

Experiment 8: Practical Exercise: Set Up a Jenkins CI Pipeline for a Maven Project, Use Ansible to Deploy Artifacts Generated by Jenkins



Step 1: Create three virtual machines on any cloud platform: one for the Jenkins server, one for the Ansible server, and one for the Ansible client server.

The screenshot shows the AWS EC2 Instances page with the following details:

- Instances (3) Info**: Shows 3 instances: ansible-server, ansible-client-server, and jenkins-server.
- Last updated**: 5 minutes ago.
- Actions**: Buttons for Connect, Instance state, Actions, and Launch instances.
- Table Headers**: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone.
- Table Data**:
 - ansible-server: i-0490ccdf290c4a560, Running, t2.micro, 2/2 checks passed, View alarms, ap-south-1b
 - ansible-client-server: i-02e8e25d43d55cb68, Running, t2.micro, 2/2 checks passed, View alarms, ap-south-1b
 - jenkins-server: i-075b9232fb736155d, Running, t2.micro, 2/2 checks passed, View alarms, ap-south-1a
- Select an instance**: A dropdown menu for selecting an instance.
- Sidebar Navigation**: Includes links for Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Snapshots, Lifecycle Manager).
- Bottom Bar**: Includes CloudShell, Feedback, Weather (23°C, Mostly cloudy), Search, and various system icons (ENG US, 02:24, 02-03-2025).

Step 2: Access all the Virtual machines

Step 2.1: Copy the Public IP of virtual machines

The screenshot shows the AWS EC2 Instances page with three running instances listed:

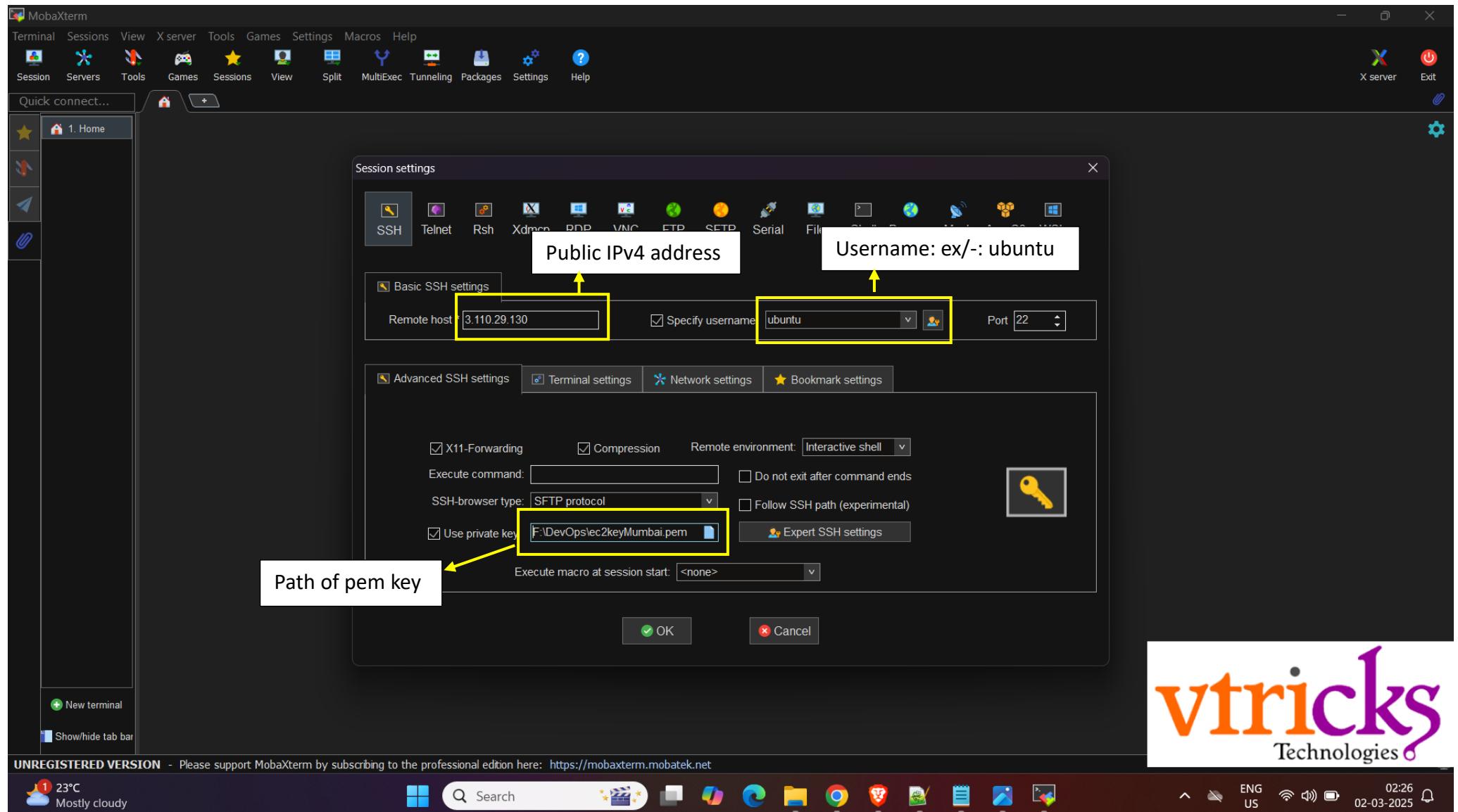
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
ansible-server	i-0490ccdf290c4a560	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b
ansible-client-server	i-02e8e25d43d55cb68	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b
jenkins-server	i-075b9232fb736155d	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a

