

The Jenkins Dashboard

The Jenkins dashboard is the central point of all operations in our pipelines and projects. Any and all operations are coordinated from this point.

From the dashboard, we are able to achieve operations not limited to the following:

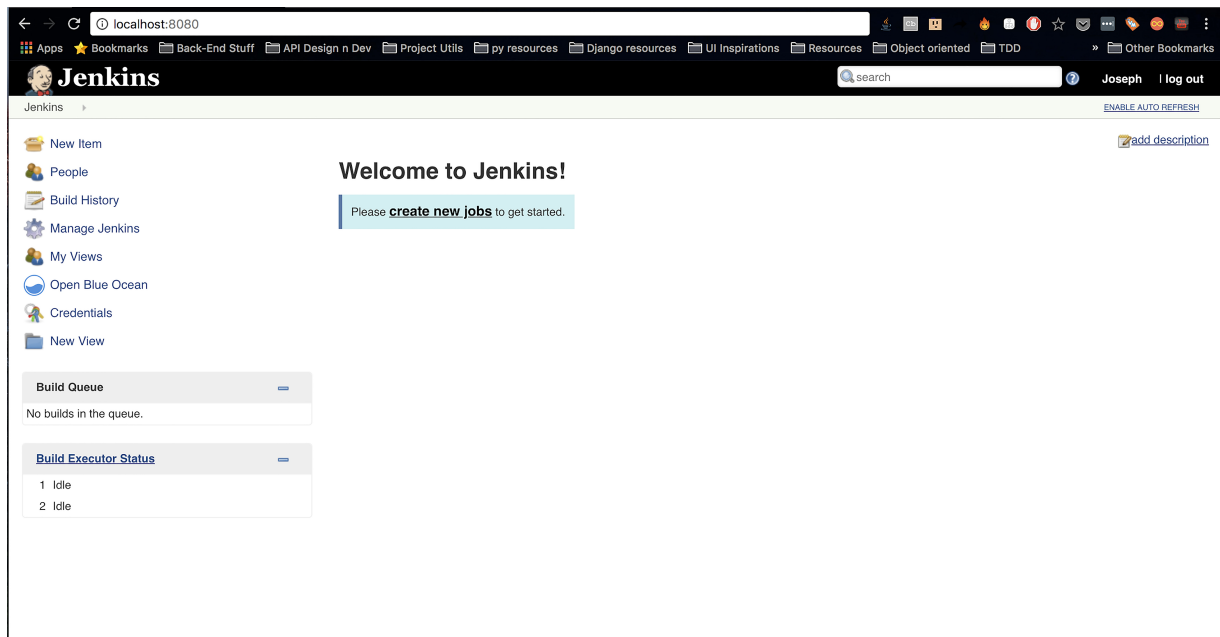
- Project and pipeline management
- Access control
- Data and resource management

Once the Jenkins setup is complete, click on **Start using Jenkins** on the landing screen, and let's get started.



*If the page still shows Jenkins is almost ready, click on **Restart**. If the page doesn't restart or update after a minute, reload your web page manually.*

If required to log in, use the credentials we just created on the UI, and you should be good to go. Take a look at this screenshot:

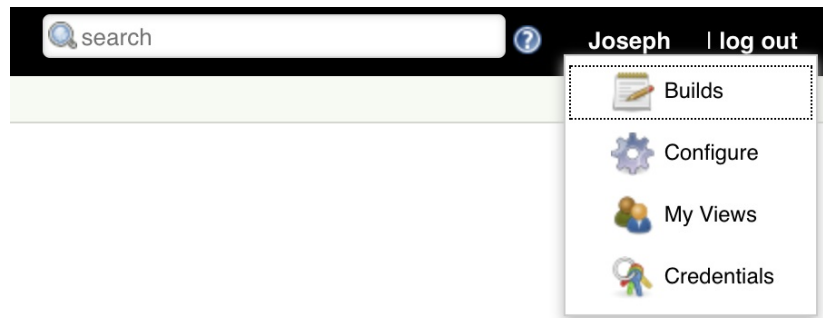


At first glance, we have a few options listed there. When you first log into Jenkins and you don't have any jobs or builds, the above will be the message displayed. In the top-right corner, there is an option to add a description. This is where you would add typical information about the server and guide, for example, some company info, as shown in the following screenshot:

In the top-right corner, there's a drop-down linked to your username. There are a few items to note here; they are as follows:

- **Builds:** Option to view all pipeline builds.
- **Configure:** Add a new project or Job.
- **User Views:** Display custom user views.
- **Credentials:** Display credentials, if authorized.

