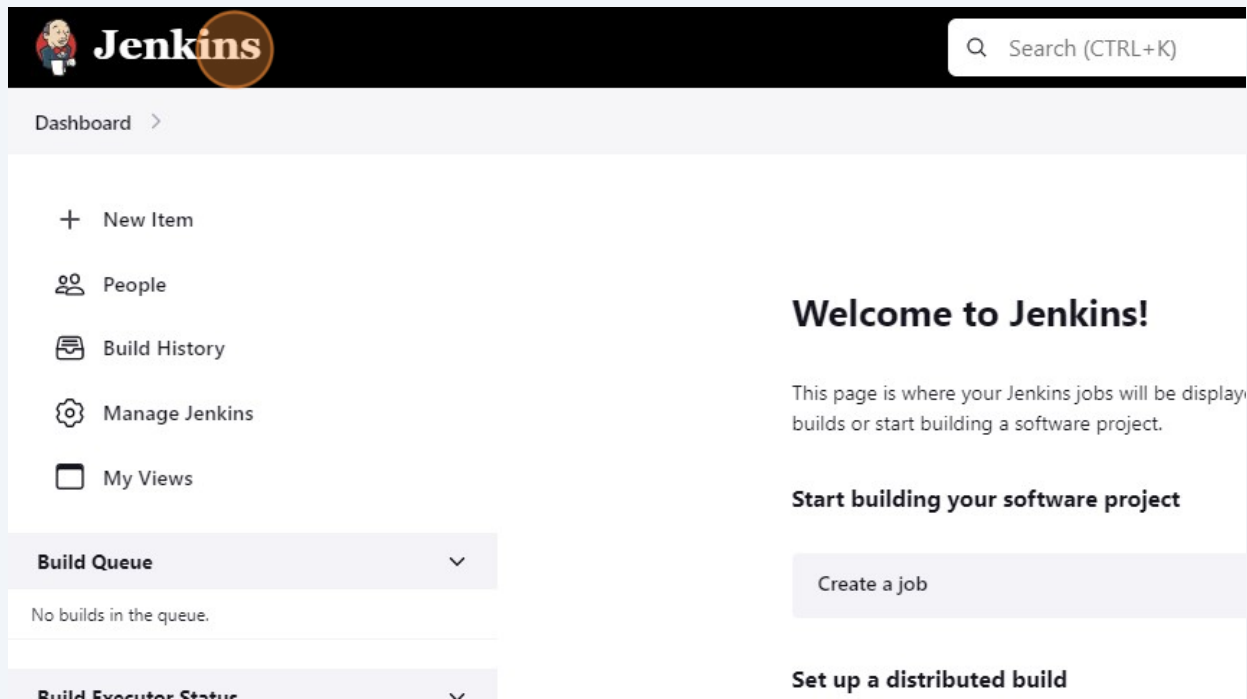
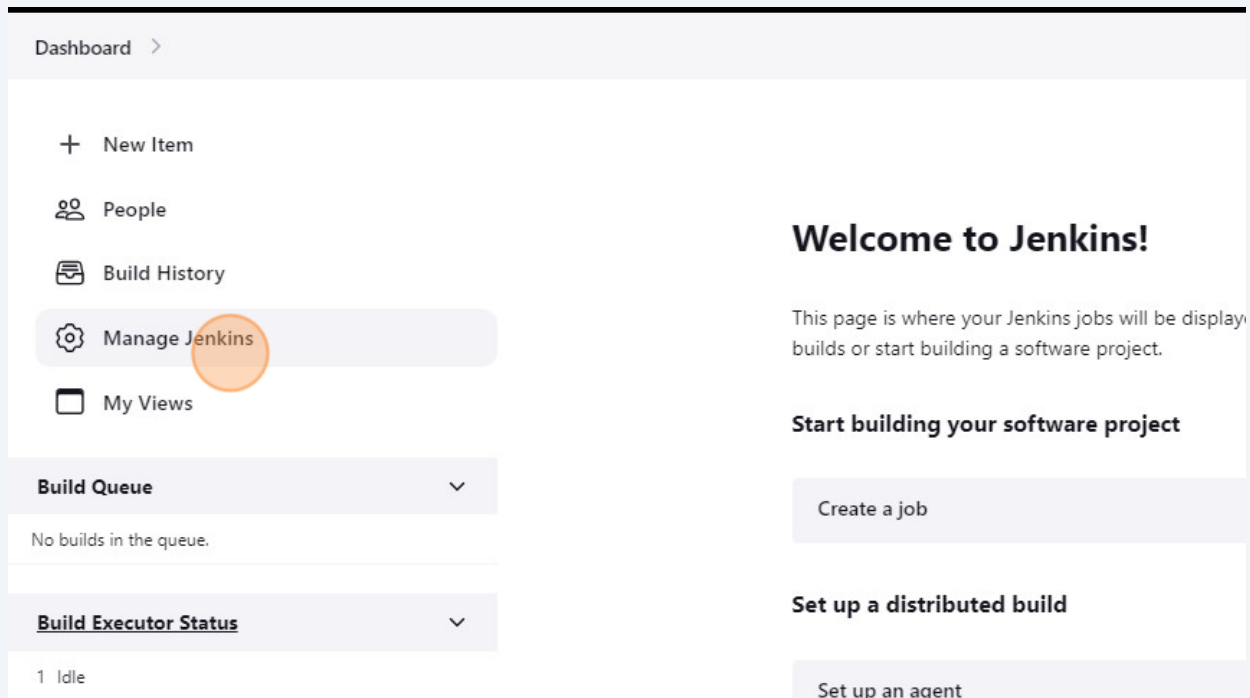


# How to Configure Jenkins Credentials and Webhook Notifications

1



## 2 Click "Manage Jenkins"



The screenshot shows the Jenkins Dashboard. On the left, there is a sidebar with navigation links: "New Item", "People", "Build History", "Manage Jenkins" (highlighted with an orange circle), and "My Views". Below the sidebar, there are two sections: "Build Queue" showing "No builds in the queue." and "Build Executor Status" showing "1 Idle". On the right, the main content area has a "Welcome to Jenkins!" heading, followed by a paragraph explaining the page's purpose. Below this, there are two sections: "Start building your software project" with a "Create a job" button, and "Set up a distributed build" with a "Set up an agent" button.

Dashboard >

- + New Item
- People
- Build History
- Manage Jenkins**
- My Views

**Build Queue** ▼

No builds in the queue.

**Build Executor Status** ▼

1 Idle

### Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed, or you can start building a software project.

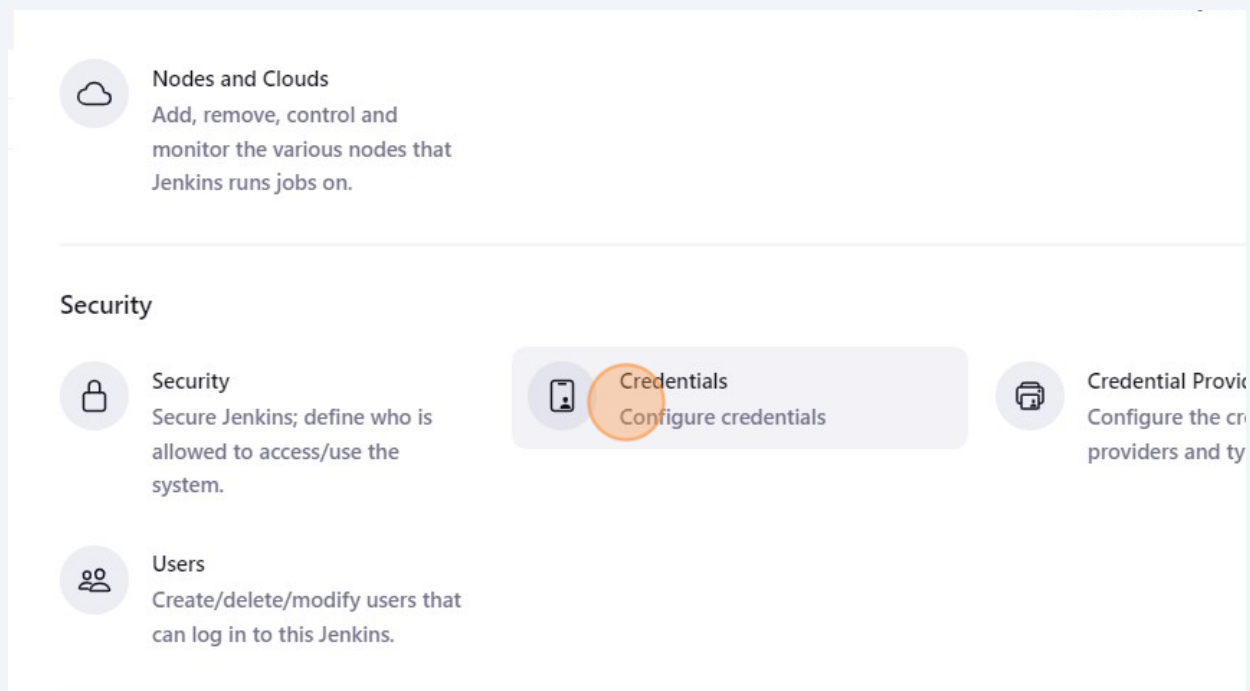
#### Start building your software project

