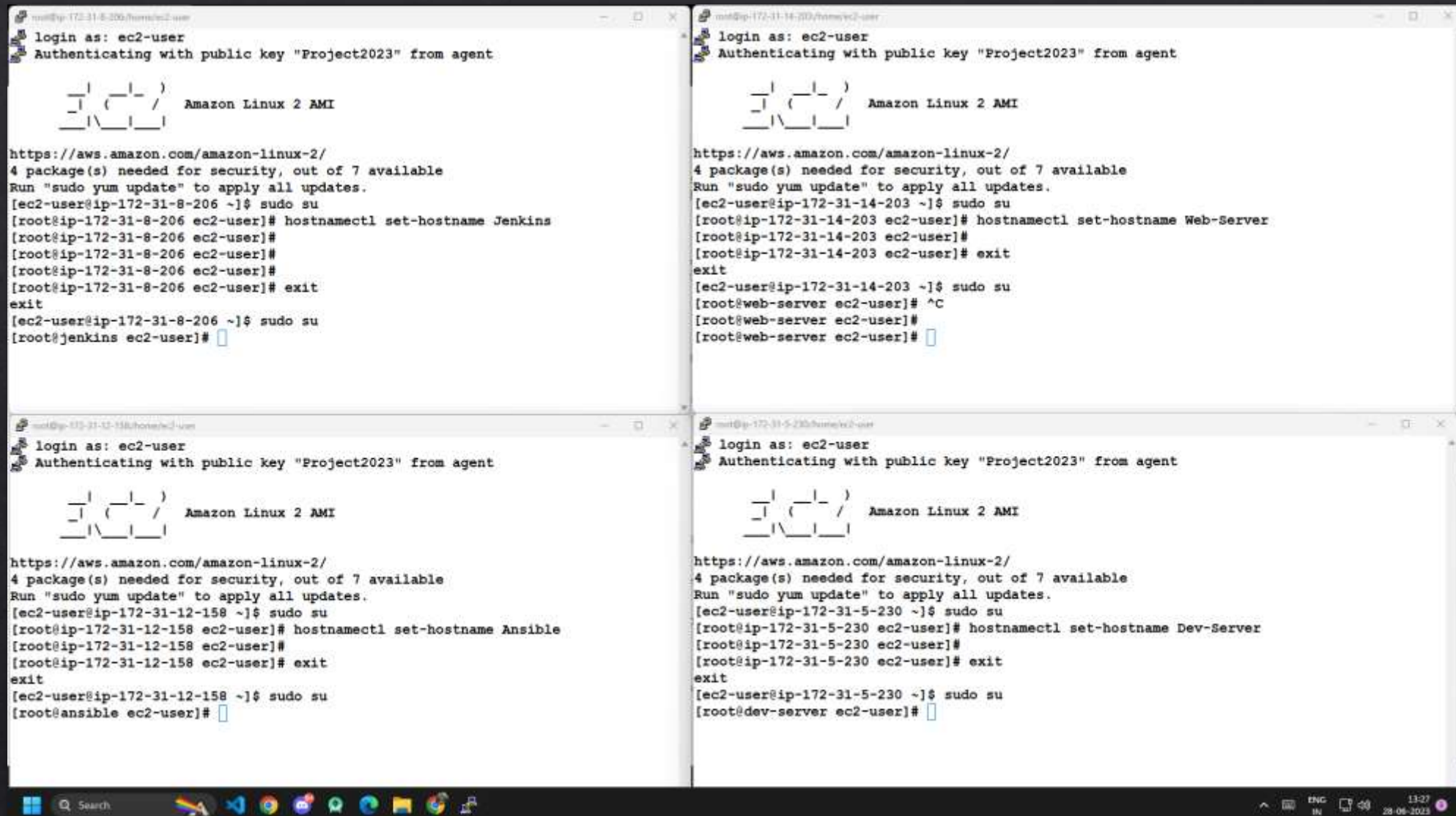
The screenshot displays the AWS Management Console interface for an EC2 instance named 'Jenkins-Server' (ID: i-0125031c45d152ab0). The instance is in a 'Running' state. A PuTTY Configuration window is overlaid on the console, showing the 'Basic options for your PuTTY session' tab. The 'Host Name (or IP address)' field is populated with '13.212.112.229', which is the public IP address of the instance. The 'Port' is set to '22' and the 'Connection type' is 'SSH'. The 'Close window on exit' option is set to 'Only on clean exit'. The background console shows various instance details such as Public IPv4 address, Private IPv4 addresses, Public IPv4 DNS, Elastic IP addresses, AWS Compute Optimizer Finding, Auto Scaling Group name, Monitoring, Termination protection, AMI location, Stop-hibernate behavior, State transition reason, and State transition message.

Open Putty And Paste the Jenkins Public IP in the Host Name.



Using Putty the Jenkins instance is connected successfully.





Likewise all the other servers are connected using putty accordingly.

Instance details | EC2 Manager - x | Change the hostname of your A x | Redhat Jenkins Packages x +

pkgs.jenkins.io/redhat/

Jenkins

Blog | Success Stories | Documentation | Plugins | Community | Subprojects | Security | About | Download

## Jenkins Redhat Packages

To use this repository, run the following command:

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat/jenkins.io-2023.key
```

If you've previously imported the key from Jenkins, the `rpm --import` will fail because you already have a key. Please ignore that and move on.

```
yum install fontconfig java-11-openjdk
yum install jenkins
```

The rpm packages were signed using this key:

```
pub rsa4096 2023-03-27 [SC] [expires: 2026-03-26]
63667EE748BA1FBA08A598725BA31D57EF5975CA
uid Jenkins Project
sub rsa4096 2023-03-27 [E] [expires: 2026-03-26]
```

You will need to explicitly install a supported Java runtime environment (JRE), either from your distribution (as described above) or another Java vendor (e.g., [Adoptium](#)).

## Weekly Release Line

Supported Java versions for the weekly release line are:

- 2.357 (June 2022) and newer**  
Java 11 or Java 17
- 2.164 (February 2019) and newer**  
Java 8 or Java 11
- 2.54 (April 2017) and newer**  
Java 8
- 1.612 (May 2015) and newer**

Windows taskbar: Search, 13:40, 28-06-2023

Install Jenkins from the official site using the wget command given in the site.

```
root@ip-172-31-0-206/home/ec2-user

44 python3.8          available [ =stable ]
45 haproxy2           available [ =stable ]
46 collectd           available [ =stable ]
47 aws-nitro-enclaves-cli available [ =stable ]
48 R4                 available [ =stable ]
   kernel-5.4         available [ =stable ]
50 selinux-ng         available [ =stable ]
51 php8.0             available [ =stable ]
52 tomcat9            available [ =stable ]
53 unbound1.13        available [ =stable ]
54 mariadb10.5        available [ =stable ]
55 kernel-5.10=latest enabled [ =stable ]
56 redis6             available [ =stable ]
57 ruby3.0            available [ =stable ]
58 postgresql12       available [ =stable ]
59 postgresql13       available [ =stable ]
60 mock2              available [ =stable ]
61 dnsmasq2.85        available [ =stable ]
62 kernel-5.15        available [ =stable ]
63 postgresql14       available [ =stable ]
64 firefox            available [ =stable ]
65 lustre             available [ =stable ]
66 php8.1             available [ =stable ]
67 awscli1            available [ =stable ]
68 php8.2             available [ =stable ]
69 dnsmasq            available [ =stable ]
70 unbound1.17        available [ =stable ]
71 golang1.19         available [ =stable ]
72 collectd-python3   available [ =stable ]

[root@jenkins ec2-user]# wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat/jenkins.repo
--2023-06-28 08:09:44-- https://pkg.jenkins.io/redhat/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 199.232.46.133, 2a04:4e42:48::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|199.232.46.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 71
Saving to: '/etc/yum.repos.d/jenkins.repo'

100%[=====>] 71 --.-K/s in 0s

2023-06-28 08:09:44 (4.78 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [71/71]

[root@jenkins ec2-user]#
[root@jenkins ec2-user]# rpm --import https://pkg.jenkins.io/redhat/jenkins.io-2023.key
[root@jenkins ec2-user]#
[root@jenkins ec2-user]# yum install jenkins -y
```

Now Jenkins successfully installed in the Jenkins-Server.

