

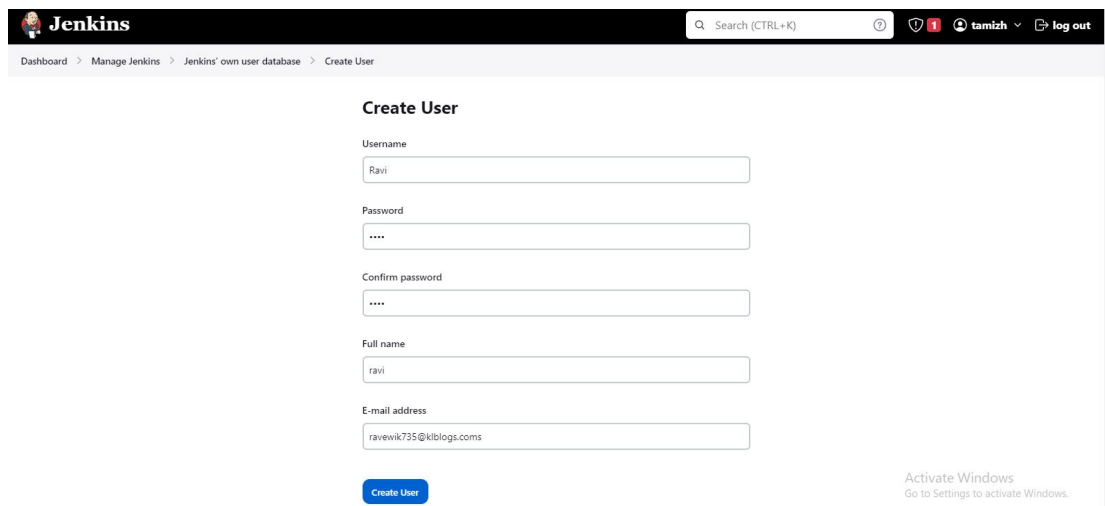
How To Manage Users and Roles in Jenkins



Jenkins Role-Based Access Control

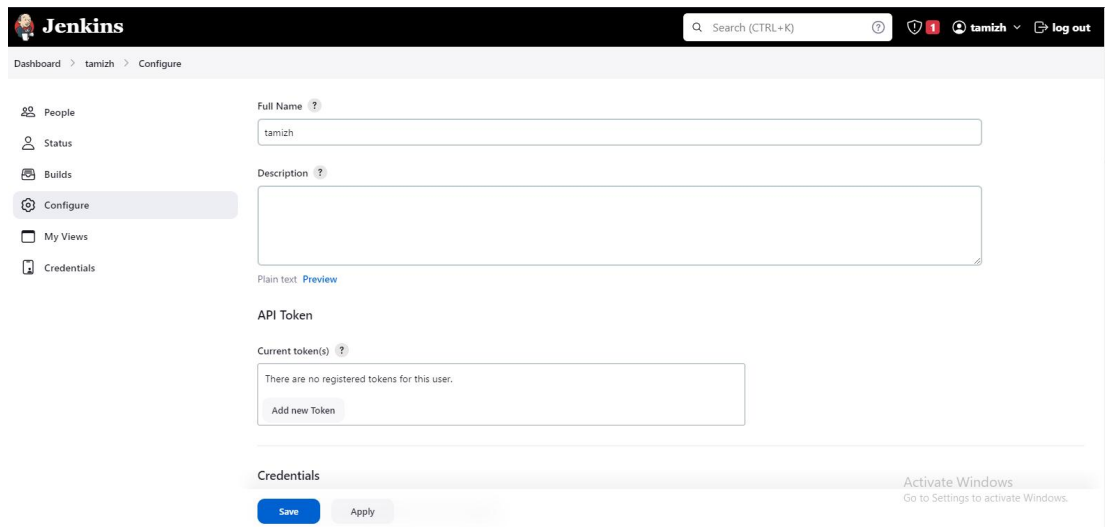
By default, when you create a user in Jenkins, it can access almost everything. In this guide, we will cover how you can create fine-grained roles for proper access control to Jenkins Server. To manage users and roles, you need to have the necessary administrative privileges. Here are the steps to manage users and roles in Jenkins:

1. Access Jenkins: open your web browser and navigate to the Jenkins login page <http://18.209.29.46:8080>
2. Log in: enter your username and password to log in to Jenkins with administrative privileges.
3. Create new user: navigate to **Jenkins Dashboard** → **Manage Jenkins** → **Users** → **Create User**. Enter the user name, password, full name and email-address and then click **Create User** button.

