**Jenkins-1**

1. Install Jenkins and run Jenkins on port number 8081.

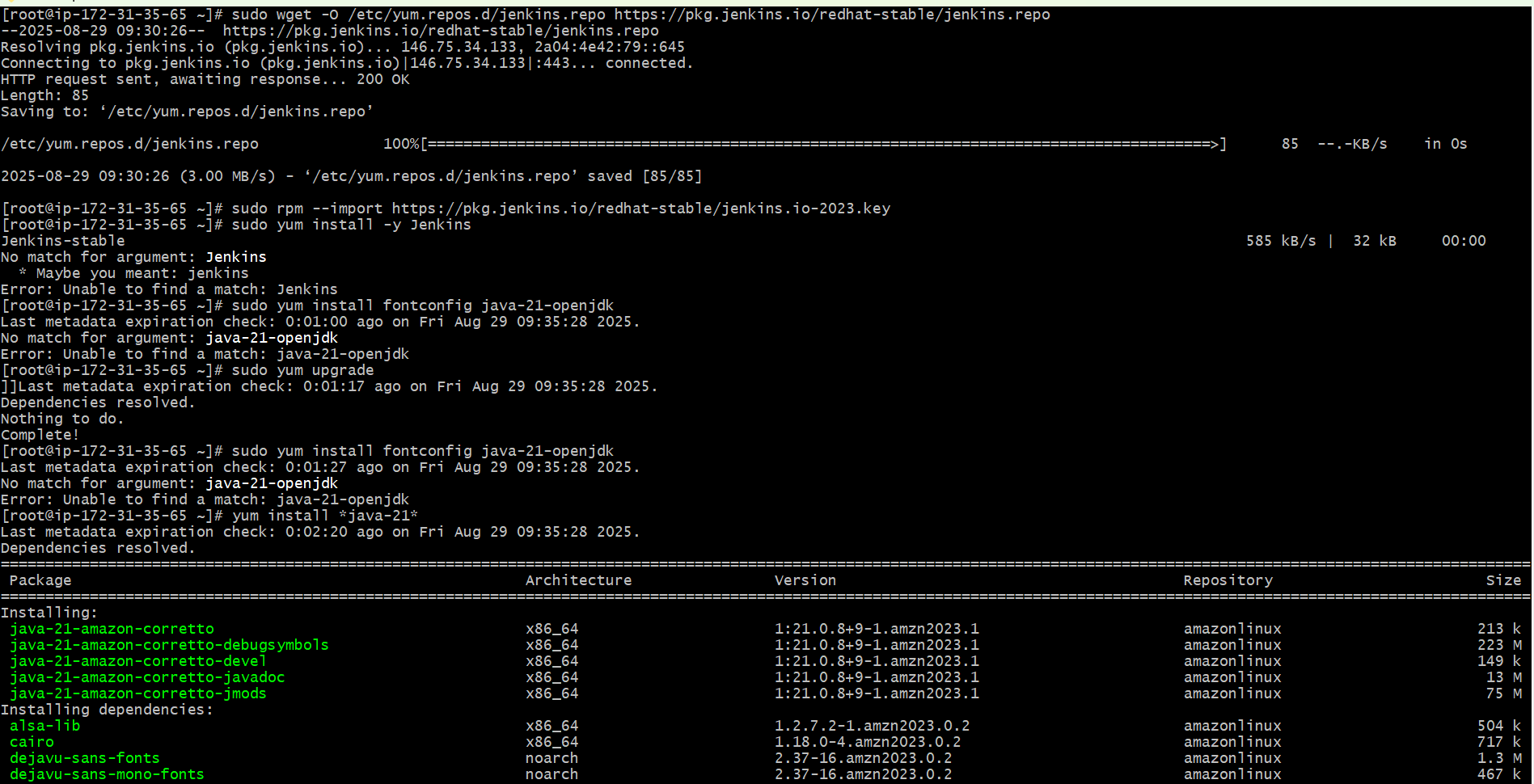
Step-1. Install Java  
(Jenkins requires Java)  
sudo dnf install -y java-17-amazon-corretto  
Verify java  
java –version  
Step-2. Add Jenkins Repository

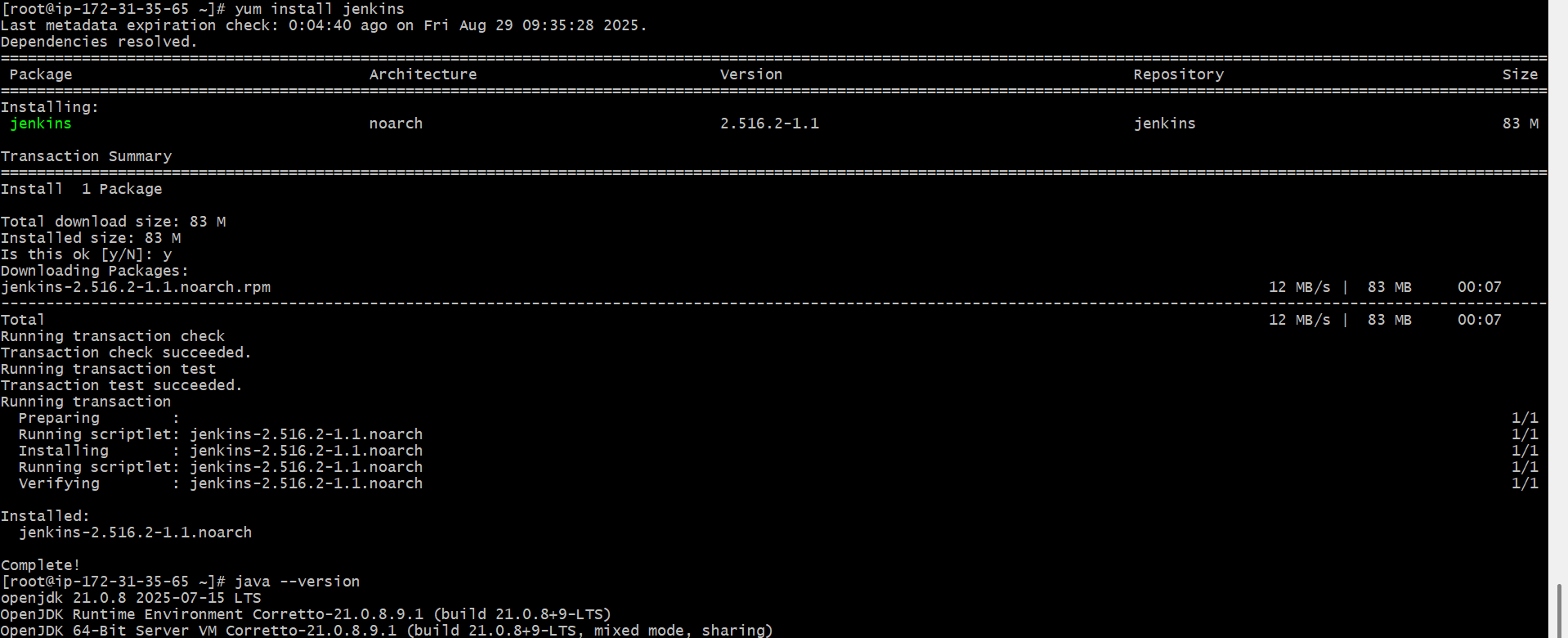
sudo wget -O /etc/yum.repos.d/jenkins.repo <https://pkg.jenkins.io/redhat-stable/jenkins.repo>