```
root@ip-172-31-12-138/home/ec2-user
[ =1.8.0_192 =1.8.0_202 =1.8.0_212 =1.8.0_222 =1.8.0_232
  =1.8.0_242 =stable ]
29 golang1.11          available \
   [ =1.11.3 =1.11.11 =1.11.13 =stable ]
30 squid4              available [ =4 =stable ]
32 lustre2.10          available \
   [ =2.10.5 =2.10.8 =stable ]
33 java-openjdk11      available [ =11 =stable ]
34 lynis               available [ =stable ]
36 BCC                 available [ =0.x =stable ]
37 mono               available [ =5.x =stable ]
38 nginx1             available [ =stable ]
40 mock               available [ =stable ]
41 postgresql11        available [ =11 =stable ]
43 livepatch           available [ =stable ]
44 python3.8           available [ =stable ]
45 haproxy2            available [ =stable ]
46 collectd            available [ =stable ]
47 aws-nitro-enclaves-cli available [ =stable ]
48 R4                 available [ =stable ]
   _ kernel-5.4        available [ =stable ]
50 selinux-ng          available [ =stable ]
51 php8.0              available [ =stable ]
52 tomcat9             available [ =stable ]
53 unbound1.13         available [ =stable ]
54 mariadb10.5         available [ =stable ]
55 kernel-5.10=latest  enabled [ =stable ]
56 redis6              available [ =stable ]
57 ruby3.0             available [ =stable ]
58 postgresql12        available [ =stable ]
59 postgresql13        available [ =stable ]
60 mock2               available [ =stable ]
61 dnsmasq2.85         available [ =stable ]
62 kernel-5.15         available [ =stable ]
63 postgresql14        available [ =stable ]
64 firefox             available [ =stable ]
65 lustre              available [ =stable ]
66 php8.1              available [ =stable ]
67 awscli1             available [ =stable ]
68 php8.2              available [ =stable ]
69 dnsmasq             available [ =stable ]
70 unbound1.17         available [ =stable ]
71 golang1.19          available [ =stable ]
72 collectd-python3    available [ =stable ]
[root@ansible ec2-user]# yum install ansible -y
```

Installing Ansible in Ansible Server.



```
root@ip-172-31-14-203:/home/ec2-user
(2/9): apr-util-1.6.3-1.amzn2.0.1.x86_64.rpm | 101 kB 00:00:00
(3/9): apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64.rpm | 22 kB 00:00:00
(4/9): generic-logos-httpd-18.0.0-4.amzn2.noarch.rpm | 19 kB 00:00:00
(5/9): httpd-filesystem-2.4.57-1.amzn2.noarch.rpm | 24 kB 00:00:00
(6/9): httpd-tools-2.4.57-1.amzn2.x86_64.rpm | 88 kB 00:00:00
(7/9): httpd-2.4.57-1.amzn2.x86_64.rpm | 1.4 MB 00:00:00
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00:00
(9/9): mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm | 149 kB 00:00:00
-----
Total | 8.9 MB/s | 1.9 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : apr-1.7.2-1.amzn2.x86_64 1/9
Installing : apr-util-1.6.3-1.amzn2.0.1.x86_64 2/9
Installing : apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64 3/9
Installing : httpd-tools-2.4.57-1.amzn2.x86_64 4/9
Installing : httpd-filesystem-2.4.57-1.amzn2.noarch 5/9
Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 6/9
Installing : mailcap-2.1.41-2.amzn2.noarch 7/9
Installing : mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
Installing : httpd-2.4.57-1.amzn2.x86_64 9/9
Verifying : apr-util-bdb-1.6.3-1.amzn2.0.1.x86_64 1/9
Verifying : apr-1.7.2-1.amzn2.x86_64 2/9
Verifying : httpd-tools-2.4.57-1.amzn2.x86_64 3/9
Verifying : apr-util-1.6.3-1.amzn2.0.1.x86_64 4/9
Verifying : mailcap-2.1.41-2.amzn2.noarch 5/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 6/9
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 7/9
Verifying : httpd-2.4.57-1.amzn2.x86_64 8/9
Verifying : httpd-filesystem-2.4.57-1.amzn2.noarch 9/9

Installed:
httpd.x86_64 0:2.4.57-1.amzn2

Dependency Installed:
apr.x86_64 0:1.7.2-1.amzn2 apr-util.x86_64 0:1.6.3-1.amzn2.0.1
apr-util-bdb.x86_64 0:1.6.3-1.amzn2.0.1 generic-logos-httpd.noarch 0:18.0.0-4.amzn2
httpd-filesystem.noarch 0:2.4.57-1.amzn2 httpd-tools.x86_64 0:2.4.57-1.amzn2
mailcap.noarch 0:2.1.41-2.amzn2 mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
[root@web-server ec2-user]# systemctl start httpd
[root@web-server ec2-user]#
```

Apache service is been installed successfully in the webserver.

```
root@jenkins/home/ec2-user
[-v] [-V] [-h] [-l] [-L] [-t] [-T] [-S] [-X]
Options:
-D name           : define a name for use in <IfDefine name> directives
-d directory      : specify an alternate initial ServerRoot
-f file           : specify an alternate ServerConfigFile
-C "directive"    : process directive before reading config files
-c "directive"    : process directive after reading config files
-e level          : show startup errors of level (see LogLevel)
-E file           : log startup errors to file
-v               : show version number
-V               : show compile settings
-h               : list available command line options (this page)
-l               : list compiled in modules
-L               : list available configuration directives
-t -D DUMP_VHOSTS : show parsed vhost settings
-t -D DUMP_RUN_CFG : show parsed run settings
-S               : a synonym for -t -D DUMP_VHOSTS -D DUMP_RUN_CFG
-t -D DUMP_MODULES : show all loaded modules
-M               : a synonym for -t -D DUMP_MODULES
-t -D DUMP_INCLUDES : show all included configuration files
-t               : run syntax check for config files
-T               : start without DocumentRoot(s) check
-X               : debug mode (only one worker, do not detach)