Create a job

#### Set up a distributed build

Set up an agent

## 3 Click "Configure credentials"



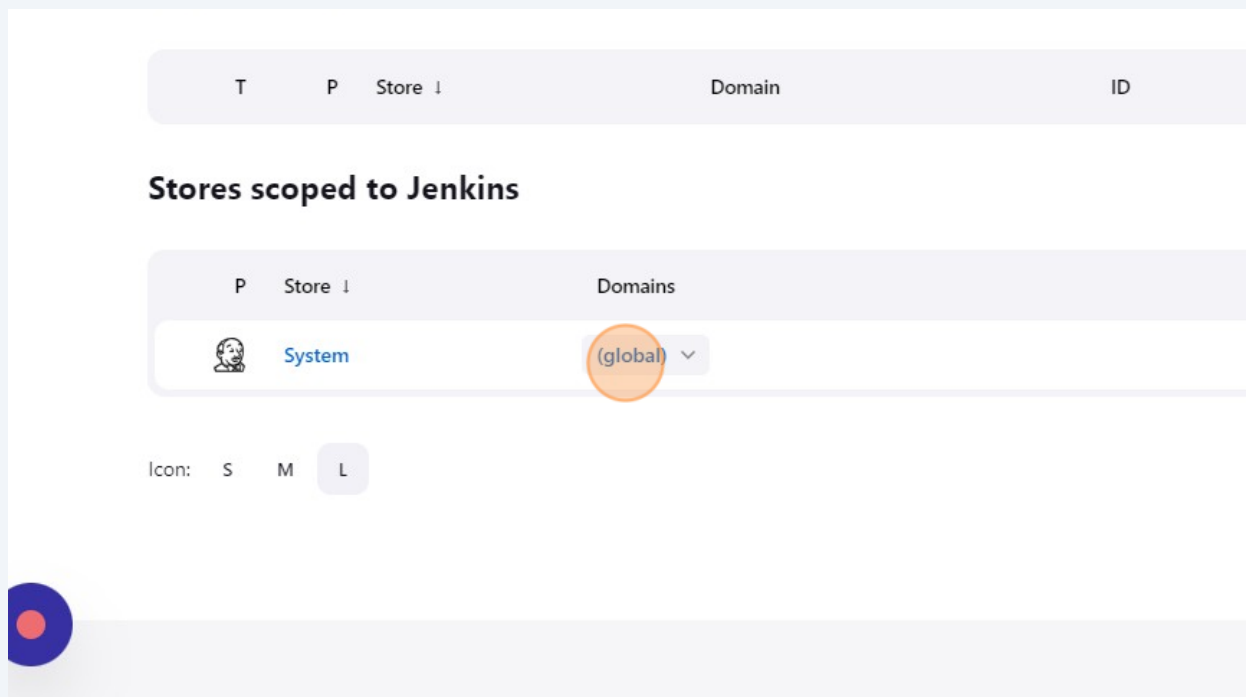
The screenshot shows the Jenkins Security Configuration page. On the left, there is a sidebar with navigation links: "Nodes and Clouds", "Security", "Users", and "Credentials" (highlighted with an orange circle). The main content area has a "Nodes and Clouds" section with a description. Below this, there is a "Security" section with three sub-sections: "Security" (Secure Jenkins; define who is allowed to access/use the system.), "Credentials" (Configure credentials), and "Credential Providers" (Configure the credential providers and types).

**Nodes and Clouds**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

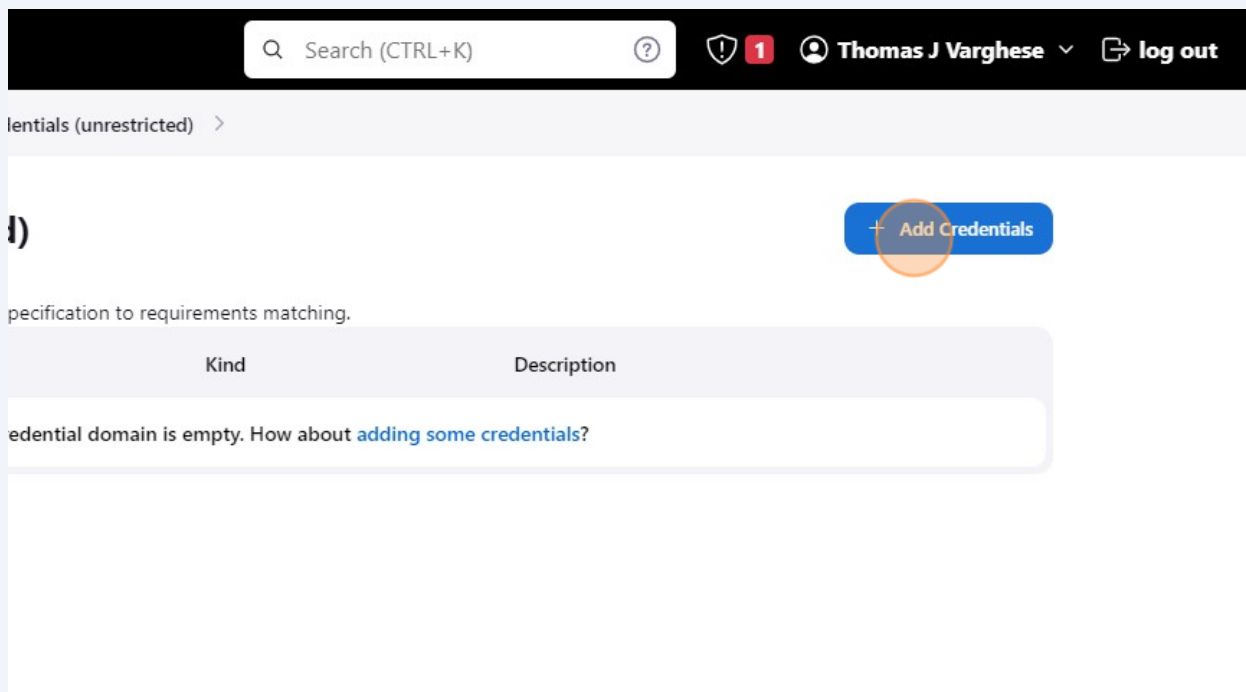
### Security

- Security**  
Secure Jenkins; define who is allowed to access/use the system.
- Credentials**  
Configure credentials
- Credential Providers**  
Configure the credential providers and types

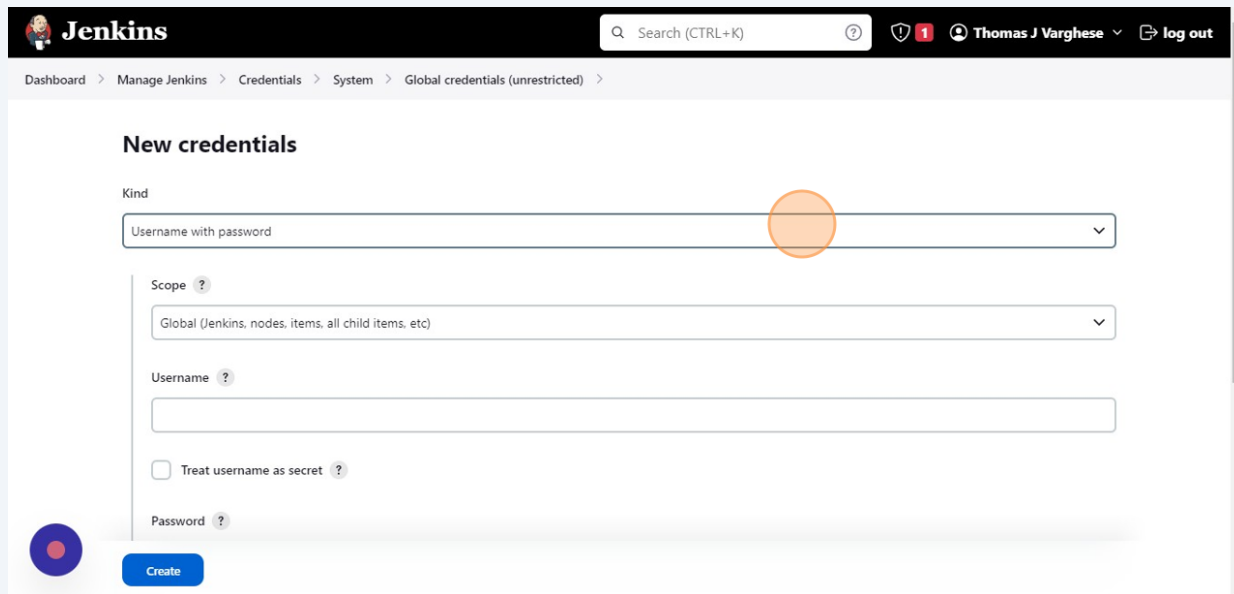
#### 4 Click "(global)"



#### 5 I am going to add authentication for jenkins to communicate with the production machine which by using the contents of PEM file that I got while creating the instance. Click "Add Credentials"

