

Advance DevOps Practical Exam

Introduction

Case Study Overview

The case study involves integrating static analysis tools with **Infrastructure-as-Code (IaC)** using Terraform. This project focuses on setting up a development environment that combines **Jenkins** for continuous integration and deployment, **SonarQube** for static code analysis, and **Terraform** for infrastructure provisioning. The goal is to create a robust pipeline that can automatically analyze Python applications for quality and security issues.

Key Features and Applications:

Key features of this case study include:

- Infrastructure deployment using Terraform
- Continuous Integration/Continuous Deployment (CI/CD) pipeline setup
- Static code analysis integration
- Cross-tool communication between Jenkins, SonarQube, and Terraform

The practical applications of this setup are:

- Improved code quality through automated static analysis
- Enhanced security by identifying potential vulnerabilities early in the development cycle
- Streamlined development workflow with automated testing and reporting
- Scalable infrastructure management using Terraform

Step-by-Step Explanation

Terraform

Step 1:

Install terraform and add it to environment variable. Now, download Amazon CLI by visiting the following website.

Visit <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

AWS Documentation AWS Command Line Interface User Guide for Version 2

Contact Us English Return to the Console

Feedback Preferences

On this page

AWS CLI install and update instructions

Troubleshooting AWS CLI install and uninstall errors
Next steps

Recently added to this guide

- Amazon ECR Public examples using AWS CLI 20 September 2024
- Route 53 Profiles examples using AWS CLI 20 September 2024
- Security Lake examples using AWS CLI 20 September 2024

Install and update requirements

- We support the AWS CLI on Microsoft-supported versions of 64-bit Windows.
- Admin rights to install software

Install or update the AWS CLI

To update your current installation of AWS CLI on Windows, download a new installer each time you update to overwrite previous versions. AWS CLI is updated regularly. To see when the latest version was released, see the [AWS CLI version 2 Changelog](#) on GitHub.

- Download and run the AWS CLI MSI installer for Windows (64-bit):
<https://awscli.amazonaws.com/AWSCLIV2.msi>
Alternatively, you can run the `msiexec` command to run the MSI installer.
`C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi`
- To confirm the installation, open the Start menu, search for `cmd` to open a command prompt

Now, click on install <https://awscli.amazonaws.com/AWSCLIV2.msi>

Complete the installation process for AWSCLIV2

Downloads

Search Downloads

New + Details

Name	Date modified	Type
AdvDevOps_practical	19-10-2024 02:45	Microsoft Word D...
AWSCLIV2	19-10-2024 02:06	Windows Installer ...
AMjenkins.pem	18-10-2024 19:23	PEM File
AMsonarqube.pem	18-10-2024 21:40	PEM File
Anuprita_Resume	13-10-2024 21:51	Microsoft Edge HT...
CNS_Assignment_4	14-10-2024 22:13	Microsoft Word D...
Siddhesh_Resume	17-10-2024 00:43	Adobe Acrobat D...
Anuprita_Resume	13-10-2024 21:55	Adobe Acrobat D...
CNS Assignment 4_Sneha	14-10-2024 22:14	Adobe Acrobat D...
CNS_Assignment_4	14-10-2024 22:13	Adobe Acrobat D...
DevOps_exp7	17-10-2024 02:44	Adobe Acrobat D...
DevOps_exp6	17-10-2024 23:34	Adobe Acrobat D...
DevOps_exp7	17-10-2024 02:43	Microsoft Word D...

Downloads (105 items)

Select a single file to get more information and share your cloud content.

Step 2:

Open AWS Academy and now click on AWS Details and then click on show button present in front of AWS CLI label. You will be shown with your cedentials

The screenshot shows the AWS Academy Learner Lab interface. At the top right, there is a 'AWS Details' button with a 'Show' link. Below it, a sidebar lists various AWS services: Environment Overview, Environment Navigation, Access the AWS Management Console, Region restriction, Service usage and other restrictions, Using the terminal in the browser, Running AWS CLI commands, Using the AWS SDK for Python, Preserving your budget, Accessing EC2 Instances, SSH Access to EC2 Instances, SSH Access from Windows, and SSH Access from a Mac. A note at the bottom says 'Instructions last updated: 2024-08-06'. The main area is titled 'Environment Overview'.

This screenshot shows the 'AWS Details' section expanded. It includes a 'Cloud Access' section with a 'Close' button. Under 'Cloud Access', there is a 'AWS CLI' section with a 'Show' button. Below that is a 'Cloud Labs' section showing session details: 'Remaining session time: 00:42:11(43 minutes)', 'Session started at: 2024-10-18T10:50:58-0700', and 'Session to end at: 2024-10-18T14:50:58-0700'. It also displays accumulated lab time: '1 day 06:28:00 (1828 minutes)'. The 'Cloud Access' section also lists 'No running instance', 'SSH key' (with 'Show' and 'Download PEM' buttons), 'Download PPK' (button), 'AWS SSO' (button), and 'Download URL' (button). At the bottom, there is a table with two rows: 'AWSAccountId' (856746069793) and 'Region' (us-east-1).

The screenshot shows the AWS Academy Learner Lab interface. On the left, there's a sidebar with icons for Account, Dashboard, Courses, Calendar, Inbox, History, and Help. The main area has a breadcrumb navigation: ALLv2EN-US... > Modules > AWS Academy... > Launch AWS Academy Learner Lab. The top right shows 'Used \$1.6 of \$50' and a timer at '00:42'. Below the breadcrumb, there are tabs for Home, Modules (selected), Discussions, Grades, and Lucid (Whiteboard). A notification icon with the number '1' is visible next to the Grades tab. The central area contains a terminal window with the command 'eee_l_3429984@runweb141049:~\$'. To the right of the terminal is a 'Cloud Access' panel with a 'Close' button. It contains instructions to copy and paste AWS CLI credentials into a file named '.aws/credentials'. The credentials themselves are a long string of characters.

Step 3:

Now, create a folder in VSCode and create a main.tf file in it with the following content.

```
# Specify the AWS provider
provider "aws" {

    region = "us-east-1" # Replace with your preferred region
}

# Jenkins instance

resource "aws_instance" "jenkins" {

    ami      = "ami-05f408238af346b4f" # Amazon Linux 2 AMI
    instance_type = "t2.micro"
    key_name   = "AMjenkins"
    tags = {
        Name = "JenkinsServer"
    }

    # User data to install Jenkins with Java 17
    user_data = <<-EOF
#!/bin/bash
sudo yum update -y
sudo dnf install -y java-17-amazon-corretto-devel # Install Java 17
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
sudo yum install -y Jenkins
sudo systemctl start Jenkins
sudo systemctl enable Jenkins
EOF
}
```

```

# SonarQube instance
resource "aws_instance" "sonarqube" {

    ami      = "ami-05f408238af346b4f" # Amazon Linux 2 AMI
    instance_type = "t2.medium"
    key_name     = "AMsonarqube"
    tags = {
        Name = "SonarQubeServer"
    }

    # User data to install SonarQube manually
    user_data = <<-EOF
#!/bin/bash
sudo yum update -y
sudo su -
cd /opt
wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-10.7.0.96327.zip
unzip sonarqube-10.7.0.96327.zip
sudo adduser sonar
sudo passwd sonar
sudo chown -R sonar:sonar /opt/sonarqube-10.7.0.96327
su - sonar -c "/opt/sonarqube-10.7.0.96327/bin/linux-x86-64/sonar.sh start"
EOF
}

```

}

```

File Edit Selection View Go Run Terminal Help ⏪ ⏩ AdvDevOps_Practical
EXPLORER ... main.tf x hello.py
main.tf
1 # Specify the AWS provider
2 provider "aws" {
3   region = "us-east-1" # Replace with your preferred region
4 }
5
6 # Jenkins instance
7 resource "aws_instance" "jenkins" {
8   ami      = "ami-05f408238af346b4f" # Amazon Linux 2 AMI
9   instance_type = "t2.micro"
10  key_name     = "AMjenkins"
11  tags = {
12    Name = "JenkinsServer"
13  }
14
15  # User data to install Jenkins with Java 17
16  user_data = <<-EOF
17  #!/bin/bash
18  sudo yum update -y
19  sudo dnf install -y java-17-amazon-corretto-devel # Install Java 17
20  sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
21  sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key
22  sudo yum install -y jenkins
23  sudo systemctl start jenkins
24  sudo systemctl enable jenkins
25
26 } # Jenkins instance
27
28 # SonarQube instance
29 resource "aws_instance" "sonarqube" {
30   ami      = "ami-05f408238af346b4f" # Amazon Linux 2 AMI
31   instance_type = "t2.medium"
32   key_name     = "AMsonarqube"
33   tags = {
34     Name = "SonarQubeServer"
35   }
36
37  # User data to install SonarQube manually
38  user_data = <<-EOF
39  #!/bin/bash
40  sudo yum update -y
41  sudo su -
42  cd /opt

```

Step 4:

Now, to ensure and run the aws cli commands in vs code terminal, run the following commands

aws –version
aws configure

(Write the content as mentioned in the figure below)

The screenshot shows a dark-themed instance of Visual Studio Code. In the Explorer sidebar, there is a folder named 'ADVDEVOPS_PRACTICAL' containing files 'hello.py' and 'main.tf'. The 'hello.py' file has the code: 'print("Hello, World!")'. The 'TERMINAL' tab is active, displaying the following command-line session:

```
PS D:\Sem5_anuprita\AdvDevops_Practical> aws --version
aws-cli/2.18.10 Python/3.12.6 Windows/11 exe/AMD64
PS D:\Sem5_anuprita\AdvDevops_Practical> aws configure
● AWS Access Key ID [None]: ASIA4O6QLR4QWMA6EBFG
AWS Secret Access Key [None]: g101Lg880R5lgoNx0OE8oX3TNhxBcJ5u8fTUHuU
Default region name [None]: us-east-1
Default output format [None]: json
○ PS D:\Sem5_anuprita\AdvDevops_Practical>
```

The status bar at the bottom indicates the terminal is in line 1, column 23, with spaces 4, CRLF, Python 3.12.0 64-bit, Go Live, and Prettier.

Step 5:

Now, run the following commands in the vs code terminal to set the credential secrets.

```
$env:AWS_ACCESS_KEY_ID="ASIA4O6QLR4QWMA6EBFG"
$env:AWS_SECRET_ACCESS_KEY="g101Lg880R5lgoNx0OE8oX3TNhxBcJ5u8fTUHuU"
$env:AWS_SESSION_TOKEN="IQoJb3JpZ2luX2VjEO//////////wEaCXVzLXdIc3QtMiJIMEYCIQDFth+aigG/gP+Z6F3r+MqGoz
mnWIMrKNwWhAdlQybeowlhANM0Vv37FoO8JPYclXeF9WdzbBVNXuzCyL3CFTilbFITKrICCFgQARoMODU2NzQ2MDY5Nzk
zlgzfb754SQVtRN0LY6gqjwLRheg15jeclfhu956B4lw1wh7nh8uAbmksRgzTFn0f/XRdNDp8umD5361F3CB0Kw0y3u2iUiappL
GRgmAls6Ipwx91OaD/IziE+J6jqA9werjjE+vPSwJZDjBAI3AOsXEqrHHOxOCMnV8tDADefFFdq4/TVCpj7XrZZbr/bI9b33kA28Y
b02vxwFpXXwCjcuPtkae+kPC5v74lAb1S0gEXudqQv1okS7d+M6hg3qOqrIswTZJ52IZHDZYbeavRrU5qv8W+ky95bOhy1fb3
HKUSGKnpS6Pu2IDqjG97SiAFiB6YCxBzq4UxFJWVreSMILwMPzbL7nq87z3lbRNHbJ3RylWA0TLNBwNNKT92DPyqMP+jzbgG
OpwBBmh89toqaGEkit4lMz7vMppk0w4Q6pgBb/qWs5QtACBlxE6MMqUMBvMquTpP7t3u48g6Z7/ebnkNnf4xtfZ+MgcGY
pwWqkHhiBg3QoX4toPicq83phWThYfOaRop4D/V7h9en8dRRMLuYkEh545h55d+dHbzgop1JpKCLCvcDO8jh3UGzDhSlvEB
5BYjnwrn24CiA3bX4yHa/2o"
```

```

resource "aws_lambda_function" "hello" {
  function_name = "hello"
  runtime      = "python3.8"
  role          = "arn:aws:iam::123456789012:role/lambdaBasicExecutionRole"
  handler       = "app.handler"
  code          = "lambda_function.zip"
}

resource "aws_s3_bucket" "aws-tutorial-bucket" {
  bucket = "aws-tutorial-bucket"
}

resource "aws_lambda_permission" "lambda_s3_trigger" {
  action        = "lambda:InvokeFunction"
  function_name = aws_lambda_function.hello.function_name
  principal    = "s3.amazonaws.com"
  source_arn   = aws_s3_bucket.aws-tutorial-bucket.arn
}

```

Step 6:

Now, to check whether the aws cli is connected to your aws account run the following command.

```
aws sts get-caller-identity
aws configure
```

```

resource "aws_lambda_function" "hello" {
  function_name = "hello"
  runtime      = "python3.8"
  role          = "arn:aws:iam::123456789012:role/lambdaBasicExecutionRole"
  handler       = "app.handler"
  code          = "lambda_function.zip"
}

resource "aws_s3_bucket" "aws-tutorial-bucket" {
  bucket = "aws-tutorial-bucket"
}

resource "aws_lambda_permission" "lambda_s3_trigger" {
  action        = "lambda:InvokeFunction"
  function_name = aws_lambda_function.hello.function_name
  principal    = "s3.amazonaws.com"
  source_arn   = aws_s3_bucket.aws-tutorial-bucket.arn
}

```

Step 7:

Now, to get the AMI ID run the following command and select any of the AMI ID and replace the AMI ID present in main.tf file in VSCode.

```
aws ec2 describe-images --owners amazon --filters "Name=name,Values=amzn2-ami-hvm-2.0.*-x86_64-gp2" --query "Images[*].[ImageId,Name]" --region us-east-1 --output table
```

The screenshot shows the VSCode interface with the terminal tab active. The terminal window displays the AWS CLI command to list AMIs and its output, which lists numerous AMI IDs and names. The code editor shows a Terraform configuration file (main.tf) with a script block that uses wget to download SonarQube and sudo to start it.

```
PS D:\Sem6\anuprita\AdvDevops_Practical> aws ec2 describe-images --owners amazon --filters "Name=name,Values=amzn2-ami-hvm-2.0.*-x86_64-gp2" --query "Images[*].[ImageId,Name]" --region us-east-1 --output table
+-----+-----+
| ImageId | Name |
+-----+-----+
| ami-007868005aea67c54 | amzn2-ami-hvm-2.0.20230119.1-x86_64-gp2 |
| ami-0241b1d769b029352 | amzn2-ami-hvm-2.0.20240620.0-x86_64-gp2 |
| ami-01e3c4a339a264cc9 | amzn2-ami-hvm-2.0.20241014.0-x86_64-gp2 |
| ami-014d544cfe21b2d2 | amzn2-ami-hvm-2.0.20240223.0-x86_64-gp2 |
| ami-02d69c34f7a0bf56a | amzn2-ami-hvm-2.0.20230418.0-x86_64-gp2 |
| ami-02b972fec07f1e659 | amzn2-ami-hvm-2.0.20221103.3-x86_64-gp2 |
| ami-04c09ef2f505b609 | amzn2-ami-hvm-2.0.20240109.0-x86_64-gp2 |
| ami-03c951bbe993ea887 | amzn2-ami-hvm-2.0.20240131.0-x86_64-gp2 |
| ami-03d1b2fca19c17cf1 | amzn2-ami-hvm-2.0.20230118.2-x86_64-gp2 |
| ami-03b8c4c9d51a0a8999 | amzn2-ami-hvm-2.0.20221119.0-x86_64-gp2 |
| ami-046eeba8a7f7b0ef0 | amzn2-ami-hvm-2.0.20240124.0-x86_64-gp2 |
| ami-045602374a1982480 | amzn2-ami-hvm-2.0.20240412.0-x86_64-gp2 |
| ami-0476f2eeba10d89bc | amzn2-ami-hvm-2.0.20230719.0-x86_64-gp2 |
| ami-052b9fbbe949f883a | amzn2-ami-hvm-2.0.20240916.0-x86_64-gp2 |
| ami-05b5badc2f7ddd88d | amzn2-ami-hvm-2.0.20230221.0-x86_64-gp2 |
```

After this run the following command

```
terraform init
```

```
terraform apply
```

The screenshot shows the VSCode interface with the terminal tab active. The terminal window displays the Terraform plan command output, which shows changes to be made to the infrastructure. A confirmation dialog is visible, asking if the user wants to proceed with the actions.

```
Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes
```

```

EXPLORER          PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
+ ADVDEVOPS PRACTICAL
> .terraform
  .terraform.lock.hcl
  hello.py
  main.tf
  terraform.tfstate
  terraform.tfstate.backup

+ capacity_reservation_specification (known after apply)
+ cpu_options (known after apply)
+ ebs_block_device (known after apply)
+ enclave_options (known after apply)
+ ephemeral_block_device (known after apply)
+ instance_market_options (known after apply)
+ maintenance_options (known after apply)
+ metadata_options (known after apply)
+ network_interface (known after apply)
+ private_dns_name_options (known after apply)
+ root_block_device (known after apply)
}

Plan: 2 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value: yes

aws_instance.sonarqube: Creating...
aws_instance.jenkins: Creating...
aws_instance.sonarqube: Still creating... [10s elapsed]
aws_instance.jenkins: Still creating... [10s elapsed]
aws_instance.sonarqube: Creation complete after 16s [id=i-09f5733c9d7c06309]
aws_instance.jenkins: Creation complete after 16s [id=i-0b2d8414cbfb34c5]

Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS D:\Sem5_anuprita\AdvDevops_Practical>

```

Ln 49, Col 1 Spaces: 2 UTF-8 Plain Text Go Live Prettier

Step 8:

After this Terraform will automatically create 2 EC2 instances on the EC2 Dashboard. To check the running instances: Visit AWS EC2 Dashboard.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
JenkinsServer	i-05f12961a9cf8cf3e	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-98-80
SonarQubeServer	i-061f29e11e3fa1a8b	Running	t2.medium	2/2 checks passed	View alarms	us-east-1c	ec2-54-21

Set up Security Groups for the given two instances

Step 1:

Go to EC2 Dashboard and select the Security Groups present in the left pane or sidebar and the click on the create security group.

a. Create a security group with name AMjenkins-security and give some description and add the inbounds rules given below.

b. Create a security group with name AMsonarqube-security and give some description and add the inbounds rules given below.