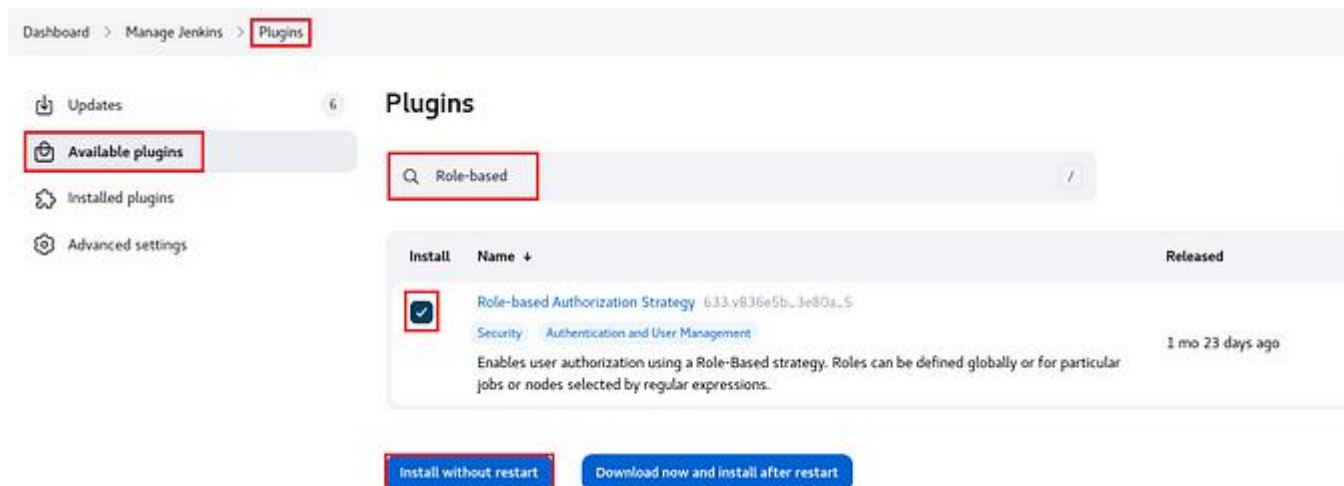


The screenshot shows the Jenkins web interface. At the top is a black header with the Jenkins logo, a search bar, and user information (tamizh) with a log out button. Below the header is a breadcrumb trail: Dashboard > Manage Jenkins > Jenkins' own user database > Create User. The main content area is titled 'Create User' and contains several input fields: Username (filled with 'Ravi'), Password (masked with dots), Confirm password (masked with dots), Full name (filled with 'Ravi'), and E-mail address (filled with 'ravevik735@kiblogs.com'). A blue 'Create User' button is at the bottom left of the form. On the bottom right, there is a 'Activate Windows' watermark.

4. Configure users: you may see the user's name in the upper right corner of Jenkins whenever you log in. Click **Configure** on the user's drop-down menu. On the configuration page, you can update the description, add new API tokens, password, terminal sessions and define time zones.



5. Download and install **Role-based Authorization Strategy** plugin: goto **Manage Jenkins** → **Plugins** → **Available plugins** and search for **Role** in the search box and in the dropdown list, select **Role-based Authentication Strategy** plugin, proceed by clicking on **Install without restart**



6. change authorization to Role-based strategy: After successful installation of plugin, Go to **Manage Jenkins** → **Security** and select the **Role-Based**