The details for the **ansible-server** instance are expanded, showing the following information:

- Public IPv4 address:** 3.7.73.2
- Private IP4 address:** 172.31.0.58
- Public IPv4 DNS:** ec2-3-7-73-2.ap-south-1.compute.amazonaws.com
- Hostname type:** IP name: ip-172-31-0-58.ap-south-1.compute.internal
- Private IP DNS name (IPv4 only):** ip-172-31-0-58.ap-south-1.compute.internal
- Instance type:** t2.micro

At the bottom of the page, there is a watermark for **vtricks Technologies**.

Step 2.2: Open mobaxterm → click on “Session” → “Session settings” window appears → click on “SSH” and fill the fields → click “OK”



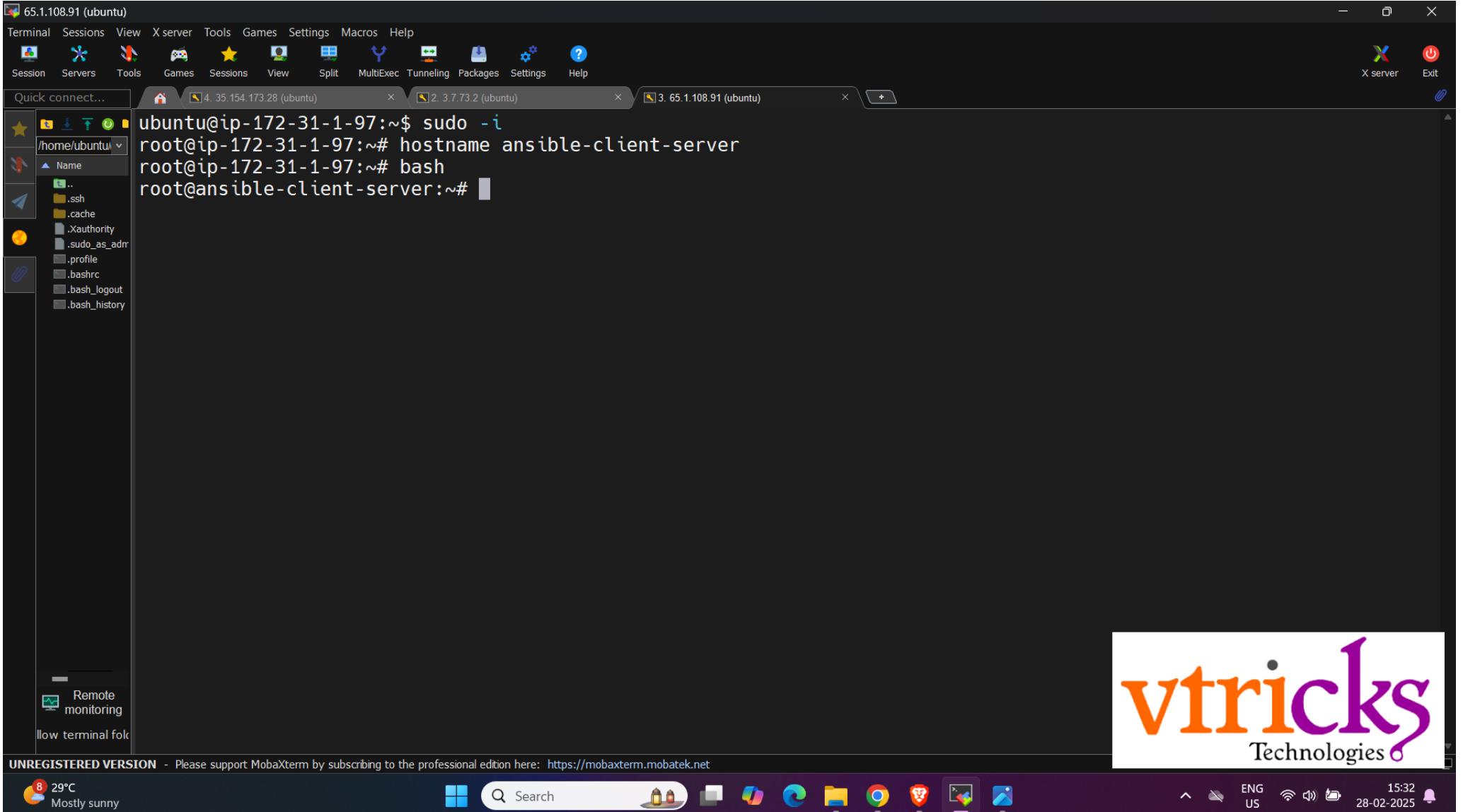
Step 3: Set the host name of controller machine to “jenkins-server” for the identification. Execute the commands as in the following image.

The screenshot shows a MobaXterm interface with multiple sessions open. The current session is titled "3.110.29.130 (ubuntu)". The terminal window displays the following command execution:

```
ubuntu@ip-172-31-45-87:~$ sudo hostname jenkins-server
ubuntu@ip-172-31-45-87:~$ bash
ubuntu@jenkins-server:~$
```

The left sidebar shows a file tree under "/home/ubuntu". The bottom status bar indicates an unregistered version and provides a link to support the software. The system tray at the bottom right shows weather information (23°C, mostly cloudy), system icons, and a date/time stamp (02-03-2025).

Step 4: Set the host name of one client machine to “ansible-client-server” for the identification. Execute the commands as in the following image.