The screenshot shows the AWS Management Console with the 'Security Groups' page open. The left sidebar includes sections for Events, Instances (Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, Reservations), Images (AMIs, AMI Catalog), Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), and Network & Security (Security Groups). The main content area displays a table titled 'Security Groups (23)'. The columns are: Name, Security group ID, Security group name, VPC ID, and Description. The table lists several security groups, such as 'launch-wizard-6', 'sg-01e6da10b21246fb', and 'allow_http'. A search bar at the top allows filtering by attribute or tag. An orange 'Create security group' button is located in the top right corner.

The screenshot shows the 'Edit inbound rules' page for the security group 'sg-06766b363e07ace3b - AMjenkins-security'. The top navigation bar includes links for EC2, Security Groups, sg-06766b363e07ace3b - AMjenkins-security, and Edit inbound rules. Below the navigation, a message states: 'Inbound rules control the incoming traffic that's allowed to reach the instance.' The main content area is titled 'Inbound rules' and contains a table with columns: Security group rule ID, Type, Protocol, Port range, Source, and Description - optional. There are three rules listed:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0d9d57127ff521991	Custom TCP	TCP	8080	Custom	49.36.105.96/32
sgr-057e1247e95e38c1c	SSH	TCP	22	Custom	0.0.0.0/0
sgr-07584f2ee4e6eebfaf	Custom TCP	TCP	8080	Custom	0.0.0.0/0

A button labeled 'Add rule' is located at the bottom left. A warning message at the bottom states: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' At the bottom right are buttons for 'Cancel', 'Preview changes', and 'Save rules'.

AWS Services Search [Alt+S] N. Virginia vocabs/user3413602=MHAPANKAR_ANUPRITA_ANAND @ 8567-4606-9793

EC2 > Security Groups > sg-04d93757f217a03e0 - AMsonarqube-security > Edit inbound rules

Edit inbound rules Info

Inbound rules control the incoming traffic that's allowed to reach the instance.

Inbound rules <small>Info</small>	Type <small>Info</small>	Protocol <small>Info</small>	Port range	Source <small>Info</small>	Description - optional <small>Info</small>
sgr-0b145544ddb02dbd0	SSH	TCP	22	Custom	0.0.0.0/0 <small>X</small>
sgr-0ca54f064e2b116c9	Custom TCP	TCP	9000	Custom	49.36.105.96/32 <small>X</small>
sgr-0b9649cd8bb7a9401	Custom TCP	TCP	9000	Custom	0.0.0.0/0 <small>X</small>

[Add rule](#)

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Preview changes](#) [Save rules](#)

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Installation for Jenkins

Reference Video: <https://www.youtube.com/watch?v=bNuAS52ebLs>

Step 1:

Click on the JenkinsServer and click on connect.

[CloudShell](#) [Feedback](#)

Instances (1/2) [Info](#)

Last updated 1 minute ago [Connect](#) Instance state [Actions](#) [Launch instances](#)

Find Instance by attribute or tag (case-sensitive) All states

Instance state = running Clear filters

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input checked="" type="checkbox"/> JenkinsServer	i-05f12961a9cf8cf3e	Running View details Logs	t2.micro	2/2 checks passed View alarms	View alarms	us-east-1a	ec2-98-80-223-40.compute-1.amazonaws.com
<input type="checkbox"/> SonarQubeSer...	i-061f29e11e3fa1a8b	Running View details Logs	t2.medium	2/2 checks passed View alarms	View alarms	us-east-1c	ec2-54-213-11-111

i-05f12961a9cf8cf3e (JenkinsServer)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary [Info](#)

Instance ID i-05f12961a9cf8cf3e (JenkinsServer)	Public IPv4 address 98.80.223.40 open address	Private IPv4 addresses 172.31.42.86
IPv6 address -	Instance state Running	Public IPv4 DNS ec2-98-80-223-40.compute-1.amazonaws.com open address
Hostname type IP name: ip-172-31-42-86.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-42-86.ec2.internal	Elastic IP addresses
Answer private resource DNS name	Instance type t2.micro	

EC2 > Instances > i-05f12961a9cf8cf3e > Connect to instance

Connect to instance [Info](#)

Connect to your instance i-05f12961a9cf8cf3e (JenkinsServer) using any of these options

EC2 Instance Connect Session Manager [SSH client](#) EC2 serial console

Instance ID [i-05f12961a9cf8cf3e \(JenkinsServer\)](#)

- Open an SSH client.
- Locate your private key file. The key used to launch this instance is AMjenkins.pem.
- Run this command, if necessary, to ensure your key is not publicly viewable.
 chmod 400 "AMjenkins.pem"
- Connect to your instance using its Public DNS:
 ec2-98-80-223-40.compute-1.amazonaws.com

Example:

ssh -i "AMjenkins.pem" ec2-user@ec2-98-80-223-40.compute-1.amazonaws.com

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

[Cancel](#)

Step 2:

Open Git Bash and go to the directory which has the Key downloaded. If you don't have the key downloaded, create a key pair and download the .pem file for the key.

Since, I have the key downloaded in Downloads directory, I used the following commands:

cd Download

dir AMjenkins.pem*

```
ssh -i "AMjenkins.pem" ec2-user@ec2-98-80-223-40.compute-1.amazonaws.com
```

```
ec2-user@ip-172-31-42-155:~  
user@DESKTOP-QOGK15A MINGW64 ~ (master)  
$ cd Downloads  
user@DESKTOP-QOGK15A MINGW64 ~/Downloads (master)  
$ dir Jenkins.pem  
Jenkins.pem  
user@DESKTOP-QOGK15A MINGW64 ~/Downloads (master)  
$ ssh -i "Jenkins.pem" ec2-user@ec2-54-91-87-54.compute-1.amazonaws.com  
The authenticity of host 'ec2-54-91-87-54.compute-1.amazonaws.com (54.91.87.54)'  
can't be established.  
ED25519 key fingerprint is SHA256:LNzVrp4MguqGgT9Kdfgnah0mojppMk06keIbzEMM.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
warning: Permanently added 'ec2-54-91-87-54.compute-1.amazonaws.com' (ED25519) t  
o the list of known hosts.  
_ _ _ _ _  
_ _ _ _ _ ####  
_ _ _ _ _ \###  
_ _ _ _ _ \#)  
_ _ _ _ _ / \_,-->  
_ _ _ _ _ V-,-->  
_ _ _ _ _ /  
_ _ _ _ _ / \/  
_ _ _ _ _ / \/  
_ /m/  
[ec2-user@ip-172-31-42-155 ~]$ |
```

Step 3:

Go to google and search for Jenkins and then click on the Download and Deploy Link. Else, navigate using the following link: <https://pkg.jenkins.io/redhat-stable/>

Google Jenkins

All Images Videos News Shopping Web Maps More Tools

Jenkins https://www.jenkins.io ...

Jenkins

Jenkins – an open source automation server which enables developers around the world to reliably build, test, and deploy their software.

Download and deploy

Jenkins – an open source automation server which ...

Installing Jenkins

Jenkins – an open source automation server which ...

Jenkins User Documentation

Installing Jenkins - Pipeline - User Handbook Overview - Blue Ocean

Tutorials overview

Jenkins – an open source automation server which ...

Pipeline

A suite of plugins which supports implementing and integrating ...

[More results from jenkins.io »](#)

People also ask :

What is Jenkins used for?

Is Jenkins free software?

Jenkins

Software ...



Jenkins is an open source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration, and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat. [Wikipedia](#)

Programming language: Java

Developer: Kohsuke Kawaguchi

Initial release: 2 February 2011

License: MIT License

Platform: Java 11, Java 17, Java 21

Repository: github.com/jenkinsci/jenkins

Stable release: 2.477 / 17 September 2024; 18 days ago

Step 4:

Now, run the initial 2 commands as it is and then run the next 2 commands using sudo word in the first; to run as root user.

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-stable/jenkins.repo
sudo rpm --import https://pkg.jenkins.io/redhat-stable/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-17-openjdk
yum install jenkins
```

The rpm packages were signed using this key:

```
pub    rsa4096 2023-03-27 [SC] [expires: 2026-03-26]
      63667EE74BBA1F0A08A698725BA31D57EF5975CA
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:

2.463 (June 2024) and newer

Java 17 or Java 21

2.419 (August 2023) and newer

OR

Run the following commands:

```
sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
```

```
sudo sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
```

```
sudo yum install fontconfig java-17-openjdk
```

```
sudo yum install jenkins
```

Step 5:

Now, in order to install java run the following commands:

```
sudo yum install java-17-amazon-corretto-headless
```

```
sudo yum install java-17-amazon-corretto
```

```
sudo dnf install java-17-amazon-corretto-devel
```

```
ec2-user@ip-172-31-42-155:~
```

```
User@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ ssh -i "Mjenkins.pem" ec2-user@ec2-54-91-87-54.compute-1.amazonaws.com
.
.
.
Amazon Linux 2023
.
.
.
https://aws.amazon.com/linux/amazon-linux-2023

Last login: Fri Oct 18 14:57:44 2024 from 49.36.105.208
[ec2-user@ip-172-31-42-155 ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo
[ec2-user@ip-172-31-42-155 ~]$ curl -L https://pkg.jenkins.io/redhat-stable/jenkins.io-2024-10-18-15:11:26-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Reading pkgs.jenkins.io (pkgs.jenkins.io) ... 146.75.34.133, 2a04:4e42:78::645
Connected to pkgs.jenkins.io (pkgs.jenkins.io)|146.75.34.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 85
Saving to: '/etc/yum.repos.d/jenkins.repo'

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

/etc/yum.repos.d/jenkins.repo
```

```
2024-10-18 15:11:26 (3.60 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

[ec2-user@ip-172-31-42-155 ~]$ sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
[ec2-user@ip-172-31-42-155 ~]$ yum install fontconfig java-17-openjdk
Error: This command has to be run with superuser privileges (under the root user on most systems).
[ec2-user@ip-172-31-42-155 ~]$ sudo yum install fontconfig java-17-openjdk
Jenkins-stable
No match for argument: java-17-openjdk
Error: Unable to find a match: java-17-openjdk
[ec2-user@ip-172-31-42-155 ~]$ sudo dnf install java-17-amazon-corretto-devel
Last metadata expiration check: 0:02:40 ago on Fri Oct 18 15:12:06 2024.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
Installing:				
java-17-amazon-corretto-devel	x86_64	1:17.0.12+7-1.amzn2023.1	amazonlinux	142 k
Installing dependencies:				
alsa-lib	x86_64	1.2.7.2-1.amzn2023.0.2	amazonlinux	504 k
cairo	x86_64	1.17.6-2.amzn2023.0.1	amazonlinux	684 k
dejavu-sans-fonts	noarch	2.37-16.amzn2023.0.2	amazonlinux	1.3 M
dejavu-sans-mono-fonts	noarch	2.37-16.amzn2023.0.2	amazonlinux	467 k
dejavu-serif-fonts	noarch	2.37-16.amzn2023.0.2	amazonlinux	1.0 M
fontconfig	x86_64	2.13.94-2.amzn2023.0.2	amazonlinux	273 k
freetype	noarch	1.12.7-12.amzn2023.0.2	amazonlinux	93.5 k
font-filesystem	x86_64	2.13.2-5.amzn2023.0.1	amazonlinux	423 k
fontepoch	noarch	20201206-2.amzn2023.0.2	amazonlinux	15 k
fontepoch-fonts-common	noarch	20201206-2.amzn2023.0.2	amazonlinux	492 k
google-noto-sans-vf-fonts	x86_64	1.3.14-7.amzn2023.0.2	amazonlinux	97 k
graphite2	x86_64	7.0.0-2.amzn2023.0.1	amazonlinux	868 k
harfbuzz	x86_64	1:17.0.12+7-1.amzn2023.1	amazonlinux	91 M
java-17-amazon-corretto-headless	x86_64	6.0.0-7.amzn2023.0.6	amazonlinux	12 k
javapackages-filesystem	noarch	3.0-21.amzn2023.0.4	amazonlinux	10 k
langpacks-core-font-en	noarch	1.8.10-2.amzn2023.0.1	amazonlinux	659 k
libX11	x86_64	1.8.10-2.amzn2023.0.1	amazonlinux	147 k
libX11-common	noarch	1.8.10-2.amzn2023.0.1	amazonlinux	33 k
libXau	x86_64	1.3.6-1.amzn2023.0.1	amazonlinux	42 k
libXext	x86_64	0.9.11-6.amzn2023.0.1	amazonlinux	29 k
libXrender	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
libbrotli	x86_64	2.1.4-2.amzn2023.0.5	amazonlinux	190 k
libjpeg-turbo	x86_64	2:1.6.37-10.amzn2023.0.6	amazonlinux	128 k
libpng	x86_64			