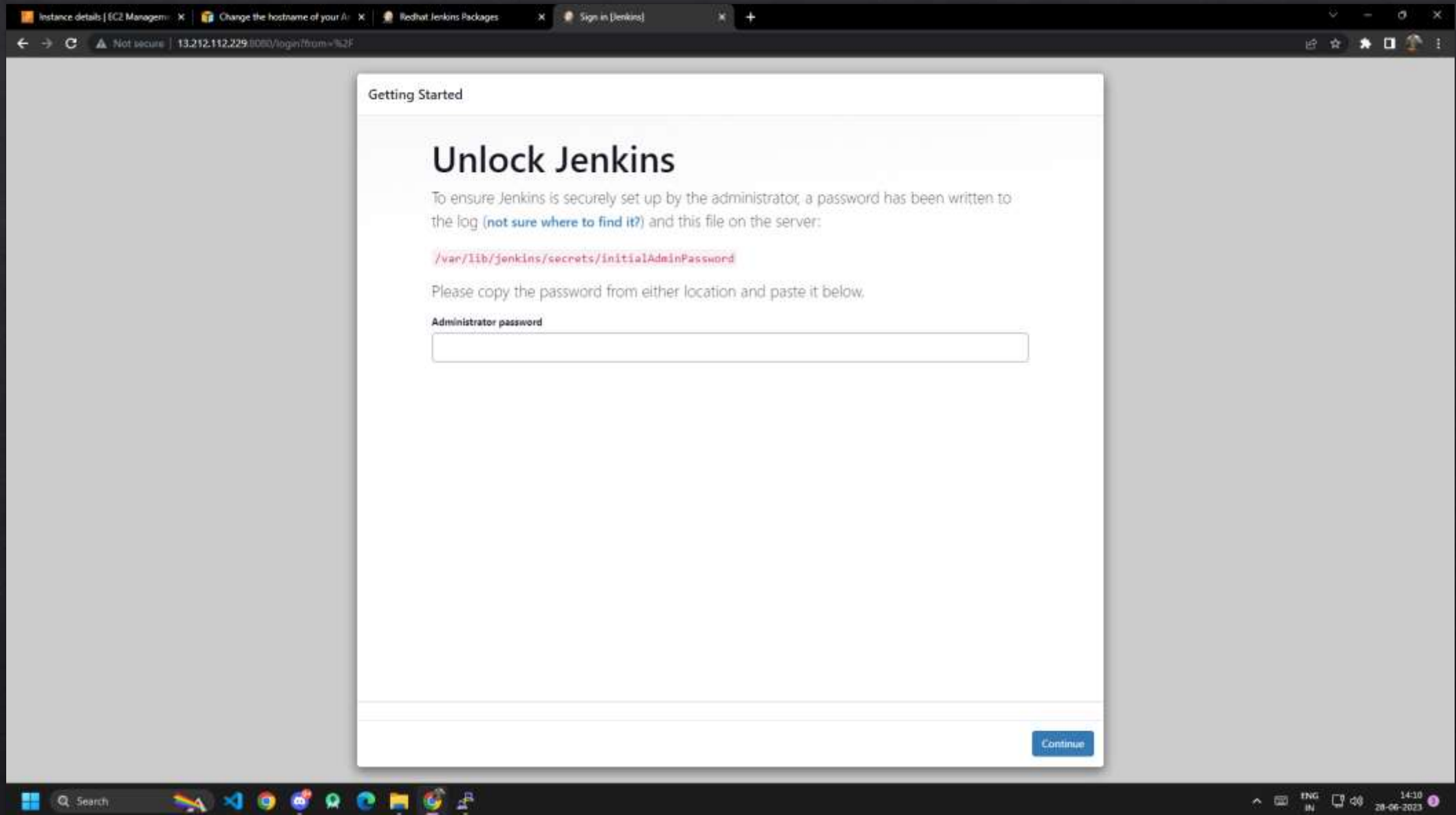
[root@jenkins ec2-user]# yum-get remove httpd -y
bash: yum-get: command not found
[root@jenkins ec2-user]# system ct
login as: ec2-user
Authenticating with public key "Project2023" from agent
Last login: Wed Jun 28 07:47:46 2023 from 49.37.212.50