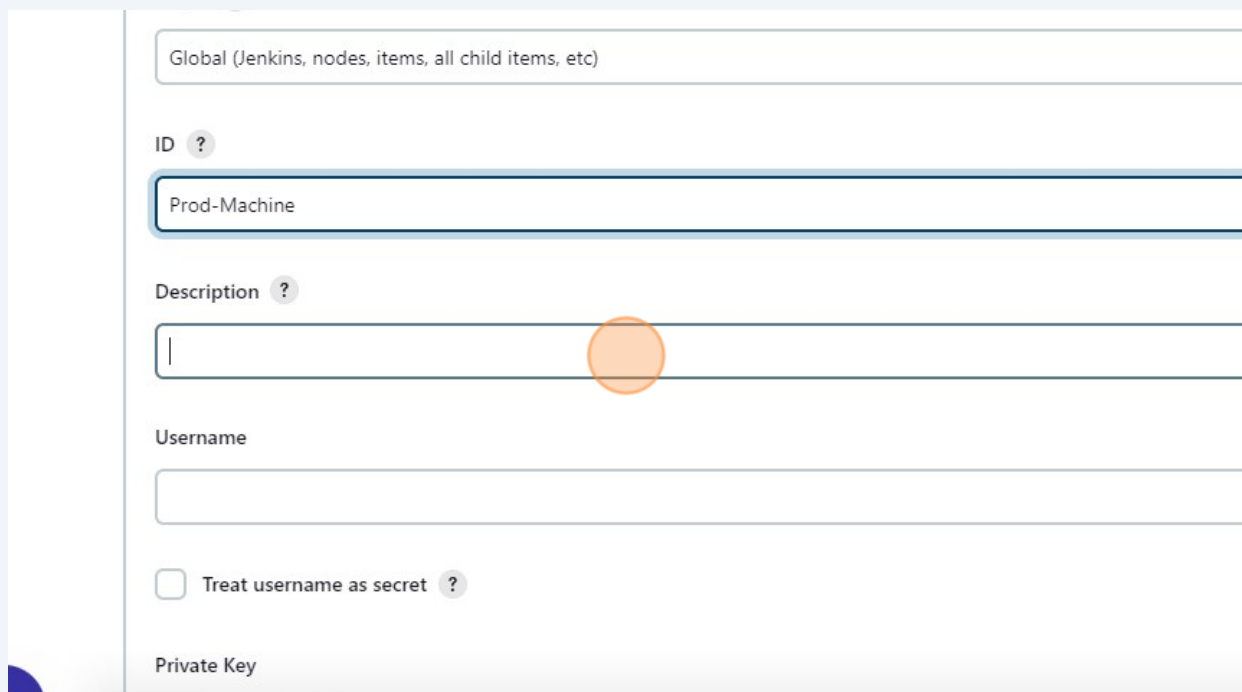


6 Click this dropdown. On their select "SSH Username with private key"



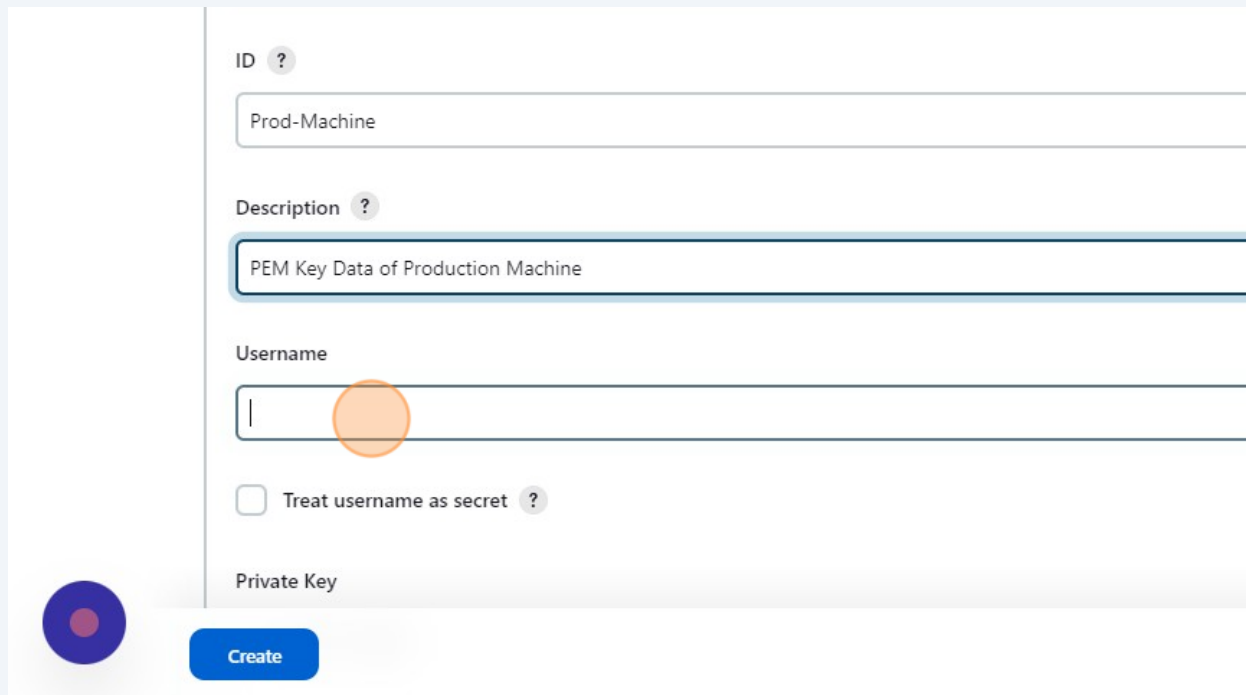
The screenshot shows the Jenkins 'New credentials' form. The 'Kind' dropdown is highlighted with an orange circle and currently shows 'Username with password'. The breadcrumb trail at the top reads: Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >. The form includes fields for 'Scope' (set to 'Global (Jenkins, nodes, items, all child items, etc)'), 'Username', a checkbox for 'Treat username as secret', and a 'Password' field. A blue 'Create' button is at the bottom left.

7 Here it is important to give the ID name related to that. Here I have added the name as "Prod-Machine". Using this ID we will be using it inside the Jenkins Pipeline. Description is optional.



This is a close-up of the 'New credentials' form, focusing on the 'ID' field which is highlighted with an orange circle and contains the text 'Prod-Machine'. The 'Scope' dropdown above it is set to 'Global (Jenkins, nodes, items, all child items, etc)'. Below the ID field is the 'Description' field, which is empty and also highlighted with an orange circle. Further down are the 'Username' field, a 'Treat username as secret' checkbox, and the 'Private Key' section.

8 Type "ubuntu" for Username field.



ID ?

Prod-Machine

Description ?

PEM Key Data of Production Machine

Username

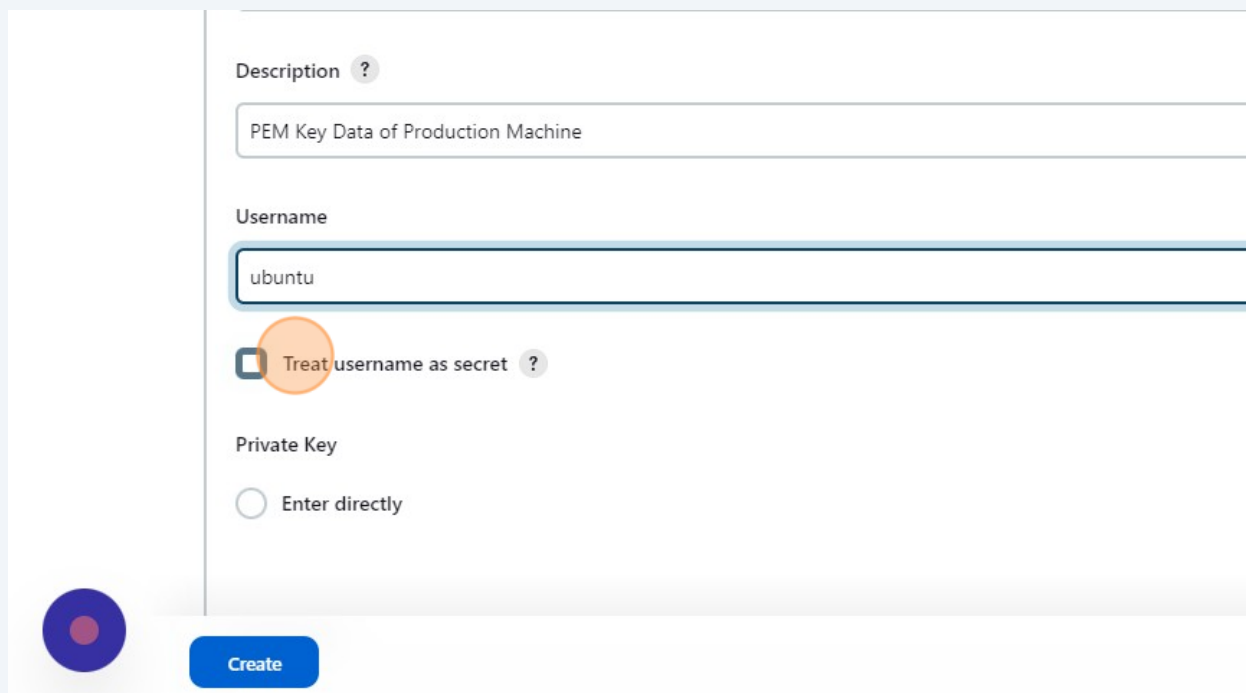
|

☐ Treat username as secret ?

Private Key

Create

9 Click "Treat username as secret"



Description ?

PEM Key Data of Production Machine

Username

ubuntu

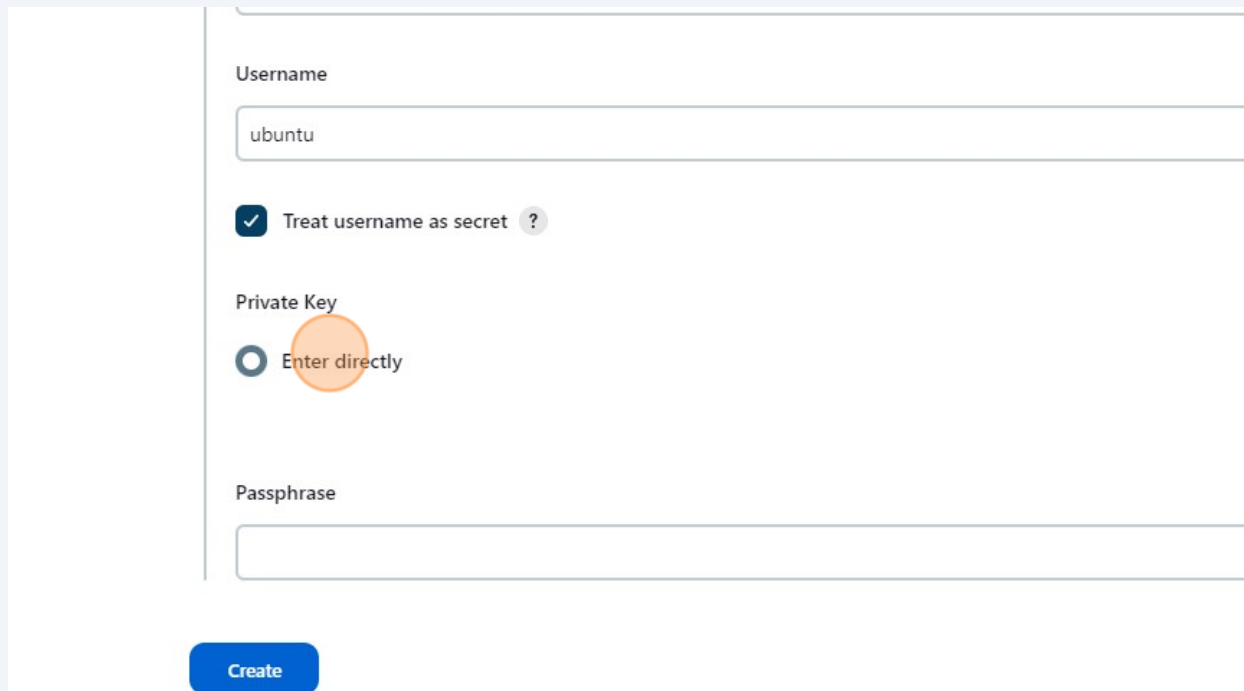
☒ Treat username as secret ?