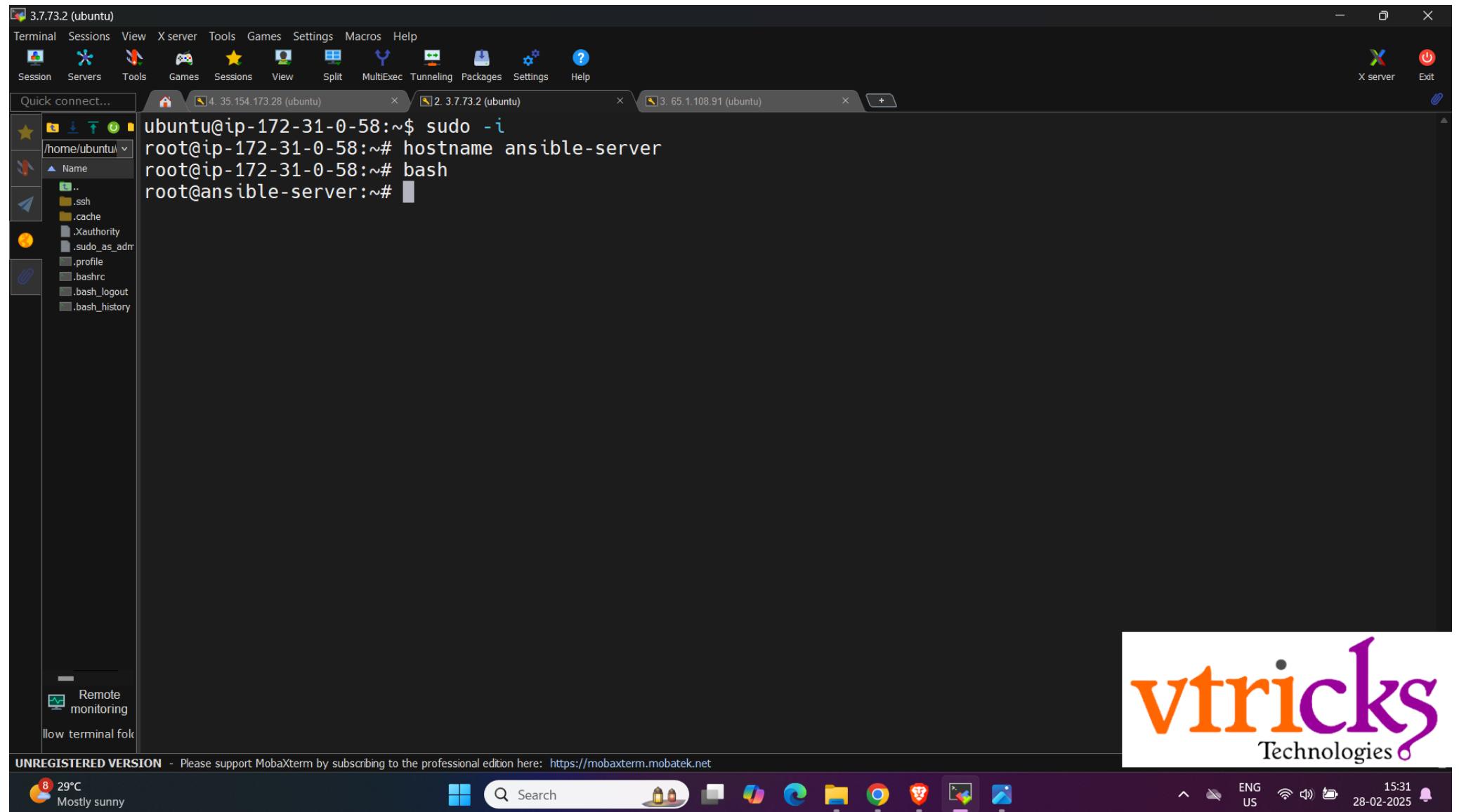
The screenshot shows a MobaXterm interface with four terminal sessions open:

- Session 1: 65.1.108.91 (ubuntu)
- Session 2: 3.7.73.2 (ubuntu)
- Session 3: 3. 65.1.108.91 (ubuntu) - This is the active session, showing a root shell. The commands entered are:

```
ubuntu@ip-172-31-1-97:~$ sudo -i
root@ip-172-31-1-97:~# hostname ansible-client-server
root@ip-172-31-1-97:~# bash
root@ansible-client-server:~#
```
- Session 4: 4. 35.154.173.28 (ubuntu)

The left sidebar shows a file tree under /home/ubuntu. A tooltip "Remote monitoring" is visible near the bottom left. The bottom status bar includes a weather icon (8°, 29°C, Mostly sunny), a search bar, and system icons for battery, signal, volume, and date/time (15:32, 28-02-2025). A watermark for "vtricks Technologies" is in the bottom right corner.

Step 5: Set the host name of one client machine to “ansible-server” for the identification. Execute the commands as in the following image.