    _ _ _ _ _
   _/   ( _ _ _ _ _
  _/_\ _/_/_/_/_

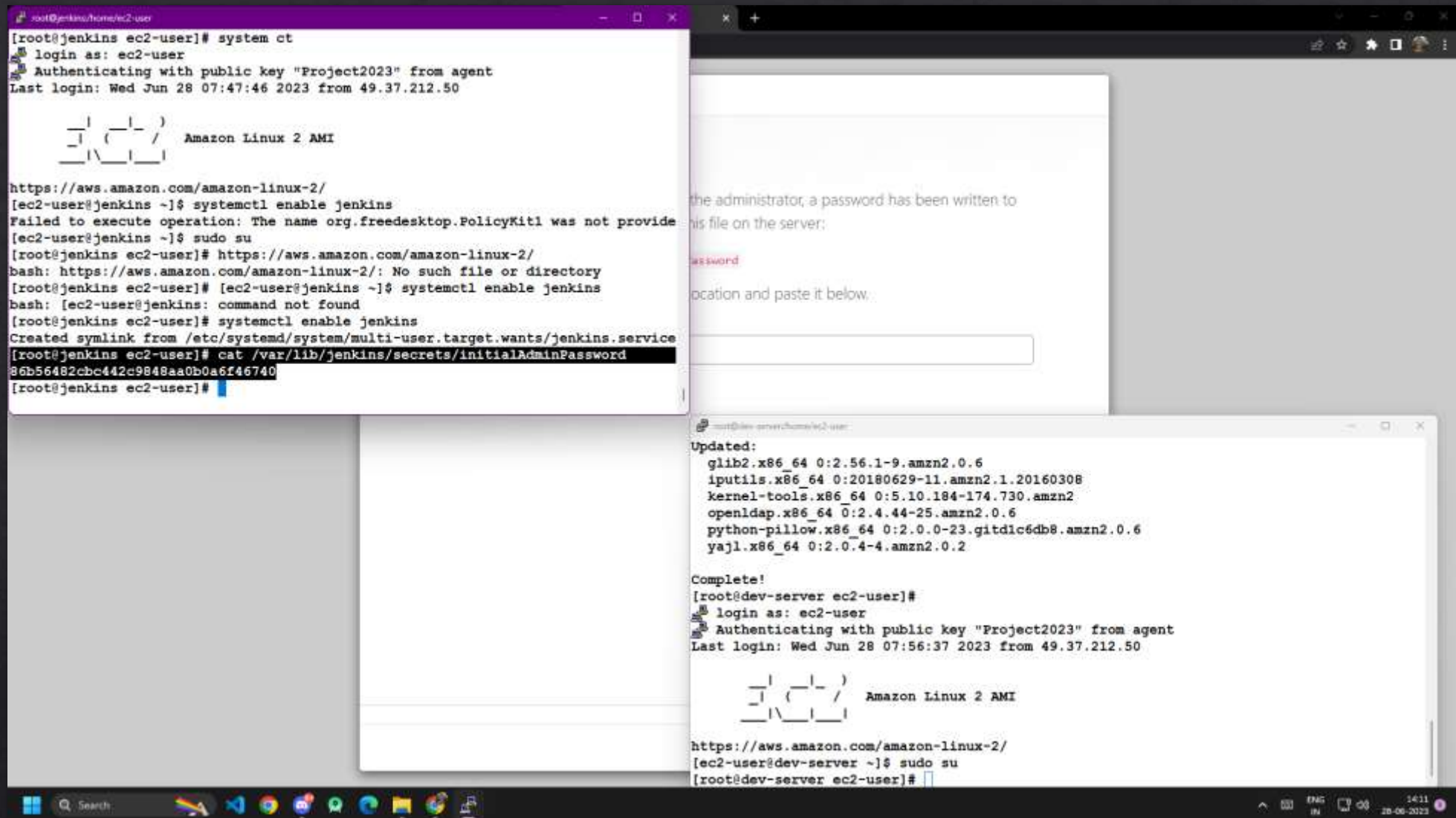
Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@jenkins ~]$ systemctl enable jenkins
Failed to execute operation: The name org.freedesktop.PolicyKit1 was not provided by any .service files
[ec2-user@jenkins ~]$ sudo su
[root@jenkins ec2-user]# https://aws.amazon.com/amazon-linux-2/
bash: https://aws.amazon.com/amazon-linux-2/: No such file or directory
[root@jenkins ec2-user]# [ec2-user@jenkins ~]$ systemctl enable jenkins
bash: [ec2-user@jenkins: command not found
[ec2-user@jenkins ~]$ systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
[root@jenkins ec2-user]#
```

Enabling Jenkins in Jenkins Server .

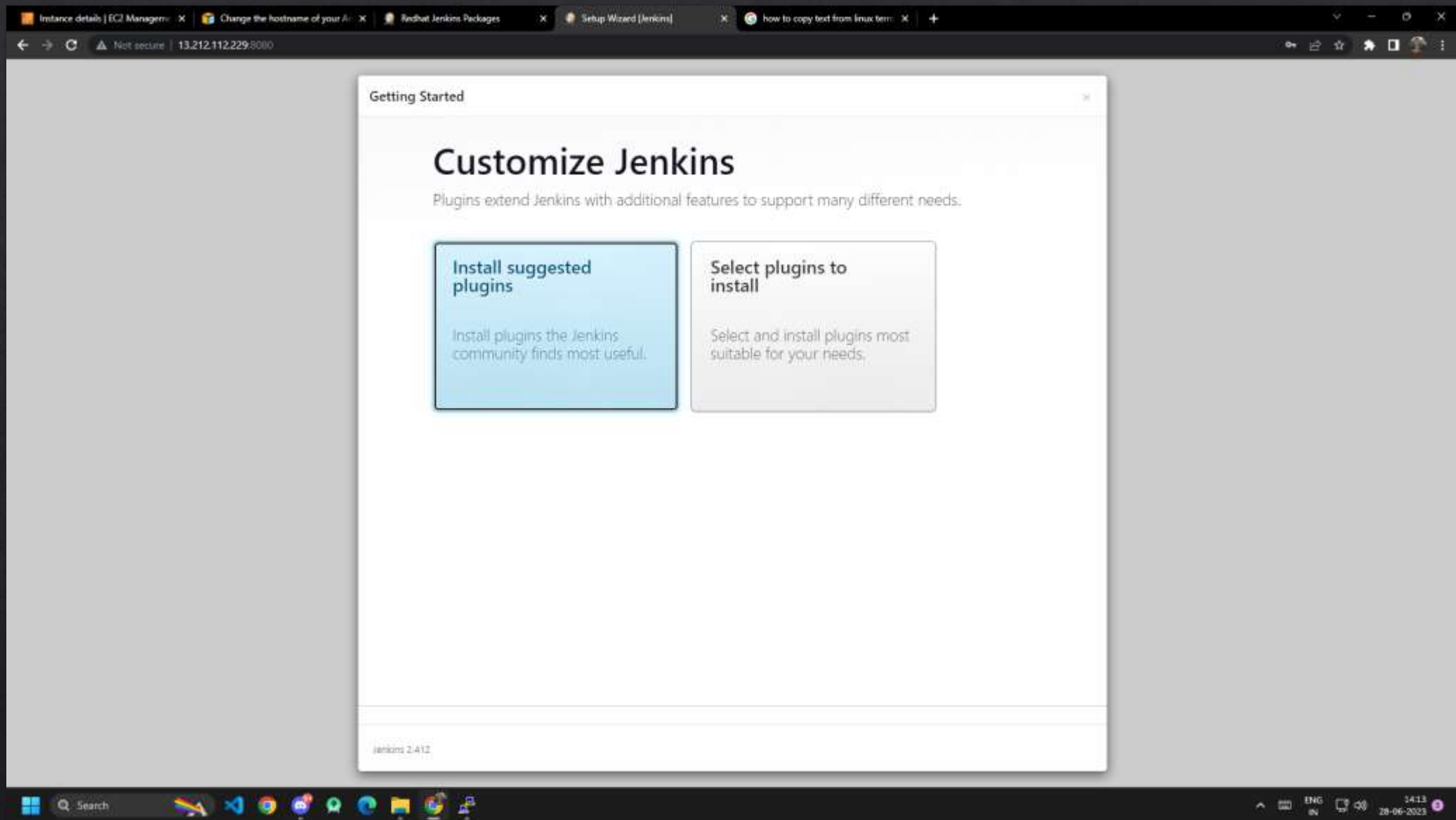


Jenkins is successfully started and enabled.



Jenkins initial Admin password is generated successfully by using this we'll be able to enter initiation page of Jenkin.





Lets gets started with Install suggested Plugins in Jenkins.

```
root@dev-server/home/ec2-user

perl-Error                noarch                1:0.17020-2.amzn2                amzn2-core                32 k
perl-Git                  noarch                2.40.1-1.amzn2.0.1              amzn2-core                41 k
perl-TermReadKey          x86_64                2.30-20.amzn2.0.2                amzn2-core                31 k

Transaction Summary

Install 1 Package (+5 Dependent packages)

Total download size: 13 M
Installed size: 44 M
Downloading packages:
(1/6): git-2.40.1-1.amzn2.0.1.x86_64.rpm                                | 54 kB 00:00:00
(2/6): git-core-doc-2.40.1-1.amzn2.0.1.noarch.rpm                      | 3.0 MB 00:00:00
(3/6): perl-Error-0.17020-2.amzn2.noarch.rpm                            | 32 kB 00:00:00
(4/6): git-core-2.40.1-1.amzn2.0.1.x86_64.rpm                          | 10 MB 00:00:00
(5/6): perl-Git-2.40.1-1.amzn2.0.1.noarch.rpm                          | 41 kB 00:00:00
(6/6): perl-TermReadKey-2.30-20.amzn2.0.2.x86_64.rpm                   | 31 kB 00:00:00
-----
Total                                                                    48 MB/s | 13 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : git-core-2.40.1-1.amzn2.0.1.x86_64                      1/6
  Installing : git-core-doc-2.40.1-1.amzn2.0.1.noarch                  2/6
  Installing : 1:perl-Error-0.17020-2.amzn2.noarch                      3/6
  Installing : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64                4/6
  Installing : perl-Git-2.40.1-1.amzn2.0.1.noarch                      5/6
  Installing : git-2.40.1-1.amzn2.0.1.x86_64                           6/6
  Verifying  : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64                1/6
  Verifying  : git-core-2.40.1-1.amzn2.0.1.x86_64                      2/6
  Verifying  : git-core-doc-2.40.1-1.amzn2.0.1.noarch                  3/6
  Verifying  : perl-Git-2.40.1-1.amzn2.0.1.noarch                      4/6
  Verifying  : 1:perl-Error-0.17020-2.amzn2.noarch                      5/6
  Verifying  : git-2.40.1-1.amzn2.0.1.x86_64                           6/6

Installed:
  git.x86_64 0:2.40.1-1.amzn2.0.1

Dependency Installed:
  git-core.x86_64 0:2.40.1-1.amzn2.0.1      git-core-doc.noarch 0:2.40.1-1.amzn2.0.1      perl-Error.noarch 1:0.17020-2.amzn2      perl-Git.noarch 0:2.40.1-1.amzn2.0.1
  perl-TermReadKey.x86_64 0:2.30-20.amzn2.0.2

Complete!
[root@dev-server ec2-user]#
```

Installing Git in Dev-Server Successfully.