Private Key

☐ Enter directly

Create

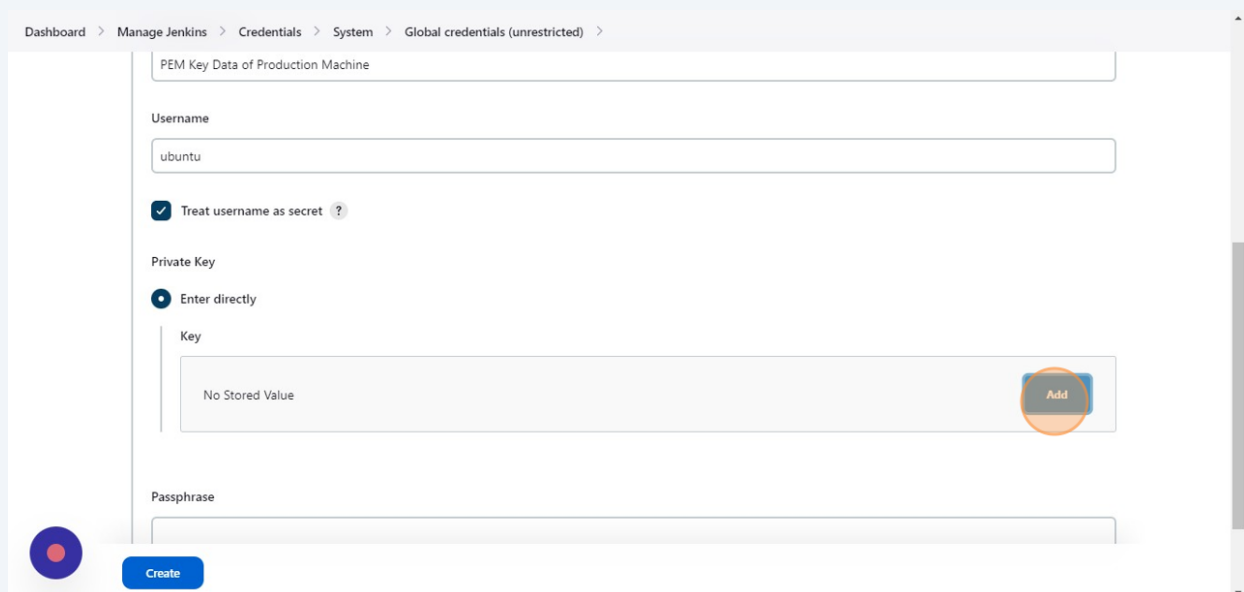
## 10 Click "Enter directly"



A screenshot of the Jenkins 'Add Credentials' form. The form is titled 'Add Credentials' and has a breadcrumb trail: 'Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >'. The form contains the following fields and options:

- Username:** A text input field containing the value 'ubuntu'.
- Treat username as secret:** A checked checkbox with a help icon.
- Private Key:** A section with two radio buttons: 'Enter directly' (selected and highlighted with an orange circle) and 'From file'.
- Passphrase:** A text input field.
- Create:** A blue button at the bottom left.

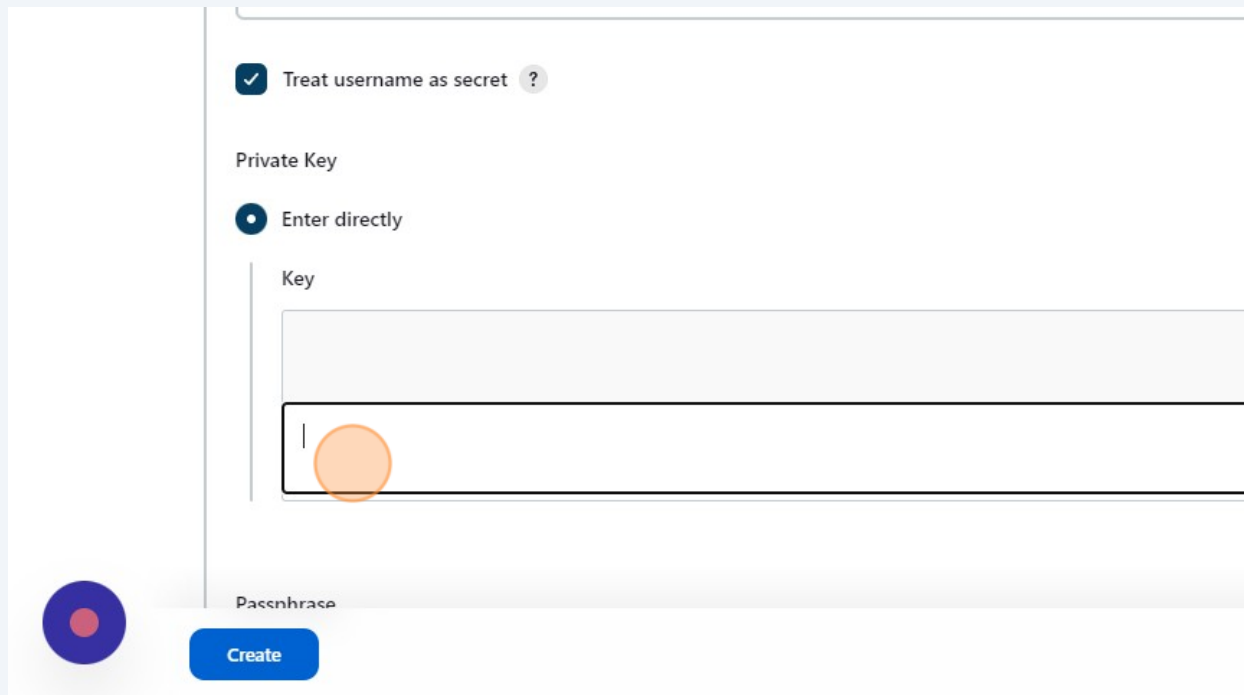
## 11 Click this button field.



A screenshot of the Jenkins 'Add Credentials' form, showing the 'Enter directly' option selected. The form is titled 'Add Credentials' and has a breadcrumb trail: 'Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >'. The form contains the following fields and options:

- Username:** A text input field containing the value 'ubuntu'.
- Treat username as secret:** A checked checkbox with a help icon.
- Private Key:** A section with two radio buttons: 'Enter directly' (selected) and 'From file'.
- Key:** A text input field containing the value 'No Stored Value'. An 'Add' button (highlighted with an orange circle) is located to the right of this field.
- Passphrase:** A text input field.
- Create:** A blue button at the bottom left.

## 12 Paste in your PEM Key contents here.



☒ Treat username as secret ?

Private Key

☒ Enter directly

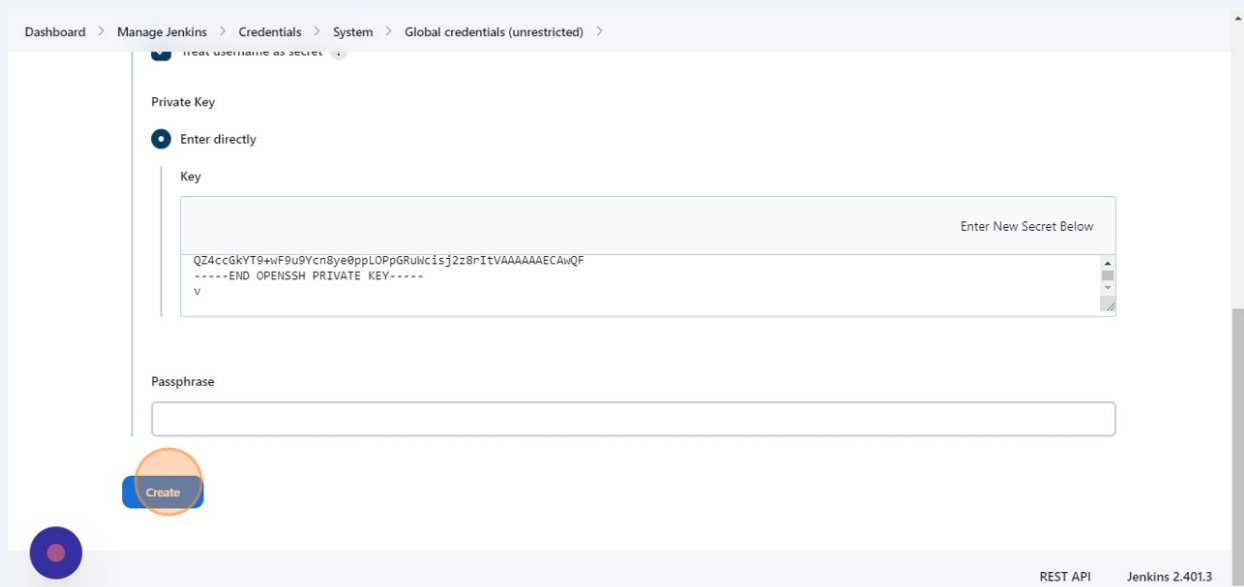
Key

Passphrase

Create

This screenshot shows the 'Create' dialog for a Private Key credential in Jenkins. The 'Treat username as secret' checkbox is checked. Under the 'Private Key' section, the 'Enter directly' radio button is selected. A large text area for the 'Key' is visible, with an orange circle highlighting the start of the input field. A 'Passphrase' field is located below the key field. A blue 'Create' button is at the bottom right.