The screenshot shows a MobaXterm interface with three terminal sessions open:

- Session 1: 3.7.73.2 (ubuntu) - Shows the user's home directory (~/.ssh, .cache, .Xauthority, .sudo_as_admin, .profile, .bashrc, .bash_logout, .bash_history).
- Session 2: 2. 3.7.73.2 (ubuntu) - The current session, showing the command history:

```
ubuntu@ip-172-31-0-58:~$ sudo -i
root@ip-172-31-0-58:~# hostname ansible-server
root@ip-172-31-0-58:~# bash
root@ansible-server:~#
```
- Session 3: 3. 65.1.108.91 (ubuntu) - Shows the command history for the third session.

At the bottom of the window, there is a status bar with the following information:

- UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>
- Weather icon: 8 29°C Mostly sunny
- Search bar: Search
- Taskbar icons: File Explorer, File Manager, Edge browser, Google Chrome, FileZilla, and others.
- System tray icons: Volume, Battery, Network, and Date/Time (15:31, 28-02-2025).

Step 6: Create a “deploy.yml” file and an “inventory” file with details as in following image

The screenshot shows a terminal session in MobaXterm with four tabs open:

- Tab 1: 3.7.73.2 (ubuntu) - Shows the directory structure in /home/ubuntu, including files like deploy.yml and inventory.
- Tab 2: 2. 3.110.29.130 (ubuntu) - Not visible in the screenshot.
- Tab 3: 4. 3.7.73.2 (ubuntu) - Displays the contents of deploy.yml:

```
ubuntu@ansible-server:~$ ls
deploy.yml  inventory
ubuntu@ansible-server:~$ cat deploy.yml
---
- name: delploy to tomcat
hosts: tomcat
gather_facts: true
become: yes
tasks:
  - name: Copy the artifact to the deployment directory
    copy:
      src: /home/ubuntu/target/R00T.war
      dest: /opt/apache-tomcat-10.1.36/webapps/
```
- Tab 4: 5. 65.1.108.91 (ubuntu) - Displays the contents of inventory:

```
ubuntu@ansible-server:~$ cat inventory
[tomcat]
65.1.108.91
```

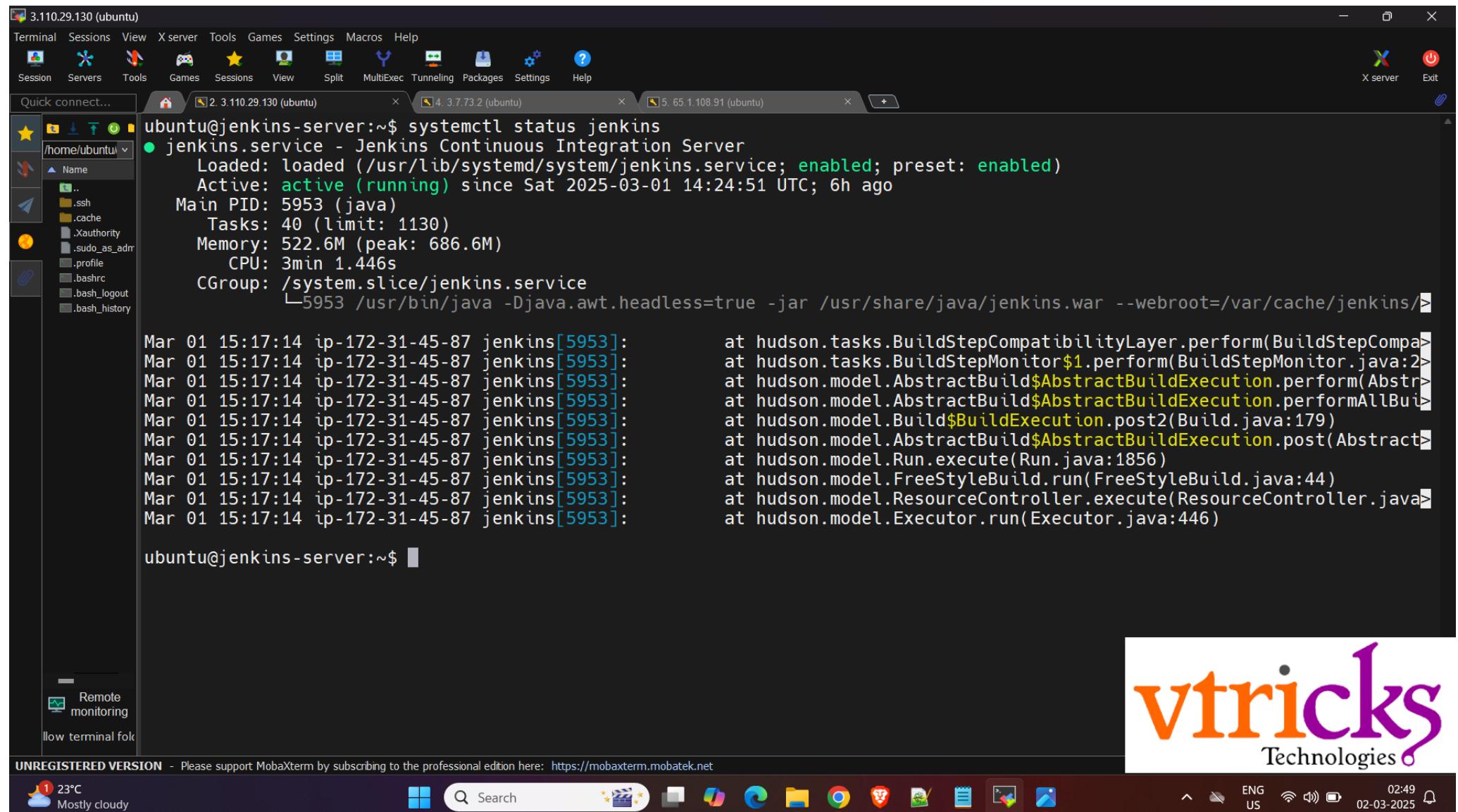
A yellow box highlights the "deploy.yml" and "inventory" files in the terminal. Another yellow box highlights the IP address "65.1.108.91". A callout bubble points to this IP address with the text "Public IP of ‘ansible-client-server’".