```
root@ansible/home/ec2-user

  _ | _ | )
  _ | ( _ /
  _ | \ _ | _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ansible ~]$ sudo su
[root@ansible ec2-user]# vi /etc/ssh/sshd_config
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]# passwd root
Changing password for user root.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]# vi /etc/ssh/sshd_config
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]# systemctl restart sshd
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2023-06-28 09:26:05 UTC; 18s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 3411 (sshd)
    CGroup: /system.slice/sshd.service
            └─3411 /usr/sbin/sshd -D

Jun 28 09:26:05 ansible systemd[1]: Starting OpenSSH server daemon...
Jun 28 09:26:05 ansible sshd[3411]: Server listening on 0.0.0.0 port 22.
Jun 28 09:26:05 ansible sshd[3411]: Server listening on :: port 22.
Jun 28 09:26:05 ansible systemd[1]: Started OpenSSH server daemon.
[root@ansible ec2-user]#
```

Editing sshd\_config to access PasswordAuthenticatin to Yes.



```

root@jenkins/home/ec2-user
|..=*o+oo. |
|o.oEo=,=. |
|o=+...* + |
|o=+ o + o |
|.B. . S |
|o . . * o |
| o o o . o |
|..+ . |
| |
+----[SHA256]-----+
[root@jenkins ec2-user]#
[root@jenkins ec2-user]#
[root@jenkins ec2-user]# ssh-copy-id -i root@172.31.8.206
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.8.206 (172.31.8.206)' can't be established.
ECDSA key fingerprint is SHA256:1B37v9EQ0a2HazI3P4c/y1Owu2QqFt/5bNx61RRcbm4.
ECDSA key fingerprint is MD5:89:e6:c3:6b:f2:45:5e:da:2a:59:47:df:e0:91:42:72.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.8.206's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.8.206'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins ec2-user]# ssh-copy -i root@172.31.12.158
bash: ssh-copy: command not found
[root@jenkins ec2-user]# ssh-copy-id -i root@172.31.12.158
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.12.158 (172.31.12.158)' can't be established.
ECDSA key fingerprint is SHA256:3fo0c6a/df5oCgcSvocSIh2dmVVb1FU0H+Smmi22qtE.
ECDSA key fingerprint is MD5:46:fb:0a:97:a9:d3:ba:2d:6d:26:99:30:7d:48:ee:ee.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.12.158's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.12.158'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins ec2-user]#

```

Establishing ssh connection with Jenkins and Ansible.

```
root@jenkins ec2-user]# ssh-copy-id -i root@172.31.8.206
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.8.206 (172.31.8.206)' can't be established.
ECDSA key fingerprint is SHA256:1B37v9EQ0a2HazI3P4c/y1Owu2QqFt/5bNx61RRcbm4.
ECDSA key fingerprint is MD5:89:e6:c3:6b:f2:45:5e:da:2a:59:47:df:e0:91:42:72.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.8.206's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.8.206'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins ec2-user]# ssh-copy -i root@172.31.12.158
bash: ssh-copy: command not found
[root@jenkins ec2-user]# ssh-copy-id -i root@172.31.12.158
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.12.158 (172.31.12.158)' can't be established.
ECDSA key fingerprint is SHA256:3fo0c6a/df5oCgcSvocSIh2dmVVb1FU0H+SmMi22qtE.
ECDSA key fingerprint is MD5:46:fb:0a:97:a9:d3:ba:2d:6d:26:99:30:7d:48:ee:ee.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.12.158's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@172.31.12.158'"
and check to make sure that only the key(s) you wanted were added.

[root@jenkins ec2-user]# ssh root@172.31.12.158
Last failed login: Wed Jun 28 10:38:46 UTC 2023 from vm1338504.contaboserver.net on ssh:notty
There were 119 failed login attempts since the last successful login.
Last login: Wed Jun 28 08:38:37 2023

  _ | _ | _ |
  _ | ( _ | _ | /
  _ | \ _ | _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 2 packages available
Run "sudo yum update" to apply all updates.
[root@ansible ~]#
```

ssh successfully Installed between the Jenkins and Ansible.

```

root@web-server:~#
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:V3zGg58fcfmtKaDa5sqjGQCGf5in4DDp6aJ5Jf7whMo root@ansible
The key's randomart image is:
+---[RSA 2048]-----+
|
| .      . o . |
| o.      + =o. |
|.+ o      . + o=|
| = = o    S o  o.o|
|+oo*.     o .  +.|
| =+oo     . .  o .|
|+oo+ +.o.   .   |
|*E .,+.+=o   |
+----[SHA256]-----+
[root@ansible ec2-user]#
[root@ansible ec2-user]#
[root@ansible ec2-user]# ssh-copy-id -i root@172.31.14.203
/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
The authenticity of host '172.31.14.203 (172.31.14.203)' can't be established.
ECDSA key fingerprint is SHA256:a3X9PLMaJ3Nqp0xMsPFxw3YUEieacUJl0Mx3cT5j1HI.
ECDSA key fingerprint is MD5:98:fb:42:3d:01:4a:76:73:6b:e0:1e:13:50:c8:ad:07.
Are you sure you want to continue connecting (yes/no)? yes
/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
root@172.31.14.203's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'root@172.31.14.203'"
and check to make sure that only the key(s) you wanted were added.

[root@ansible ec2-user]# ssh root@172.31.14.203
Last login: Wed Jun 28 08:38:26 2023