Strategy under the **Authorization** section, click on **Apply** and **Save**

Dashboard > Manage Jenkins > Security

Security

Authentication

☐ Disable remember me

Security Realm

Jenkins' own user database

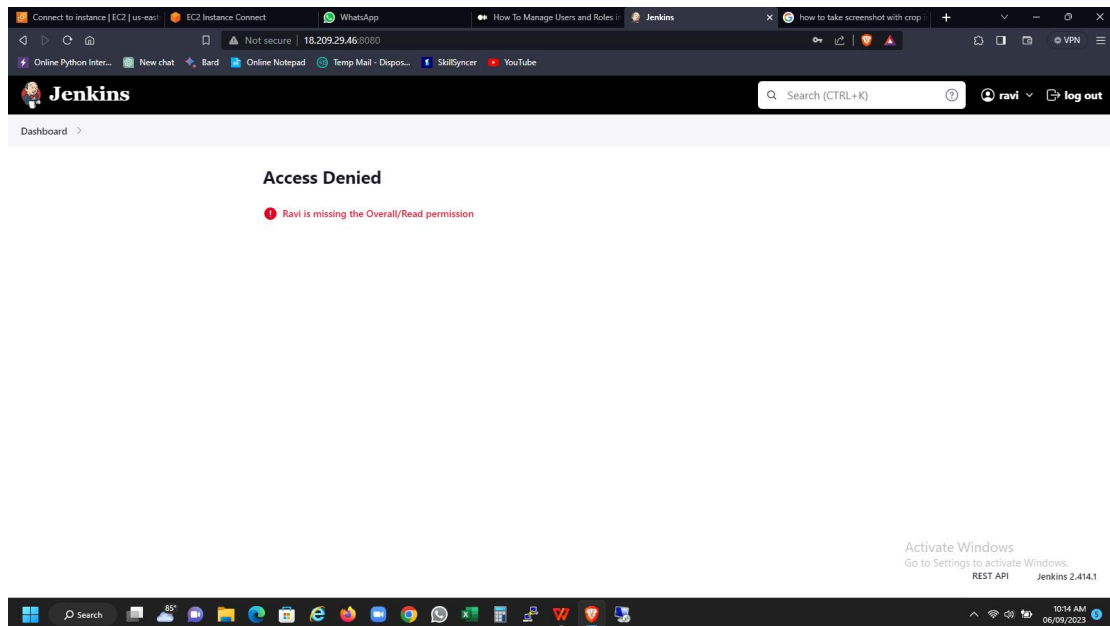
☐ Allow users to sign up ?

Authorization

Role-Based Strategy

Save Apply

NOTE: 'Logged-in users can do anything' option would already be selected before selecting Role-Based Strategy.



7. Creating roles and granting privileges to role: go to **Manage Jenkins** → **Manage and Assign Roles**

There are three options:

1. Manage Roles
2. Assign Roles and
3. Role Strategy Macros

Click on the manage role, you can see that there is only one role admin, but three sessions:

1. Global roles
2. Item roles
3. Agent roles

Global roles → provide authorization and access to the global level. Let's create **employee** role and assign access like read, view, etc.

The screenshot shows the Jenkins 'Manage Roles' interface. The 'Global roles' section is active, displaying a table of permissions for the 'admin' and 'employee' roles. The 'admin' role has all permissions checked, while the 'employee' role has only 'Read', 'View', and 'Tag' permissions checked. The 'Item roles' section is currently empty.

Role	Overall	Credentials	Agent	Build	Configuration	Discover	Read	Workspace	Workflow	Update	Configure	Tag
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
employee	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Role to add: employee

Add

Item roles

Save Apply

Item roles → provides roles specific to a project. For a particular project, you can assign permissions like build, create, delete, configure. Let's create **developer** role and add pattern like **dev.*** ie. project name starting from dev.

Item roles

Role	Pattern	Credentials				Job							Run		View			SCM					
		Create	Delete	ManageDomains	Update	View	Build	Cancel	Configure	Create	Delete	Discover	Move	Read	Workspace	Delete	Replay	Update	Configure	Create	Delete	Read	Tag
developer	/ "dev.*"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
qa	/ "test.*"	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Role to add

qa

Pattern ?