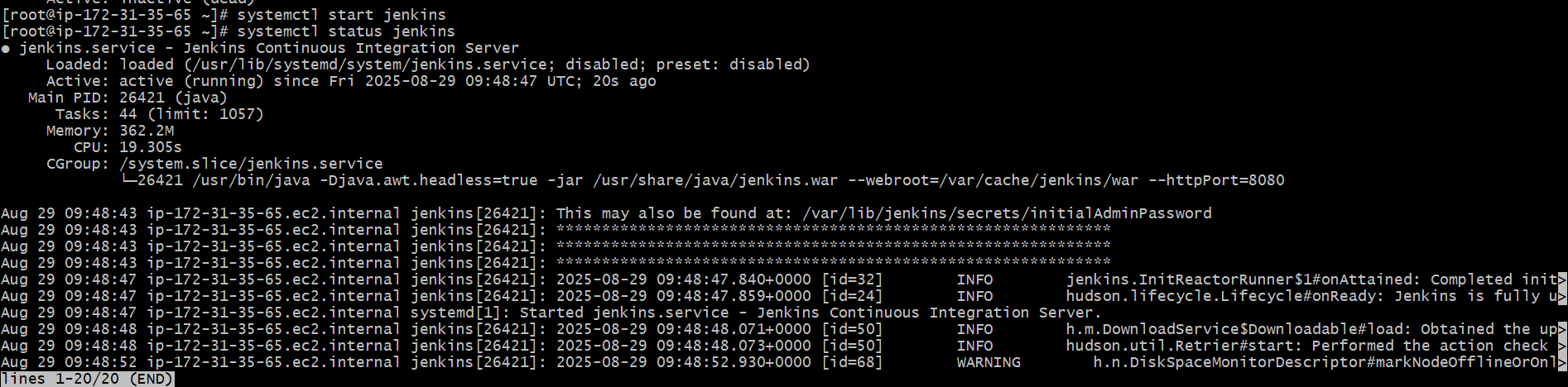
* import rpm : sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key

Install Jenkins

sudo yum install -y Jenkins  
Start and Enable Jenkins Service  
sudo systemctl start jenkins  
sudo systemctl enable Jenkins  
Check status:  
systemctl status Jenkins