```
ec2-user@ip-172-31-42-155:~
```

Transaction Summary			
Install	27 Packages		
Total	download size: 100 M		
Installed	size: 261 M		
Is this ok [y/N]: y			
Downloading Packages:			
(1/27): cairo-1.17.6-2.amzn2023.0.1.x86_64.rpm		8.3 MB/s 684 kB 00:00	
(2/27): dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch.rpm		1.5 MB/s 1.3 MB 00:00	
(3/27): alsa-lib-1.2.7-2-1.amzn2023.0.2.x86_64.rpm		4.2 MB/s 504 kB 00:00	
(4/27): dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch.rpm		2.0 MB/s 1.0 MB 00:00	
(5/27): fonts-filesystem-2.0.3-12.amzn2023.0.2.noarch.rpm		276 kB/s 1.0 MB 00:00	
(6/27): dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch.rpm		5.7 MB/s 467 kB 00:00	
(7/27): fontconfig-2.13.94-2.amzn2023.0.2.x86_64.rpm		4.6 MB/s 273 kB 00:00	
(8/27): google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch.rpm		22 MB/s 492 kB 00:00	
(9/27): graphite2-1.3.14-7.amzn2023.0.2.x86_64.rpm		3.5 MB/s 97 kB 00:00	
(10/27): freetype-2.13.2-5.amzn2023.0.1.x86_64.rpm		5.4 MB/s 423 kB 00:00	
(11/27): harfbuzz-7.0.0-2.amzn2023-0.1.x86_64.rpm		26 MB/s 868 kB 00:00	
(12/27): google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch.rpm		129 kB/s 15 kB 00:00	
(13/27): java-17-amazon-corretto-devel-17.0.12+7-1.amzn2023.1.x86_64.rpm		2.0 MB/s 142 kB 00:00	
(14/27): javapackages-filesystem-6.0-0.amzn2023.0.6.noarch.rpm		359 kB/s 12 kB 00:00	
(15/27): langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch.rpm		313 kB/s 10 kB 00:00	
(16/27): libX11-1.8.10-2.amzn2023.0.1.x86_64.rpm		6.4 MB/s 659 kB 00:00	
(17/27): libX11-common-1.8.10-2.amzn2023.0.1.noarch.rpm		1.6 MB/s 147 kB 00:00	
(18/27): libXau-1.0.11-6.amzn2023.0.1.x86_64.rpm		1.1 MB/s 33 kB 00:00	
(19/27): libXext-1.3.6-1.amzn2023.0.1.x86_64.rpm		1.2 MB/s 42 kB 00:00	
(20/27): libXrender-0.9.11-6.amzn2023.0.1.x86_64.rpm		724 kB/s 29 kB 00:00	
(21/27): libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64.rpm		6.0 MB/s 190 kB 00:00	
(22/27): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm		3.4 MB/s 315 kB 00:00	
(23/27): libpng-1.6.37-10.amzn2023.0.6.x86_64.rpm		3.4 MB/s 128 kB 00:00	
(24/27): libxcb-1.17-0-1.amzn2023.0.1.x86_64.rpm		7.8 MB/s 235 kB 00:00	
(25/27): xml-common-0.6.3-56.amzn2023.0.2.noarch.rpm		1.0 MB/s 32 kB 00:00	
(26/27): pixman-0.45.4-1.amzn2023.0.4.x86_64.rpm		3.1 MB/s 298 kB 00:00	
(27/27): java-17-amazon-corretto-headless-17.0.12+7-1.amzn2023.1.x86_64.rpm		59 MB/s 91 MB 00:01	

```
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch
Installing  : dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch
Installing  : dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch
Installing  : libpng-2.1.6.37-10.amzn2023.0.6.x86_64
Installing  : dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch
Running scriptlet: xml-common-0.6.3-56.amzn2023.0.2.noarch
Installing  : xml-common-0.6.3-56.amzn2023.0.2.noarch
Installing  : pixman-0.43.4-1.amzn2023.0.4.x86_64
Installing  : libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
Installing  : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Installing  : libXau-1.0.11-6.amzn2023.0.1.x86_64
Installing  : libXext-1.3.6-1.amzn2023.0.1.x86_64
Installing  : libXrender-0.9.11-6.amzn2023.0.1.x86_64
Installing  : javapackages-filesystem-6.0-0-7.amzn2023.0.6.noarch
Installing  : graphite2-1.3.14-7.amzn2023.0.2.x86_64
Installing  : google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
Installing  : google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
Installing  : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
```

```
Prepared:
```

```
1/1
1/27
2/27
3/27
4/27
5/27
6/27
7/27
8/27
9/27
10/27
11/27
12/27
13/27
14/27
15/27
16/27
17/27
18/27
19/27
20/27
21/27
22/27
23/27
24/27
25/27
26/27
27/27
```

```

ec2-user@ip-172-31-42-155:~-
Installing : libX11-1.8.10-2.amzn2023.0.1.x86_64
Installing : libXext-1.3.6-1.amzn2023.0.1.x86_64
Installing : libXcursor-0.4.11-6.amzn2023.0.2.x86_64
Installing : jvoicemeeting-filters-6.0.0-7.amzn2023.0.6.noarch
Installing : graphite2-1.3.14-7.amzn2023.0.2.x86_64
Installing : google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
Installing : google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
Installing : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
Installing : cairo-1.17.6-2.amzn2023.0.1.x86_64
Installing : harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
Installing : freetype-2.13.2-5.amzn2023.0.1.x86_64
Installing : fontconfig-2.13.94-2.amzn2023.0.2.x86_64
Running scriptlet: fontconfig-2.13.94-2.amzn2023.0.2.x86_64
Installing : alsa-lib-1.2.7.2-1.amzn2023.0.1.x86_64
Installing : java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64
Running scriptlet: java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64
Installing : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
Running scriptlet: java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
Running scriptlet: fontconfig-2.13.94-2.amzn2023.0.2.x86_64
Running scriptlet: java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
Verifying : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
Verifying : alsalib-1.2.7.2-1.amzn2023.0.1.x86_64
Verifying : cairo-1.17.6-2.amzn2023.0.1.x86_64
Verifying : dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch
Verifying : dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch
Verifying : dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch
Verifying : fontconfig-2.13.14-2.amzn2023.0.2.x86_64
Verifying : graphite2-1.3.14-7.amzn2023.0.2.x86_64
Verifying : google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch
Verifying : google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch
Verifying : harfbuzz-7.0.0-2.amzn2023.0.1.x86_64
Verifying : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64
Verifying : java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64
Verifying : java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.0.6.noarch
Verifying : javapackages-filesystem-6.0.0-7.amzn2023.0.6.noarch
Verifying : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch
Verifying : libX11-1.8.10-2.amzn2023.0.1.noarch
Verifying : libXau-1.0.11-6.amzn2023.0.1.x86_64
Verifying : libXext-1.3.6-1.amzn2023.0.1.x86_64
Verifying : libXrender-0.9.11-6.amzn2023.0.1.x86_64
Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Verifying : libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64
Verifying : libpng-2:1.6.37-10.amzn2023.0.6.x86_64
Verifying : libxcb-1.17.0-1.amzn2023.0.1.x86_64
Verifying : pixman-0.43.4-1.amzn2023.0.4.x86_64
Verifying : xml-common-0.6.3-56.amzn2023.0.2.noarch
13/27
14/27
15/27
16/27
17/27
18/27
19/27
20/27
21/27
22/27
23/27
24/27
25/27
26/27
27/27
28/27
29/27
30/27
31/27
32/27
33/27
34/27
35/27
36/27
37/27
38/27
39/27
40/27
41/27
42/27
43/27
44/27
45/27
46/27
47/27
48/27
49/27
50/27
51/27
52/27
53/27
54/27
55/27
56/27
57/27
58/27
59/27
60/27
61/27
62/27
63/27
64/27
65/27
66/27
67/27
68/27
69/27
70/27
71/27
72/27
73/27
74/27
75/27
76/27
77/27
78/27
79/27
80/27
81/27
82/27
83/27
84/27
85/27
86/27
87/27
88/27
89/27
90/27
91/27
92/27
93/27
94/27
95/27
96/27
97/27
98/27
99/27
100/27
101/27
102/27
103/27
104/27
105/27
106/27
107/27
108/27
109/27
110/27
111/27
112/27
113/27
114/27
115/27
116/27
117/27
118/27
119/27
120/27
121/27
122/27
123/27
124/27
125/27
126/27
127/27
128/27
129/27
130/27
131/27
132/27
133/27
134/27
135/27
136/27
137/27
138/27
139/27
140/27
141/27
142/27
143/27
144/27
145/27
146/27
147/27
148/27
149/27
150/27
151/27
152/27
153/27
154/27
155/27
156/27
157/27
158/27
159/27
160/27
161/27
162/27
163/27
164/27
165/27
166/27
167/27
168/27
169/27
170/27
171/27
172/27
173/27
174/27
175/27
176/27
177/27
178/27
179/27
180/27
181/27
182/27
183/27
184/27
185/27
186/27
187/27
188/27
189/27
190/27
191/27
192/27
193/27
194/27
195/27
196/27
197/27
198/27
199/27
200/27
201/27
202/27
203/27
204/27
205/27
206/27
207/27
208/27
209/27
210/27
211/27
212/27
213/27
214/27
215/27
216/27
217/27
218/27
219/27
220/27
221/27
222/27
223/27
224/27
225/27
226/27
227/27
228/27
229/27
230/27
231/27
232/27
233/27
234/27
235/27
236/27
237/27
238/27
239/27
240/27
241/27
242/27
243/27
244/27
245/27
246/27
247/27
248/27
249/27
250/27
251/27
252/27
253/27
254/27
255/27
256/27
257/27
258/27
259/27
260/27
261/27
262/27
263/27
264/27
265/27
266/27
267/27
268/27
269/27
270/27
271/27
272/27
273/27
274/27
275/27
276/27
277/27
278/27
279/27
280/27
281/27
282/27
283/27
284/27
285/27
286/27
287/27
288/27
289/27
290/27
291/27
292/27
293/27
294/27
295/27
296/27
297/27
298/27
299/27
300/27
301/27
302/27
303/27
304/27
305/27
306/27
307/27
308/27
309/27
310/27
311/27
312/27
313/27
314/27
315/27
316/27
317/27
318/27
319/27
320/27
321/27
322/27
323/27
324/27
325/27
326/27
327/27
328/27
329/27
330/27
331/27
332/27
333/27
334/27
335/27
336/27
337/27
338/27
339/27
340/27
341/27
342/27
343/27
344/27
345/27
346/27
347/27
348/27
349/27
350/27
351/27
352/27
353/27
354/27
355/27
356/27
357/27
358/27
359/27
360/27
361/27
362/27
363/27
364/27
365/27
366/27
367/27
368/27
369/27
370/27
371/27
372/27
373/27
374/27
375/27
376/27
377/27
378/27
379/27
380/27
381/27
382/27
383/27
384/27
385/27
386/27
387/27
388/27
389/27
390/27
391/27
392/27
393/27
394/27
395/27
396/27
397/27
398/27
399/27
400/27
401/27
402/27
403/27
404/27
405/27
406/27
407/27
408/27
409/27
410/27
411/27
412/27
413/27
414/27
415/27
416/27
417/27
418/27
419/27
420/27
421/27
422/27
423/27
424/27
425/27
426/27
427/27
428/27
429/27
430/27
431/27
432/27
433/27
434/27
435/27
436/27
437/27
438/27
439/27
440/27
441/27
442/27
443/27
444/27
445/27
446/27
447/27
448/27
449/27
450/27
451/27
452/27
453/27
454/27
455/27
456/27
457/27
458/27
459/27
460/27
461/27
462/27
463/27
464/27
465/27
466/27
467/27
468/27
469/27
470/27
471/27
472/27
473/27
474/27
475/27
476/27
477/27
478/27
479/27
480/27
481/27
482/27
483/27
484/27
485/27
486/27
487/27
488/27
489/27
490/27
491/27
492/27
493/27
494/27
495/27
496/27
497/27
498/27
499/27
500/27
501/27
502/27
503/27
504/27
505/27
506/27
507/27
508/27
509/27
510/27
511/27
512/27
513/27
514/27
515/27
516/27
517/27
518/27
519/27
520/27
521/27
522/27
523/27
524/27
525/27
526/27
527/27
528/27
529/27
530/27
531/27
532/27
533/27
534/27
535/27
536/27
537/27
538/27
539/27
540/27
541/27
542/27
543/27
544/27
545/27
546/27
547/27
548/27
549/27
550/27
551/27
552/27
553/27
554/27
555/27
556/27
557/27
558/27
559/27
560/27
561/27
562/27
563/27
564/27
565/27
566/27
567/27
568/27
569/27
570/27
571/27
572/27
573/27
574/27
575/27
576/27
577/27
578/27
579/27
580/27
581/27
582/27
583/27
584/27
585/27
586/27
587/27
588/27
589/27
590/27
591/27
592/27
593/27
594/27
595/27
596/27
597/27
598/27
599/27
600/27
601/27
602/27
603/27
604/27
605/27
606/27
607/27
608/27
609/27
610/27
611/27
612/27
613/27
614/27
615/27
616/27
617/27
618/27
619/27
620/27
621/27
622/27
623/27
624/27
625/27
626/27
627/27
628/27
629/27
630/27
631/27
632/27
633/27
634/27
635/27
636/27
637/27
638/27
639/27
640/27
641/27
642/27
643/27
644/27
645/27
646/27
647/27
648/27
649/27
650/27
651/27
652/27
653/27
654/27
655/27
656/27
657/27
658/27
659/27
660/27
661/27
662/27
663/27
664/27
665/27
666/27
667/27
668/27
669/27
670/27
671/27
672/27
673/27
674/27
675/27
676/27
677/27
678/27
679/27
680/27
681/27
682/27
683/27
684/27
685/27
686/27
687/27
688/27
689/27
690/27
691/27
692/27
693/27
694/27
695/27
696/27
697/27
698/27
699/27
700/27
701/27
702/27
703/27
704/27
705/27
706/27
707/27
708/27
709/27
710/27
711/27
712/27
713/27
714/27
715/27
716/27
717/27
718/27
719/27
720/27
721/27
722/27
723/27
724/27
725/27
726/27
727/27
728/27
729/27
730/27
731/27
732/27
733/27
734/27
735/27
736/27
737/27
738/27
739/27
740/27
741/27
742/27
743/27
744/27
745/27
746/27
747/27
748/27
749/27
750/27
751/27
752/27
753/27
754/27
755/27
756/27
757/27
758/27
759/27
760/27
761/27
762/27
763/27
764/27
765/27
766/27
767/27
768/27
769/27
770/27
771/27
772/27
773/27
774/27
775/27
776/27
777/27
778/27
779/27
780/27
781/27
782/27
783/27
784/27
785/27
786/27
787/27
788/27
789/27
790/27
791/27
792/27
793/27
794/27
795/27
796/27
797/27
798/27
799/27
800/27
801/27
802/27
803/27
804/27
805/27
806/27
807/27
808/27
809/27
810/27
811/27
812/27
813/27
814/27
815/27
816/27
817/27
818/27
819/27
820/27
821/27
822/27
823/27
824/27
825/27
826/27
827/27
828/27
829/27
830/27
831/27
832/27
833/27
834/27
835/27
836/27
837/27
838/27
839/27
840/27
841/27
842/27
843/27
844/27
845/27
846/27
847/27
848/27
849/27
850/27
851/27
852/27
853/27
854/27
855/27
856/27
857/27
858/27
859/27
860/27
861/27
862/27
863/27
864/27
865/27
866/27
867/27
868/27
869/27
870/27
871/27
872/27
873/27
874/27
875/27
876/27
877/27
878/27
879/27
880/27
881/27
882/27
883/27
884/27
885/27
886/27
887/27
888/27
889/27
890/27
891/27
892/27
893/27
894/27
895/27
896/27
897/27
898/27
899/27
900/27
901/27
902/27
903/27
904/27
905/27
906/27
907/27
908/27
909/27
910/27
911/27
912/27
913/27
914/27
915/27
916/27
917/27
918/27
919/27
920/27
921/27
922/27
923/27
924/27
925/27
926/27
927/27
928/27
929/27
930/27
931/27
932/27
933/27
934/27
935/27
936/27
937/27
938/27
939/27
940/27
941/27
942/27
943/27
944/27
945/27
946/27
947/27
948/27
949/27
950/27
951/27
952/27
953/27
954/27
955/27
956/27
957/27
958/27
959/27
960/27
961/27
962/27
963/27
964/27
965/27
966/27
967/27
968/27
969/27
970/27
971/27
972/27
973/27
974/27
975/27
976/27
977/27
978/27
979/27
980/27
981/27
982/27
983/27
984/27
985/27
986/27
987/27
988/27
989/27
990/27
991/27
992/27
993/27
994/27
995/27
996/27
997/27
998/27
999/27
1000/27

```

Step 6:

Run the following commands:

```
sudo yum install Jenkins
```

```
sudo systemctl status jenkins
```

```
sudo systemctl enable jenkins
```

```
sudo systemctl status jenkins
```

```
sudo systemctl start jenkins
```

```
sudo systemctl status jenkins
```

```
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
jenkins	noarch	2.462.3-1.1	jenkins	89 M
Transaction Summary				
Install:	jenkins-2.462.3-1.1.noarch.rpm			
Total download size: 89 M				
Installed size: 89 M				
Is this ok [y/N]: y				
Downloading Packages:				
jenkins-2.462.3-1.1.noarch.rpm				13 MB/s 89 MB 00:06
Total				
Running transaction check				
Transaction check succeeded.				
Running transaction test				
Transaction test succeeded.				
Running transaction				
Preparing...				1/1
Running scriptlet: jenkins-2.462.3-1.1.noarch				1/1
Installing : jenkins-2.462.3-1.1.noarch				1/1
Running scriptlet: jenkins-2.462.3-1.1.noarch				1/1
Verifying : jenkins-2.462.3-1.1.noarch				1/1
Installed:	jenkins-2.462.3-1.1.noarch			
Complete!	[ec2-user@ip-172-31-42-155 ~]\$			

```

ec2-user@ip-172-31-42-155:~-
=====
                         Architecture          Version           Repository          Size
=====
Package                noarch             2.462.3-1.1      jenkins
=====
Installing: Jenkins
=====
Transaction Summary
=====
Install 1 Package
=====
Total download size: 89 M
Installed size: 89 M
Is this ok [y/N]: y
Downloaded Packages:
jenkins-2.462.3-1.1.noarch.rpm
=====
Total: 13 MB/s | 89 MB 00:06
=====
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Running scriptlet: jenkins-2.462.3-1.1.noarch 1/1
  Installing : jenkins-2.462.3-1.1.noarch 1/1
  Running scriptlet: jenkins-2.462.3-1.1.noarch 1/1
  Verifying   : jenkins-2.462.3-1.1.noarch 1/1
=====
Installed:
  jenkins-2.462.3-1.1.noarch
=====
Complete!
[ec2-user@ip-172-31-42-155 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; disabled; preset: disabled)
    Active: inactive (dead)
[ec2-user@ip-172-31-42-155 ~]$ sudo systemctl enable jenkins
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service → /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-42-155 ~]$ sudo systemctl status jenkins
● Jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
    Active: inactive (dead)
Main PID: 28188 (java)
  Tasks: 4 (limit: 1112)
    Memory: 3277M
      CPU: 14.868s
      CGroup: /system.slice/jenkins.service
           └─28188 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080
Oct 18 15:22:22 ip-172-31-42-155.ec2.internal jenkins[28188]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Oct 18 15:22:22 ip-172-31-42-155.ec2.internal jenkins[28188]: ****
Oct 18 15:22:22 ip-172-31-42-155.ec2.internal jenkins[28188]: ****
Oct 18 15:22:22 ip-172-31-42-155.ec2.internal jenkins[28188]: ****
Oct 18 15:22:28 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:28.706+0000 [id=31] INFO jenkins.InitReactorRunner$1#onAttained: Completed initialization
Oct 18 15:22:28 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:28.730+0000 [id=24] INFO hudson.lifecycle.Lifecycle$OnReady: Jenkins is fully up and running
Oct 18 15:22:28 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:28.730+0000 [id=24] INFO hudson.util.Retryer$Start: Jenkins is fully up and running
Oct 18 15:22:28 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:28.824+0000 [id=47] INFO h.m.DownloadService$Downloadable#load: Obtained the updated data file for hudson.tasks.Maven
Oct 18 15:22:28 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:28.825+0000 [id=47] INFO hudson.util.Retryer$Start: Performed the action check updates server successfully at the ats
Oct 18 15:22:33 ip-172-31-42-155.ec2.internal jenkins[28188]: 2024-10-18 15:22:33.834+0000 [id=62] WARNING h.n.DiskSpaceMonitorDescriptor#markNodeOfflineOrOnline: Making Built-in Node offline tem
Lines 1-20/20 (END)

```

Step 7:

Now, go to EC2 dashboard and select Jenkins server and copy its public address and visit <http://<public-address>:8080>

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity, and Reservations. Below that is another sidebar for Images, AMIs, and AMI Catalog. Further down are sections for Elastic Block Store, Network & Security, and Network & Security. At the bottom are CloudShell and Feedback buttons.

The main area displays a table of instances. The JenkinsServer instance (with ID i-05f12961a9cf8cf3e) is selected and highlighted in blue. Its details are shown in a modal window below:

- Details:**
 - Public IPv4 address copied:** 98.80.223.40 (with a link to open address)
 - Instance summary:**
 - Instance ID: i-05f12961a9cf8cf3e (JenkinsServer)
 - IPv6 address: -
 - Hostname type: IP name: ip-172-31-42-86.ec2.internal
 - Answer private resource DNS name: -
 - Public IPv4 DNS:** ec2-98-80-223-40.compute-1.amazonaws.com (with a link to open address)
 - Private IP4 addresses:** 172.31.42.86
 - Private IP4 DNS:** ip-172-31-42-86.ec2.internal
 - Instance type:** t2.micro
 - Status check:** 2/2 checks passed
 - Alarm status:** +
 - Availability Zone:** us-east-1a
 - Public IP:** ec2-98-80-223-40
- SonarQubeServer** instance (with ID i-061f29e11e3fa1a8b) is also listed with its details.

Step 8:

You will be redirected to this page on successful installation of Jenkins and visiting the public address url with port 8080.

← ⌂ Not secure | 98.80.223.40:8080/login?from=%2F

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

Continue

Step 9:

Now, come back to gitbash run the command

```
sudo more /var/lib/jenkins/secrets/initialAdminPassword
```

And, copy the content in the output and paste it in the input of Administrator password.

```
ec2-user@ip-172-31-42-86:~$ aliases: install-n, install-na, install-nevra
[ec2-user@ip-172-31-42-86 ~]$ sudo yum install client_loop: send disconnect: Connection reset by peer
User@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ ssh -i "Mjenkins.pem" ec2-user@ec2-98-80-223-40.compute-1.amazonaws.com
Last login: Sat Oct 19 16:16:12 2024 from 49.36.105.96

      _###_
     /###\   Amazon Linux 2
    /##\|   AL2 End of Life is 2025-06-30.
   /#/\|_>
  /#/\|_> A newer version of Amazon Linux is available!
 /#/\|_>
/_m/|_> Amazon Linux 2023, GA and supported until 2028-03-15.
          https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-42-86 ~]$ sudo dnf install java-17-amazon-corretto-devel
sudo: dnf: command not found
[ec2-user@ip-172-31-42-86 ~]$ AC
[ec2-user@ip-172-31-42-86 ~]$ sudo yum install java-17-amazon-corretto-devel
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Package 1:java-17-amazon-corretto-devel-17.0.12+7-1.amzn2.1.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl status jenkins
sudo: systemctl: command not found
[ec2-user@ip-172-31-42-86 ~]$ AC
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; disabled; vendor preset: disabled)
     Active: inactive (dead)
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl enable jenkins
Failed to execute operation: No such file or directory
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
     Active: inactive (dead)
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-42-86 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
     Active: active (running) since Sat 2024-10-19 16:37:17 UTC; 10s ago
       Main PID: 13065 (java)
      CGroup: /system.slice/jenkins.service
              └─13065 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=%C/jenkins/war --httpPort=8080

Oct 19 16:37:10 ip-172-31-42-86.ec2.internal jenkins[13065]: 174B8D24318F44658530f65beb39ff2
Oct 19 16:37:10 ip-172-31-42-86.ec2.internal jenkins[13065]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Oct 19 16:37:10 ip-172-31-42-86.ec2.internal jenkins[13065]: ****
Oct 19 16:37:10 ip-172-31-42-86.ec2.internal jenkins[13065]: ****
Oct 19 16:37:17 ip-172-31-42-86.ec2.internal jenkins[13065]: 2024-10-19 16:37:17.478+0000 [id:32]      INFO    jenkins.InitReactorRunner$1#onAttained: Completed initialization
Oct 19 16:37:17 ip-172-31-42-86.ec2.internal jenkins[13065]: 2024-10-19 16:37:17.516+0000 [id:24]      INFO    hudson.lifecycle.Lifecycle$OnReady: Jenkins is fully up and running
Oct 19 16:37:17 ip-172-31-42-86.ec2.internal systemd[1]: Started Jenkins Continuous Integration Server.
Oct 19 16:37:17 ip-172-31-42-86.ec2.internal jenkins[13065]: 2024-10-19 16:37:17.584+0000 [id:48]      INFO    h.m.DownloadService$Downloadable#load: Obtained the updated data file for huds...venInstaller
Oct 19 16:37:17 ip-172-31-42-86.ec2.internal jenkins[13065]: 2024-10-19 16:37:17.584+0000 [id:48]      INFO    hudson.util.Retrier#start: Performed the action check updates server successfully attempt #1
[ec2-user@ip-172-31-42-86 ~]$ sudo more /var/lib/jenkins/secrets/initialAdminPassword
174B8D24318F44658530f65beb39ff2
[ec2-user@ip-172-31-42-86 ~]$ AC
[ec2-user@ip-172-31-42-86 ~]$
```

Not secure | 98.80.223.40:8080/login?from=%2F

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:

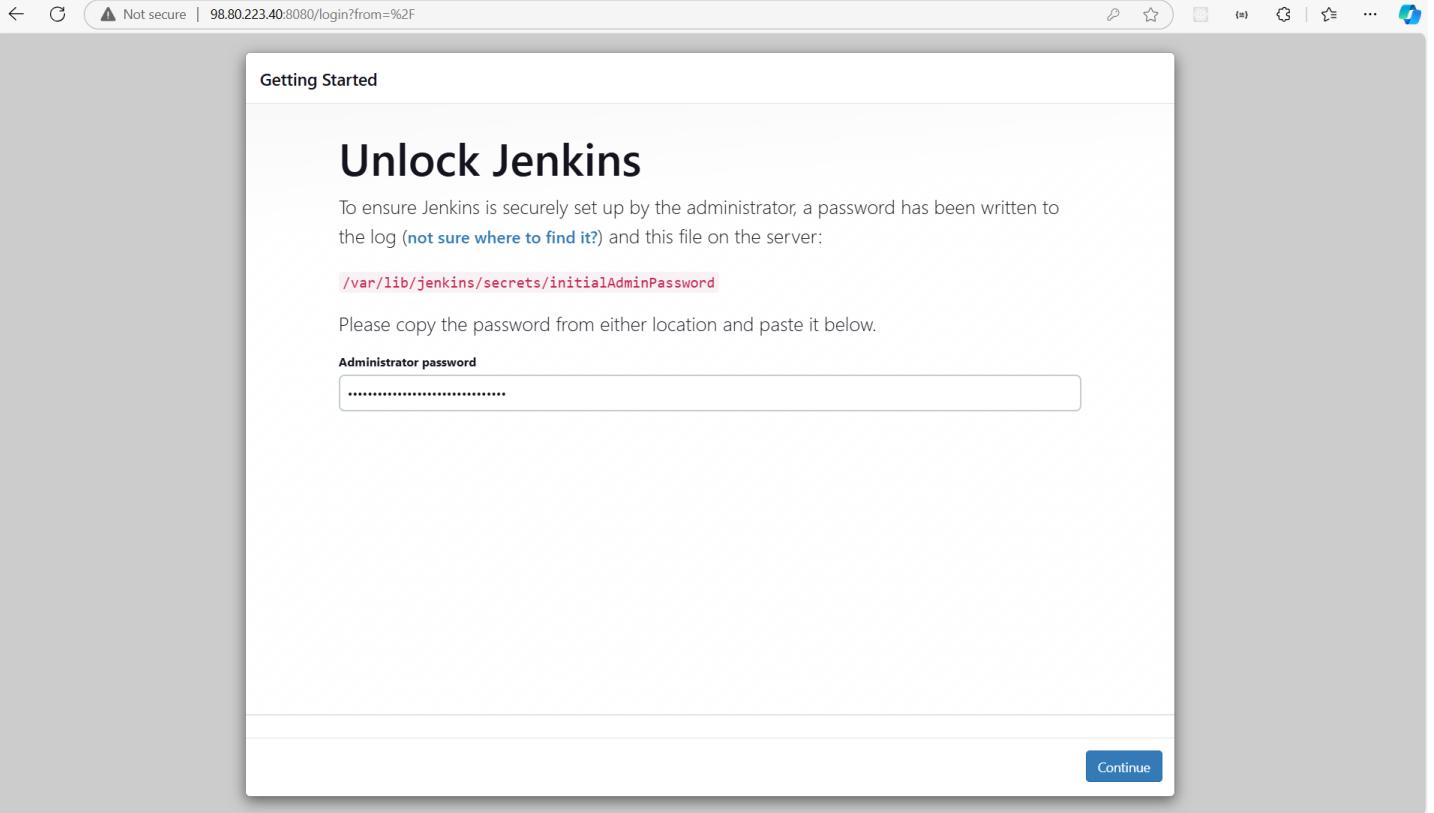
`/var/lib/jenkins/secrets/initialAdminPassword`

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

Administrator password

```
.....
```

Continue



Step 10:

Select install suggested plugins and complete the installation and initial configurations.

Not secure | 98.80.223.40:8080

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

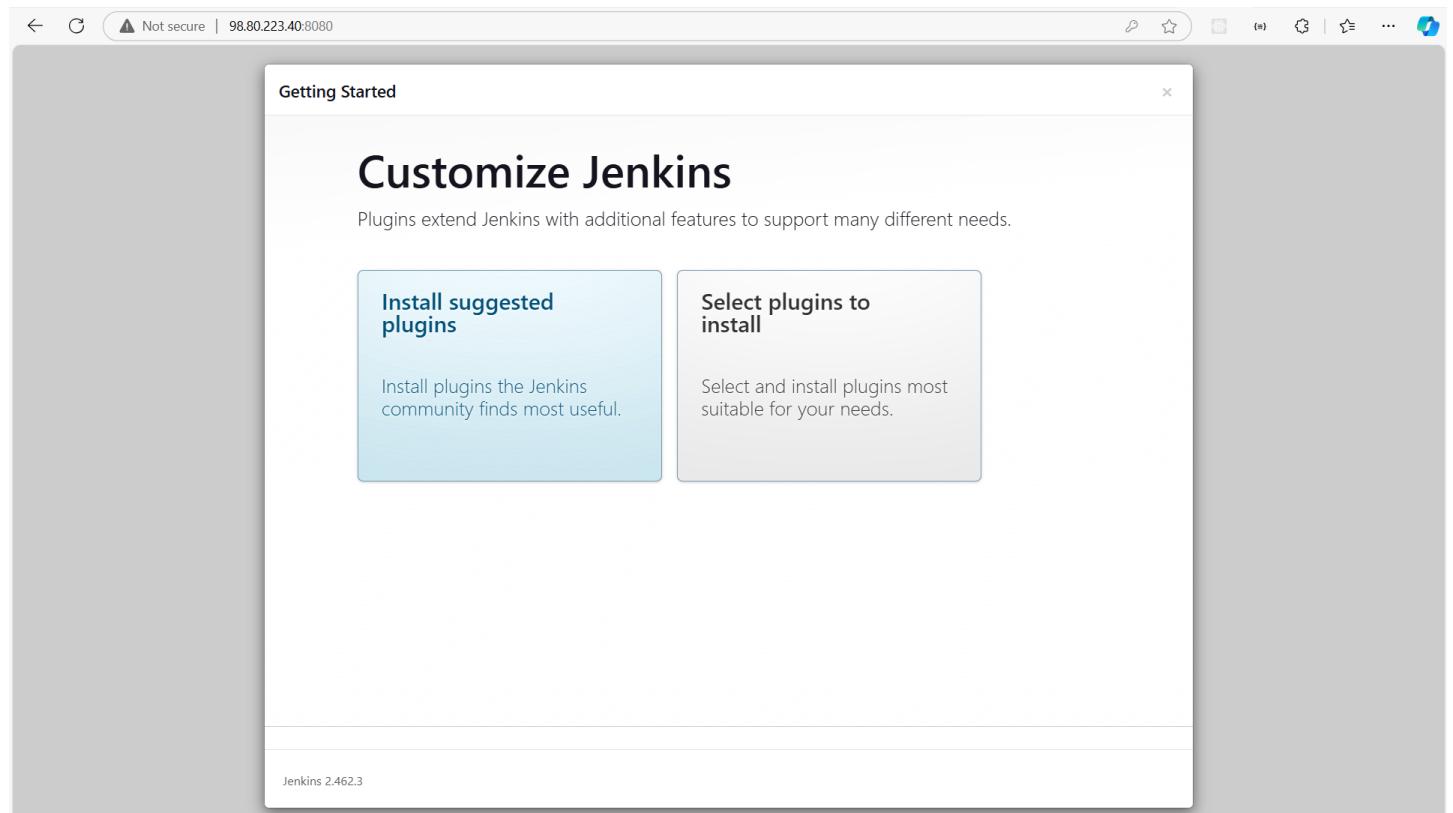
Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.462.3