  _ | _ | _ |
 _ | ( _ | _ | /
 _ | \ _ | _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[root@web-server ~]#

```

Establishing ssh connection with Ansible and Web-Server Successfully.

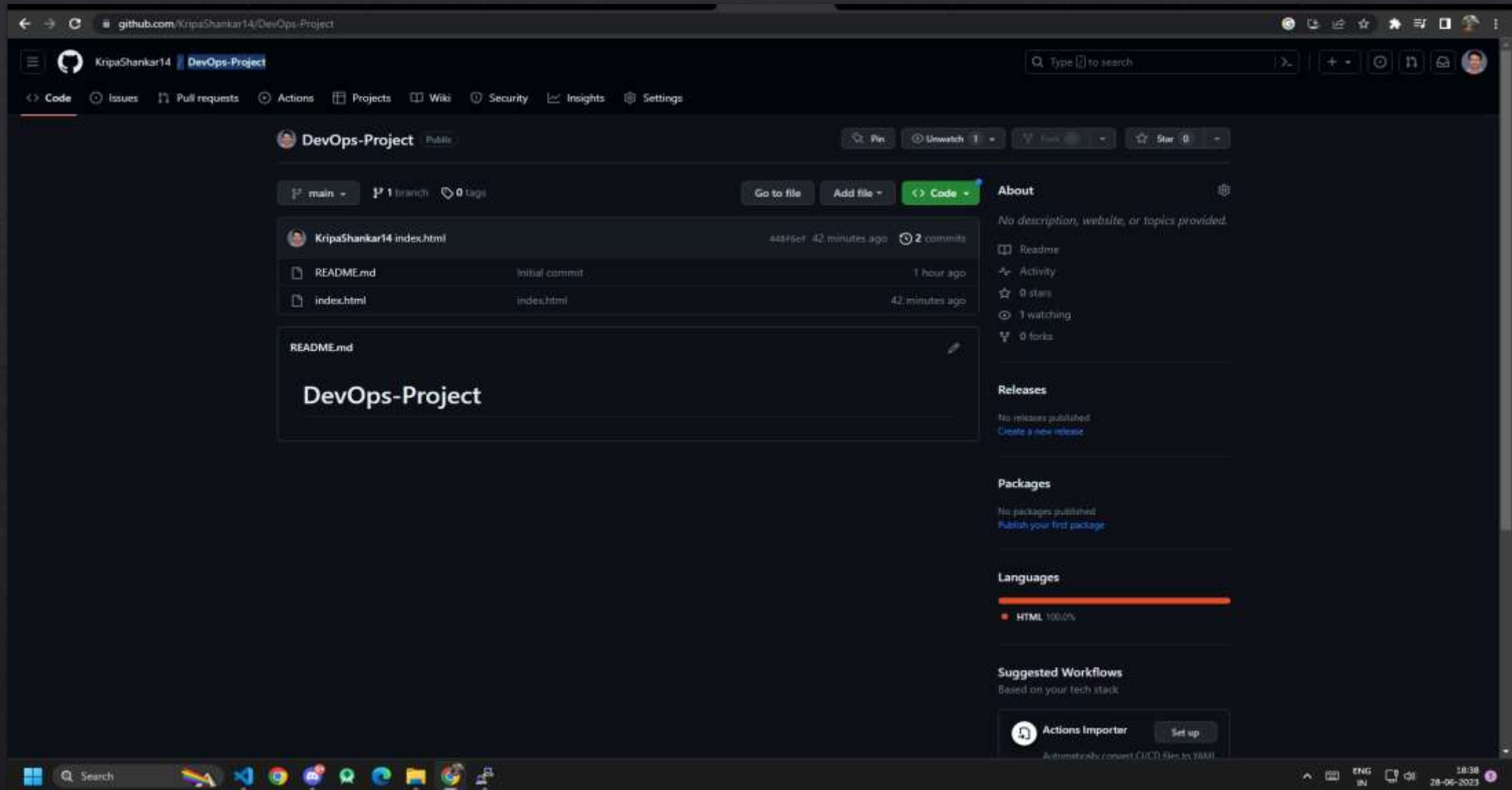


