

Jenkins Installation and Management on AWS

Table of Contents

1. Introduction
 2. Prerequisites
 3. Installing Jenkins on AWS
 4. Configuring Jenkins
 5. Managing Users & Permissions
 6. Security Best Practices
 7. Troubleshooting
 8. Conclusion
-

Introduction

[Jenkins](#) is an open-source automation server used for **Continuous Integration and Continuous Deployment (CI/CD)**. This guide provides detailed steps for installing, setting up, and managing Jenkins on an **AWS EC2 instance (Amazon Linux 2)**.

Prerequisites

Before installing Jenkins, ensure you have the following:

- AWS Account with access to **EC2**
 - **Amazon Linux 2** instance with at least **t2.micro**
 - **Security Group** allowing inbound traffic on **port 8080** (Jenkins UI)
 - SSH access to the EC2 instance
-

Installing Jenkins on AWS

1. Launch an EC2 Instance

1. Go to **AWS EC2 Dashboard** → Click **Launch Instance**
2. Select **Amazon Linux 2 AMI**
3. Choose **t2.micro** (Free Tier) or a higher instance type
4. Configure **Security Group**:
 - Allow **port 8080** (for Jenkins)
 - Allow **port 22** (for SSH)
5. Launch the instance and connect via SSH:
6. ssh -i your-key.pem ec2-user@your-instance-ip

The screenshot shows the AWS EC2 Instances page. In the left sidebar, under 'Instances', there are sections for 'Instances', 'Images', 'Elastic Block Store', 'Network & Security', and 'CloudShell'. The main content area displays a table titled 'Instances (1/1)'. A single row is selected for an instance named 'jenkins' with the ID 'i-0a71cfdcc4d99f5d3'. The instance is shown as 'Running' with the type 't2.micro'. It has a public IPv4 address '3.80.153.252' and a private IPv4 address '172.31.80.201'. The public DNS name is 'ec2-3-80-153-252.compute-1.amazonaws.com'. The instance was last updated less than a minute ago. There are buttons for 'Connect', 'Actions', and 'Launch instances' at the top right of the table.

2. Install Jenkins

1. Update the system:

- sudo yum update -y

2. Add the Jenkins repository and install Jenkins :

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

3. Import a key file from Jenkins-CI to enable installation from the package:

- sudo rpm --import <https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.keys>
- yum install upgrade

4. Install Java (Amazon Linux 2023):

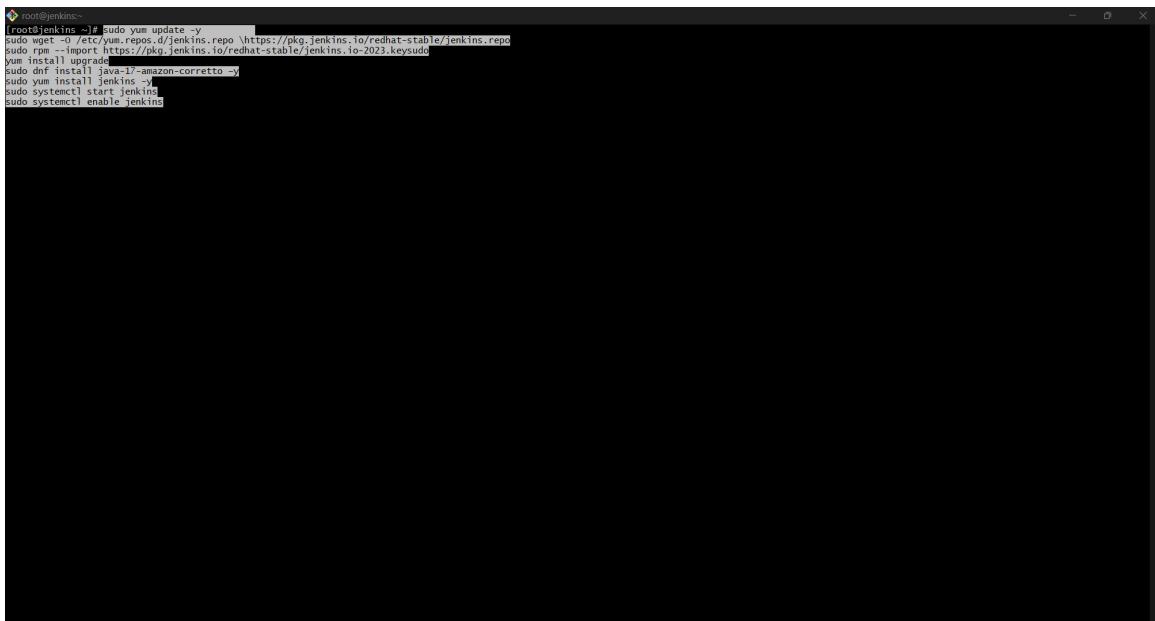
- sudo dnf install java-17-amazon-corretto -y

