# Practice for Lesson 3: Assign Roles to User and Build Master-slave configuration in Jenkins Instance

# Practices for Lesson 3

## Overview

In these practices, you will assign Role based authorization plugins to users created in Jenkins instance. Further, you will learn how to build a master-slave configuration on a Jenkins instances and followed by Backing up the configuration files of Jenkins in the AWS EC2 instance.

Practice for Lesson 3: Assign Roles to User and Build Master-slave configuration in Jenkins Instance

# Practice 3-1: Assign Role Based Authorization Plugin to New Users
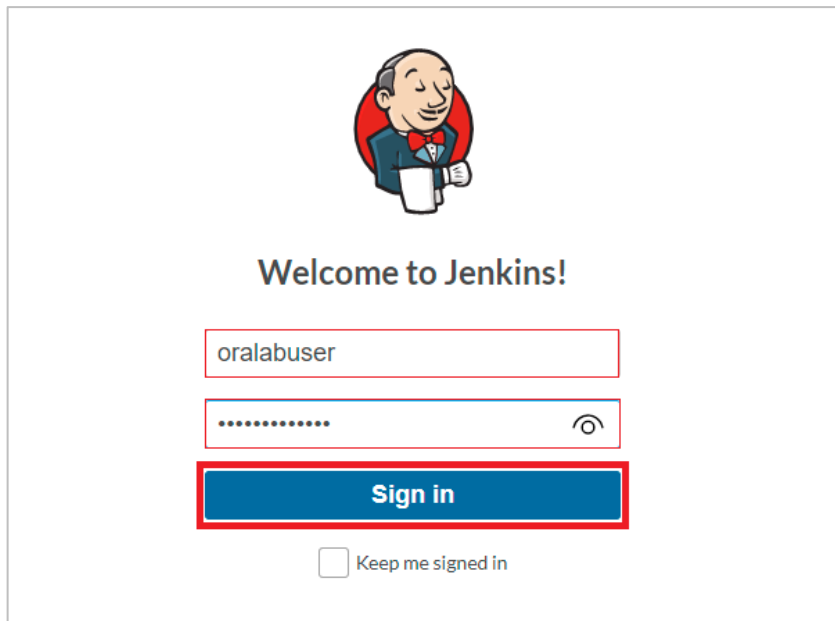
## Overview

In this practice, you will learn to assign Role based authorization plugins to new users created in Jenkins Instance.

## Assumptions

You should have completed the Practice of Lesson 2.

## Tasks

1.  Sign in to **Jenkins Instance Dashboard**.

    a.  In a browser on your local machine, enter the **Public IP** address of the EC2 instance followed by the IP address to sign in to the Jenkins Dashboard (for example, **34.208.77.168:8080**).

    

    b.  Enter the user name and password provided.

    c.  You will have access to the Jenkins Dashboard.

2.  Install Role-based plugin on Jenkins AWS Instance.

    a.  In the Jenkins Instance Dashboard, navigate to **Main menu** and select **Manage Jenkins** as shown below.