```
root@ansible:/home/ec2-user#  
# This is the default ansible 'hosts' file.  
#  
# It should live in /etc/ansible/hosts  
#  
# - Comments begin with the '#' character  
# - Blank lines are ignored  
# - Groups of hosts are delimited by [header] elements  
# - You can enter hostnames or ip addresses  
# - A hostname/ip can be a member of multiple groups  
#  
# Ex 1: Ungrouped hosts, specify before any group headers.  
## green.example.com  
## blue.example.com  
## 192.168.100.1  
## 192.168.100.10  
[webserver]  
172.31.14.203  
# Ex 2: A collection of hosts belonging to the 'webservers' group  
## [webservers]  
## alpha.example.org  
## beta.example.org  
## 192.168.1.100  
## 192.168.1.110  
# If you have multiple hosts following a pattern you can specify  
# them like this:  
## www[001:006].example.com  
# Ex 3: A collection of database servers in the 'dbservers' group  
## [dbservers]  
##  
## db01.intranet.mydomain.net  
## db02.intranet.mydomain.net  
## 10.25.1.56  
## 10.25.1.57  
# Here's another example of host ranges, this time there are no  
# leading 0s:  
## db-[99:101]-node.example.com  
-- INSERT --
```

18,1 Top

In the ansible cfg file add Webserver's Private IP.





Open Github create a new repo and name its DevOps-Project  
And add index.html file contains a content.

← → ↻ ⚠ Not secure | 13.212.112.229:8080/job/DevOps-Project/configure

Dashboard > DevOps-Project > Configuration

### Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps
- Post-build Actions

#### Source Code Management

☐ None

☒ Git

Repositories

Repository URL

`https://github.com/KripaShankar14/DevOps-Project.git`

Credentials

- none -

Add

Advanced

Add Repository

Branches to build

Branch Specifier (blank for 'any')

`*/master`

Add Branch

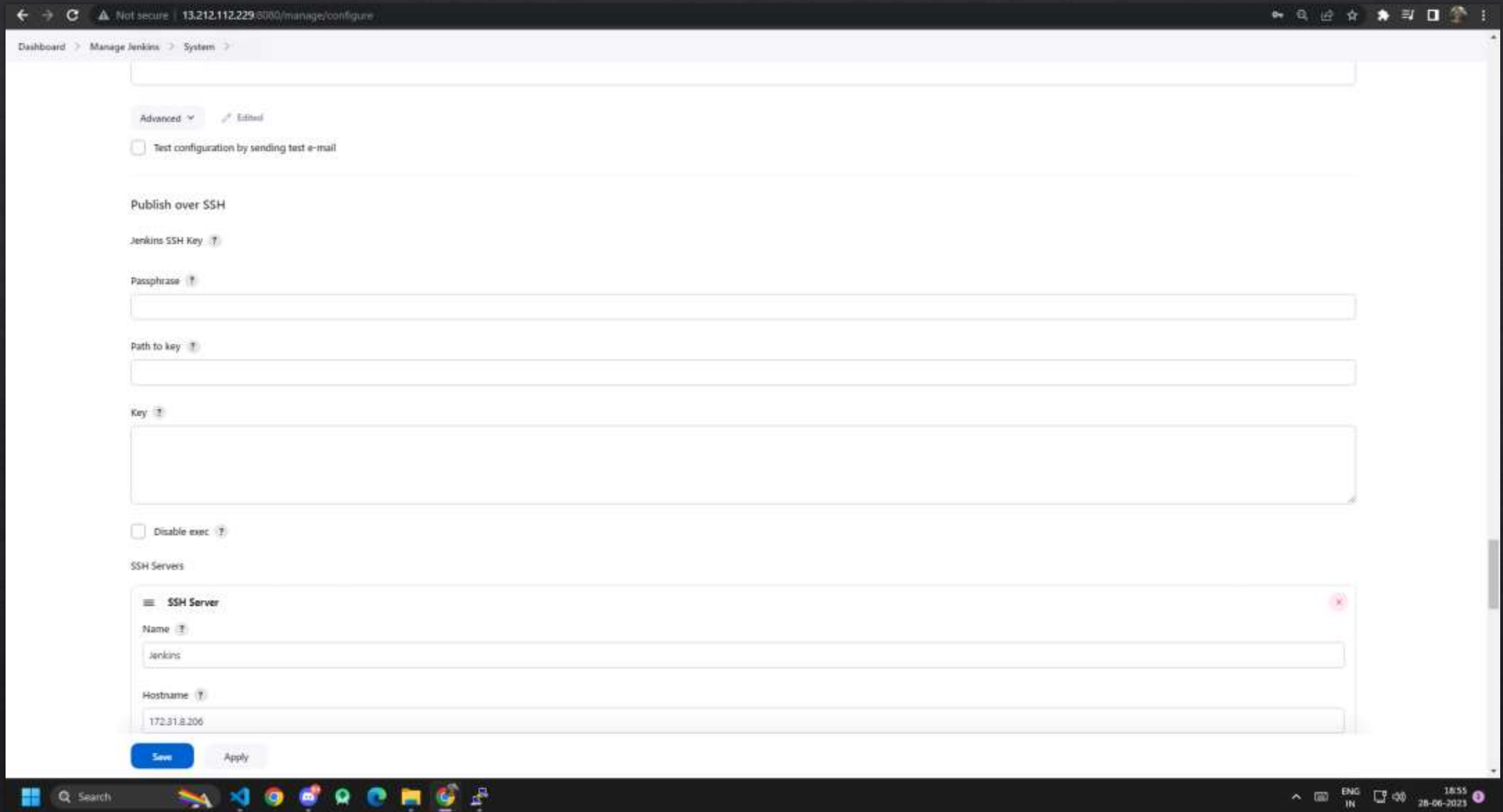
Repository browser

(Auto)

Save Apply

Windows taskbar: Search, Task View, Edge, Teams, File Explorer, Settings, DevOps-Project, System Tray: ENG IN, 14:38, 28-06-2023

Open Jenkins > new Item > add the git url in the Repo url.



Adding SSH server – jenkins @ privatekey of jenkins.

Instance details | EC2 Managem... x DevOps-Project/ at main - Kri... x DevOps-Project Config [Jenkins] x jenkins.plugins.publish\_over.Bapi... x (B64) DevOps Live Project -1 in | x +

← → ↻ ⚠ Not secure 13.212.112.229:8080/job/DevOps-Project/configure

Dashboard > DevOps-Project > Configuration

### Configure

- General
- Source Code Management
- Build Triggers
- Build Environment**
- Build Steps
- Post-build Actions

☐ Send files or execute commands over SSH before the build starts ⓘ  
☐ Send files or execute commands over SSH after the build runs ⓘ  
☐ Add timestamps to the Console Output  
☐ Inspect build log for published build scans  
☐ Terminate a build if it's stuck  
☐ With Ant ⓘ

### Build Steps

Send files or execute commands over SSH ⓘ

SSH Publishers

SSH Server:  
Name ⓘ  
Jenkins

Advanced ⓘ

Transfers

Transfer Set  
Source files ⓘ  

Either Source files, Exec command or both must be supplied

Remove prefix ⓘ

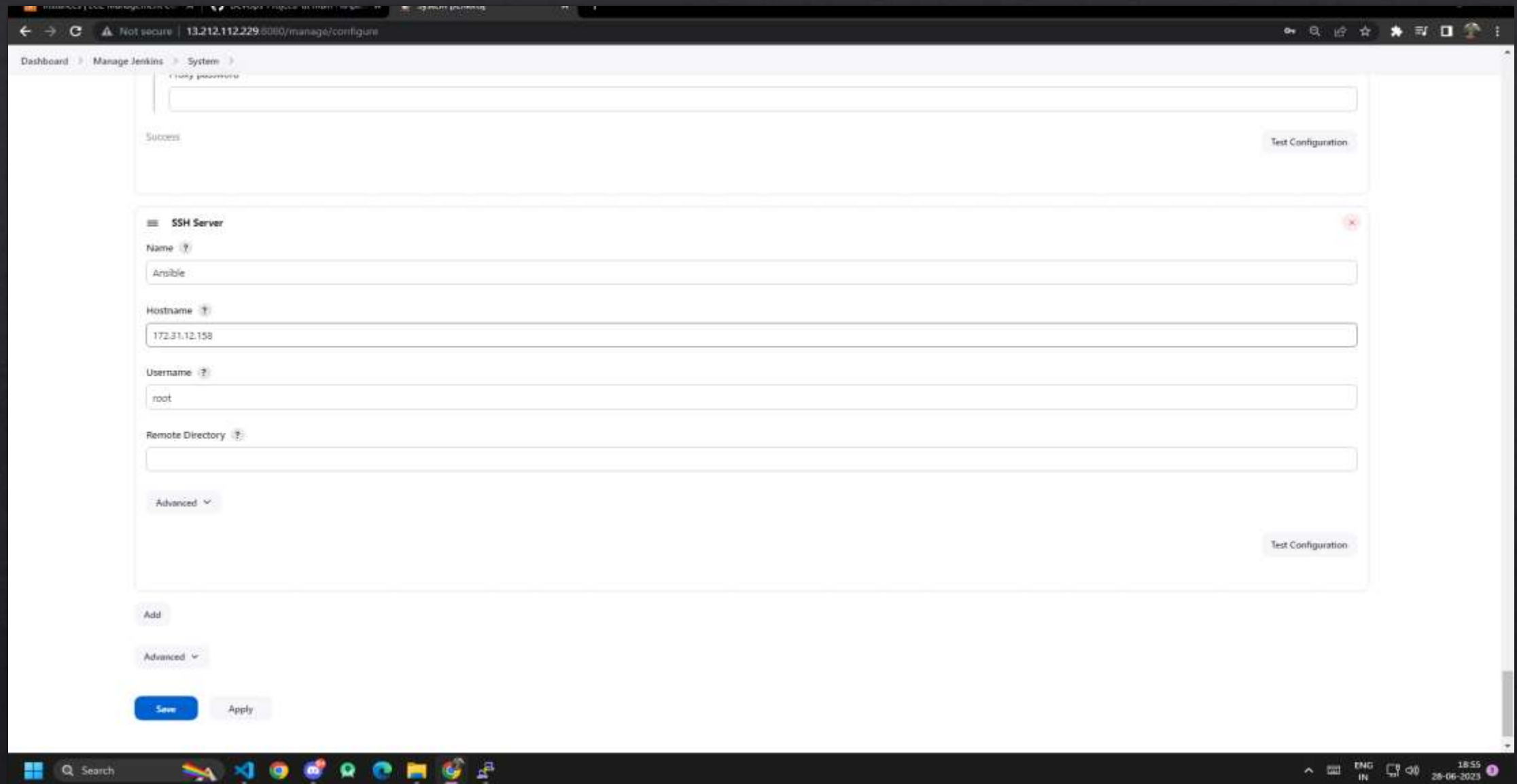
Remote directory ⓘ

Save Apply

Windows taskbar: Search, File Explorer, Edge, Teams, OneDrive, etc. System tray: ENG IN, 19:14, 28-06-2023

At Built Steps add SSH Publishers –Jenkins.





Similarly add Ansible at the SSH server with Host name @ privatekey.

← → ↻ Not secure | 13.212.112.229:8080/job/DevOps-Project/configure

Dashboard > DevOps-Project > Configuration

## Configure

- General
- Source Code Management
- Build Triggers
- Build Environment
- Build Steps**
- Post-build Actions

### Build Steps

Send files or execute commands over SSH ?

SSH Publishers

SSH Server Name ?

Jenkins

Advanced ▾

Transfers

Transfer Set ?

Source files ?

Either Source files, Exec command or both must be supplied

Remove prefix ?

Remote directory ?

Exec command ?

```
rsync -aH /var/lib/jenkins/workspace/DevOps-Project/*.html root@172.31.12.158:/opt/index.html
```

Save Apply

Windows taskbar: Search, 19:21, 28-06-2023

Add Exec – Command to call the repo.