5. Install Jenkins:

- sudo yum install jenkins -y

6. Start and enable Jenkins :

- sudo systemctl start jenkins
- sudo systemctl enable jenkins



A terminal window titled 'root@jenkins:' showing the command history of the Jenkins installation process. The commands listed are:

```
root@jenkins: ~# sudo yum update -y
[sudo] password for root: 
[sudo] password for root: 
root@jenkins: ~# sudo rpm -Uvh https://pkg.jenkins.io/redhat-stable/jenkins.repo
[sudo] password for root: 
root@jenkins: ~# curl -fsSL https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key | sudo tee /etc/pki/rpm-gpg/RPM-GPG-KEY-jenkins.io
[sudo] password for root: 
root@jenkins: ~# sudo yum install jenkins
[sudo] password for root: 
root@jenkins: ~# sudo systemctl start jenkins
[sudo] password for root: 
root@jenkins: ~# sudo systemctl enable jenkins
```

7. You can check the status of the Jenkins service using the command:

- sudo systemctl status Jenkins

```
root@jenkins:~#
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.keysudo
sudo dnf install java-17-amazon-corretto -y
sudo yum install jenkins -y
sudo curl -L https://jenkins.io/war/146.75.34.133_2a044e4278a:645
sudo systemctl enable jenkins
Last metadata expiration check: 0:07:30 ago on Tue Feb 11 11:59:29 2023.
Dependencies resolved.
Nothing to do.
Complete!
- 2023-02-11 12:07:05 - https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving Dependencies
Connecting to 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'

/etc/yum.repos.d/jenkins.repo    100%[=====]   85 --.-KB/s    in 0s

2023-02-11 12:07:05 (8.42 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

curl: (22) The requested URL returned error: 404
curl: (22) Could not resolve host: jenkins.io-2023.keysudo; import read failed(2).
Last metadata expiration check: 0:07:30 ago on Tue Feb 11 11:59:29 2023.
No match for argument: upgrade
Error: Unable to find a match: upgrade
Error: Nothing to do.
Feb 11 12:07:05 jenkins java[26401]: Last metadata expiration check: 0:07:37 ago on Tue Feb 11 11:59:29 2023.
Package java-17-amazon-corretto-11.0.14+7-1.amzn2023.1.x86_64 is already installed.
No match for argument: -y
Error: Nothing to do.
Feb 11 12:07:05 jenkins java[26401]: Last metadata expiration check: 0:07:38 ago on Tue Feb 11 11:59:29 2023.
Package jenkins-2.492.1-1.1.noarch is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[root@jenkins ~]# systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: disabled)
     Active: active (running) since Tue 2023-02-11 12:04:35 UTC; 2min 48s ago
       Main PID: 26401 (java)
          Tasks: 1 (since Feb 11 12:04)
            Memory: 343.3M
              CPU: 14.838s
             CGroup: /system.slice/jenkins.service
                     └─26401 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

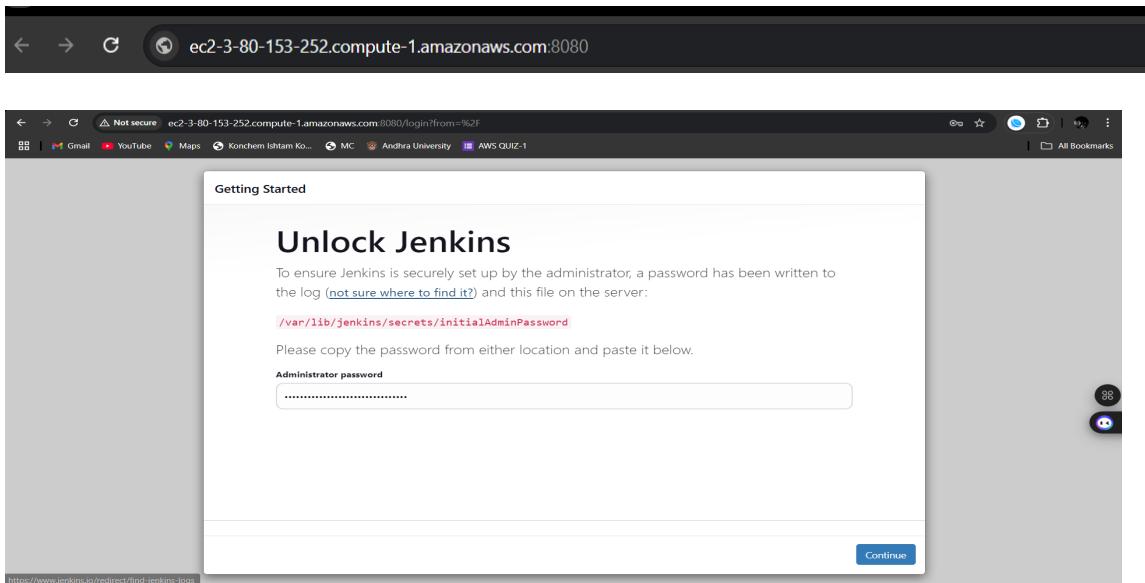
Feb 11 12:04:28 jenkins jenkins[26401]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Feb 11 12:04:28 jenkins jenkins[26401]: ****
Feb 11 12:04:28 jenkins jenkins[26401]: ****
Feb 11 12:04:28 jenkins jenkins[26401]: ****
Feb 11 12:04:35 jenkins jenkins[26401]: ****
Feb 11 12:04:35 jenkins jenkins[26401]: 2023-02-11 12:04:35.096+0000 [id=30] INFO  jenkins.InitReactorRunner$1@onAttained: Completed initialization
Feb 11 12:04:35 jenkins jenkins[26401]: 2023-02-11 12:04:35.138+0000 [id=30] INFO  jenkins.HudsonLifecycleListener@onReady: Jenkins is fully up and running
Feb 11 12:04:35 jenkins jenkins[26401]: 2023-02-11 12:04:35.602+0000 [id=46] INFO  h.m.downloadService$DownloadableLoad@: obtained the updated data file for hudson.tasks.MavenInstaller
Feb 11 12:04:35 jenkins jenkins[26401]: 2023-02-11 12:04:35.603+0000 [id=46] INFO  hudson.util.Retrigger#start: Performed the action check updates server successfully at the attempt #1
Feb 11 12:04:40 jenkins jenkins[26401]: 2023-02-11 12:04:40.241+0000 [id=68] WARNING h.a.DiskSpaceMonitorDescriptor#markNodeOffline@: Making Built-in Node offline temporarily due to the low disk space available
[lines 1-20/20 (END)]
```