---

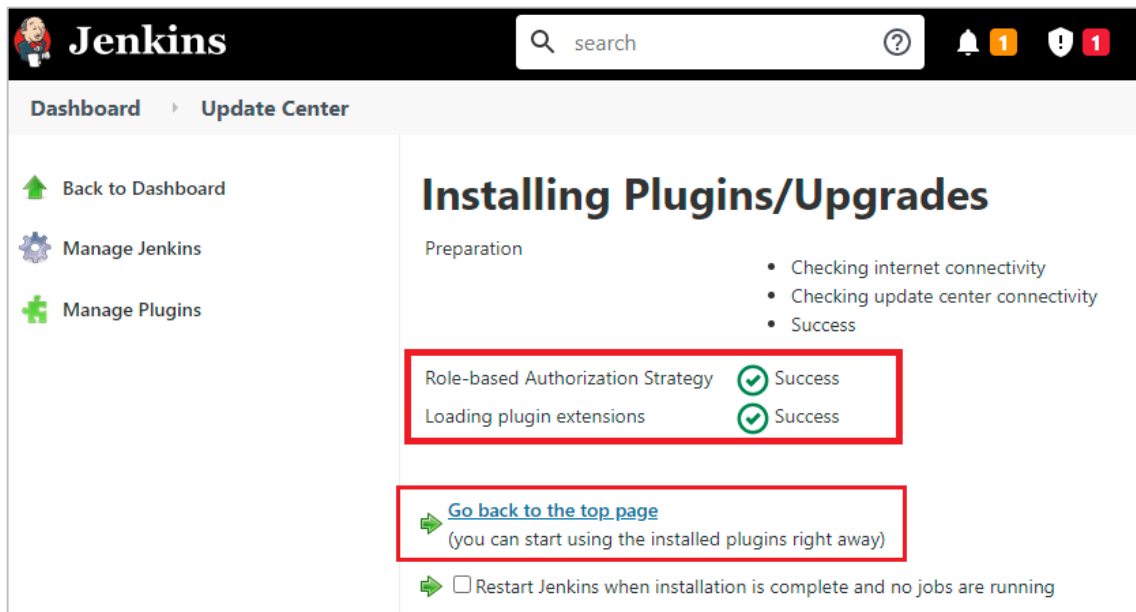b. Under **System Configuration** select **Manage Plugins** as shown below.