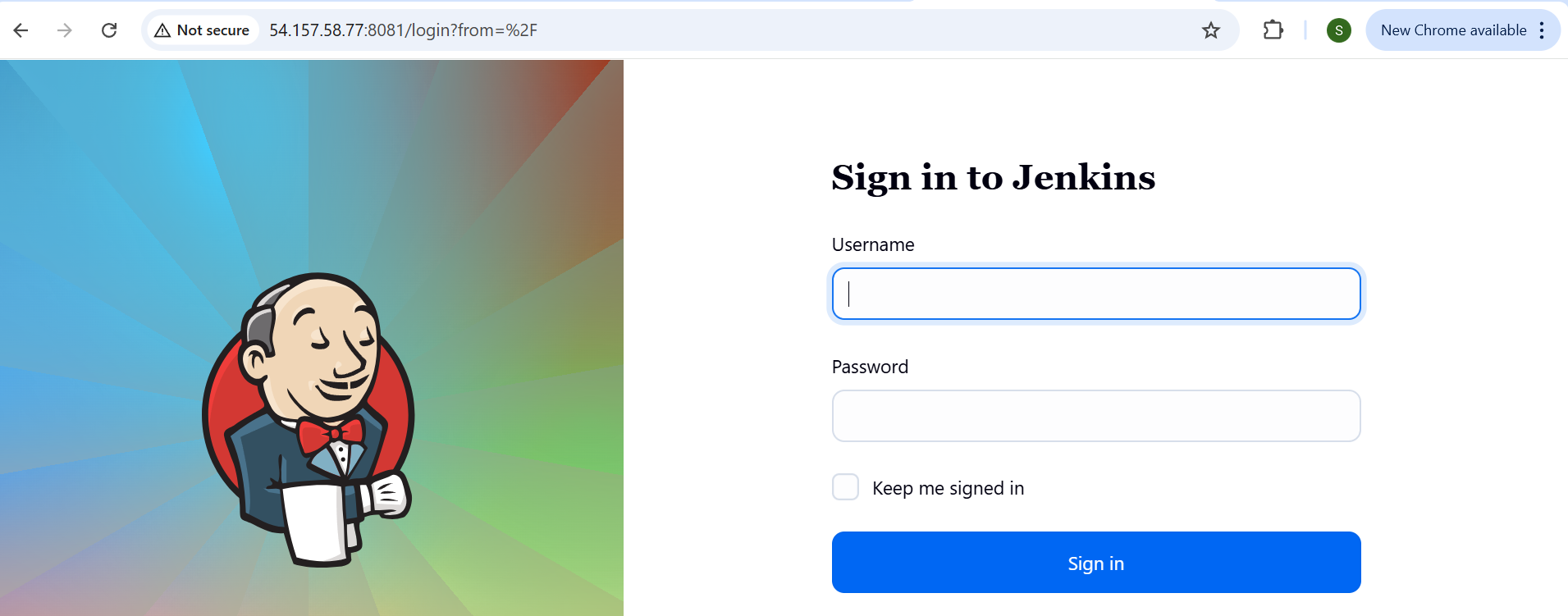




Change Jenkins Port to 8081

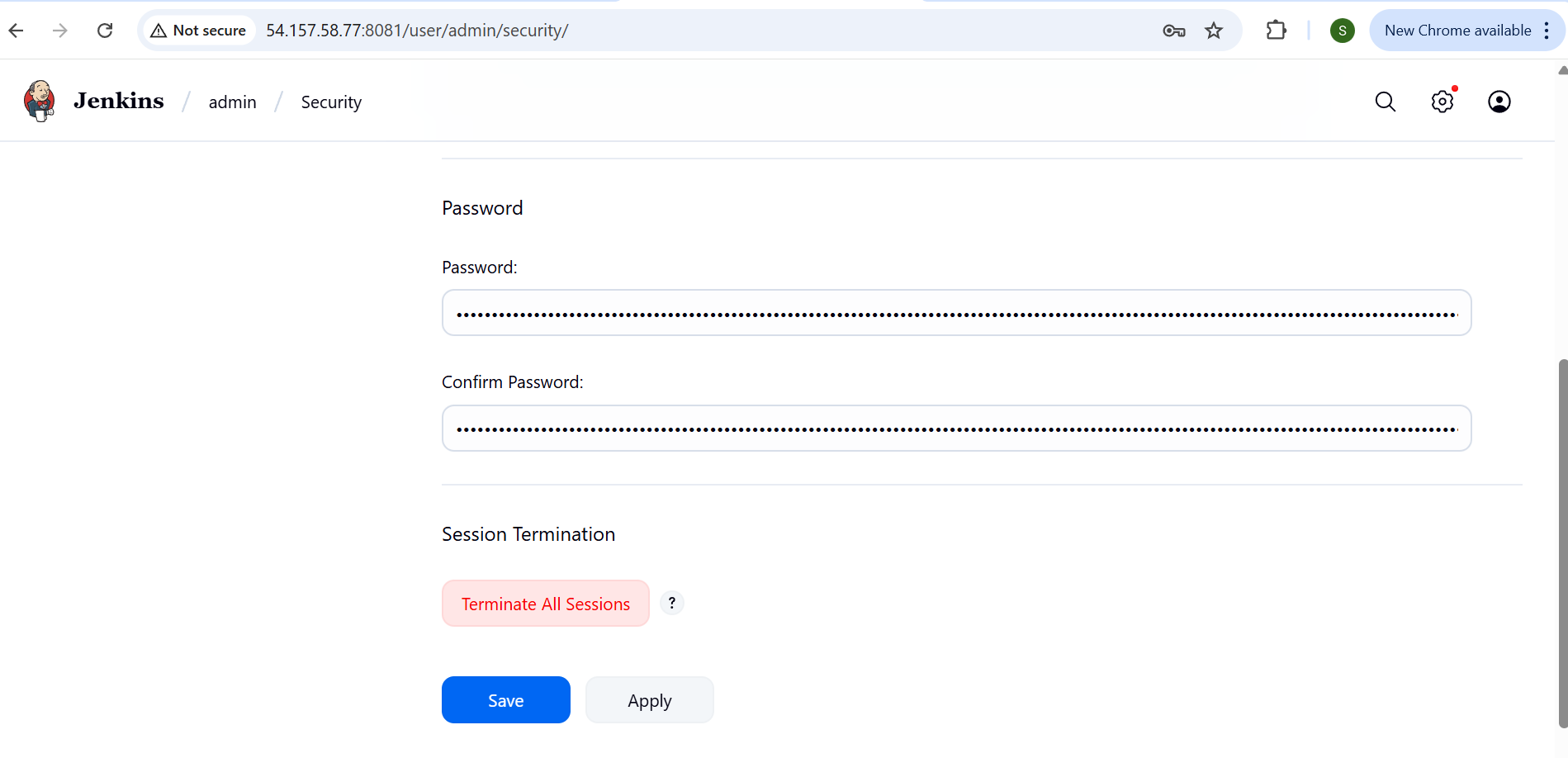
By default Jenkins runs on 8080, so we edit its systemd config:  
Now go to → sudo vi /usr/lib/systemd/system/jenkins.service  
Update → Environment="JENKINS\_PORT=8081"

sudo systemctl daemon-reexec  
sudo systemctl daemon-reload  
sudo systemctl restart Jenkins