Not secure | 98.80.223.40:8080

Getting Started

✓ Folders	✓ OWASP Markup Formatter	✓ Build Timeout	✓ Credentials Binding
✓ Timestamper	⌚ Workspace Cleanup	⌚ Ant	⌚ Gradle
⌚ Pipeline	⌚ GitHub Branch Source	⌚ Pipeline: GitHub Groovy Libraries	⌚ Pipeline Graph View
⌚ Git	⌚ SSH Build Agents	⌚ Matrix Authorization Strategy	⌚ PAM Authentication
⌚ LDAP	⌚ Email Extension	⌚ Mailer	⌚ Dark Theme

Ionicons API
Folders
OWASP Markup Formatter
** ASH API
** JSON Path API
** Structs
** Pipeline: Step API
** Token Macro
Build Timeout
** poureycastle API
** Credentials
** Pipeline Credentials
** Variant
** SSH Credentials
Credentials Binding
** SCM API
** Pipeline: API
** commons-lang3 v3.x Jenkins API
Timestamper
** Geforce API
** Script Security
** JavaBeans Activation Framework (JAF) API
** JAXB
** SnakeYAML API
** JSON API
** Jackson 2 API
** - required dependency

Jenkins 2.462.3

Not secure | 98.80.223.40:8080

Getting Started

Create First Admin User

Username

Password

Confirm password

Full name

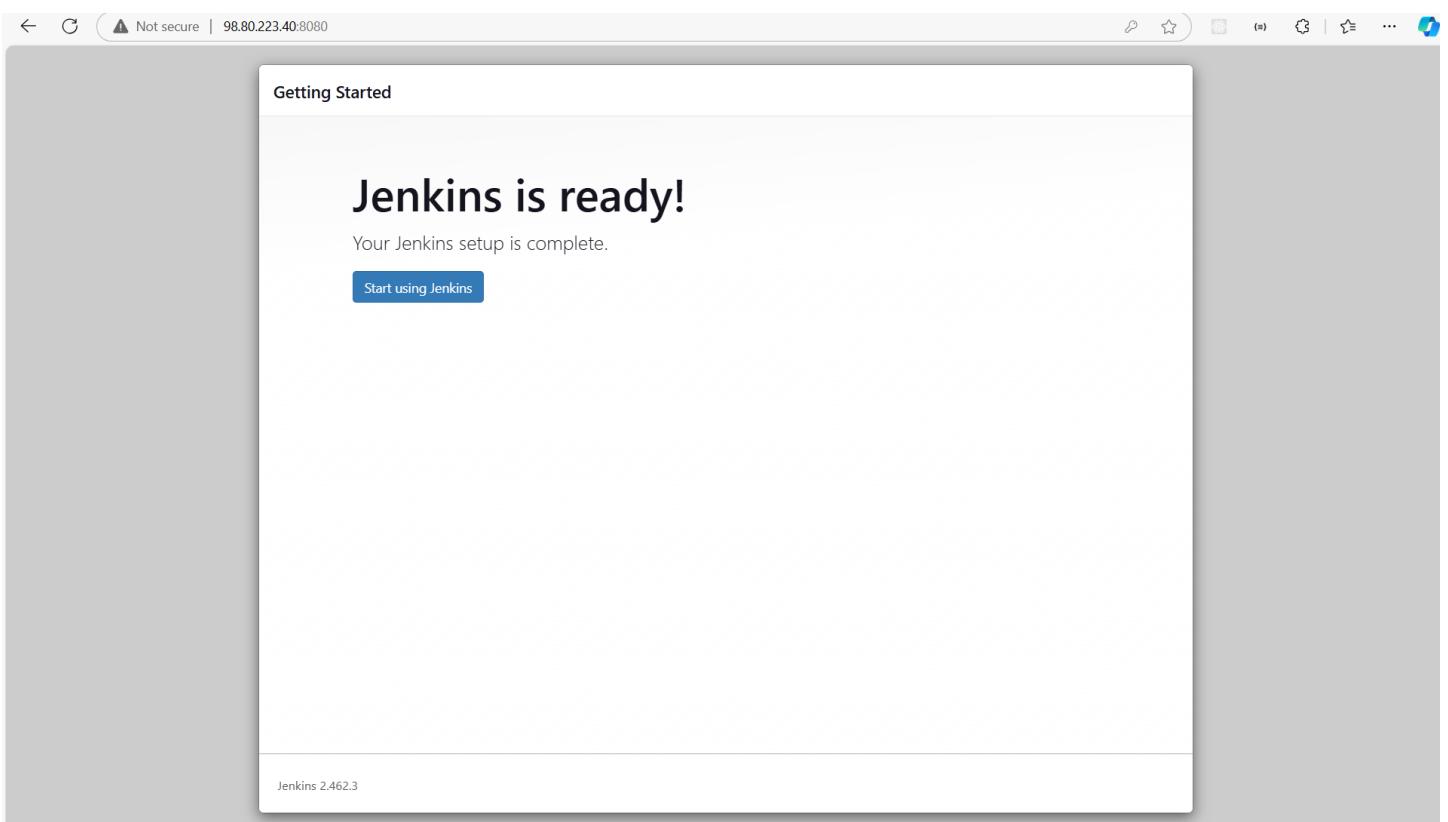
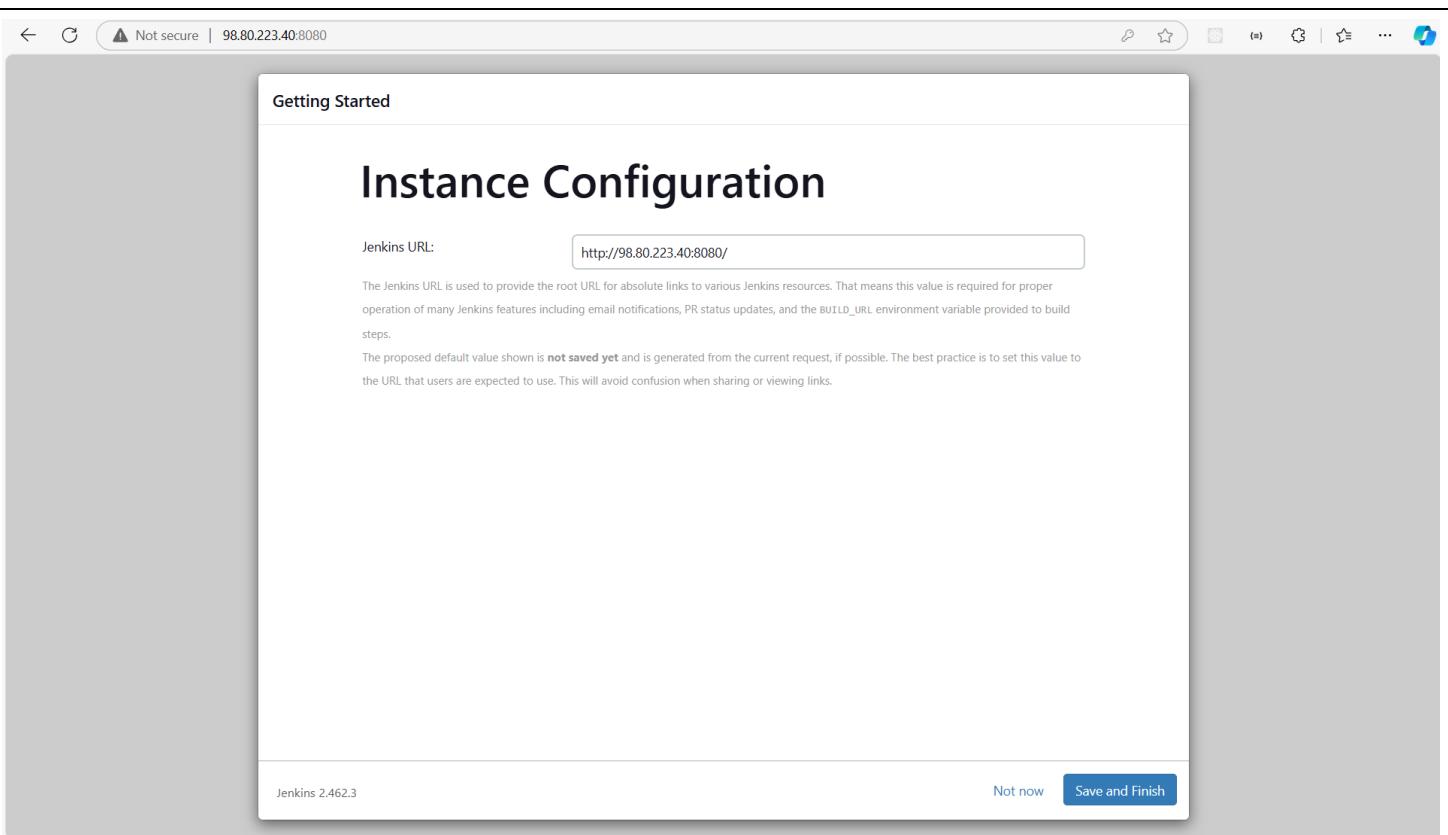
E-mail address

Jenkins 2.462.3

Skip and continue as admin

Save and Continue

Password: Anuprita@4321



Step 11:

After proper initial configuration you will be redirected to this page.

The screenshot shows the Jenkins dashboard at the URL <http://98.80.223.40:8080>. The top navigation bar includes a 'Not secure' warning, the IP address, and user information for 'Anuprita'. The main content area features a 'Welcome to Jenkins!' message and a 'Start building your software project' section. On the left, there are sections for 'Build Queue' (empty), 'Build Executor Status' (2 Idle), 'Create a job' (button with '+'), 'Set up a distributed build' (with 'Set up an agent' and 'Configure a cloud' options), and a link to 'Learn more about distributed builds'. The bottom right corner shows 'REST API' and 'Jenkins 2.46.3'.

Step 12:

Install more plugins which will be required for this experiment.

- a. SonarQube Scanner
- b. Pipeline: Stage View

The screenshot shows the 'Manage Jenkins' page at the URL <http://98.80.223.40:8080/manage/>. The top navigation bar includes a 'Not secure' warning, the IP address, and user information for 'Anuprita'. The main content area features a 'Manage Jenkins' header and a 'System Configuration' section. A prominent red warning box states: 'Building on the built-in node can be a security issue. You should set up distributed builds. See [the documentation](#)'. It also mentions: 'You are running Jenkins on Amazon Linux 2. Jenkins stopped supporting Amazon Linux 2 as of 2023-11-16. Please upgrade to a supported operating system. Refer to [the documentation](#) for details.' Buttons for 'More Info' and 'Ignore' are present. Below this, the 'System Configuration' section includes links for 'System', 'Tools', 'Plugs', 'Nodes', 'Clouds', and 'Appearance'.

Not secure | 98.80.223.40:8080/manage/pluginManager/available

Plugins

Search: sonarqube

Install Name ↓ Released

	Name	Released
<input checked="" type="checkbox"/>	SonarQube Scanner 2.17.2 External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	8 mo 3 days ago
<input type="checkbox"/>	Sonar Gerrit 388.v9b_f1cb_e42306 External Site/Tool Integrations This plugin allows to submit issues from SonarQube to Gerrit as comments directly.	4 mo 17 days ago
<input type="checkbox"/>	SonarQube Generic Coverage 1.0 TODO	5 yr 2 mo ago

Updates Available plugins Installed plugins Advanced settings Download progress REST API Jenkins 2.462.3

Not secure | 98.80.223.40:8080/manage/pluginManager/available

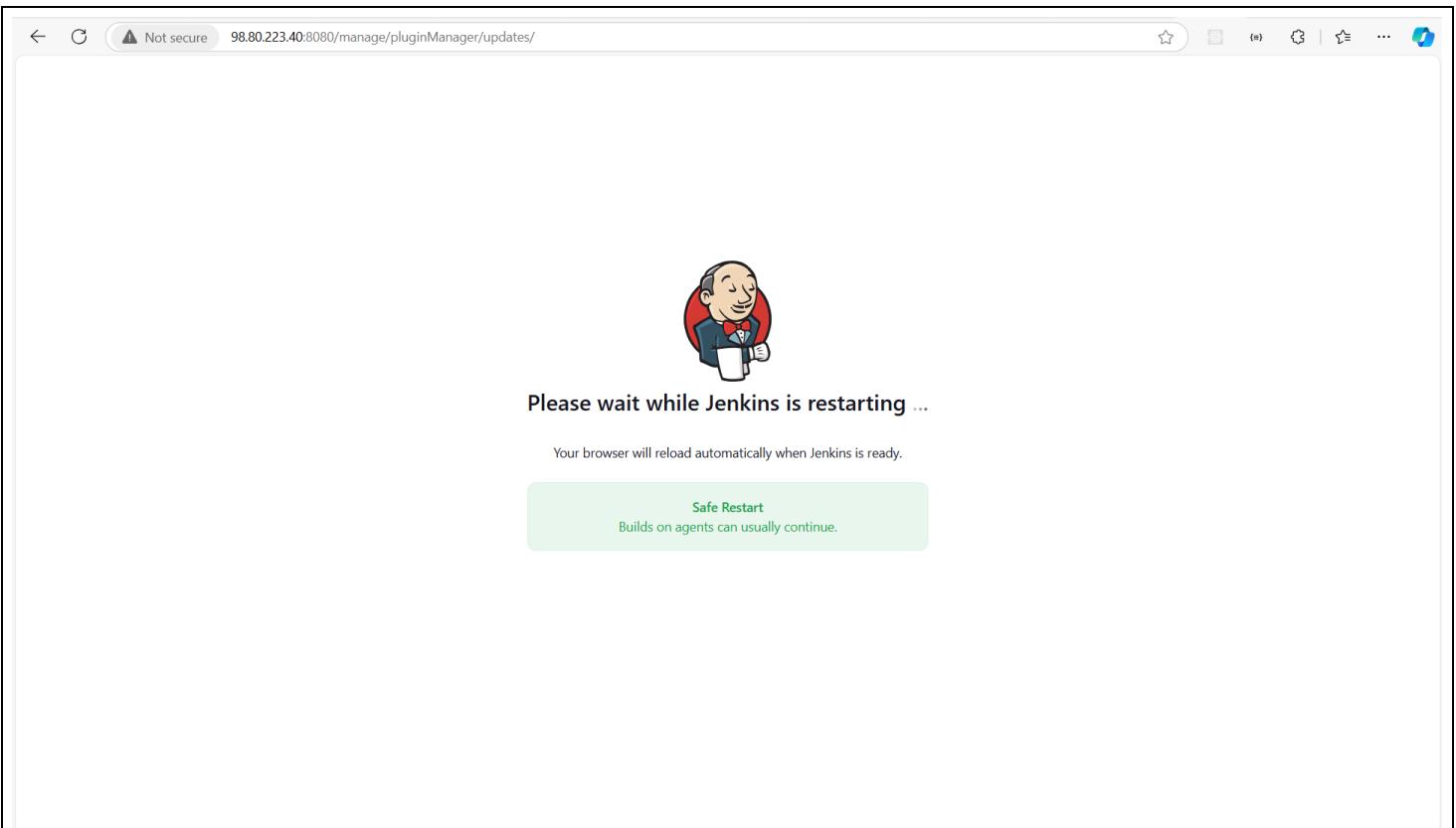
Plugins

Search: pipeline

Install Name ↓ Released

	Name	Released
<input type="checkbox"/>	Pipeline: REST API 2.34 User Interface Provides a REST API to access pipeline and pipeline run data.	11 mo ago
<input checked="" type="checkbox"/>	Pipeline: Stage View 2.34 User Interface Pipeline Stage View Plugin.	11 mo ago
<input type="checkbox"/>	Docker Pipeline 580.vc0c340686b_54 pipeline DevOps Deployment docker Build and use Docker containers from pipelines.	5 mo 0 days ago
<input type="checkbox"/>	Lockable Resources 1320.v1f0dff578476 pipeline Cluster Management Agent Management This plugin allows to define external resources (such as printers, phones, computers) that can be locked by builds. If a build requires an external resource which is already locked, it will wait for the resource to be free.	3 days 22 hr ago
<input type="checkbox"/>	Pipeline: Deprecated Groovy Libraries 612.v55f2f80781ef Miscellaneous Hosting of Pipeline Groovy libraries inside a Jenkins Git server. Deprecated . Use Pipeline: Groovy Libraries instead. If you see this plugin installed just because you upgraded, you can probably uninstall it now. This plugin should only be used if you have historically pushed libraries to a Git server inside Jenkins.	9 mo 4 days ago

Updates Available plugins Installed plugins Advanced settings Download progress REST API Jenkins 2.462.3



Step 13:

Go to Manage Jenkins > Tools. Scroll down to SonarQube Scanner installations and add the SonarQube Scanner and then click on the save button.