At the bottom of the terminal window, there is a footer bar with the following information:

- UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>
- Remote monitoring
- Show terminal log
- 23°C Mostly cloudy
- Search bar
- Minimize, Maximize, Close buttons
- Icons for File, Copy, Paste, Find, Replace, etc.

On the right side of the image, there is a watermark for "vtricks Technologies".

Step 7: Install jenkins in Jenkins-server



The screenshot shows a MobaXterm interface with four terminal sessions open:

- Session 3.110.29.130 (ubuntu) displays the command `systemctl status jenkins` output, showing the Jenkins service is active (running).
- Session 2.3.110.29.130 (ubuntu) shows Jenkins logs from March 1, 2025, at 15:17:14. The logs indicate a stack trace for a build step compatibility layer error.
- Session 4.3.7.73.2 (ubuntu) is visible in the background.
- Session 5.65.1.108.91 (ubuntu) is also visible in the background.

The left sidebar of the MobaXterm window includes icons for Session, Servers, Tools, Games, Sessions, View, Split, MultiExec, Tunneling, Packages, Settings, Help, X server, and Exit. A "Remote monitoring" icon is also present.

At the bottom of the screen, there is a taskbar with various application icons (Windows Start, Search, File Explorer, Edge, Google Chrome, Mozilla Firefox, Microsoft Edge, Notepad, FileZilla, and others). The system tray shows the date (02-03-2025), time (02:49), battery level (23°C), and connectivity status (ENG US).