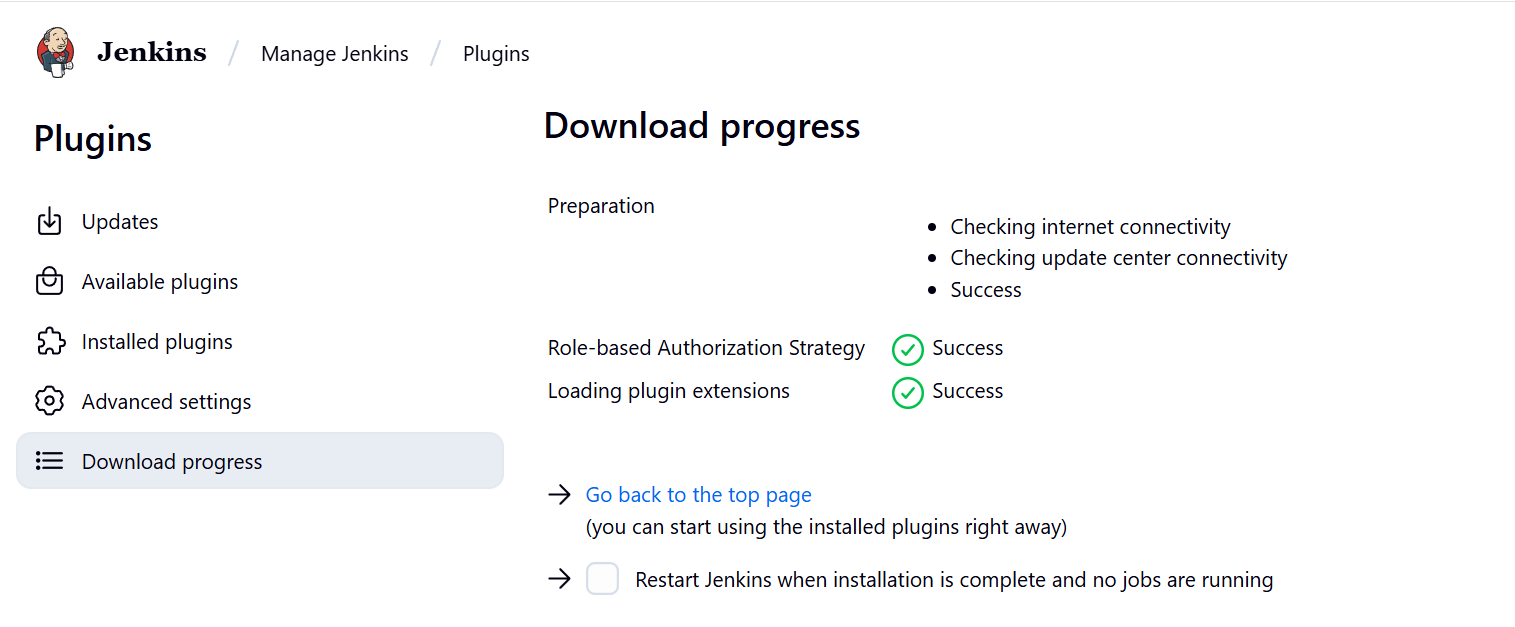


**2) Secure Jenkins server**

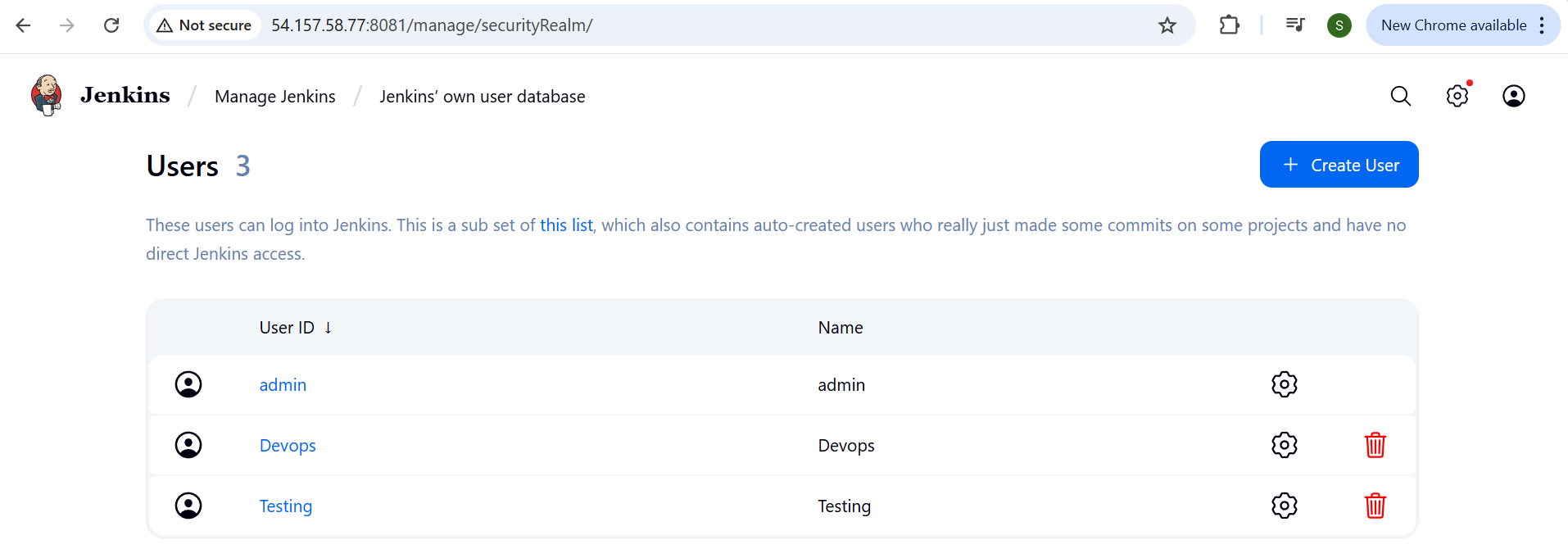
Securing Jenkins Server  
Step -1. Change Default Admin Password  
First login: Jenkins asks for the initial admin password  
/var/lib/jenkins/secrets/initialAdminPassword



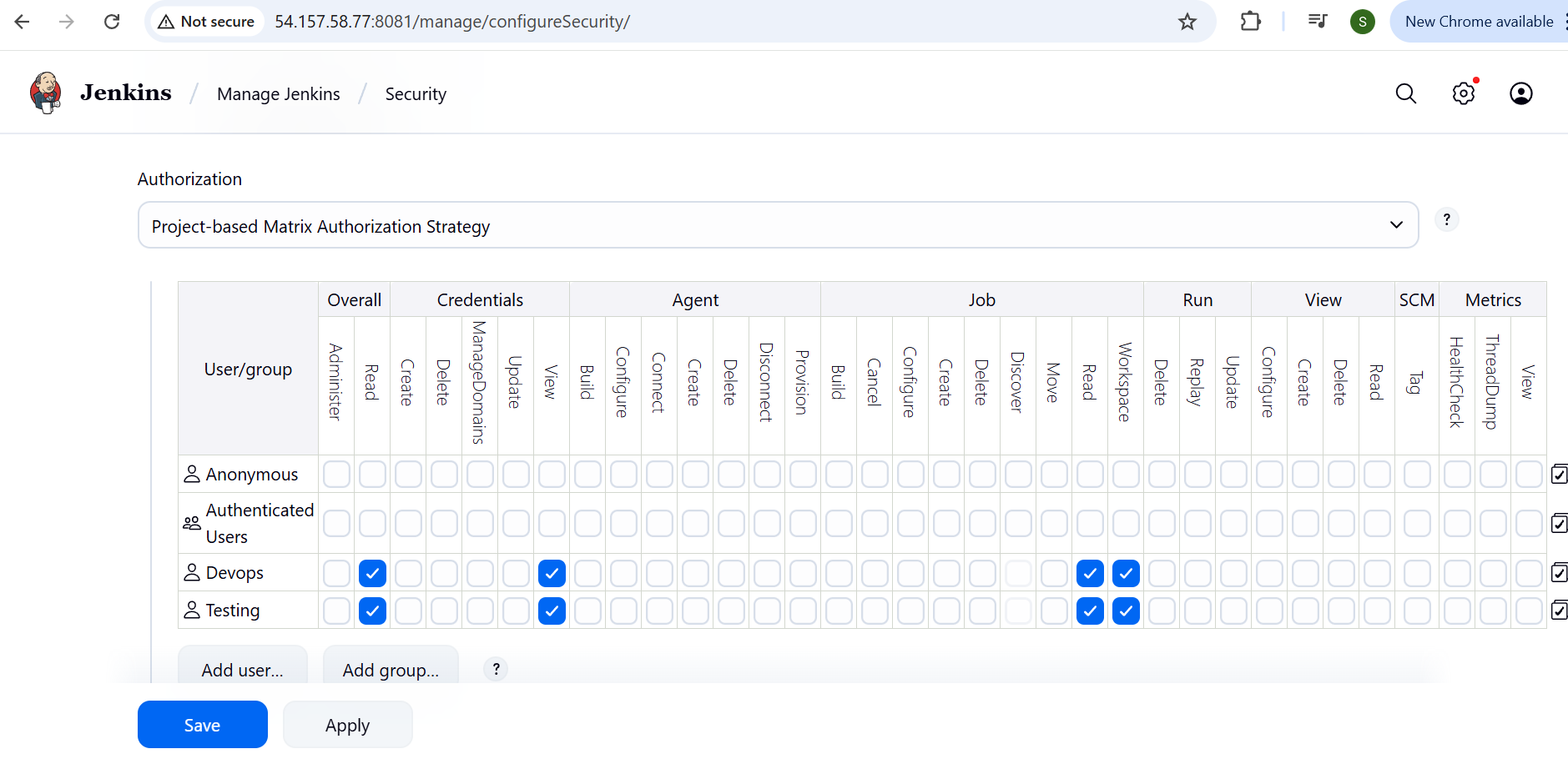
**3) Create users called DevOps, Testing in Jenkins with Limited access.**