The drop-down panel gives us a few items to note too. Take a look at this screenshot:



The drop-down panel lists the following:

Selection	Explanation
New Item	Allows the user to create a new item, which could be a project, pipeline, and so on.
People	Lists all the users available.
Build History	Shows all builds.
Manage Jenkins	Lists all configurations related to the Jenkins server.
My Views	Lists all custom user views.
Credentials	Lists all of the user and server credentials available.
New View	Allows you to create a new view.



Some of the options will change when access is controlled for different users.

In the bottom-right corner, there's a REST API button, with some information on how to interact with the Jenkins API. Take a look at this screenshot:



Now take a look at this screenshot:

REST API

Many objects of Jenkins provide the remote access API. They are available at `.../api/` where `"..."` portion is the object for which you'd like to access.

XML API

Access data exposed in [HTML](#) as XML for machine consumption. [Schema](#) is also available.

You can also specify optional XPath to control the fragment you'd like to obtain (but see [below](#)). For example, `.../api/xml?xpath=/*/*[0]`.

For XPath that matches multiple nodes, you need to also specify the "wrapper" query parameter to specify the name of the root XML element to be create so that the resulting XML becomes well-formed.

Similarly `exclude` query parameter can be used to exclude nodes that match the given XPath from the result. This is useful for trimming down the amount of data you fetch (but again see [below](#)). This query parameter can be specified multiple times.

XPath filtering is powerful, and you can have it only return a very small data, but note that the server still has to build a full DOM of the raw data, which could cause a large memory spike. To avoid overloading the server, consider using the `tree` parameter, or use the `xpath` parameter in conjunction with the `tree` parameter. When used together, the result of the `tree` parameter filtering is built into DOM, then the XPath is applied to compute the final return value. In this way, you can often substantially reduce the size of DOM built in memory.

JSON API

Access the same data as JSON for JavaScript-based access. `tree` may be used.

Python API

Access the same data as Python for Python clients. This can be parsed into Python object as `eval(urllib.urlopen("...").read())` and the resulting object tree is identical to that of JSON. However, when you do this, beware of the security implication. If you are connecting to a non-trusted Jenkins, the server can send you malicious Python programs.

In Python 2.6 or later you can safely parse this output using `ast.literal_eval(urllib.urlopen("...").read())`

For more information about remote API in Jenkins, see [the documentation](#).

We will learn more about this when designing and building a pipeline, as there are some instances where you would be required to programmatically interact with Jenkins or run some scripts to get metrics or artifacts.



It is important to understand the dashboard as much as possible, so as to know where you would get and/or use a particular service. For now, we will solely focus on General Management and the setup of Jenkins, but as we progress, we will delve deeper into the service, giving us more information regarding the handling operations in Jenkins.

User Management

User Management describes the ability of administrators or super users to control access to resources and services, such as creating, stopping, or deleting Jenkins' pipelines.

User management is a core security essential and one of the enablers of security audits. Controlled access, as we discussed earlier, limits human error to production servers, for example, someone accidentally deleting a host or, in this case, a build and eventually deploying a buggy service, causing downtime.

User Management

Now we'll analyze how Jenkins handles user management and security.

1. Open **Manage Jenkins** and then select the **Configure Global Security** settings as follows. Take a look at this screenshot:

Jenkins

search

Joseph | log out

Jenkins » Configure Global Security

Configure Global Security

☒ Enable security
Disable remember me

Access Control

Security Realm

☐ Delegate to servlet container

☒ Jenkins' own user database

☐ Allow users to sign up

☐ LDAP

☐ Unix user/group database

Authorization

☐ Anyone can do anything

☐ Legacy mode

☒ Logged-in users can do anything

☐ Allow anonymous read access

☐ Matrix-based security

☐ Project-based Matrix Authorization Strategy

Markup Formatter

Save

Apply



i By default, we should have security enabled and the Security Realm as Jenkins's own user database. Authorization is also set to the default, allowing all users to do anything when logged in. While this might be OK if only one user exists, having more users can be hazardous if no control exists.

Let's have a look at what some of the options under Authorization perform:

- **Matrix-based security:** Allows the admin to grant permissions in groups, divided into contexts. Take a look at this screenshot:

Authorization

☐ Anyone can do anything
☐ Legacy mode
☐ Logged-in users can do anything
☒ Matrix-based security

User/group	Overall	Credentials				Agent								Job								
	Administer	Read	Create	Delete	Manage	Domains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect	Build	Cancel	Configure	Create	Delete	Discover	Move	Reassign
 Anonymous Users	<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 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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Authenticated Users	<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 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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

User/group to add:

☐ Project-based Matrix Authorization Strategy

- **Legacy mode:** Grants the administrator all rights, and limits everyone else to read-only access.
- **Project-based Matrix Authorization Strategy:** Gives control over group permissions per project.