```
rust@ansible:sourcecode
- name: Pick the file from the source and push it into the destination
- hosts: webserver
  tasks:
  - copy:
    src: /opt/index.html
    dest: /var/www/html
```

-- INSERT --

6,7 All

Write the playbook in the Ansible server at the mkdir /sourcecode  
named deployment.yal .

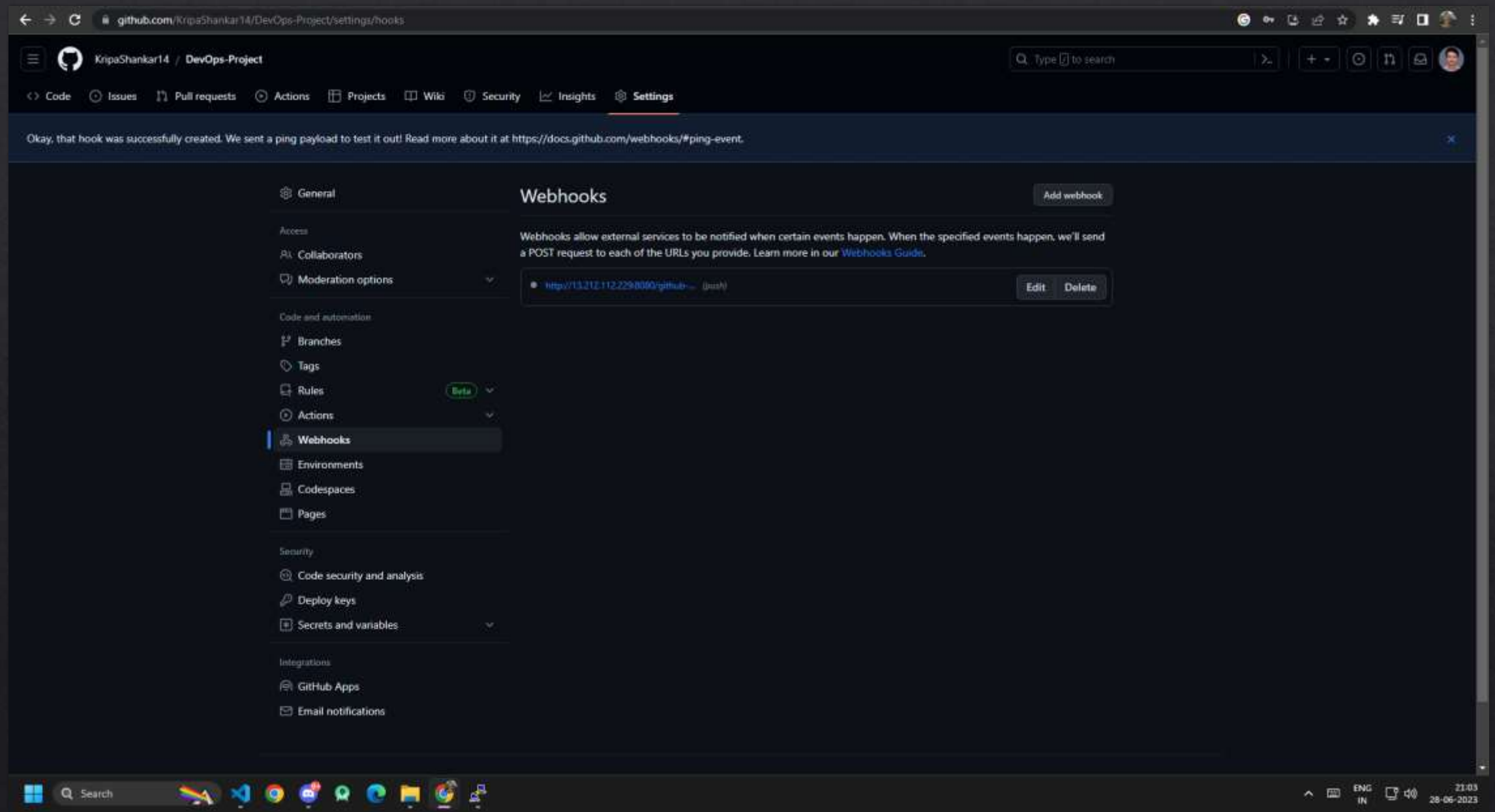
The screenshot shows the GitHub web interface for a repository named 'DevOps-Project' under the user 'KripaShankar14'. The 'Settings' tab is selected, and the 'Webhooks' section is highlighted in the left sidebar. The main content area is titled 'Webhooks / Add webhook' and contains the following fields and options:

- Payload URL:** A text input field containing the URL `http://13.212.112.229:8080/github-webhook/`.
- Content type:** A dropdown menu set to `application/json`.
- Secret:** An empty text input field for a secret key.
- Which events would you like to trigger this webhook?** Three radio button options:
  - ☒ Just the push event.
  - ☐ Send me everything.
  - ☐ Let me select individual events.
- Active:** A checkbox that is checked, with a subtext: 'We will deliver event details when this hook is triggered.'
- Add webhook:** A green button to save the configuration.

At the bottom of the page, there is a footer with copyright information '© 2023 GitHub, Inc.' and various links like Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About. The Windows taskbar is visible at the very bottom, showing the time as 21:01 on 28-06-2023.

Add Webhook at the git by payloading the url of home screen of jenkins.





Webhook successfully added.

← → ↻ ⚠ Not secure | 13.212.112.229:8080/job/DevOps-Project/configure

Dashboard > DevOps-Project > Configuration

### Configure

- General
- Source Code Management
- Build Triggers**
- Build Environment
- Build Steps
- Post-build Actions

#### Build Triggers

- ☐ Trigger builds remotely (e.g., from scripts) ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☒ GitHub hook trigger for GITScm polling ?
- ☐ Poll SCM ?

#### Build Environment

- ☐ Delete workspace before build starts
- ☐ Use secret text(s) or file(s) ?
- ☐ Send files or execute commands over SSH before the build starts ?
- ☐ Send files or execute commands over SSH after the build runs ?
- ☐ Add timestamps to the Console Output
- ☐ Inspect build log for published build scans
- ☐ Terminate a build if it's stuck
- ☐ With Ant ?

#### Build Steps

☰ Send files or execute commands over SSH ?


SSH Publishers

Save


Apply


Turn on the GitHub hook trigger.


← → ↻ ⚠ Not secure | 13.212.112.229:8080/job/DevOps-Project-Kripa/10/console


 **Jenkins** 🔍 Search (CTRL+K) 🛡️ 1 👤 Kripashankar 🚪 log out


Dashboard > DevOps-Project-Kripa > #10 > Console Output


 Status


 Changes


 Console Output


 View as plain text

 Edit Build Information

 Delete build '#10'

 Previous Build

 Next Build

 **Console Output**

```
Started by user Kripashankar
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/DevOps-Project
The recommended git tool is: NONE
No credentials specified
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/DevOps-Project/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/KripaShankar14/DevOps-Project.git # timeout=10
Fetching upstream changes from https://github.com/KripaShankar14/DevOps-Project.git
> git --version # timeout=10
> git --version # 'git version 2.40.1'
> git fetch --tags --force --progress -- https://github.com/KripaShankar14/DevOps-Project.git +refs/heads/*:refs/remotes/origin/* # timeout=10
> git rev-parse refs/remotes/origin/master^{commit} # timeout=10
> git rev-parse origin/master^{commit} # timeout=10
```

Finally the content in the index.html has showed up .

This is how the CI/CD pipeline works.

# Things Learned From This Project

## *Learnings:*

- *Creating instances and installing Custom Os*
- *Installing the appropriate services accordingly*
- *Configuring Jenkins and connecting with Git*
- *Installing SSH connection Between the Servers*
- *push and pull data from each other accordingly*