Session 3.110.29.130 (ubuntu) Output:

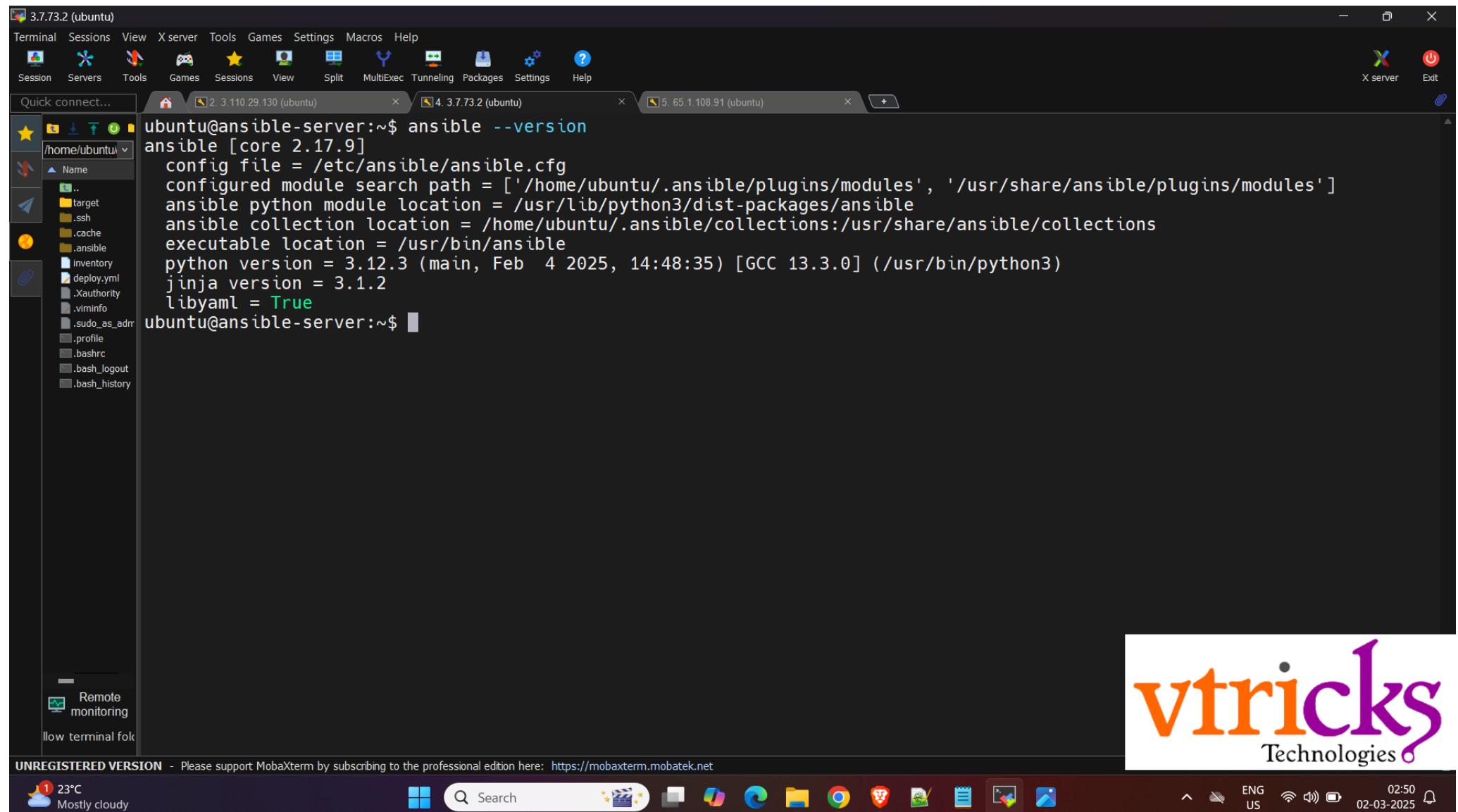
```
ubuntu@jenkins-server:~$ systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
  Active: active (running) since Sat 2025-03-01 14:24:51 UTC; 6h ago
    Main PID: 5953 (java)
      Tasks: 40 (limit: 1130)
     Memory: 522.6M (peak: 686.6M)
        CPU: 3min 1.446s
       CGroup: /system.slice/jenkins.service
               └─5953 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/...

Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.tasks.BuildStepCompatibilityLayer.perform(BuildStepCompa>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.tasks.BuildStepMonitor$1.perform(BuildStepMonitor.java:2>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.perform(Absr>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.performAllBui>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Build$BuildExecution.post2(Build.java:179)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.post(Abstract>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Run.execute(Run.java:1856)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.FreeStyleBuild.run(FreeStyleBuild.java:44)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.ResourceController.execute(ResourceController.java:>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Executor.run(Executor.java:446)

ubuntu@jenkins-server:~$
```

vtricks Technologies

Step 8: Install ansible in ansible-server.