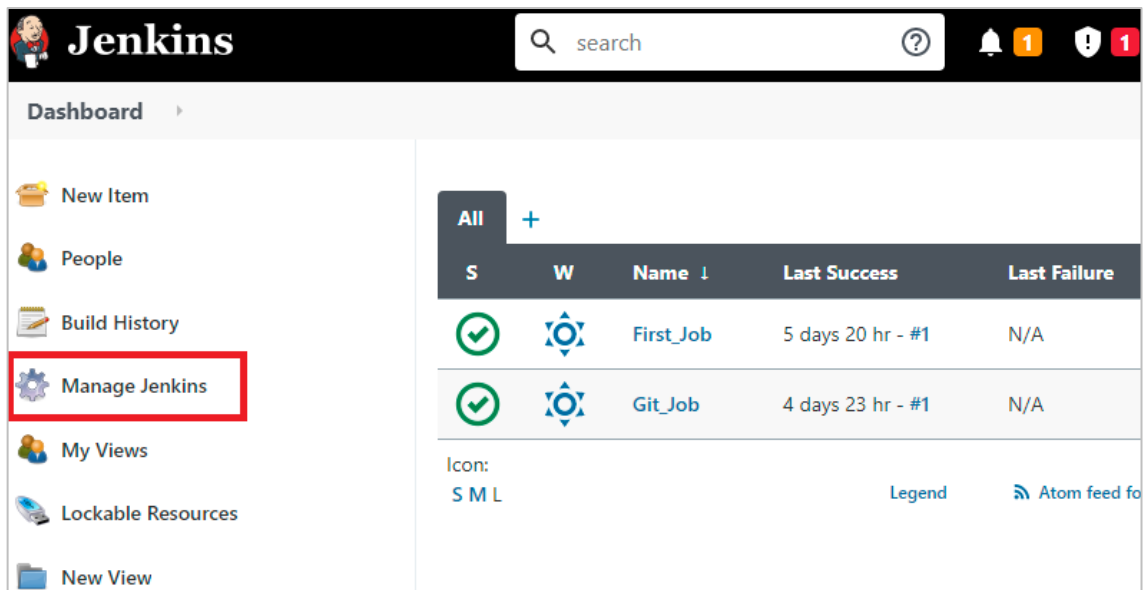c. Select **Available** and type Role-based in the search bar (press Enter).
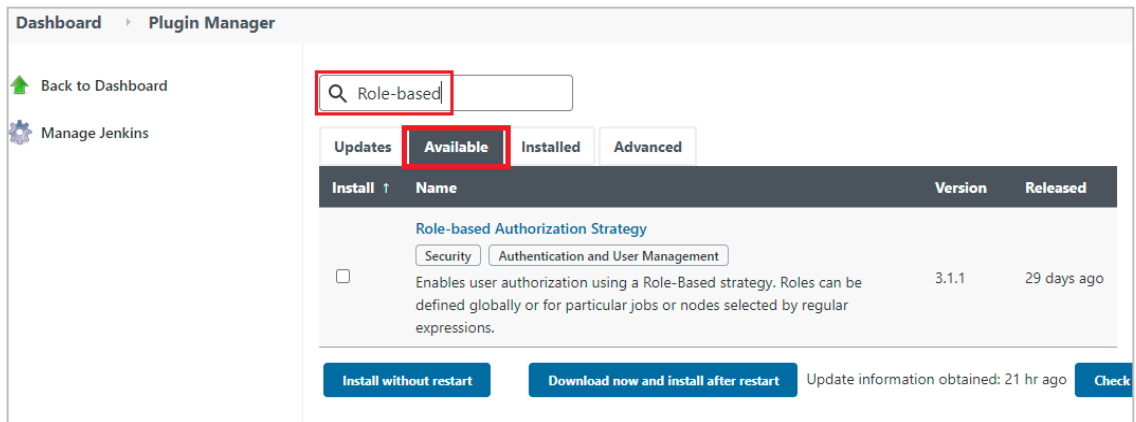
d. The **Role-based Authorization Strategy** plugin appears, select the check box and click **Install without restart** as shown below.
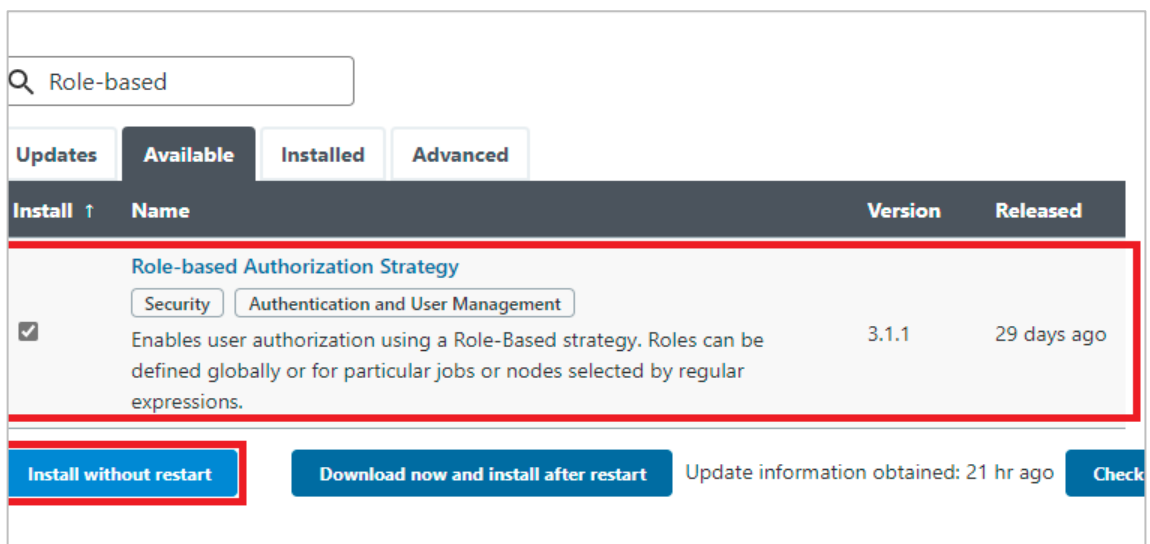


e. The installation process will proceed and Success message will be displayed as shown below.

3. Verify and enable the Role-based plugin in the Jenkins instance.
   a. Navigate to **Main menu** and select **Manage Jenkins** as shown below.



   b. Select **Configure Global Security** under **Security** as shown below.