Step 1: Install Role-Based Authorization Plugin1. Go to Manage Jenkins → Plugins → Available Plugins.  
2. Search for Role-based Authorization Strategy and install it.  
****

Steps to Create Users in Jenkins  
1. Log in to Jenkins as an administrator.  
2. From the left menu, go to:  
Manage Jenkins → Security → Manage Users.  
3. Click Create User.  
4. Enter details for each user:  
oUsername: DevOps  
oPassword: (choose a strong one)  
oFull Name: DevOps  
Repeat for user Testing

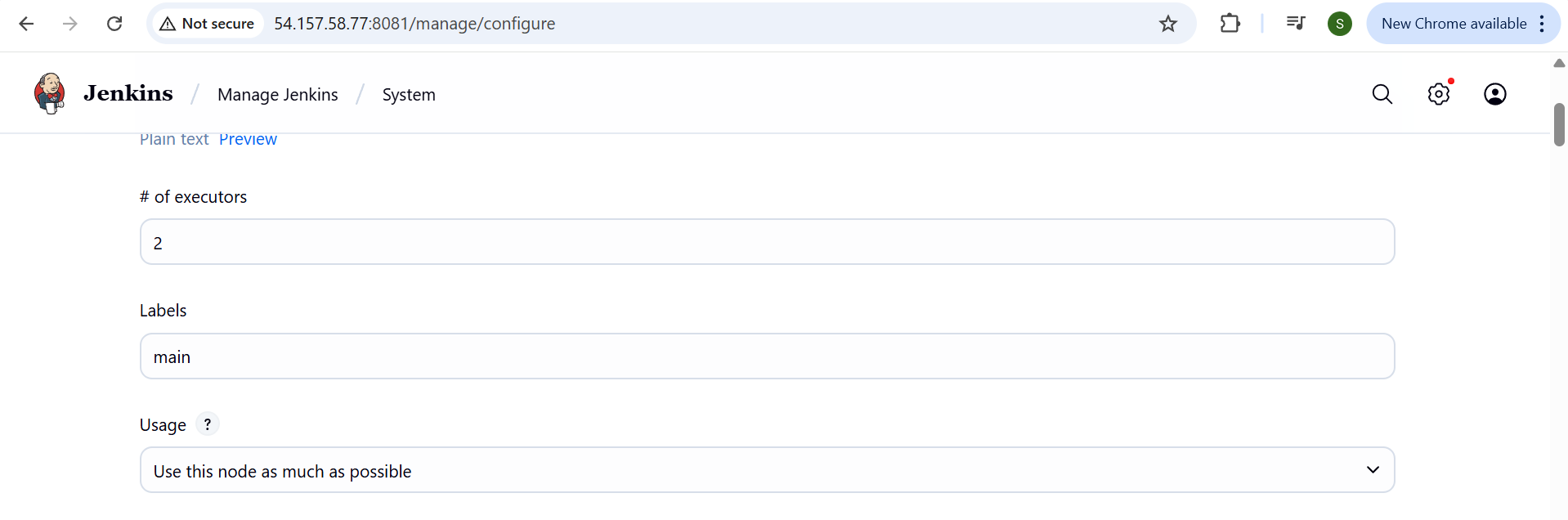
****

After creating users  
Go to  
Manage jenkins – security  
In authorization select –project based matrix  
And give the limited access