2. Select **Matrix-based security**, and add a new user to your current account.



If you added admin, when you signed up, type in `admin` as follows.

3. Click on **Add**, and your user first and last names should appear on the list.
4. Select all checkboxes for the administrator, and then select **Apply** and save the changes. Take a look at this screenshot:

Anonymous Users	<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 type="checkbox"/>
Authenticated Users	<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 type="checkbox"/>
Joseph muli	<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 type="checkbox"/>

User/group to add:

Add

Activity: User Management and Security

Scenario

You have been asked to create two user accounts: an Ops engineer and a developer account. Both users should have different access levels from the administrator account. The Ops engineer will only not be able to do the following:

- Administer
- Create credentials
- View credentials
- Delete credentials
- Update credentials

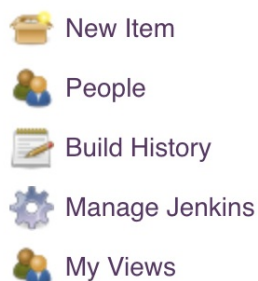
The developer will only be able to have overall **Read** permissions.

Aim

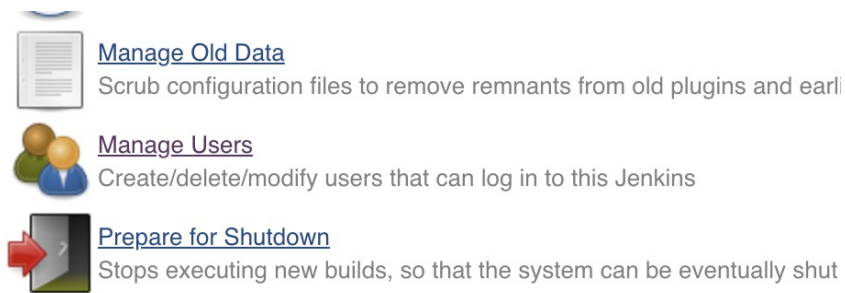
To create users with different access permissions and resource controls.

Steps for Completion

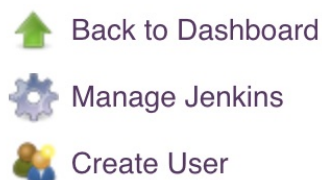
1. Select the **Manage Jenkins** option from the configuration panel. Take a look at this screenshot:



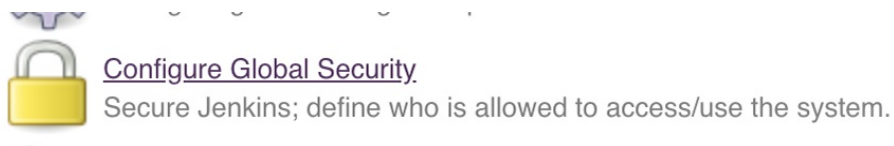
2. Select the **Manage Users** option. Take a look at this screenshot:



3. Select the **Create User** option on the left-hand side navigation bar. Take a look at this screenshot:








4. Go to the **Configure Global Security** option. Take a look at this screenshot:



5. Under **Authorization**, select **Matrix-based security**.
6. Add the new users.

Authorization

- ☐ Anyone can do anything
- ☐ Legacy mode
- ☐ Logged-in users can do anything
- ☒ Matrix-based security






User/group	Overall	Credentials					Agent							
	Administer	Read	Create	Delete	Manage	Domains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect
 Anonymous Users	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Authenticated Users	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Joseph muli	<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 checked="" type="checkbox"/>
 Ops Engineer	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Dev One	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

User/group to add:

Add

7. Update the users' permissions as shown in the following screenshot:

- ☒ Matrix-based security

User/group	Overall		Credentials					Agent						
	Administer	Read	Create	Delete	Manage	Domains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect
 Anonymous Users	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Authenticated Users	<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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Joseph muli	<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 checked="" type="checkbox"/>
 Ops Engineer	<input type="checkbox"/>	<input checked="" 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"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 Dev One	<input type="checkbox"/>	<input checked="" 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 type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

We have successfully created two user accounts: an ops engineer account and a developer account with different access permissions and resource controls.