## 13 Click "Create"



Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

Treat username as secret ?

Private Key

☒ Enter directly

Key

Enter New Secret Below

```
QZ4ccgKYT9+uF9u9Ycn8ye@pplOPpGRuWcisj2z8rItVAAAAAECauQF
-----END OPENSSH PRIVATE KEY-----
v
```

Passphrase

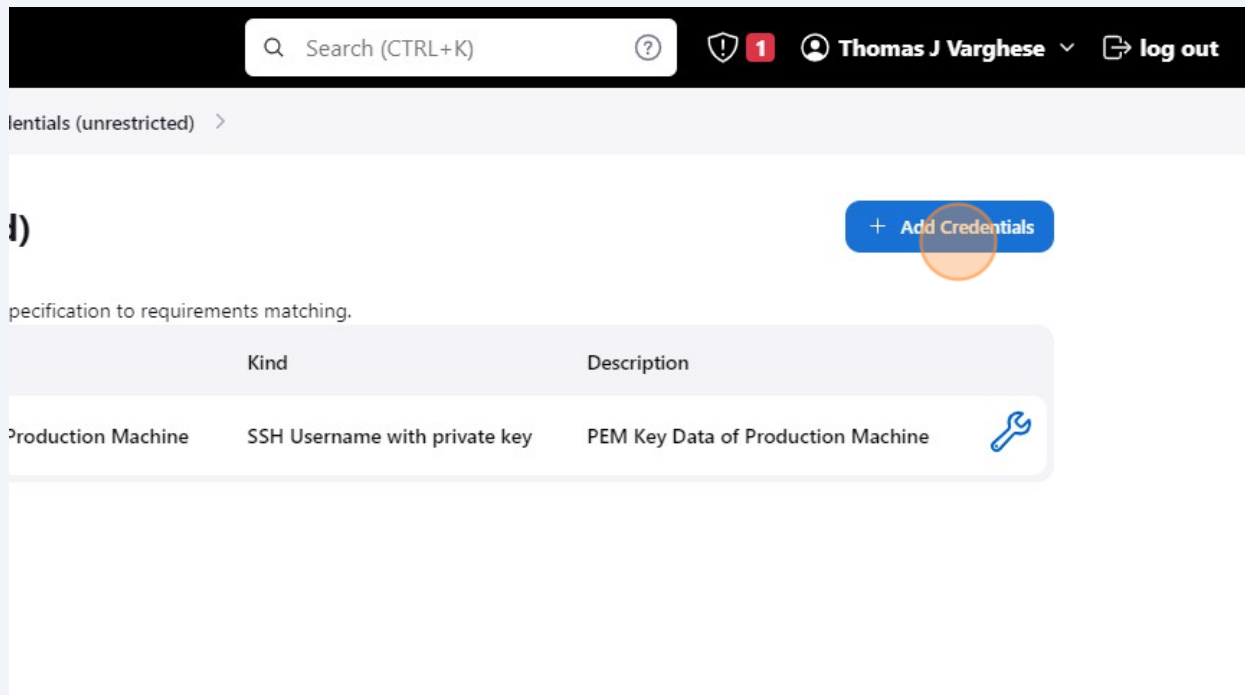
Create

REST API Jenkins 2.401.3

This screenshot shows the 'Create' dialog for a Private Key credential in Jenkins, with the 'Key' field populated. The breadcrumb trail at the top indicates the path: Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >. The 'Treat username as secret' checkbox is checked. Under the 'Private Key' section, the 'Enter directly' radio button is selected. The 'Key' field contains a PEM-formatted private key. An orange circle highlights the 'Create' button. The bottom right corner shows 'REST API' and 'Jenkins 2.401.3'.

14

Now we will do the same thing by pasting the SSH private key of our Machine to Jenkins for authentication. Click "Add Credentials"

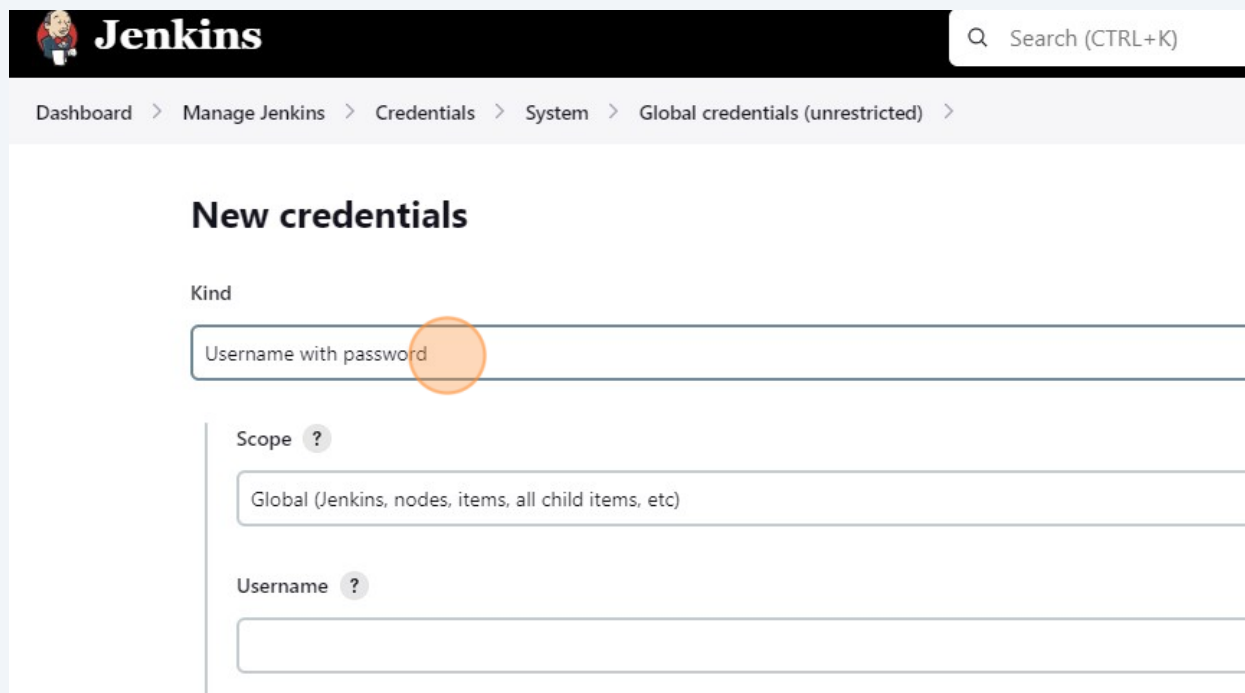


The screenshot shows the Jenkins interface. At the top, there is a search bar with the text "Search (CTRL+K)" and a user profile for "Thomas J Varghese" with a "log out" button. Below the search bar, the breadcrumb "Credentials (unrestricted) >" is visible. The main content area shows a table of credentials. The table has two columns: "Kind" and "Description". There is one entry in the table: "Production Machine" with the description "SSH Username with private key" and "PEM Key Data of Production Machine". A blue button labeled "+ Add Credentials" is located in the top right corner of the table. An orange circle highlights the "+ Add Credentials" button.

Kind	Description
Production Machine	SSH Username with private key PEM Key Data of Production Machine

15

Click this dropdown.



The screenshot shows the Jenkins "New credentials" form. The Jenkins logo and name are at the top left, and a search bar with "Search (CTRL+K)" is at the top right. The breadcrumb "Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >" is visible. The form has a section titled "New credentials". Under the "Kind" label, there is a dropdown menu with "Username with password" selected. An orange circle highlights the dropdown menu. Below the "Kind" section, there is a "Scope" section with a question mark icon and a dropdown menu with "Global (Jenkins, nodes, items, all child items, etc)" selected. Below the "Scope" section, there is a "Username" section with a question mark icon and an empty text input field.

Kind

Username with password

Scope ?

Global (Jenkins, nodes, items, all child items, etc)

Username ?



16

ID is important here, I am passing "Github-PrivateKey". This is on the assumption that you have already set the public key in github.

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

SSH Username with private key

Scope ?  
Global (Jenkins, nodes, items, all child items, etc)

ID ?

Description ?

Username

17

Fill in the rest of the fields, make sure the User name is "ubuntu".

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

Github-PrivateKey

Description ?  
PrivateSSH Key of SSH

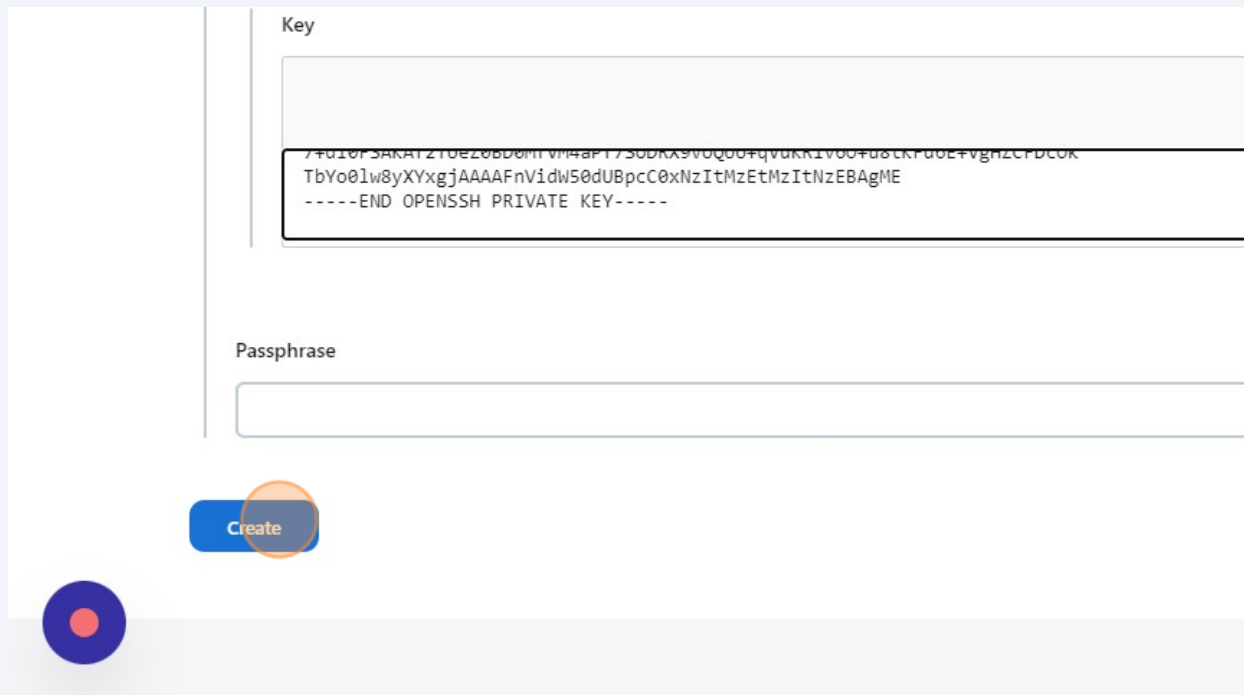
Username  
ubuntu

☒ Treat username as secret ?