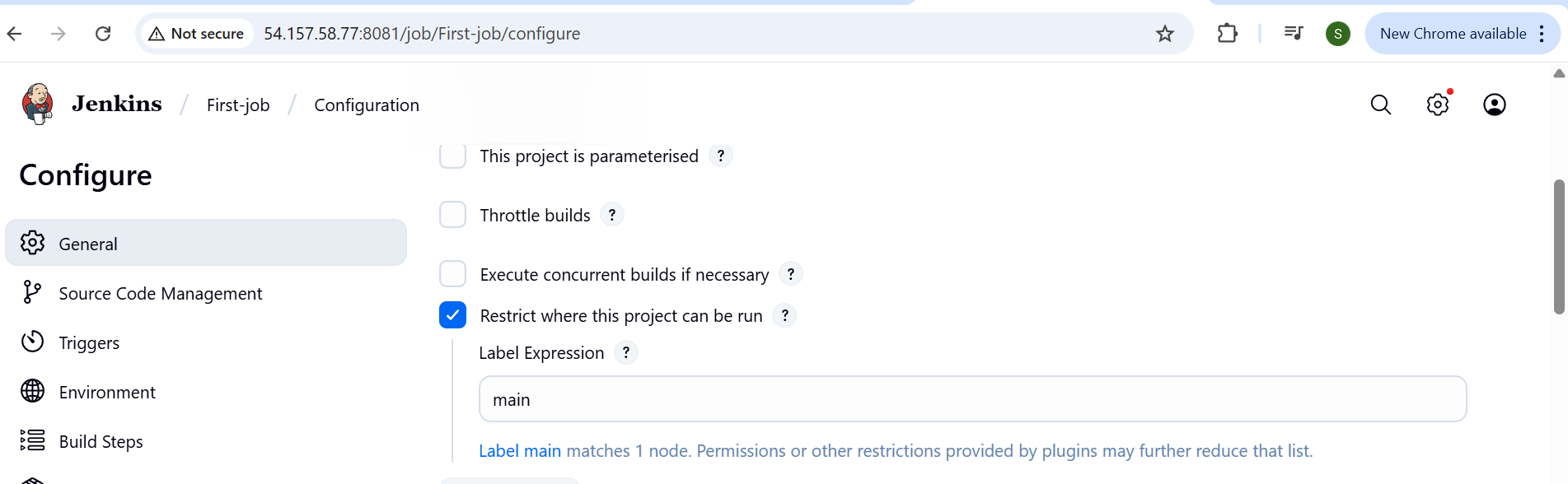
****

**4) Configure labels and restrict the jobs to execute based on label only.**

Step 1 : Assign Labels to Nodes  
Go to manage Jenkins – system  
In labels give any label name

****

Step 2: Restrict Job to Run on a Label  
1. Open the Jenkins job (Freestyle or Pipeline).  
2. Click Configure.  
3. Under General, check Restrict where this project can be run.  
4. In the text box, enter the label (e.g., Main).  
5. Save.  
Now this job will only run on nodes with that label

****

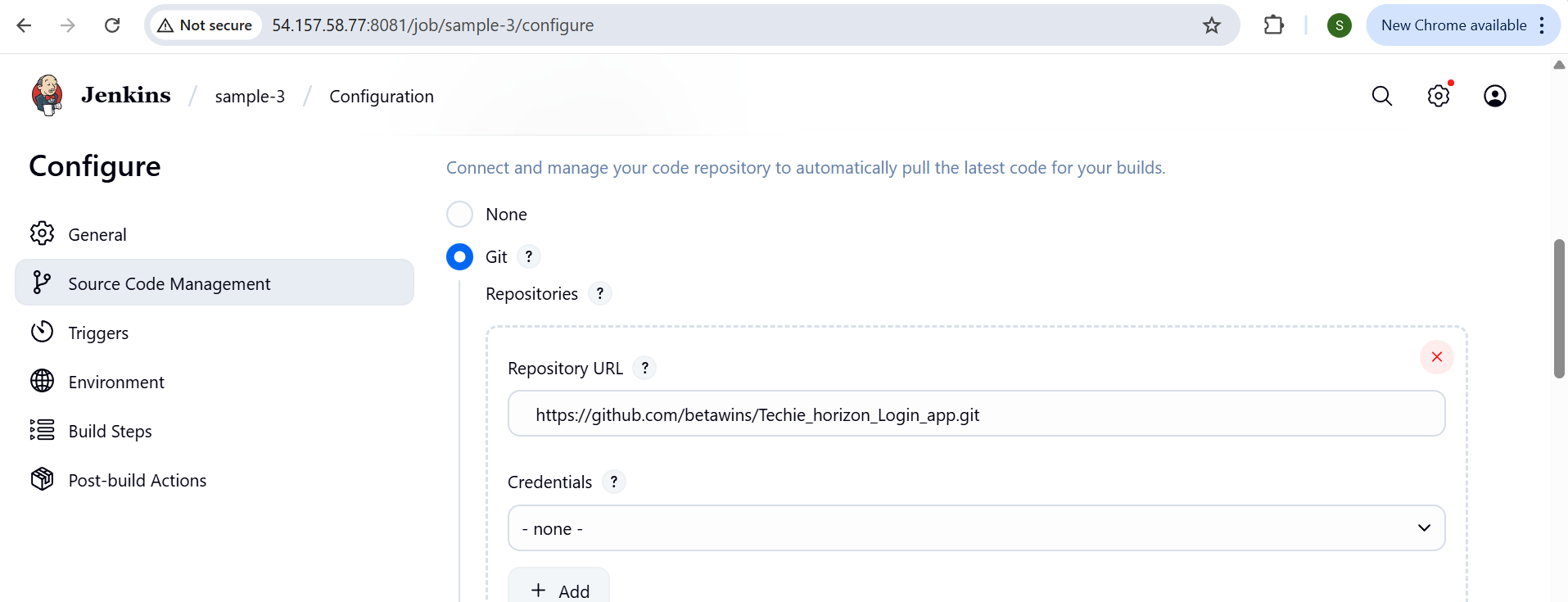
**5) Create Three sample jobs using the below URL.**[**https://github.com/betawins/Techie\_horizon\_Login\_app.git**](https://github.com/betawins/Techie_horizon_Login_app.git)