8. Retrieve the initial admin password

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

```
[root@jenkins:~#
[root@jenkins ~]# sudo cat /var/lib/jenkins/secrets/initialAdminPassword
6ef9788d46214a2390b850a8c8fcdbbee
[root@jenkins ~]# ...
```

9. Access Jenkins UI Open http://your-ec2-instance-ip:8080 in your browser.



Configuring Jenkins

1. System Configuration

Go to **Manage Jenkins > Configure System** to modify:

- Build execution settings
- Email notifications
- Environment variables

2. Global Tool Configuration

Go to **Manage Jenkins > Global Tool Configuration** to configure:

- **JDK**
- **Git**
- **Maven**

3. Managing Plugins

Go to **Manage Jenkins > Manage Plugins** to install essential plugins:

- **Pipeline Plugin**
- **Git Plugin**
- **Role-Based Access Control Plugin**

Managing Users & Permissions

1. Creating Users

1. Go to **Manage Jenkins > Manage Users**
2. Click **Create User** and define:
 - Username
 - Password
 - Email

Jenkins

Dashboard > Manage Jenkins > Jenkins' own user database > Create User

Create User

Username

Password

Confirm password

Full name

E-mail address

Create User

→ Create user and try to login with above url after creating user

Not secure ec2-3-80-153-252.compute-1.amazonaws.com:8080/securityRealm/

Gmail YouTube Maps Konchem Ishtam Ko... MC Andhra University AWS QUIZ-1

Jenkins

PadmakarPilladi log out

Dashboard > Jenkins' own user database

Users 2

+ Create User

These users can log into Jenkins. This is a sub set of [this list](#), which also contains auto-created users who really just made some commits on some projects and have no direct Jenkins access.