Private Key  
☐ Enter directly

Key  
No Stored Value

## 18 Add the SSH Private Key of our Production Machine and Click "Create"



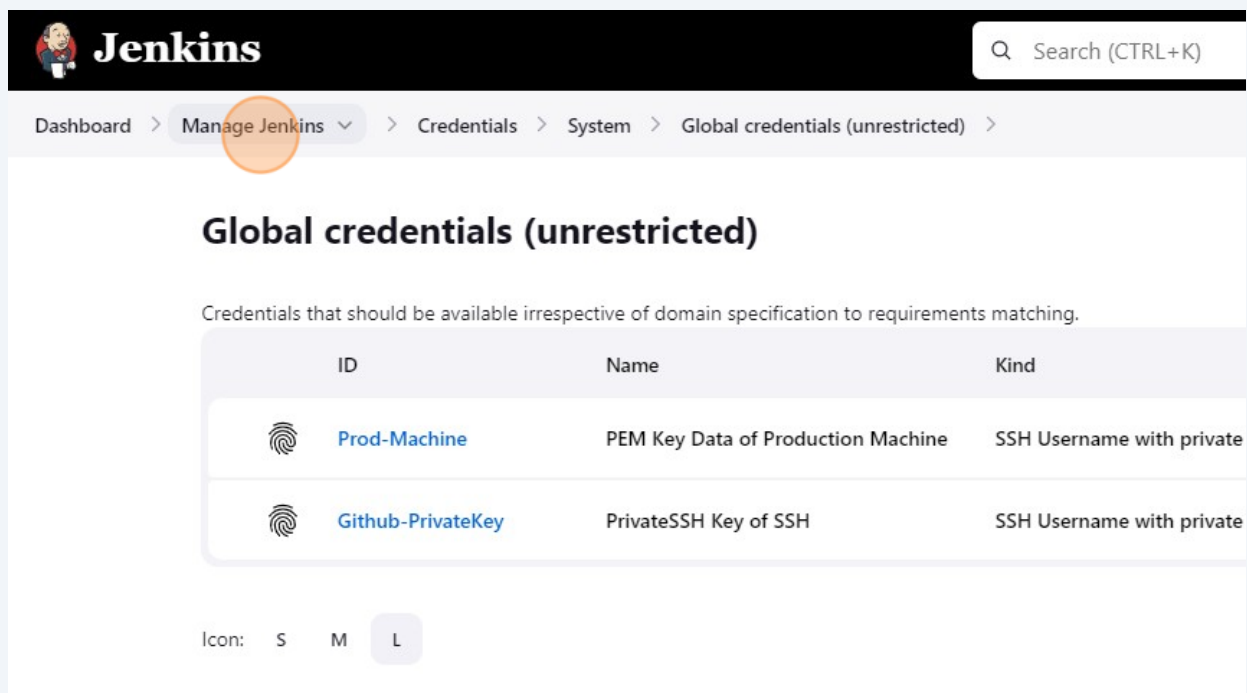
Key

```
7#d10P5AKATZT0E208D0M VM4aP1730DKA9V0Q00#qvukK1v00#d8CKF00E#vGHZCPDCOK  
TbYo0lw8yXYxgjAAAAFnVidW50dUBpcC0xNzItMzEtMzItNzEBAgME  
-----END OPENSSH PRIVATE KEY-----
```

Passphrase

Create

## 19 Once that is done we need to install 'SSH Agent' extension. For that Click "Manage Jenkins"





**Jenkins** Search (CTRL+K)

Dashboard > Manage Jenkins > Credentials > System > Global credentials (unrestricted) >

### Global credentials (unrestricted)

Credentials that should be available irrespective of domain specification to requirements matching.

ID	Name	Kind
 <a href="#">Prod-Machine</a>	PEM Key Data of Production Machine	SSH Username with private
 <a href="#">Github-PrivateKey</a>	PrivateSSH Key of SSH	SSH Username with private

Icon: S M L

20

Click "Add, remove, disable or enable plugins that can extend the functionality of Jenkins."

It-in node can be a security issue. You should set up distributed builds. See [the](#)

[Set up agent](#)[Set up cloud](#)[Dismiss](#)

ation

global settings and



Tools

Configure tools, their locations and automatic installers.



Plugins

Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

I Clouds

ve, control and  
ie various nodes that  
ns jobs on.

21

Click "Available plugins"



# Jenkins

[Dashboard](#) > [Manage Jenkins](#) > [Plugins](#)



Updates



Available plugins



Installed plugins



Advanced settings



Download progress

## Plugins



Name ↓

Released

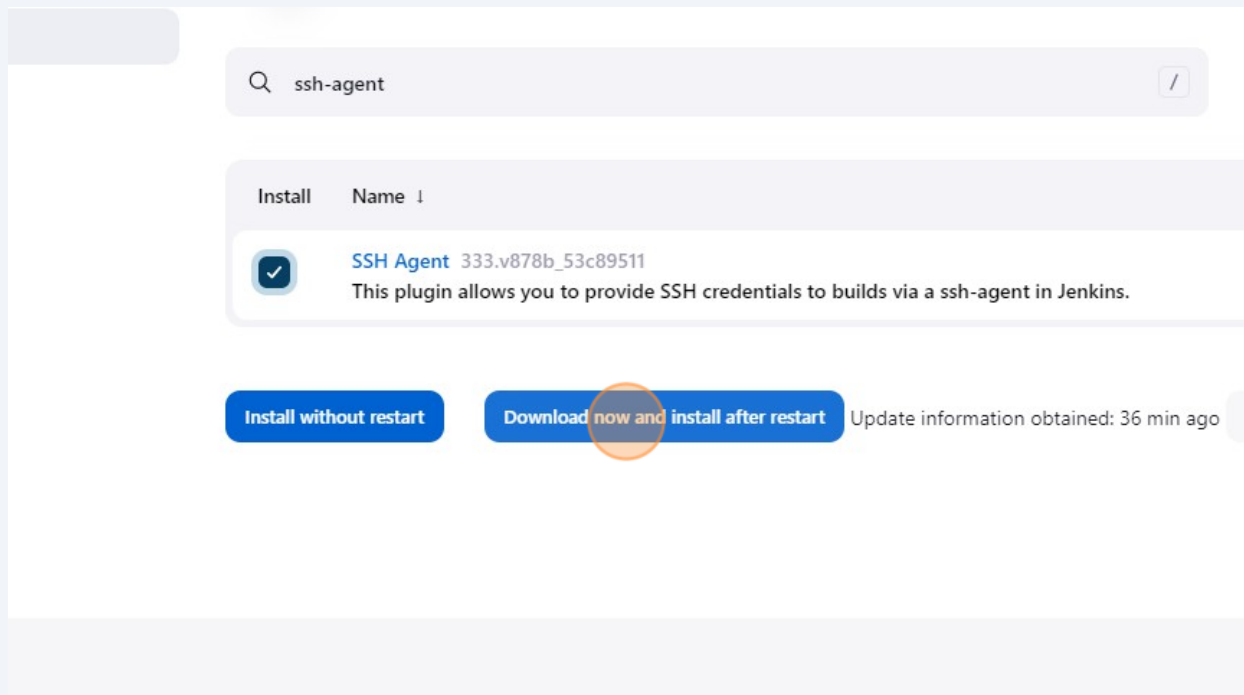
No up

Update information obtained: 35 min ago

[Check now](#)

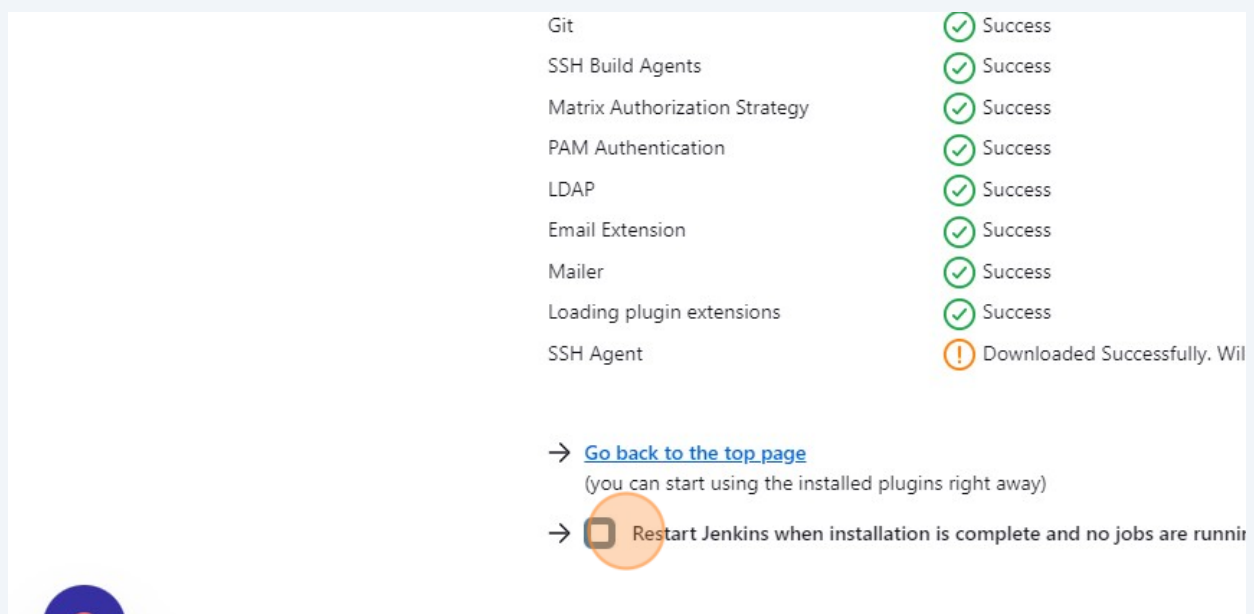
22

Search for "ssh-agent". Check the box and click the Download now and install after restart button.