A screenshot of the Jenkins 'Configure Tools' page. The URL is 98.80.223.40:8080/manage/configureTools/. The page shows the 'SonarQube Scanner installations' section. A new entry is being added with the name 'sonarqube'. The 'Install automatically' checkbox is checked. Under 'Install from Maven Central', the version is set to 'SonarQube Scanner 6.2.1.4610'. There are 'Save' and 'Apply' buttons at the bottom.

Sonarqube installation

Reference video: <https://www.youtube.com/watch?v=E5hMOGeBT-o&t=38s>

Step 1:

Click on the SonarQubeServer and click on connect.

The screenshot shows the AWS EC2 Instances page. There are two instances listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
JenkinsServer	i-05f12961a9cf8cf3e	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1a	ec2-98-80-223-40.compute-1.amazonaws.com
SonarQubeServer	i-061f29e11e3fa1a8b	Running	t2.medium	2/2 checks passed	View alarms +	us-east-1c	ec2-54-210-13-192.compute-1.amazonaws.com

Below the instances, the details for the selected instance ('i-061f29e11e3fa1a8b (SonarQubeServer)') are displayed. The 'Details' tab is selected, showing the following information:

- Instance ID: i-061f29e11e3fa1a8b (SonarQubeServer)
- Public IPv4 address: 54.210.13.192 [open address]
- Private IPv4 addresses: 172.31.92.157
- IPv6 address: -
- Instance state: Running
- Public IPv4 DNS: ec2-54-210-13-192.compute-1.amazonaws.com [open address]
- Hostname type: IP name: ip-172-31-92-157.ec2.internal
- Private IP DNS name (IPv4 only): ip-172-31-92-157.ec2.internal
- Answer private resource DNS name: -
- Instance type: t2.medium
- Elastic IP addresses: -

At the bottom of the page, there is a 'Connect to instance' button with options for EC2 Instance Connect, Session Manager, SSH client, and EC2 serial console. The 'SSH client' tab is selected.

Step 2:

Open Git Bash and go to the directory which has the Key downloaded. If you don't have the key downloaded, create a key pair and download the .pem file for the key.

Since, I have the key downloaded in Downloads directory, I used the following commands:

```
cd Download
```

```
dir AMsonarqube.pem*
```

```
ssh -i "AMsonarqube.pem" ec2-user@ec2-98-80-223-40.compute-1.amazonaws.com
```

```
ec2-user@ip-172-31-92-157:~
```

```
User@DESKTOP-QQGK15A MINGW64 ~ (master)
$ cd Downloads

User@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ dir AMsonarqube.pem
AMsonarqube.pem

User@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ ssh -i "AMsonarqube.pem" ec2-user@ec2-54-210-13-192.compute-1.amazonaws.com
The authenticity of host 'ec2-54-210-13-192.compute-1.amazonaws.com (54.210.13.192)' can't be established.
ED25519 key fingerprint is SHA256:1npBgiusutDBAlldNoAcPyZQhmXvYRxxIskaITxGQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-210-13-192.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

          #
          #####      Amazon Linux 2
~~\ #####\                                AL2 End of Life is 2025-06-30.
~~ \###)                                A newer version of Amazon Linux is available!
~~ \#)                                Amazon Linux 2023, GA and supported until 2028-03-15.
~~ \#)                                https://aws.amazon.com/linux/amazon-linux-2023/
/m/` 

[ec2-user@ip-172-31-92-157 ~]$ sudo wget -O /etc/yum.repos.d/sonar.repo http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo
--2024-10-19 16:56:06-- http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 204.68.111.105
Connecting to downloads.sourceforge.net (downloads.sourceforge.net)|204.68.111.105|:80... connected.
HTTP request sent, awaiting response... 304 Found
Location: http://psychz.dl.sourceforge.net/project/sonar-pkg/rpm/sonar.repo?viafs=1 [following]
--2024-10-19 16:56:06-- http://psychz.dl.sourceforge.net/project/sonar-pkg/rpm/sonar.repo?viafs=1
Resolving psychz.dl.sourceforge.net (psychz.dl.sourceforge.net)... 208.87.241.191
Connecting to psychz.dl.sourceforge.net (psychz.dl.sourceforge.net)|208.87.241.191|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 93 [application/octet-stream]
Saving to: '/etc/yum.repos.d/sonar.repo'

100%[=====] 93           --.-K/s   in 0s

2024-10-19 16:56:07 (16.7 MB/s) - '/etc/yum.repos.d/sonar.repo' saved [93/93]

[ec2-user@ip-172-31-92-157 ~]$ sudo yum install sonar -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2-repos
sonar/primary_db
Resolving Dependencies
--> Running transaction check
--> Package sonar.noarch 0:7.1-1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
          Package          Arch        Version       Repository    Size
=====
Installing:
  sonar                noarch     7.1-1         sonar        142 M

Transaction Summary
=====
Install 1 Package
```

Step 3:

Now, in order to install java run the following commands:

sudo su

```
sudo yum install java-17-amazon-corretto-headless
```

```
sudo yum install java-17-amazon-corretto
```

```
sudo dnf install java-17-amazon-corretto-devel
```

```

ec2-user@ip-172-31-42-155:~$ Transaction Summary
=====  

Install 27 Packages  

Total download size: 100 M  

Installed size: 261 M  

Is this ok [y/N]: y  

Downloading Packages:  

(1/27): cairo-1.17.6-2.amzn2023.0.1.x86_64.rpm  

(2/27): dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch.rpm  

(3/27): alsalib-1.2.7-2.1.amzn2023.0.2.x86_64.rpm  

(4/27): dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch.rpm  

(5/27): fonts-filesystem-1.0.5-12.amzn2023.0.2.noarch.rpm  

(6/27): fontconfig-2.13.94-2.amzn2023.0.2.noarch.rpm  

(7/27): fontconfig-2.13.94-2.amzn2023.0.2.x86_64.rpm  

(8/27): google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch.rpm  

(9/27): graphite2-1.3.14-7.amzn2023.0.2.x86_64.rpm  

(10/27): freetype-2.13.2-5.amzn2023.0.1.x86_64.rpm  

(11/27): harfbuzz-7.0-0-2.amzn2023.0.1.x86_64.rpm  

(12/27): google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch.rpm  

(13/27): java-17-amazon-corretto-devel-17.0.12+7-1.amzn2023.1.x86_64.rpm  

(14/27): javapackages-fsitem-6.0.0-7.amzn2023.0.6.noarch.rpm  

(15/27): langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch.rpm  

(16/27): libX11-1.8.10-2.amzn2023.0.1.x86_64.rpm  

(17/27): libX11-common-1.8.10-2.amzn2023.0.1.noarch.rpm  

(18/27): libXext-1.3.6-1.amzn2023.0.2.x86_64.rpm  

(19/27): libXext-1.3.6-1.amzn2023.0.1.x86_64.rpm  

(20/27): libXrender-0.9.11-6.amzn2023.0.1.x86_64.rpm  

(21/27): libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64.rpm  

(22/27): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm  

(23/27): libpng-1.6.37-10.amzn2023.0.6.x86_64.rpm  

(24/27): libxcb-1.17.0-1.amzn2023.0.1.x86_64.rpm  

(25/27): xml-common-0.6.3-56.amzn2023.0.2.noarch.rpm  

(26/27): pixman-0.43.4-1.amzn2023.0.4.x86_64.rpm  

(27/27): java-17-amazon-corretto-headless-17.0.12+7-1.amzn2023.1.x86_64.rpm  

Total 54 MB/s | 100 MB 00:01  

Running transaction check  

Transaction check succeeded.  

Running transaction test  

Transaction test succeeded.  

Running transaction  

Preparing : 1/1  

Installing : fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch 1/27  

Installing : dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch 2/27  

Installing : dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch 3/27  

Installing : dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch 4/27  

Installing : fontconfig-2.13.94-2.amzn2023.0.2.noarch 5/27  

Running scriptlet: xml-common-0.6.3-56.amzn2023.0.2.noarch 6/27  

Installing : xml-common-0.6.3-56.amzn2023.0.2.noarch 7/27  

Installing : pixman-0.43.4-1.amzn2023.0.4.x86_64 8/27  

Installing : libjpeg-turbo-2.1.4-2.amzn2023.0.5.x86_64 9/27  

Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/27  

Installing : libpng-1.6.37-10.amzn2023.0.6.x86_64 11/27  

Installing : libxcb-1.17.0-1.amzn2023.0.1.x86_64 12/27  

Installing : libX11-1.8.10-2.amzn2023.0.1.noarch 13/27  

Installing : libXext-1.3.6-1.amzn2023.0.1.x86_64 14/27  

Installing : libXrender-0.9.11-6.amzn2023.0.1.x86_64 15/27  

Installing : javapackages-fsitem-6.0.0-7.amzn2023.0.6.noarch 16/27  

Installing : graphite2-1.3.14-7.amzn2023.0.2.x86_64 17/27  

Installing : google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch 18/27  

Installing : google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch 19/27  

Installing : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch 20/27  

ec2-user@ip-172-31-42-155:~$  

Installing : libX11-1.8.10-2.amzn2023.0.1.x86_64 1/27  

Installing : libXext-1.3.6-1.amzn2023.0.1.x86_64 2/27  

Installing : libXrender-0.9.11-6.amzn2023.0.1.x86_64 3/27  

Installing : dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch 4/27  

Installing : dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch 5/27  

Installing : dejavu-serif-fonts-2.37-16.amzn2023.0.2.noarch 6/27  

Installing : fontconfig-2.13.94-2.amzn2023.0.2.x86_64 7/27  

Running scriptlet: xml-common-0.6.3-56.amzn2023.0.2.noarch 8/27  

Installing : graphite2-1.3.14-7.amzn2023.0.2.x86_64 9/27  

Installing : google-noto-fonts-common-20201206-2.amzn2023.0.2.noarch 10/27  

Installing : google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch 11/27  

Installing : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch 12/27  

Running scriptlet: java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64 13/27  

Running scriptlet: java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64 14/27  

Installing : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64 15/27  

Running scriptlet: java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64 16/27  

Running scriptlet: fontconfig-2.13.94-2.amzn2023.0.2.x86_64 17/27  

Running scriptlet: java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64 18/27  

Verifying : alsalib-1.2.7.2-2.1.amzn2023.0.2.x86_64 19/27  

Verifying : java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64 20/27  

Running scriptlet: java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64 21/27  

Verifying : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64 22/27  

Verifying : fontconfig-2.13.94-2.amzn2023.0.2.x86_64 23/27  

Verifying : fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch 24/27  

Verifying : freetype-2.13.2-5.amzn2023.0.1.x86_64 25/27  

Verifying : dejavu-sans-fonts-2.37-16.amzn2023.0.2.noarch 26/27  

Running scriptlet: java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64 27/27  

Verifying : java-17-amazon-corretto-devel-1:17.0.12+7-1.amzn2023.1.x86_64 28/27  

Verifying : javapackages-fsitem-6.0.0-7.amzn2023.0.6.noarch 29/27  

Verifying : langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch 30/27  

Verifying : libX11-1.8.10-2.amzn2023.0.1.x86_64 31/27  

Verifying : libX11-common-1.8.10-2.amzn2023.0.1.noarch 32/27  

Verifying : libXext-1.3.6-1.amzn2023.0.1.x86_64 33/27  

Verifying : libXrender-0.9.11-6.amzn2023.0.1.x86_64 34/27  

Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 35/27  

Verifying : libpng-2.1.6.37-10.amzn2023.0.6.x86_64 36/27  

Verifying : pixman-0.43.4-1.amzn2023.0.4.x86_64 37/27  

Verifying : xml-common-0.6.3-56.amzn2023.0.2.noarch 38/27  

Installed:  

alsalib-1.2.7.2-2.1.amzn2023.0.2.x86_64  

dejavu-sans-mono-fonts-2.37-16.amzn2023.0.2.noarch  

fonts-filesystem-1:2.0.5-12.amzn2023.0.2.noarch  

google-noto-sans-vf-fonts-20201206-2.amzn2023.0.2.noarch  

java-17-amazon-corretto-headless-1:17.0.12+7-1.amzn2023.1.x86_64  

langpacks-core-font-en-3.0-21.amzn2023.0.4.noarch  

libXau-1.0.11-6.amzn2023.0.1.x86_64  

libbrotli-1.0.9-4.amzn2023.0.2.x86_64  

libxcb-1.17.0-1.amzn2023.0.1.x86_64  

Completed!  

[ec2-user@ip-172-31-42-155 ~]$ java --version  

openjdk 17.0.2 2024-07-16 LTS  

OpenJDK Runtime Environment Corretto-17.0.12.7.1 (build 17.0.12+7-LTS)  

OpenJDK 64-bit Server VM Corretto-17.0.12.7.1 (build 17.0.12+7-LTS, mixed mode, sharing)  

[ec2-user@ip-172-31-42-155 ~]$
```

Step 4:

Now, run the following command to install sonarqube:

```
sudo wget -O /etc/yum.repos.d/sonar.repo http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo  

sudo yum install sonar -y
```

```

ec2-user@ip-172-31-92-157:~$ cd Downloads
ec2-user@DESKTOP-QQGK15A MINGW64 ~ (master)
$ cd Downloads
ec2-user@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ dir AMsonarqube.pem
AMsonarqube.pem

User@DESKTOP-QQGK15A MINGW64 ~/Downloads (master)
$ ssh -i "AMsonarqube.pem" ec2-user@ec2-54-210-13-192.compute-1.amazonaws.com
The authenticity of host 'ec2-54-210-13-192.compute-1.amazonaws.com (54.210.13.192)' can't be established.
ED25519 key fingerprint is SHA256:1np8qisuh0tDBAUdn0AcPzQhmxvwHRxx1skaiTxQQ.
This key is not known by any other name.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
warning: Permanently added 'ec2-54-210-13-192.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

  _\###_
 /#\###\ Amazon Linux 2
  ~~ \### AL2 End of Life is 2025-06-30.
  ~~ \#/ V~' ->
  ~~ A newer version of Amazon Linux is available!
  ~~ /_/_ Amazon Linux 2023, GA and supported until 2028-03-15.
  /m/  https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-92-157 ~]$ sudo wget -O /etc/yum.repos.d/sonar.repo http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo
--2024-10-19 16:56:06 -- http://downloads.sourceforge.net/project/sonar-pkg/rpm/sonar.repo
Resolving downloads.sourceforge.net (downloads.sourceforge.net)... 204.68.111.105
Connecting to downloads.sourceforge.net (downloads.sourceforge.net)|204.68.111.105|:80... connected.
HTTP request sent, awaiting response... 200 OK
Location: http://psychz.dl.sourceforge.net/project/sonar-pkg/rpm/sonar.repo?viasf=1 [following]
--2024-10-19 16:56:06 -- http://psychz.dl.sourceforge.net/project/sonar-pkg/rpm/sonar.repo?viasf=1
Resolving psychz.dl.sourceforge.net (psychz.dl.sourceforge.net)... 208.87.241.191
Connecting to psychz.dl.sourceforge.net (psychz.dl.sourceforge.net)|208.87.241.191|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 93 [application/octet-stream]
Saving to: '/etc/yum.repos.d/sonar.repo'

100%[=====] 93      --.-K/s   in 0s

2024-10-19 16:56:07 (16.7 MB/s) - '/etc/yum.repos.d/sonar.repo' saved [93/93]

[ec2-user@ip-172-31-92-157 ~]$ sudo yum install sonar -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
sonar
sonar/primary_db
Resolving Dependencies
>--> Running transaction check
-->> Package sonar.noarch 0:7.1-1 will be installed
-->> Finished Dependency Resolution
Dependencies Resolved

=====
Package           Arch       Version        Repository      Size
=====
Installing:
sonar            noarch    7.1-1          sonar          142 M
Transaction Summary
Install 1 Package