---

c. In **Configure Global Security** page scroll down to **Authorization** and select radio button of **Role-Based Strategy**, and click **Save.**

4. Create Developer Role in Jenkins instances.

   a. Navigate to **Manage Jenkins** and select **Manage and Assign Roles** under **Security** as shown below.
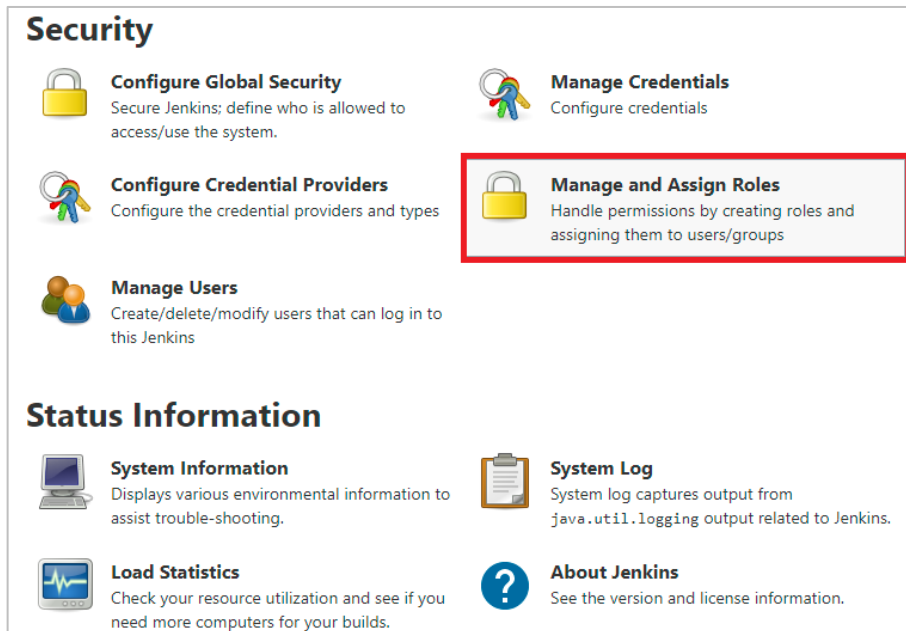
   

   b. In **Manage and Assign Roles** select **Manage Roles** to create a new role as shown below.
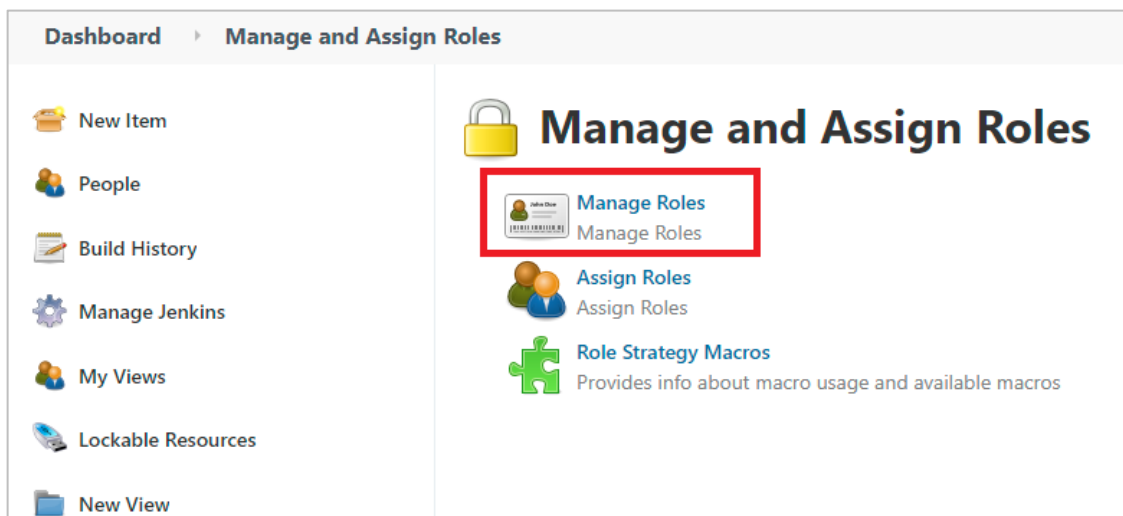
   

   c. In Manage Roles page, under **Global roles** click **Add** and provide the Name for the role in **Role to add** as **Developer**.

Practice for Lesson 3: Assign Roles to User and Build Master-slave configuration in Jenkins Instance

d. As shown below, select the check boxes for the Developer role.



e. Select the check boxes by scrolling to right of the page as shown below.



f. Scroll down and select **Save** to save the new role.