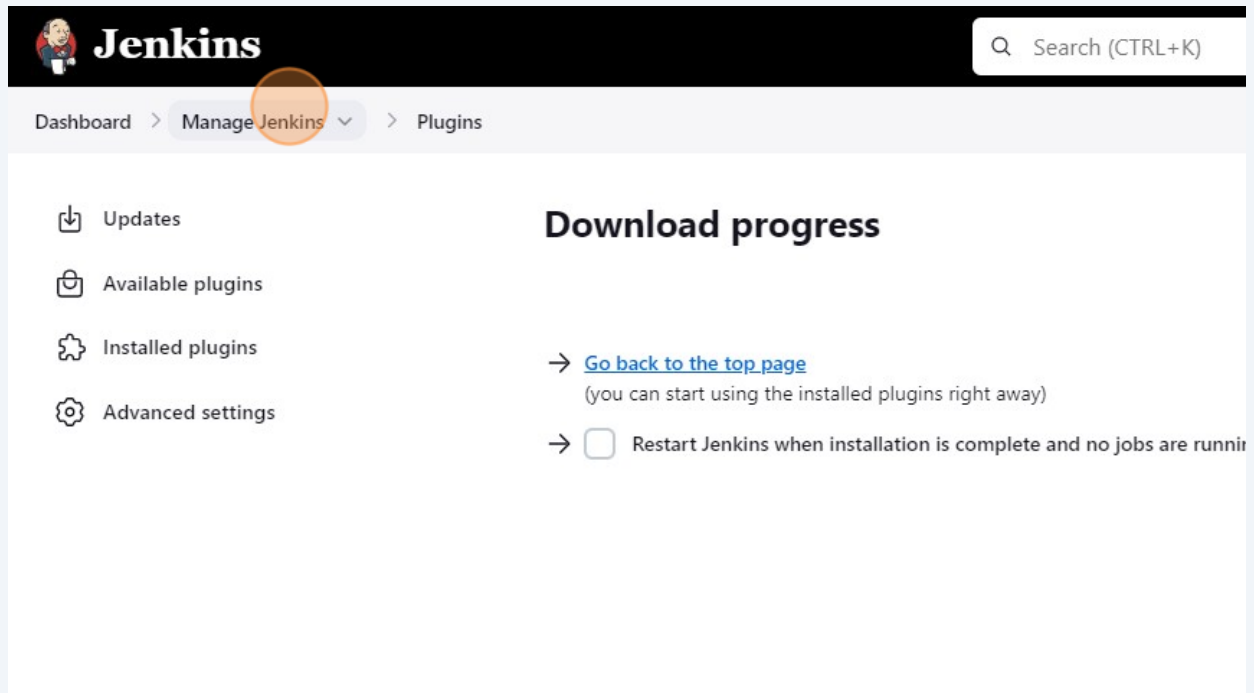
23

Then it will take you to a page and go below and Click "Restart Jenkins when installation is complete and no jobs are running"



24 This will restart the Jenkins and we will asked to give username and password.

25 Now we need to edit settings for Git. For that Click "Manage Jenkins"



## 26 Click "Secure Jenkins"

Dashboard > Manage Jenkins

add, remove, control and monitor the various nodes that Jenkins runs jobs on.

### Security



#### Security

Secure Jenkins; define who is allowed to access/use the system.



#### Credentials

Configure credentials



#### Users

Create/delete/modify users that can log in to this Jenkins.

### Status Information

## 27 Go below and Click this dropdown.

Require a legacy API token for each newly created user (Not recommended) ?

Require users to manually create a legacy API token (Not recommended) ?

Legacy API Token usage statistics ?

### Legacy Verification Configuration

Verification Strategy ?

File

Apply

## 28 Change it to 'Accept first connection' and Click "Save"

☐ Generate a legacy API token for each newly created user (Not recommended) ?

☐ Allow users to manually create a legacy API token (Not recommended) ?

☒ Enable API Token usage statistics ?

---

### Git Host Key Verification Configuration

Host Key Verification Strategy ?

Accept first connection

**Save** Apply

## 29 Now go to your Github Repository and Click "Settings"

thomasjv799 / react-sample-app

Type to search

<> Code Issues Pull requests Actions Projects Wiki Security Insights **Settings**

react-sample-app Public

Pin Unwatch 1 Fork 0 Star 0

master 1 branch 0 tags

Go to file Add file <> Code

thomasjv799 Delete .github/workflows directory 127bd0f last week 3 commits

public	New Repo	last week
src	New Repo	last week
.gitignore	New Repo	last week
README.md	New Repo	last week
package-lock.json	New Repo	last week
package.json	New Repo	last week

README.md

### Getting Started with Create React App

About

No description, website, or topics provided.