```

Step 5:

Now, run the following commands to install the sonarqube

```

sudo su
cd /opt
||
wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-10.7.0.96327.zip
unzip sonarqube-10.7.0.96327.zip
||
cd sonarqube-10.7.0.96327
||
cd conf
||
cat sonar.properties
cd ..
cd bin
cd linux-x86-64
||
./sonar.sh start

sudo adduser sonar
sudo passwd sonar
sudo chown -R sonar:sonar /opt/sonarqube-10.7.0.96327
su - sonar
cd /opt/sonarqube-10.7.0.96327/bin/linux-x86-64/
./sonar.sh start
./sonar.sh status

```

```
MINGW64: C:/Users/User/Downloads -    
inflate: sonarqube-10.7.0.96327/web/WEB-INF/web.xml
creating: sonarqube-10.7.0.96327/lib/jdbc/mssql/
inflate: sonarqube-10.7.0.96327/lib/jdbc/mssql/mssql-jdbc-12.6.3.jre11.jar
creating: sonarqube-10.7.0.96327/lib/jdbc/postgresql/
inflate: sonarqube-10.7.0.96327/lib/jdbc/postgresql/postgresql-42.7.3.jar
creating: sonarqube-10.7.0.96327/lib/jdbc/h2/
inflate: sonarqube-10.7.0.96327/lib/jdbc/h2/h2-2.2.224.jar
inflate: sonarqube-10.7.0.96327/lib/sonar-shutdowner-10.7.0.96327.jar
creating: sonarqube-10.7.0.96327/elasticsearch/plugins/
[root@ip-172-31-82-31 opt]# 11
total 741384
drwxr-xr-x 4 root root 33 Oct 10 21:23 aws
drwxr-xr-x 11 sonar sonar 144 Oct 18 18:12 sonar
drwxr-xr-x 12 root root 384 Sep 27 2016 sonarqube-10.7.0.96327
drwxr-xr-x 1 root root 759175712 Sep 27 15:52 sonarqube-10.7.0.96327.zip
[root@ip-172-31-82-31 opt]# cd sonarqube-10.7.0.96327
[root@ip-172-31-82-31 sonarqube-10.7.0.96327]# 11
total 112
-rw-r--r-- 1 root root 7651 Sep 27 15:13 COPYING
drwxr-xr-x 6 root root 117 Sep 27 15:13 bin
drwxr-xr-x 2 root root 30 Sep 27 15:13 conf
drwxr-xr-x 2 root root 24 Sep 27 15:13 data
-rw-r--r-- 1 root root 73376 Sep 27 15:14 dependency-license.json
drwxr-xr-x 7 root root 132 Sep 27 15:16 elasticsearch
drwxr-xr-x 4 root root 40 Sep 27 15:13 extensions
drwxr-xr-x 2 root root 16384 Sep 27 15:16 jre
drwxr-xr-x 2 root root 134 Sep 27 15:13 logs
drwxr-xr-x 2 root root 24 Sep 27 15:13 temp
drwxr-xr-x 6 root root 16384 Sep 27 15:16 web
[root@ip-172-31-82-31 sonarqube-10.7.0.96327]# cd conf
[root@ip-172-31-82-31 conf]# 11
total 24
-rw-r--r-- 1 root root 21761 Sep 27 15:13 sonar.properties
[root@ip-172-31-82-31 conf]# cat sonar.properties
#-----
# IMPORTANT:
# This file will *not* be reloaded when using the api/system/restart endpoint.
# In order for any change made to this file to be taken into account, you must perform a full
# restart of the main SonarQube service.
#-----

# Property values can:
# - be overridden by environment variables. The name of the corresponding environment variable is the
# upper-cased name of the property where all the dot ('.') and dash ('-') characters are replaced by
# underscores ('_'). For example, to override 'sonar.web.systemPasscode' use 'SONAR_WEB_SYSTEMPASSCODE'.
# - be encrypted. See https://docs.sonarsource.com/sonarqube/latest/instance-administration/security/#settings-encryption
#-----

# DATABASE
#-----
# IMPORTANT:
# The embedded H2 database is used by default. It is recommended for tests but not for
# production use. Supported databases are Oracle, PostgreSQL and Microsoft SQLServer.
# - Changes to database connection URL (sonar.jdbc.url) can affect SonarSource licensed products.

# User credentials.
# Permissions to create tables, indices and triggers must be granted to JDBC user.
# The schema must be created first.
#sonar.jdbc.username=
#sonar.jdbc.password=

#----- Embedded Database (default)
# H2 embedded database server listening port, defaults to 9092
```

```

MINGW64:/c/Users/User/Downloads
#---- Oracle 19c/21c
# The Oracle JDBC driver must be copied into the directory extensions/jdbc-driver/oracle/.
# Only the thin client is supported, and we recommend using the latest Oracle JDBC driver. See
# https://jira.sonarsource.com/browse/SONAR-9758 for more details.
# If you need to set the schema, please refer to http://jira.sonarsource.com/browse/SONAR-5000
#sonar.jdbc.url=jdbc:oracle:thin:@localhost:1521/XE

#----- PostgreSQL 11 or greater
# By default the schema named "public" is used. It can be overridden with the parameter "currentSchema".
#sonar.jdbc.url=jdbc:postgresql://localhost/sonarqube?currentSchema=my_schema

#----- Microsoft SQL Server 2014/2016/2017/2019/2022 and SQL Azure
# A database named sonar must exist and its collation must be case-sensitive (CS) and accent-sensitive (AS)
# Use the following connection string if you want to use integrated security with Microsoft Sql Server
# Do not set sonar.jdbc.username or sonar.jdbc.password property if you are using Integrated Security
# For Integrated Security to work, you have to install the Microsoft SQL JDBC Auth package
# Please refer to the online documentation https://docs.sonarsource.com/sonarqube
# for the exact procedure for this version of SonarQube.
#sonar.jdbc.url=jdbc:sqlserver://localhost:databaseName=sonar;integratedSecurity=true

# Use the following connection string if you want to use SQL Auth while connecting to MS Sql Server.
# Set the sonar.jdbc.username and sonar.jdbc.password appropriately.
#sonar.jdbc.url=jdbc:sqlserver://localhost:databaseName=sonar

#----- Connection pool settings
# The maximum number of active connections that can be allocated
# at the same time, or negative for no limit.
# The recommended value is 1.2 * max sizes of HTTP pools. For example if HTTP ports are
# enabled with default sizes (50, see property sonar.web.http.maxThreads)
[root@ip-172-31-82-31 conf]\# cd ..
[root@ip-172-31-82-31 sonarqube-10.7.0.96327]\# ll
total 112
-rw-r--r--. 1 root root 7651 Sep 27 15:13 COPYING
drwxr-xr-x. 6 root root 117 Sep 27 15:13 bin
drwxr-xr-x. 2 root root 30 Sep 27 15:13 conf
drwxr-xr-x. 2 root root 24 Sep 27 15:13 data
-rw-r--r--. 1 root root 73376 Sep 27 15:14 dependency-license.json
drwxr-xr-x. 7 root root 132 Sep 27 15:16 elasticsearch
drwxr-xr-x. 4 root root 40 Sep 27 15:13 extensions
drwxr-xr-x. 2 root root 16384 Sep 27 15:16 jres
drwxr-xr-x. 5 root root 134 Sep 27 15:16 lib
drwxr-xr-x. 2 root root 24 Sep 27 15:13 logs
drwxr-xr-x. 2 root root 24 Sep 27 15:13 temp
drwxr-xr-x. 6 root root 16384 Sep 27 15:16 web
[root@ip-172-31-82-31 sonarqube-10.7.0.96327]\# cd bin
[root@ip-172-31-82-31 bin]\# ll
total 4
-rw-r--r--. 1 root root 101 Jun 10 23:34 elasticsearch
drwxr-xr-x. 2 root root 22 Sep 27 15:13 linux-x86-64
drwxr-xr-x. 2 root root 22 Sep 27 15:13 macosx-universal-64
drwxr-xr-x. 3 root root 63 Sep 27 15:13 windows-x86-64
drwxr-xr-x. 2 root root 25 Sep 27 15:13 winsw-license
[root@ip-172-31-82-31 bin]\# cd Linux-x86-64
[root@ip-172-31-82-31 linux-x86-64]\# ll
total 8
-rw-r--r--. 1 root root 7192 Sep 27 15:13 sonar.sh
[root@ip-172-31-82-31 linux-x86-64]\# ./sonar.sh
/usr/bin/java
Usage: ./sonar.sh { console | start | stop | force-stop | restart | status | dump }
[root@ip-172-31-82-31 linux-x86-64]\# ./sonar.sh start
/usr/bin/java
Starting SonarQube...

```

```

MINGW64:/c/Users/User/Downloads
[root@ip-172-31-82-31 bin]\# cd linux-x86-64
[root@ip-172-31-82-31 linux-x86-64]\# ll
total 8
-rw-r--r--. 1 root root 7192 Sep 27 15:13 sonar.sh
[root@ip-172-31-82-31 linux-x86-64]\# ./sonar.sh
/usr/bin/java
Usage: ./sonar.sh { console | start | stop | force-stop | restart | status | dump }
[root@ip-172-31-82-31 linux-x86-64]\# ./sonar.sh start
/usr/bin/java
Starting SonarQube...
Started SonarQube.
[root@ip-172-31-82-31 linux-x86-64]\# ./sonar.sh status
/usr/bin/java
Removed stale pid file: ./SonarQube.pid
SonarQube is not running.
[root@ip-172-31-82-31 linux-x86-64]\# cd /opt/sonarqube-10.7.0.96327/logs
[root@ip-172-31-82-31 logs]\# cat sonar.log
2024.10.18 19:13:39 INFO app[] [o.s.a.AppFileSystem] Cleaning or creating temp directory /opt/sonarqube-10.7.0.96327/temp
2024.10.18 19:13:39 INFO app[] [o.s.a.EsSettings] Elasticsearch listening on [HTTP:127.0.0.1:9001, TCP:127.0.0.1:4529]
2024.10.18 19:13:39 INFO app[] [o.s.a.ProcessLauncherImpl] Launch process[Elasticsearch] from /opt/sonarqube-10.7.0.96327/elasticsearch: /usr/lib/jvm/java-17-amazon-corretto.x86_64/bin/java -Xms4m -Xmx64m -XX:Desdistribution-type=tar -cp /opt/sonarqube-10.7.0.96327/elasticsearch/lib/*;/opt/sonarqube-10.7.0.96327/elasticsearch-des.path.conf=/opt/sonarqube-10.7.0.96327/temp/conf/esDesdistribution-type=tar -cp /opt/sonarqube-10.7.0.96327/elasticsearch/lib/*;/opt/sonarqube-10.7.0.96327/elasticsearch-des.path.conf=/opt/sonarqube-10.7.0.96327/elasticsearch/lib/cli-launcher/* org.elasticsearch.cli.toolLauncher
2024.10.18 19:13:39 INFO app[] [o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running
2024.10.18 19:13:43 WARN app[] [o.s.a.p.AbstractManagedProcess] Process exited with exit value [ElasticSearch]: 1
2024.10.18 19:13:43 INFO app[] [o.s.a.SchedulerImpl] Process[ElasticSearch] is stopped
2024.10.18 19:13:43 INFO app[] [o.s.a.SchedulerImpl] SonarQube is stopped
[root@ip-172-31-82-31 logs]\# cat /opt/sonarqube-10.7.0.96327/logs/es.log
2024.10.18 19:13:43 INFO es[] [o.e.NativeAccess] Using [jna] native provider and native methods for [Linux]
2024.10.18 19:13:43 ERROR es[] [o.e.b.Elasticsearch] Fatal exception while booting Elasticsearch
java.lang.RuntimeException: cannot run elasticsearch as root
    at org.elasticsearch.bootstrap.Elasticsearch.initializeNatives(Elasticsearch.java:286) ~[elasticsearch-8.14.1.jar:?]
    at org.elasticsearch.bootstrap.Elasticsearch.initPhase2(Elasticsearch.java:169) ~[elasticsearch-8.14.1.jar:?]
    at org.elasticsearch.bootstrap.Elasticsearch.main(Elasticsearch.java:74) ~[elasticsearch-8.14.1.jar:?]
[root@ip-172-31-82-31 logs]\# cd ..
[root@ip-172-31-82-31 opt]\# cd ..
[root@ip-172-31-82-31 ]\# sudo adduser sonar
adduser: user `sonar` already exists
[root@ip-172-31-82-31 ]\# sudo passwd sonar
Changing password for user sonar.
New password:
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
Retype new password:
password: all authentication tokens updated successfully.
[root@ip-172-31-82-31 ]\# sudo chown -R sonar:sonar /opt/sonarqube-10.7.0.96327
[root@ip-172-31-82-31 ]\# su - sonar
last login: Fri Oct 18 18:26:31 UTC 2024 on pts/1
[sonar@ip-172-31-82-31 ~]\$ cd /opt/sonarqube-10.7.0.96327/bin/linux-x86-64
[sonar@ip-172-31-82-31 linux-x86-64]\$ ./sonar.sh start
/usr/bin/java
Starting SonarQube...
Started SonarQube.
[sonar@ip-172-31-82-31 linux-x86-64]\$ ./sonar.sh status
/usr/bin/java
SonarQube is running (29499).
[sonar@ip-172-31-82-31 linux-x86-64]\$ client_loop: send disconnect: Connection reset by peer

```

User@DESKTOP-QOK15A MINGW64 ~ /Downloads (master)

Step 6:

Now, go to EC2 dashboard and select SonarQubeServer and copy its public address and visit <http://<public-address>:9000>

Screenshot of the AWS EC2 Instances page showing two running instances: JenkinsServer (t2.micro) and SonarQubeServer (t2.medium). The SonarQubeServer instance is selected.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
JenkinsServer	i-05f12961a9cf8cf3e	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a	ec2-98-80
SonarQubeServer	i-061f29e11e3fa1a8b	Running	t2.medium	2/2 checks passed	View alarms	us-east-1c	ec2-54-21

i-061f29e11e3fa1a8b (SonarQubeServer)

Details | Status and alarms | Monitoring | Security | Networking | Storage | Tags

Instance summary

- Instance ID: i-061f29e11e3fa1a8b (SonarQubeServer)
- IPv6 address: -
- Hostname type: IP name: ip-172-31-92-157.ec2.internal
- Answer private resource DNS name
- Public IPv4 address copied: 54.210.13.192 | [open address](#)
- Instance state: Running
- Private IP DNS name (IPv4 only): ip-172-31-92-157.ec2.internal
- Instance type: t2.medium

Private IPv4 addresses: 172.31.92.157

Public IPv4 DNS: ec2-54-210-13-192.compute-1.amazonaws.com | [open address](#)

Elastic IP addresses:

Step 7:

You will be redirected to this page on successful installation of SonarQube and visiting the public address url with port 9000. Login the username=admin and password=admin.

Not secure 54.210.13.192:9000/sessions/new?return_to=%2F

sonar

Welcome to SonarQube

Login *

Password *

Go back [Login](#)

SonarQube™ technology is powered by SonarSource SA

LGPL v3 Community Documentation Plugins

Step 8:

Now, set up the initial configurations by setting up new password.
password=Anuprita@4321

Not secure 54.210.13.192:9000/account/reset_password

Update your password

⚠️ This account should not use the default password.

Enter a new password

All fields marked with * are required

Old Password *

Password *

Confirm Password *

Update

Step 9:

Now, click on the Create a local project link and name the project Hello-World and choose use the global setting

Not secure 54.210.13.192:9000/projects/create

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More ? A

How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)?
Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.

Import from Azure DevOps
Setup

Import from Bitbucket Cloud
Setup

Import from Bitbucket Server
Setup

Import from GitHub
Setup

Import from GitLab
Setup

Are you just testing or have an advanced use-case? Create a local project.

[Create a local project](#)

⚠️ **Embedded database should be used for evaluation purposes only**
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

Not secure 54.210.13.192:9000/projects/create?mode=manual

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More

1 of 2

Create a local project

Project display name *

Project key *

Main branch name *

The name of your project's default branch [Learn More](#)

[Cancel](#) [Next](#)

⚠️ Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA [Community Edition v10.7 \(96327\) ACTIVE](#) [LGPL v3](#) [Community](#) [Documentation](#) [Plugins](#) [Web API](#)

Not secure 54.210.13.192:9000/projects/create?mode=manual&ssetncd=true

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More

2 of 2

Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)

Choose the baseline for new code for this project

Use the global setting

Previous version
Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Define a specific setting for this project

Previous version
Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Number of days
Any code that has changed in the last x days is considered new code. If no action is taken on a new issue after x days, this issue will become part of the overall code.
Recommended for projects following continuous delivery.