5. Create a new user on the Jenkins instance.

   a. Navigate to **Manage Jenkins** and select **Manage Users** under **Security** as shown below.



   b. Select **Create User** and provide the following details as shown below.

Practice for Lesson 3: Assign Roles to User and Build Master-slave configuration in Jenkins Instance

c. Once the details are provided, click **Create User** as shown below.



d. The new user is created successfully as shown below.
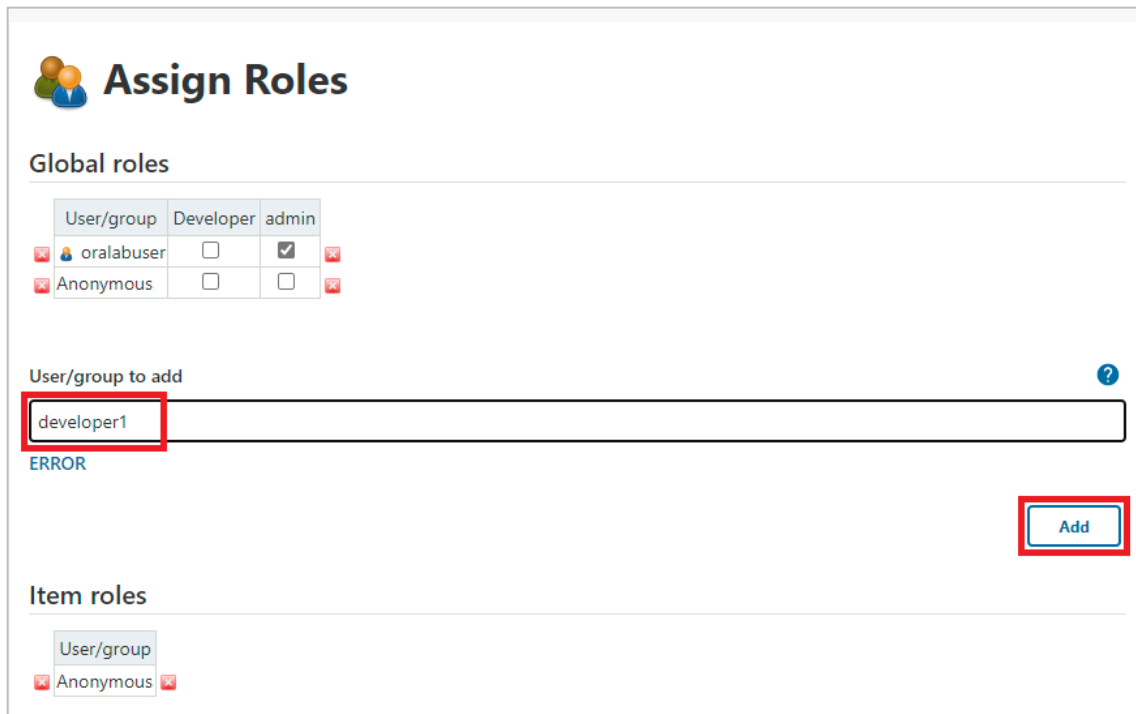
6. Set role for the new developer user created in Jenkins instance.