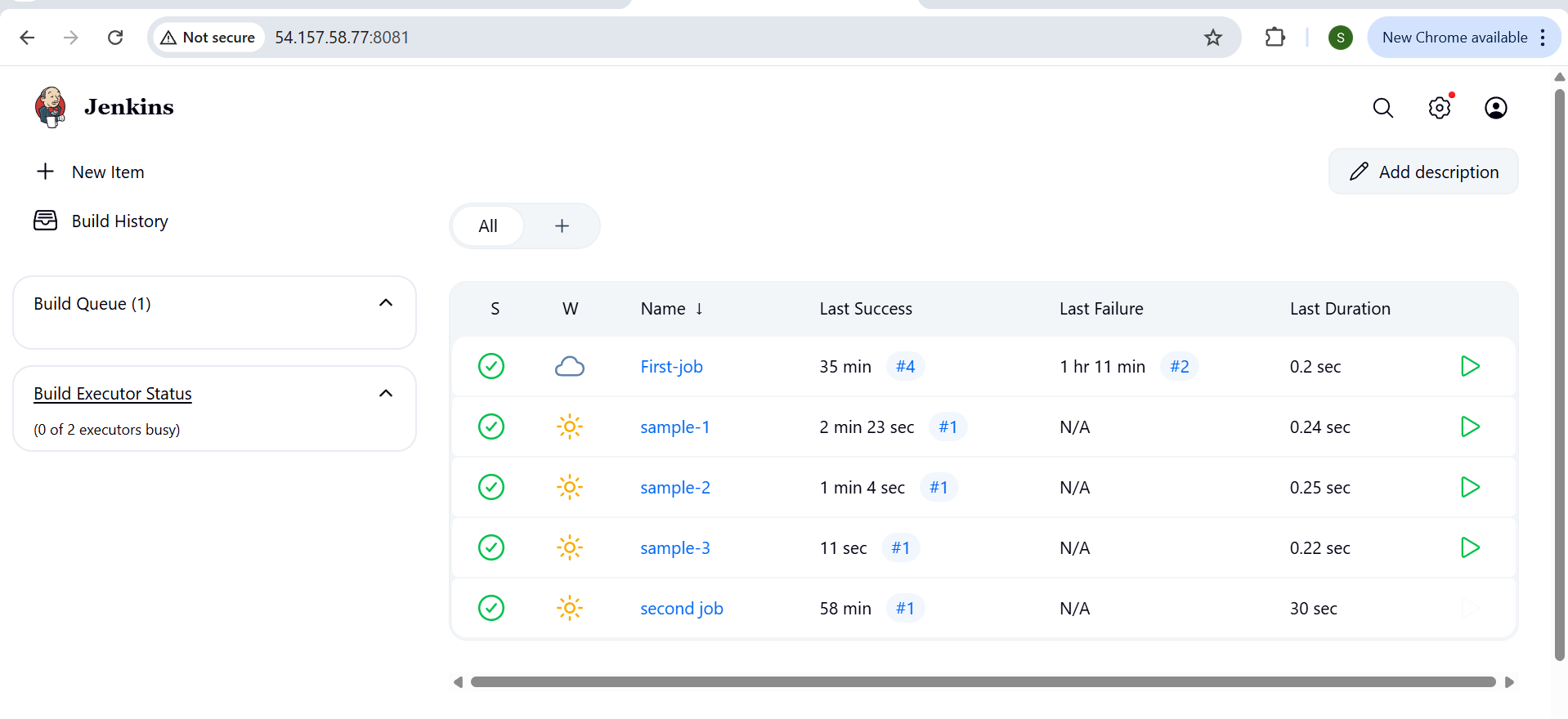
Step 1: Pre-Requisites  
Jenkins server should have Git installed  
yum install git  
(git --version).

Step 2: Create 3 Sample Jobs  
  
Go to New Item → Freestyle Project → Name: Sample-  
Job-1,2,3.  
Under Source Code Management → Git, enter repo URL:

[**https://github.com/betawins/Techie\_horizon\_Login\_app.git**](https://github.com/betawins/Techie_horizon_Login_app.git)

Save and Build

****

****

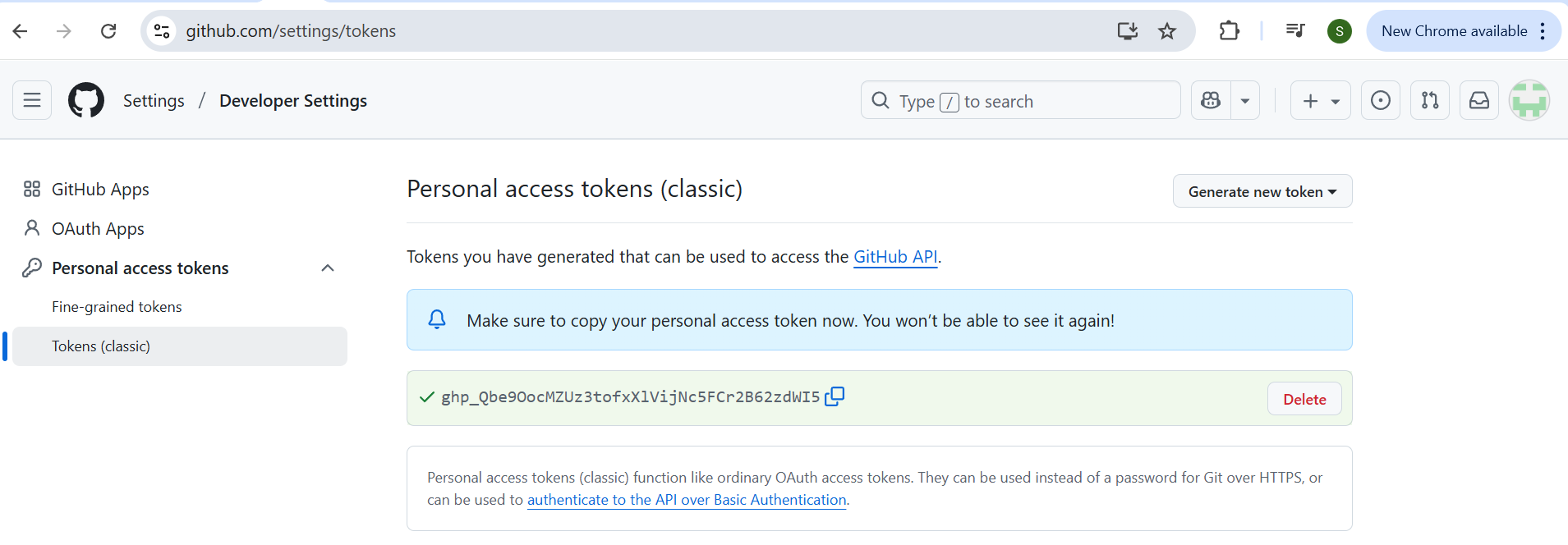
**6) Create one Jenkins job using git hub Private repository.**

**GitHubGitHub**

[**GitHub - betawins/Techie\_horizon\_Login\_app**](https://github.com/betawins/Techie_horizon_Login_app.git)