Reference branch
Choose a branch as the baseline for the new code.

Not secure 54.210.13.192:9000/tutorials?id=Hello-World

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More ? A

Hello World / main

Overview Issues Security Hotspots Measures Code Activity Project Settings Project Information

Analysis Method

Use this page to manage and set-up the way your analyses are performed.

How do you want to analyze your repository?

With Jenkins With GitHub Actions With Bitbucket Pipelines

With GitLab CI With Azure Pipelines Other CI
SonarQube integrates with your workflow no matter which CI tool you're using.

Locally
Use this for testing or advanced use-case. Other modes are recommended to help you set up your CI environment.

⚠️ Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.

SonarQube™ technology is powered by SonarSource SA [Community Edition v10.7 \(96327\) ACTIVE](#) [LGPL v3](#) [Community](#) [Documentation](#) [Plugins](#) [Web API](#)

Step 10:

Click on Administration > Security > Users. And then give a token name and click on the generate token button and copy the token number and save it somewhere. For now, I have saved it in notepad.

Not secure | 54.210.13.192:9000/admin/settings

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More ? A

Administration

Configuration Security Projects System Marketplace

General Users Groups Edit global se Global Permissions Find in Permission Templates

Analysis Scope Authentication DevOps Platform Integrations Email Notification External Analyzers General Housekeeping JaCoCo Languages

Duplications

Cross project duplication detection
DEPRECATED - By default, SonarQube detects duplications at project level. This means that a block duplicated on two different projects won't be reported. Setting this parameter to "true" allows to detect duplicates across projects. Note that activating this property will significantly increase each SonarQube analysis time, and therefore badly impact the performances of report processing as more and more projects are getting involved in this cross project duplication mechanism.
Key: sonar.cpd.cross_project

The screenshot shows the SonarQube Administration interface. A modal window titled "Tokens of Administrator" is open, showing the "Generate Tokens" section. It has fields for "Name" (with placeholder "Enter Token Name") and "Expires in" (set to "30 days"). A "Generate" button is present. Below the form, a message states: "New token "admin-token" has been created. Make sure you copy it now, you won't be able to see it again!" followed by the token value "squ_c395eea8275923706da6ddb29f77d673a9a51668" and a copy icon. A table lists tokens: "admin-token" (User type, Never last used, October 19, 2024 created, November 18, 2024 expiration). A "Revoke" button is next to the token row. A "Close" button is at the bottom right of the modal. At the bottom of the page, a warning about the embedded database is visible: "⚠️ Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine." The footer includes "SonarQube™ technology is powered by SonarSource SA" and "Community Edition v10.7 (96327) ACTIVE" along with links for "LGPL v3", "Community", "Documentation", "Plugins", and "Web API".

GitHub

Step 1:

Make a repository and upload you files in the repository.

The screenshot shows the Visual Studio Code (VS Code) interface. The workspace is named "AdvDevOps_Practical". The Explorer sidebar shows a folder named "ADVDEVOPS_PRACTICAL" containing ".terraform", ".terraform.lock.hcl", "hello.py", "main.tf", and "terraform.tfstate". The "hello.py" file is open in the editor, displaying the following Python code:

```
print("Hello, World!")
```

About
No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases
No releases published
[Create a new release](#)

Packages
No packages published
[Publish your first package](#)

Languages
Python 100.0%

Suggested workflows

Pipeline

Step 1:

Open the git bash for Jenkins and run the following commands in the terminal to install git.

```
sudo yum install git
git --version
```

```
user@DESKTOP-0OK15A: MINGW64 ~/Downloads (master)
$ ssh -i "jenkins.pem" ec2-user@ec2-18-223-40.compute-1.amazonaws.com
Last login: Sat Oct 19 17:56:46 2024 from 49.36.105.96
#
# Amazon Linux 2
# AL2 End of Life is 2025-06-30.
#
# A newer version of Amazon Linux is available!
# Amazon Linux 2023, GA and supported until 2028-03-15.
# https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-42-86 ~]$ sudo yum install git
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package git.x86_64 0:2.40.1-1.amzn2.0.3 will be installed
--> Processing Dependency: git-core = 2.40.1-1.amzn2.0.3 for package: git-2.40.1-1.amzn2.0.3.x86_64
--> Processing Dependency: git-core-doc.noarch 0:2.40.1-1.amzn2.0.3 will be installed
--> Processing Dependency: perl-Git.noarch 0:2.40.1-1.amzn2.0.3 will be installed
--> Processing Dependency: perl(Error) for package: perl-Git-2.40.1-1.amzn2.0.3.noarch
--> Processing Dependency: perl-TermReadkey.x86_64 0:2.30-20.amzn2.0.2 will be installed
--> Running transaction check
-->> Package perl-Error.noarch 1:0.17020-2.amzn2 will be installed
-->> Finished Dependency Resolution

Dependencies Resolved

=====
Package           Arch      Version          Repository        Size
=====
Installing:
git              x86_64   2.40.1-1.amzn2.0.3    amzn2-core       54 k
Installing for dependencies:
git-core          x86_64   2.40.1-1.amzn2.0.3    amzn2-core       10 M
git-core-doc     noarch   2.40.1-1.amzn2.0.3    amzn2-core      3.0 M
perl-Error        noarch   1:0.17020-2.amzn2      amzn2-core      32 k
perl-Git          noarch   2.40.1-1.amzn2.0.3    amzn2-core      42 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
Is this ok [y/d/N]: y
Downloading packages:
(1/6): git-2.40.1-1.amzn2.0.3.x86_64.rpm | 54 kB 00:00:00
(2/6): git-core-doc-2.40.1-1.amzn2.0.3.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.amzn2.0.3.x86_64.rpm | 10 MB 00:00:00
(5/6): perl-Git-2.40.1-1.amzn2.0.3.noarch.rpm | 42 kB 00:00:00
(6/6): perl-TermReadKey-2.30-20.amzn2.0.2.x86_64.rpm | 31 kB 00:00:00
```

Step 2:

Go to Manage Jenkins > System. Scroll down to SonarQube Servers section and name it as SonarQube Server and copy the <http://<public-address-of-sonarqube>:9000>

Also, copy the token as secret here in secret text.

The screenshot shows the Jenkins Manage Jenkins > System configuration page. In the SonarQube servers section, there is a configuration card for a SonarQube installation. The Name is set to "SonarQube Server". The Server URL is set to "http://54.210.13.192:9000". Under Server authentication token, the dropdown is set to "- none -". There is a "+ Add" button for adding more tokens. At the bottom of the card are "Save" and "Apply" buttons.

The screenshot shows the Jenkins Manage Jenkins > System configuration page with a modal dialog titled "Jenkins Credentials Provider: Jenkins". The dialog is titled "Add Credentials". It has fields for Domain (set to "Global credentials (unrestricted)"), Kind (set to "Secret text"), Scope (set to "Global (Jenkins, nodes, items, all child items, etc)"), Secret (containing a redacted value), ID (empty), and Description (empty). At the bottom of the dialog are "Save" and "Apply" buttons.

Not secure 98.80.223.40:8080/manage/configure

Dashboard > Manage Jenkins > System >

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

Environment variables

SonarQube installations

List of SonarQube installations

Name	<input type="text" value="SonarQube"/>	
Server URL	Default is http://localhost:9000 <input type="text" value="http://54.210.13.192:9000"/>	
Server authentication token	SonarQube authentication token. Mandatory when anonymous access is disabled. <input type="password" value="Secret text"/> + Add	
Advanced		

[Add SonarQube](#)

[Save](#) [Apply](#)

Step 3:

Go to Manage Jenkins > Credentials. Copy this id and you will need to paste in the Pipeline Script later.

Not secure | 98.80.223.40:8080/manage/credentials/

Jenkins

Dashboard > Manage Jenkins > Credentials

Credentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	6e0ad648-6931-48d0-a2eb-938a55db6234	Secret text

Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global)

Icon: S M L

REST API Jenkins 2.462.3

Step 4:

Create a new pipeline and name it pipeline1.

Enter an item name
pipeline1

Select an item type

- Freestyle project**
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.
- Pipeline**
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.
- Multi-configuration project**
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.
- Folder**
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- Multibranch Pipeline**
Creates a set of Pipeline projects according to detected branches in one SCM repository.
- Organization Folder**
Creates a set of multibranch project subfolders by scanning for repositories.

OK

Step 5:

Now select on the Git project and paste your GitHub url.

Dashboard > pipeline1 > Configuration

Configure General Enabled

General

Description

Plain text: [Preview](#)

Discard old builds ?

Do not allow concurrent builds

Do not allow the pipeline to resume if the controller restarts

GitHub project

Project url ?
https://github.com/Anuprita2022-26/helloworld_python/

Advanced ▾

Pipeline speed/durability override ?

Save Apply

Step 6:

Now, write the following Pipeline Script.

pipeline {

```
agent any
stages {
```

```
stage('Clone Repository') {
    steps {
        git branch: 'main', url: 'https://github.com/Anuprita2022-26/helloworld_python.git'
    }
}
stage('SonarQube Analysis') {
    environment {
        scannerHome = tool 'SonarQubeScanner' // Ensure SonarQube Scanner is installed
    }
    steps {
        withSonarQubeEnv('SonarQube') { // Name of SonarQube server configured in Jenkins
            withCredentials([string(credentialsId: '6e0ad648-6931-48d0-a2eb-938a55db6234', variable: 'SONAR_TOKEN')]) {
                sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=Hello-World -Dsonar.sources=. -Dsonar.login=$SONAR_TOKEN"
            }
        }
    }
}
post {
    always {
        echo 'Pipeline completed'
    }
}
```

Not secure | 98.80.223.40:8080/job/pipeline1/configure

Dashboard > pipeline1 > Configuration

Configure

Advanced Project Options

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

```

1 pipeline {
2   agent any
3   stages {
4     stage('Clone Repository') {
5       steps {
6         git branch: 'main', url: 'https://github.com/Anuprita2022-26/helloworld_python.git'
7       }
8     }
9     stage('SonarQube Analysis') {
10    environment {
11      scannerHome = tool 'SonarQubeScanner' // Ensure SonarQube Scanner is installed
12    }
13    steps {
14      withSonarQubeEnv('SonarQube') { // Name of SonarQube server configured in Jenkins
15        withCredentials([string(credentialsId: '6e0ad648-6931-48d0-a2eb-938a55db6234', variable: 'SONAR_TOKEN')]) {
16          sh "${scannerHome}/bin/sonar-scanner -Dsonar.projectKey=Hello-World -Dsonar.sources=. -Dsonar.login=$SONAR_TOKEN"
17        }
18      }
19    }
20  }
21}

```

Use Groovy Sandbox ?

Pipeline Syntax

Save Apply

Step 7:

Build run the pipeline. It gives success. Also, check the console.

Jenkins

Search (CTRL+K) Anuprita log out

Dashboard > pipeline1 >

Status

pipeline1

Add description

Changes Build Now Configure Delete Pipeline Full Stage View GitHub Stages Rename Pipeline Syntax

Stage View

Average stage times: (Average full run time: ~28s)

	Clone Repository	SonarQube Analysis	Declarative: Post Actions
#5 Oct 20 00:28 No Changes	344ms	26s	58ms Success
#4 Oct 20 00:15 No Changes	340ms	2s failed	
#3 Oct 20 00:08 No Changes	3s	4s failed	75ms
#2 Oct 19 23:59 No Changes	1s failed	204ms failed	110ms
#1 Oct 19 23:51 No Changes	458ms failed	179ms failed	108ms

Build History trend Filter... #5 Oct 19, 2024, 6:58 PM #4 Oct 19, 2024, 6:45 PM #3 Oct 19, 2024, 6:38 PM

Not secure | 98.80.223.40:8080/job/pipeline1/5/console

Jenkins

Dashboard > pipeline1 > #5

Console Output

Started by user Anuprita
 [Pipeline] Start of Pipeline
 [Pipeline] node
 Running on Jenkins in /var/lib/jenkins/workspace/pipeline1
 [Pipeline] {
 [Pipeline] stage
 [Pipeline] { (Clone Repository)
 [Pipeline] git
 The recommended git tool is: NONE
 No credentials specified
 > git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/pipeline1/.git # timeout=10
 Fetching changes from the remote Git repository
 > git config remote.origin.url https://github.com/Anuprita2022-26/helloworld_python.git # timeout=10
 Fetching upstream changes from https://github.com/Anuprita2022-26/helloworld_python.git
 > git --version # timeout=10
 > git --version # 'git version 2.40.1'
 > git fetch --tags --force --progress -- https://github.com/Anuprita2022-26/helloworld_python.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git rev-parse refs/remotes/origin/main^{commit} # timeout=10
 Checking out Revision 28b08a8b749d48fe74988ca608ac5cde40bd23bd (refs/remotes/origin/main)
 > git config core.sparsecheckout # timeout=10
 > git checkout -f 28b08a8b749d48fe74988ca608ac5cde40bd23bd # timeout=10
 > git branch -a -v --no-abbrev # timeout=10
 > git branch -D main # timeout=10
 > git checkout -b main 28b08a8b749d48fe74988ca608ac5cde40bd23bd # timeout=10
 Commit message: "Add files via upload"
 > git rev-list --no-walk 28b08a8b749d48fe74988ca608ac5cde40bd23bd # timeout=10
 [Pipeline] }
 [Pipeline] // stage

Not secure | 98.80.223.40:8080/job/pipeline1/5/console

Dashboard > pipeline1 > #5

```
18:59:20.467 INFO CPU Executor CPU calculation finished (done) | time=0ms
18:59:20.483 INFO SCM revision ID '28b08a8b749d48fe74988ca608ac5cde40bd23bd'
18:59:20.715 INFO Analysis report generated in 223ms, dir size=221.6 kB
18:59:20.760 INFO Analysis report compressed in 47ms, zip size=22.5 kB
18:59:20.823 INFO Analysis report uploaded in 53ms
18:59:20.825 INFO ANALYSIS SUCCESSFUL, you can find the results at: http://54.210.13.192:9000/dashboard?id=Hello-World
18:59:20.826 INFO Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
18:59:20.826 INFO More about the report processing at http://54.210.13.192:9000/api/ce/task?id=d88de4d6-21ba-4b74-922f-410cda6d30e8
18:59:20.852 INFO Analysis total time: 13.793 s
18:59:20.855 INFO SonarScanner Engine completed successfully
18:59:20.925 INFO EXECUTION SUCCESS
18:59:20.939 INFO Total time: 24.932s
[Pipeline] }
[Pipeline] // withCredentials
[Pipeline] }
[Pipeline] // withSonarQubeEnv
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Declarative: Post Actions)
[Pipeline] echo
Pipeline completed
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

REST API Jenkins 2.462.3

SonarQube Analysis and Results

Step 1:

Visit back to the <http://<public-address-of-sonarqube>:9000>

Now, go to projects section and you can see the analysis of the python project.

Not secure | 54.210.13.192:9000/projects

sonarqube Projects Issues Rules Quality Profiles Quality Gates Administration More Search Create Project

My Favorites All

Filters

Quality Gate

	Passed	Failed
Passed	1	0
Failed	0	1

Security

	≥ 0 info issues	≥ 1 minor issue	≥ 1 major issue	≥ 1 critical issue	≥ 1 blocker issue
A	1	0	0	0	0
B	0	1	0	0	0
C	0	0	1	0	0
D	0	0	0	1	0
E	0	0	0	0	1

Reliability

	≥ 0 info issues	≥ 1 minor issue	≥ 1 major issue	≥ 1 critical issue
A	1	0	0	0
B	0	1	0	0
C	0	0	1	0
D	0	0	0	1

Search for projects... Perspective Overall Status Sort by Name ↴ 1 project(s) Home

☆ Hello World PUBLIC Passed

Last analysis: 4 minutes ago - 1 Lines of Code - Python

Security Reliability Maintainability Hotspots Reviewed Coverage Duplications

1 of 1 shown

⚠ Embedded database should be used for evaluation purposes only
The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for migrating your data out of it into a different database engine.