  a. Navigate to **Manage Jenkins** and select **Manage and Assign Roles** under **Security** as shown below.
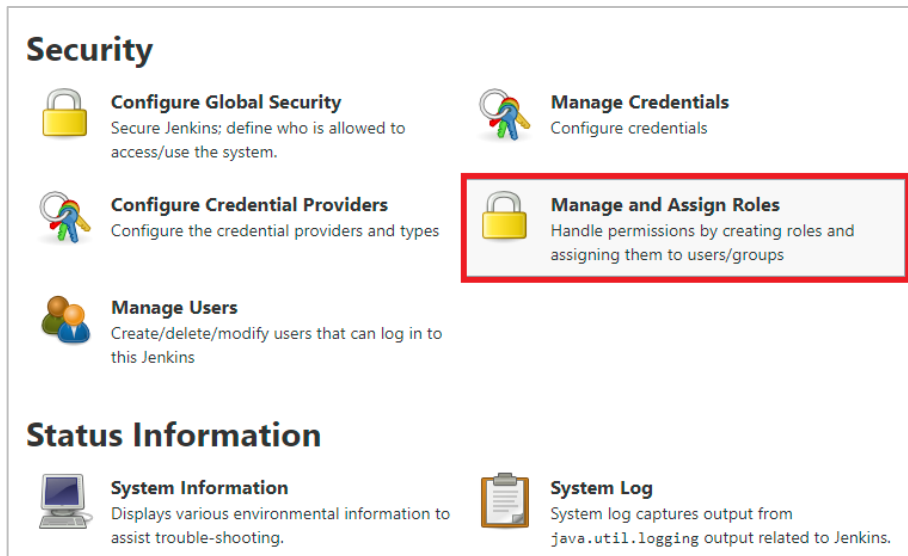


  b. In **Manage and Assign Roles** page, select **Assign Roles** as shown below.
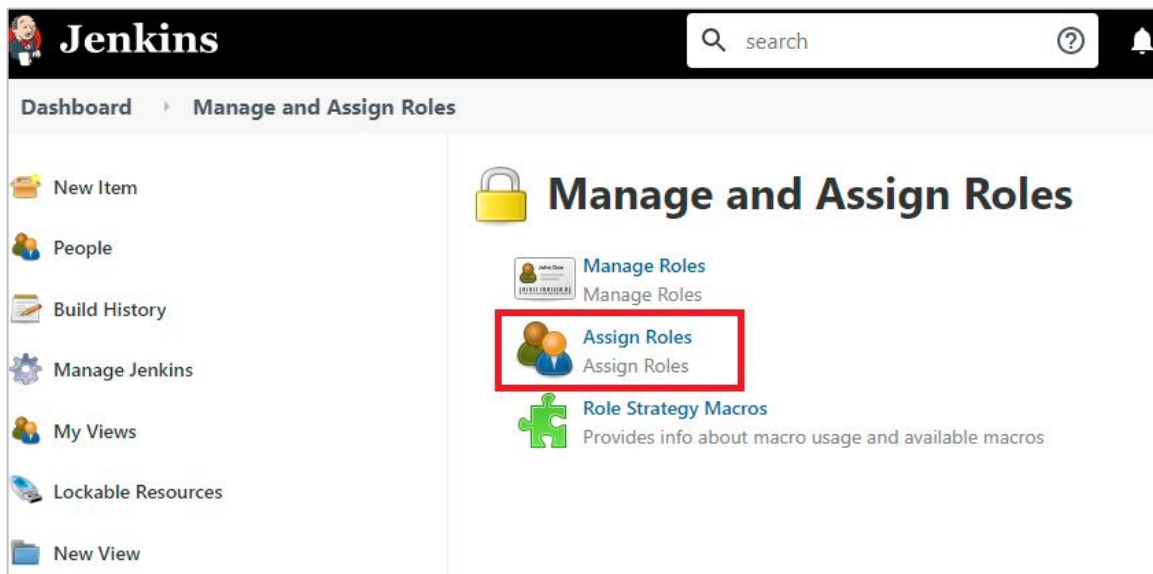


  c. Type the user name in the **User/group to add** and click **Add** as shown below.

---

d. The new user is being listed under **Global roles** successfully. Select the **Developer** check box to assign the **Developer** role to the new user.



e. Scroll down and click **Save** to assign role to the new user created in the Jenkins.

---

**Node roles**

User/group
❌ Anonymous ❌

User/group to add ❓

ERROR

Add

**Save** Apply

7. Verify the permissions of the new developer user with the assigned role.

   a. Logout from the current admin user and login as developer user with the new credentials as shown below.



**Welcome to Jenkins!**

developer1

•••••••

**Sign in**

   b. On successful Sign in, the Jenkins Dashboard is shown as below. The developer1 user will not able to create a new job or trigger an existing Job as developer1 user does not have the required privileges, the required role is not added to this user.

8. Keep the Jenkins Dashboard and the AWS Management Console open for the next practice.

---