test.*

Add

Save Apply

Item roles

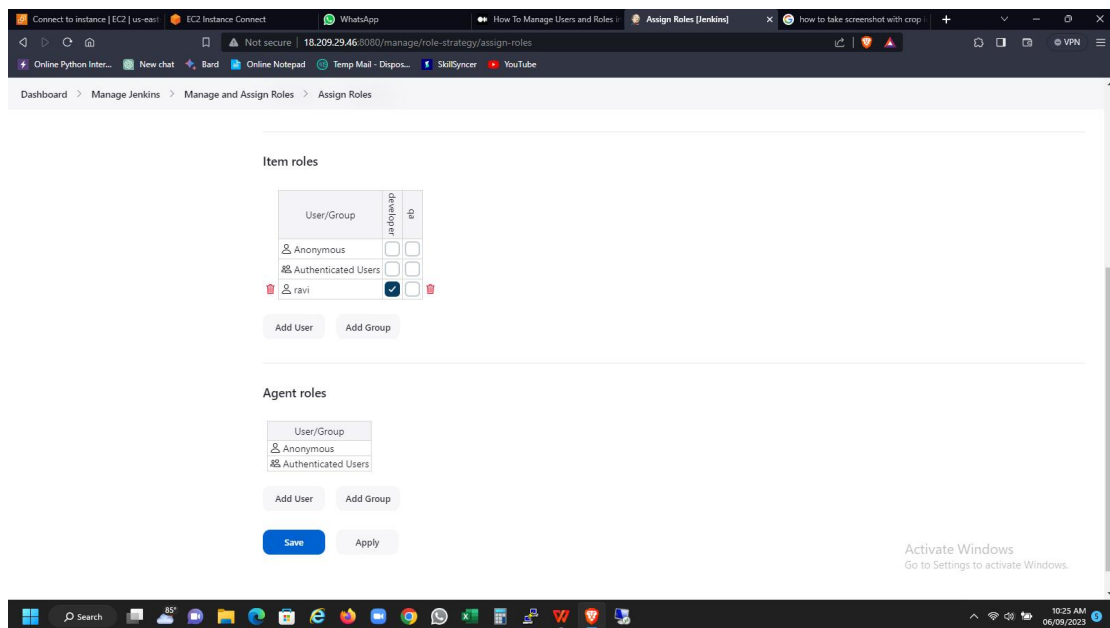
8. Assigning role to users: go to **Manage Jenkins → Manage and Assign Roles → Assign Roles**

Global roles → add Tamizh and Ravi user to the global roles and assign the **employee** role to both users.

Assign Roles

Global roles

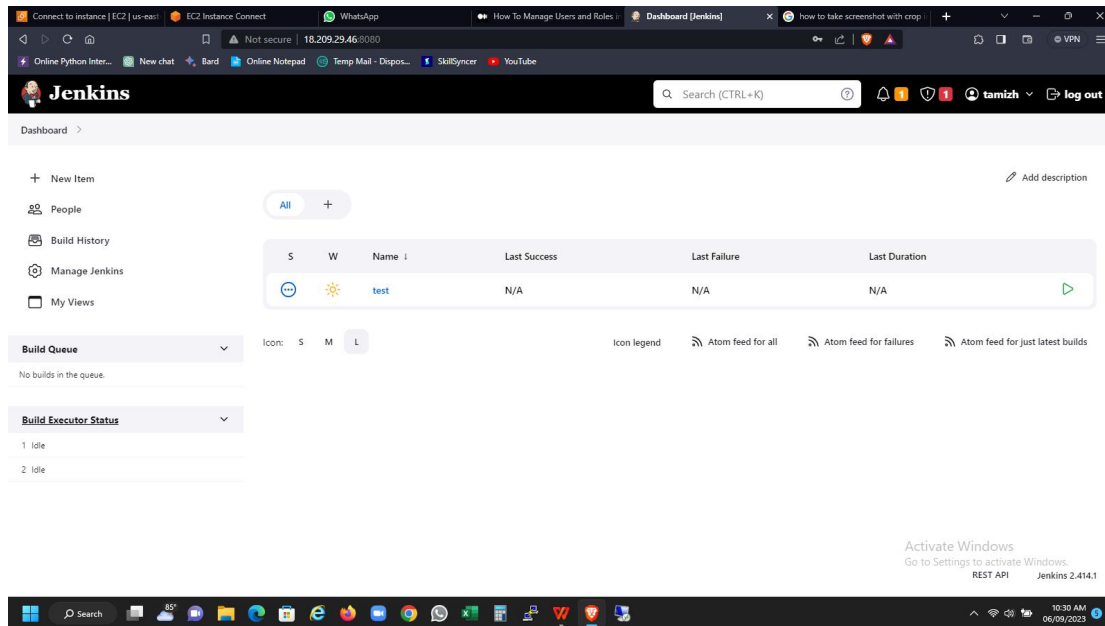
User/Group	admin	employee
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>
Authenticated Users	<input type="checkbox"/>	<input type="checkbox"/>
tamizh	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ravi	<input type="checkbox"/>	<input checked="" type="checkbox"/>



9. Create different projects to test roles: Let's create 2 projects and check the access according to the users. Go to **Jenkins**

Dashboard → New Item → Create freestyle project called devproj1 and testproj1.

Login as user with assigned role. Only projects granted should be visible.



For any new user created without being assigned a role, access denied message should be shown.