User ID	Name	Action
paddu	Paddu	
Padmakar	PadmakarPilladi	

Jenkins 2.492.1

The screenshot shows the Jenkins User Management interface. At the top, there's a navigation bar with links like 'Import favorites', 'Remote Desktop Re...', 'Home | Mynauki', 'LinkedIn', 'mc', 'Git assignment quiz', 'Linux Quiz', 'Shell Scripting Quiz', and 'yaytext - Search'. Below the navigation is the Jenkins logo and the word 'Jenkins'. A search bar and a user dropdown are also present. The main content area is titled 'Users 2' and contains a table with two rows:

User ID	Name	Action
paddu	Paddu	
Padmakar	PadmakarPilladi	

A note at the bottom says: "These users can log into Jenkins. This is a sub set of [this list](#), which also contains auto-created users who really just made some commits on some projects and have no direct Jenkins access." At the bottom right, it says "Jenkins 2.492.1".

2. Role-Based Access Control (RBAC)

1. Install the Role-Based Authorization Strategy Plugin

The screenshot shows the Jenkins Plugins management page. The left sidebar has options: 'Updates', 'Available plugins' (which is selected), 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area has a search bar with 'role' typed in. A table lists available plugins:

Install	Name	Released	Action
<input checked="" type="checkbox"/>	Role-based Authorization Strategy 756.v978cb_392eb_d3	14 days ago	
<input type="checkbox"/>	AWS Credentials 243.v41c19a_fb_5dcf	14 days ago	

At the bottom right, it says "REST API Jenkins 2.492.1".

2. Go to Manage Jenkins > Configure Global Security

3. Select Role-Based Strategy

The screenshot shows the Jenkins Security configuration page. At the top, there are links to various external services like Gmail, YouTube, Maps, Konchem Ishtam Ko..., MC, Andhra University, and AWS QUIZ-1. On the right, there's a 'All Bookmarks' link. Below the header, the breadcrumb navigation shows 'Dashboard > Manage Jenkins > Security'. The main section is titled 'Security' under 'Authentication'. It includes options for 'Disable "Keep me signed in"' (unchecked), 'Security Realm' set to 'Jenkins' own user database', and 'Allow users to sign up' (unchecked). Under 'Authorization', 'Role-Based Strategy' is selected. In the 'Markup Formatter' section, 'Plain text' is chosen. At the bottom are 'Save' and 'Apply' buttons.

4. Define roles and permissions based on user needs

1. After adding Role-Based Strategy, in securities we get manage and assign roles.
2. There we can give permissions according to roles and assign users to respective roles.

The screenshot shows the Jenkins 'Manage Roles' page. The header includes links to 'Dashboard', 'Manage Jenkins', 'Manage and Assign Roles', and 'Manage Roles'. The main area is titled 'Manage Roles' and contains two tables: 'Global roles' and 'Item roles'.
Global roles: A table where columns represent Jenkins features (View, Threading, HealthCheck, Bin, Job, Run, SCM, Metrics) and rows represent roles (Overall, Credentials, Agent, Job, Run, View, SCM, Metrics). The 'admin' role has checkboxes checked for most features.
Item roles: A similar table for item-level roles, showing roles for 'Role Pattern', 'Template', 'Credentials', 'Job', 'Run', 'View', 'SCM', and 'Metrics'. The 'admin' role has checkboxes checked for most features.
At the bottom, there are 'Role to add' input fields and 'Save' and 'Apply' buttons.

Security Best Practices

- **Enable Authentication** (Avoid anonymous access)
 - **Use Role-Based Access Control (RBAC)**
 - **Secure Jenkins with HTTPS**
 - **Limit Public Access** (Restrict access via AWS Security Groups)
 - **Use API Tokens instead of passwords**
-

Troubleshooting

Jenkins Not Starting?

Check logs using:

```
sudo journalctl -u jenkins -xe
```

Port 8080 Not Accessible?

Ensure Security Group allows **inbound traffic on port 8080**.

Permission Issues?

Check **Manage Jenkins > Configure Global Security** and update user roles.

Conclusion

This guide covered everything from **installing Jenkins on AWS** to **managing users, security, and troubleshooting**. You can now use Jenkins effectively in your CI/CD pipeline!