- Readme
- Activity
- 0 stars
- 1 watching
- 0 forks

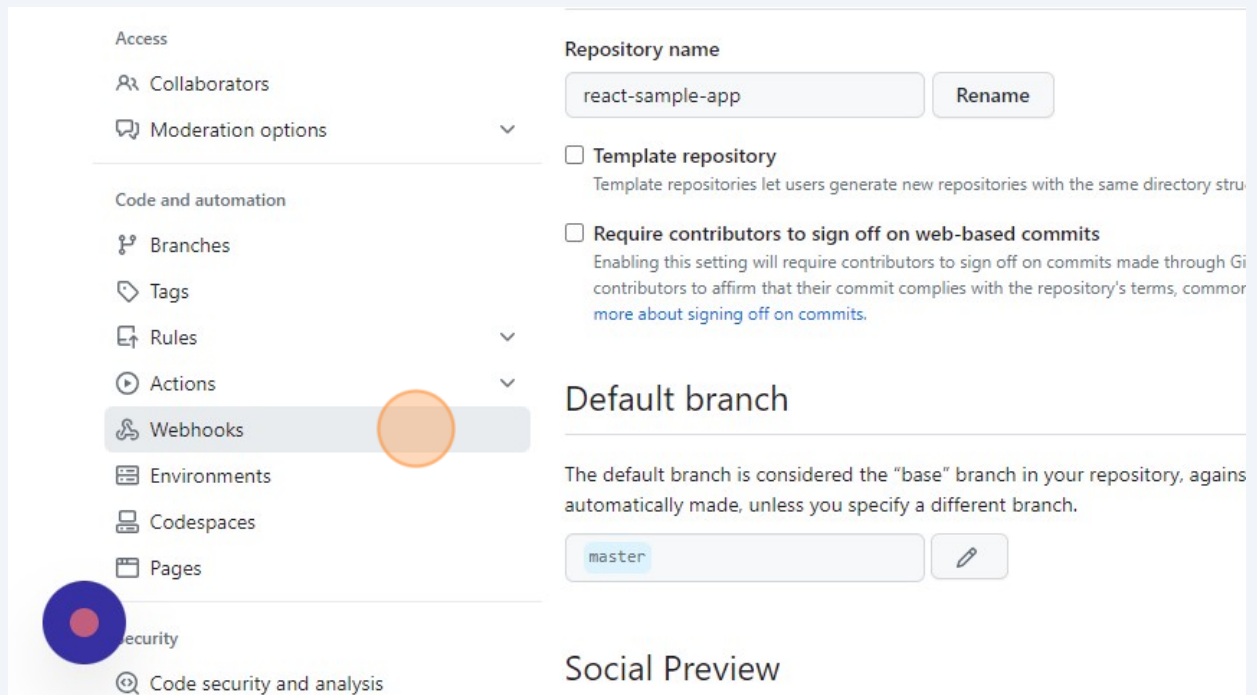
Releases

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

Packages

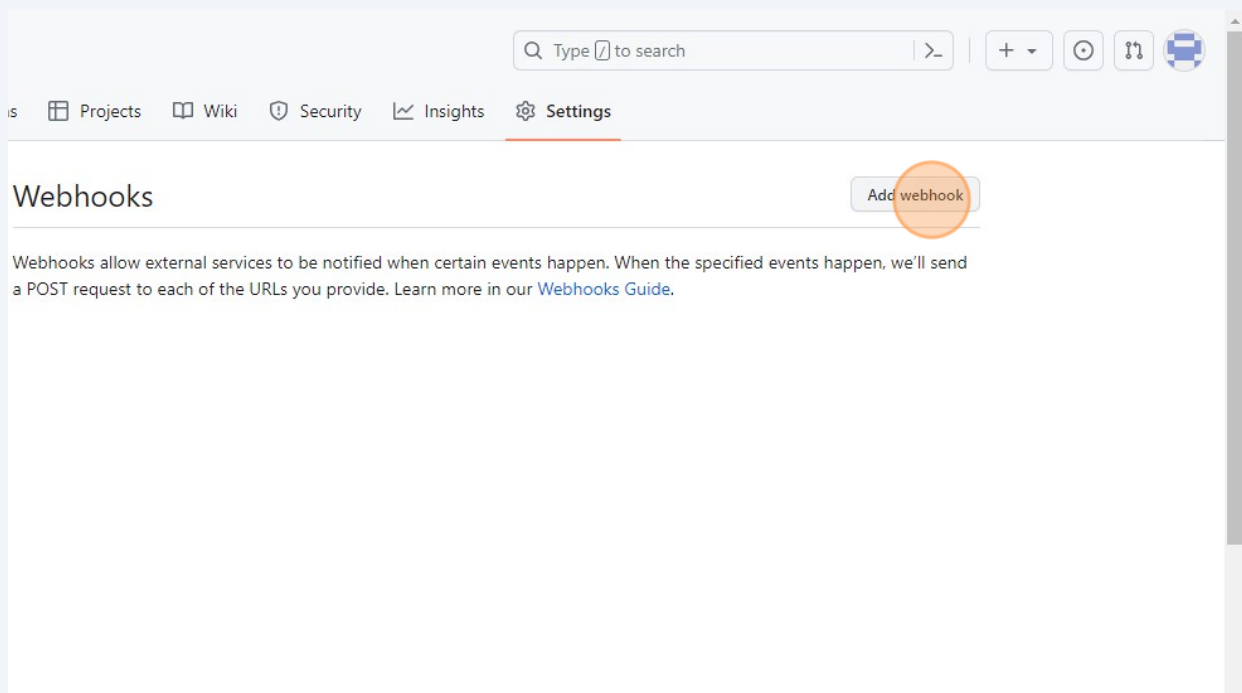
No packages published  
[Publish your first package](#)

## 30 Click "Webhooks"



The screenshot shows the GitHub repository settings page for a repository named 'react-sample-app'. The left sidebar contains several sections: 'Access' (Collaborators, Moderation options), 'Code and automation' (Branches, Tags, Rules, Actions, **Webhooks**, Environments, Codespaces, Pages), and 'Security' (Code security and analysis). The 'Webhooks' tab is highlighted with an orange circle. The main content area shows the 'Repository name' as 'react-sample-app' with a 'Rename' button. Below this are two checkboxes: 'Template repository' and 'Require contributors to sign off on web-based commits'. The 'Default branch' is set to 'master'. A 'Social Preview' section is partially visible at the bottom.

## 31 Click "Add webhook"

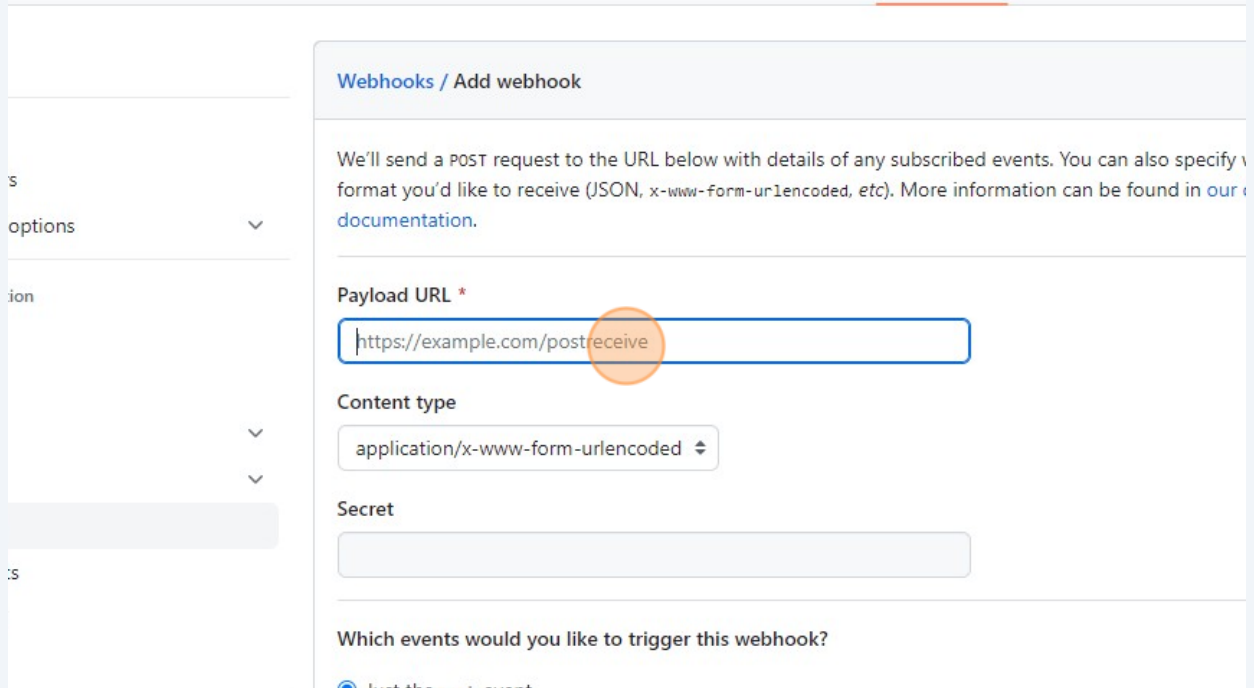


The screenshot shows the GitHub 'Webhooks' settings page. At the top, there is a search bar and navigation links for 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The 'Webhooks' section is active, and the 'Add webhook' button is highlighted with an orange circle. Below the button, there is a brief explanation of webhooks: 'Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).'



32

Click the "Payload URL" field. Paste the Jenkins URL followed by "github-webhook".  
Eg: 15.206.100.210:8080/github-webhook



**Webhooks / Add webhook**

We'll send a POST request to the URL below with details of any subscribed events. You can also specify a format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in our [documentation](#).

**Payload URL \***

`https://example.com/postreceive`

**Content type**

application/x-www-form-urlencoded

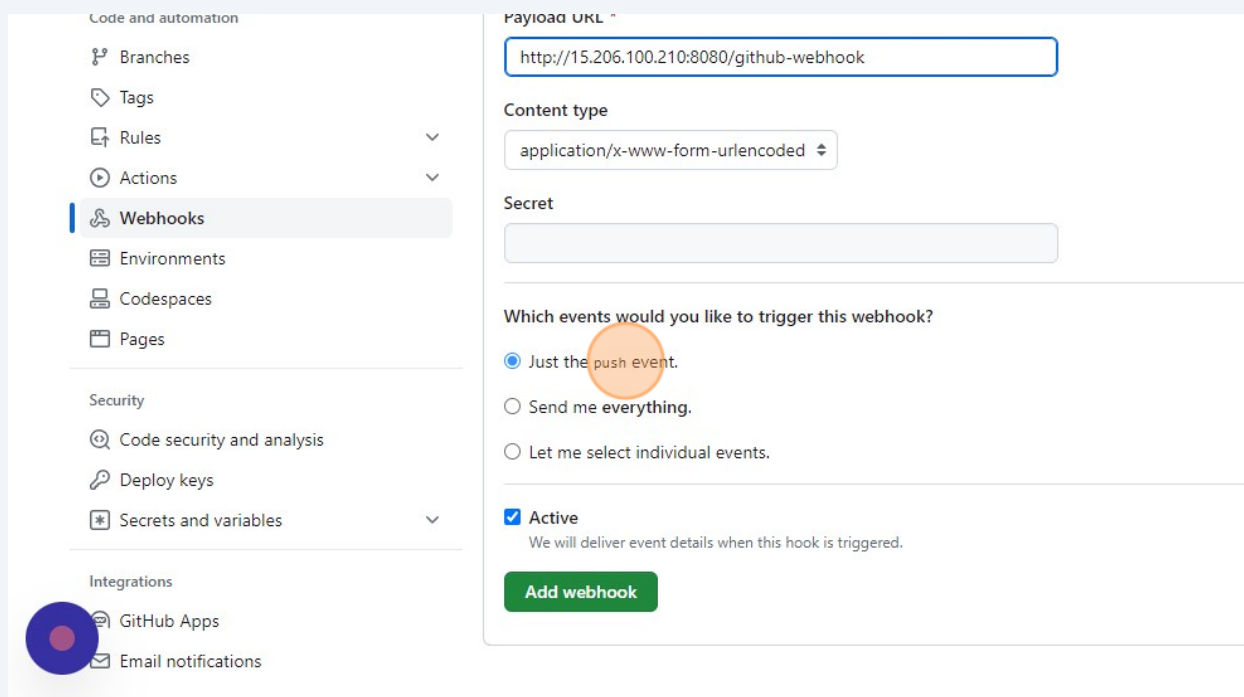
**Secret**

Which events would you like to trigger this webhook?

☒ Just the push event

33

Click "Just the push event." This will monitor the repo for any push event. If you click. For other options click the other options below.



**Code and automation**

- Branches
- Tags
- Rules
- Actions
- Webhooks**
- Environments
- Codespaces
- Pages

**Security**

- Code security and analysis
- Deploy keys
- Secrets and variables

**Integrations**

- GitHub Apps
- Email notifications

**Payload URL \***

`http://15.206.100.210:8080/github-webhook`

**Content type**

application/x-www-form-urlencoded

**Secret**

Which events would you like to trigger this webhook?

☒ Just the push event.

☐ Send me everything.

☐ Let me select individual events.

☒ **Active**  
We will deliver event details when this hook is triggered.

**Add webhook**

### 34 Click "Add webhook"

The screenshot shows the GitHub 'Add webhook' configuration page. On the left sidebar, the 'Webhooks' menu item is highlighted. The main content area contains the following fields and options:

- Content type:** A dropdown menu showing 'application/x-www-form-urlencoded'.
- Secret:** A text input field for the webhook secret.
- Which events would you like to trigger this webhook?:** Three radio button options:
  - ☒ Just the push event.
  - ☐ Send me **everything**.
  - ☐ Let me select individual events.
- Active:** A checked checkbox with the text 'We will deliver event details when this hook is triggered.'
- Add webhook:** A green button with the text 'Add webhook', which is highlighted with an orange circle.

### 35 This will be the way how github will be communicating with Jenkins for any change in the repo.