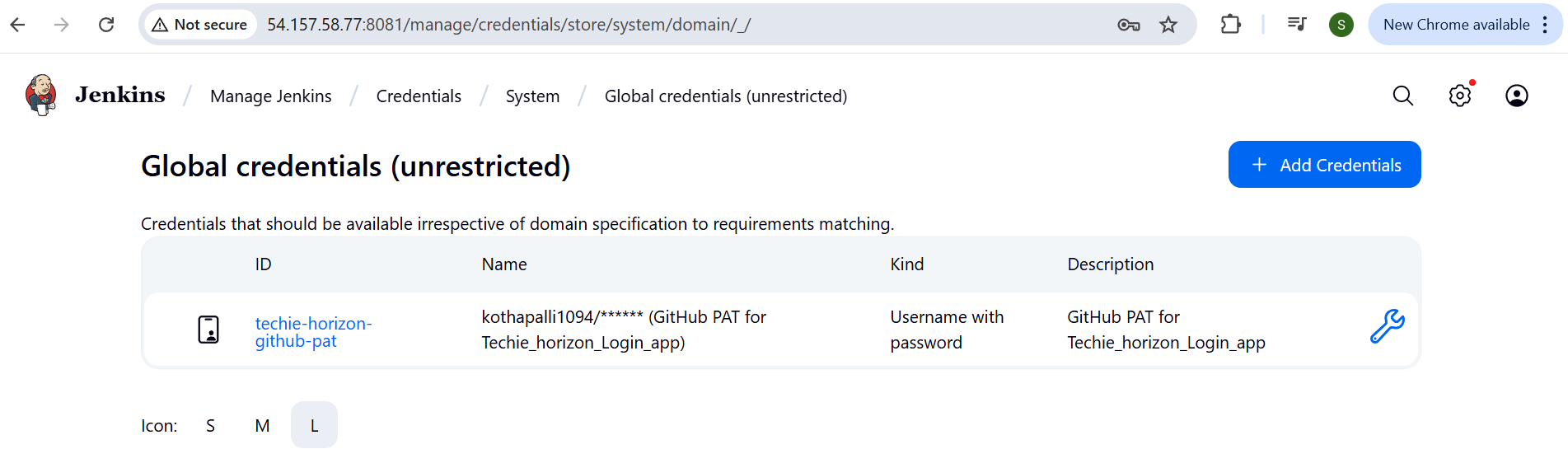
**Contribute to betawins/Techie\_horizon\_Login\_app development by creating an account on GitHub. (33 kB)**

Step 1: Generate GitHub Credentials  
1. Go to GitHub → Settings → Developer settings → Personal   
Access Tokens → Tokens (classic).  
2. Generate a token with permissions:  
3. Copy the token

****

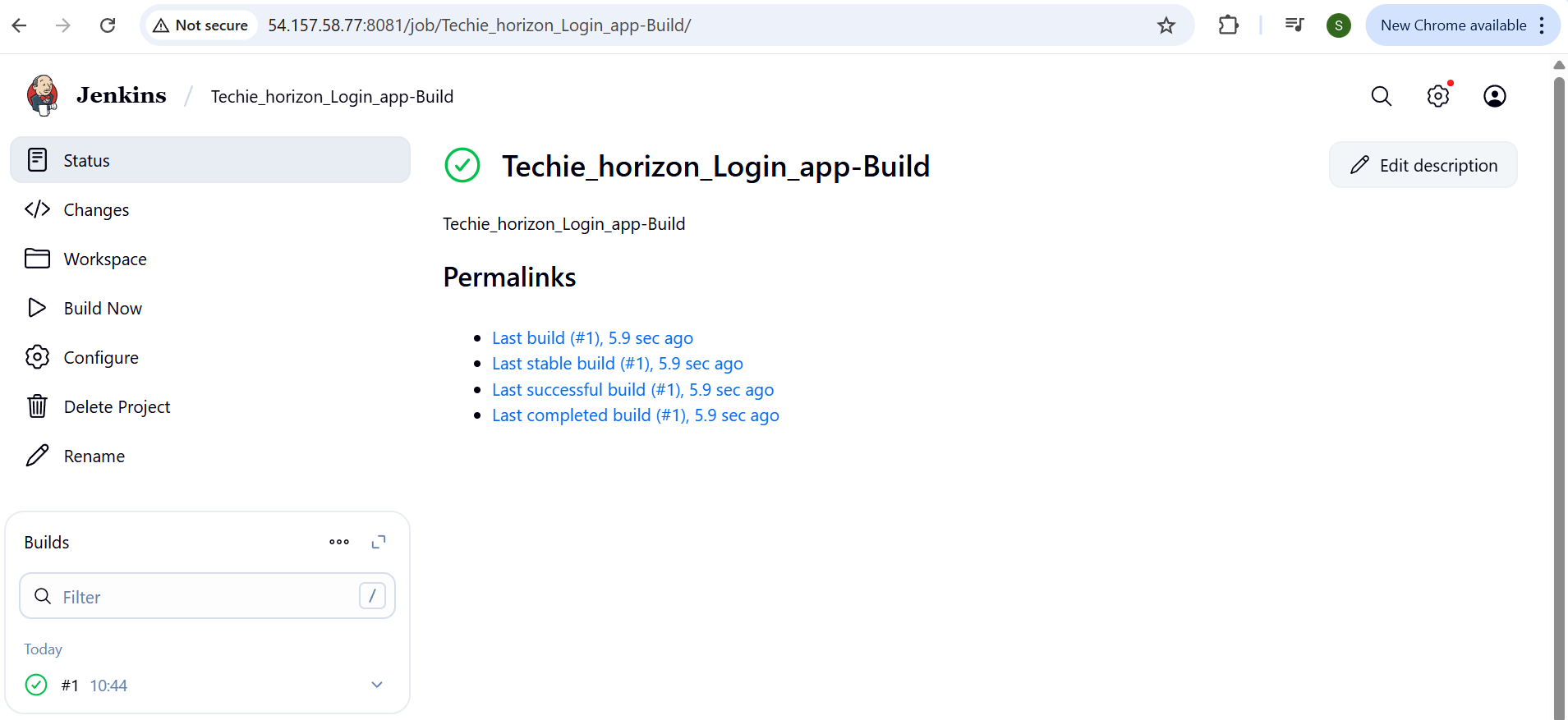
Step 2: Add Credentials to Jenkins  
1. In Jenkins → Manage Jenkins → Credentials → (Global).

2. Click Add Credentials:  
oKind: Username with password  
oUsername: GitHub username or token  
oPassword:  
oID:

****

Step 3: Create Jenkins Job  
1. New Item → Freestyle Project → Name: Private-Repo-Job.  
2. Under Source Code Management → Git:

oRepository URL:

****