The screenshot shows the MobaXterm interface with four terminal sessions open:

- Session 1: 3.7.73.2 (ubuntu) - Shows the output of the 'ansible --version' command, indicating version 2.17.9.
- Session 2: 2. 3.110.29.130 (ubuntu)
- Session 3: 4. 3.7.73.2 (ubuntu)
- Session 4: 5. 65.1.108.91 (ubuntu)

The left sidebar displays the file structure of the current session, which is located at /home/ubuntu/.ansible. The terminal window shows the following output:

```
ubuntu@ansible-server:~$ ansible --version
ansible [core 2.17.9]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['~/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb 4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@ansible-server:~$
```

At the bottom of the interface, there is a status bar with various icons and information, including the weather (23°C, Mostly cloudy), system tray icons, and system status (ENG US, 02:50, 02-03-2025).



Step 9: Install apache-tomcat in ansible-client-server.

The screenshot shows a MobaXterm window with four terminal sessions:

- Session 1: 65.1.108.91 (ubuntu) - Root shell
- Session 2: 2. 3.110.29.130 (ubuntu) - User shell
- Session 3: 4. 3.7.73.2 (ubuntu) - User shell
- Session 4: 5. 65.1.108.91 (ubuntu) - Root shell

The root shell in Session 4 shows the command output:

```
ubuntu@ansible-client-server:~$ sudo -i
root@ansible-client-server:~# cd /opt/
root@ansible-client-server:/opt# ls
apache-tomcat-10.1.36
root@ansible-client-server:/opt#
```

The desktop interface at the bottom includes:

- Weather icon: 23°C Mostly cloudy
- Search bar
- Taskbar icons: File Explorer, Edge, Chrome, FileZilla, Notepad, Paint, Task View
- System tray: ENG US, Wi-Fi, Battery, 02:53, 02-03-2025, Notifications

A watermark for "vtricks Technologies" is visible in the bottom right corner.

Step 10: Access Jenkins using public IP of Jenkins-server

The screenshot shows the AWS EC2 Instances page. In the left sidebar, under the 'Instances' section, the 'jenkins-server' instance is selected. The main pane displays the 'Instances (1/3)' table with one row. The table columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. The 'jenkins-server' row is highlighted with a blue border. Below the table, the specific instance details for 'i-075b9232fb736155d (jenkins-server)' are shown. The 'Details' tab is selected. Under the 'Instance summary' section, the 'Public IPv4 address' field is highlighted with a red box and contains the value '3.110.29.130'. Other fields visible include Instance ID (i-075b9232fb736155d), IPv6 address (empty), Hostname type (IP name: ip-172-31-45-87.ap-south-1.compute.internal), Instance state (Running), Private IP DNS name (ip-172-31-45-87.ap-south-1.compute.internal), and Private IPv4 addresses (172.31.45.87). The bottom of the screen shows the Windows taskbar with various pinned icons.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
ansible-server	i-0490ccdf290c4a560	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b
ansible-client-server	i-02e8e25d43d55cb68	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1b
jenkins-server	i-075b9232fb736155d	Running	t2.micro	2/2 checks passed	View alarms	ap-south-1a

i-075b9232fb736155d (jenkins-server)

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

Instance ID: i-075b9232fb736155d

IPv6 address: -

Hostname type: IP name: ip-172-31-45-87.ap-south-1.compute.internal

Public IPv4 address: 3.110.29.130 | [open address](#)

Private IP DNS name (IPv4 only): ip-172-31-45-87.ap-south-1.compute.internal

Private IPv4 addresses: 172.31.45.87

Public IPv4 DNS: ec2-3-110-29-130.ap-south-1.compute.amazonaws.com | [open address](#)

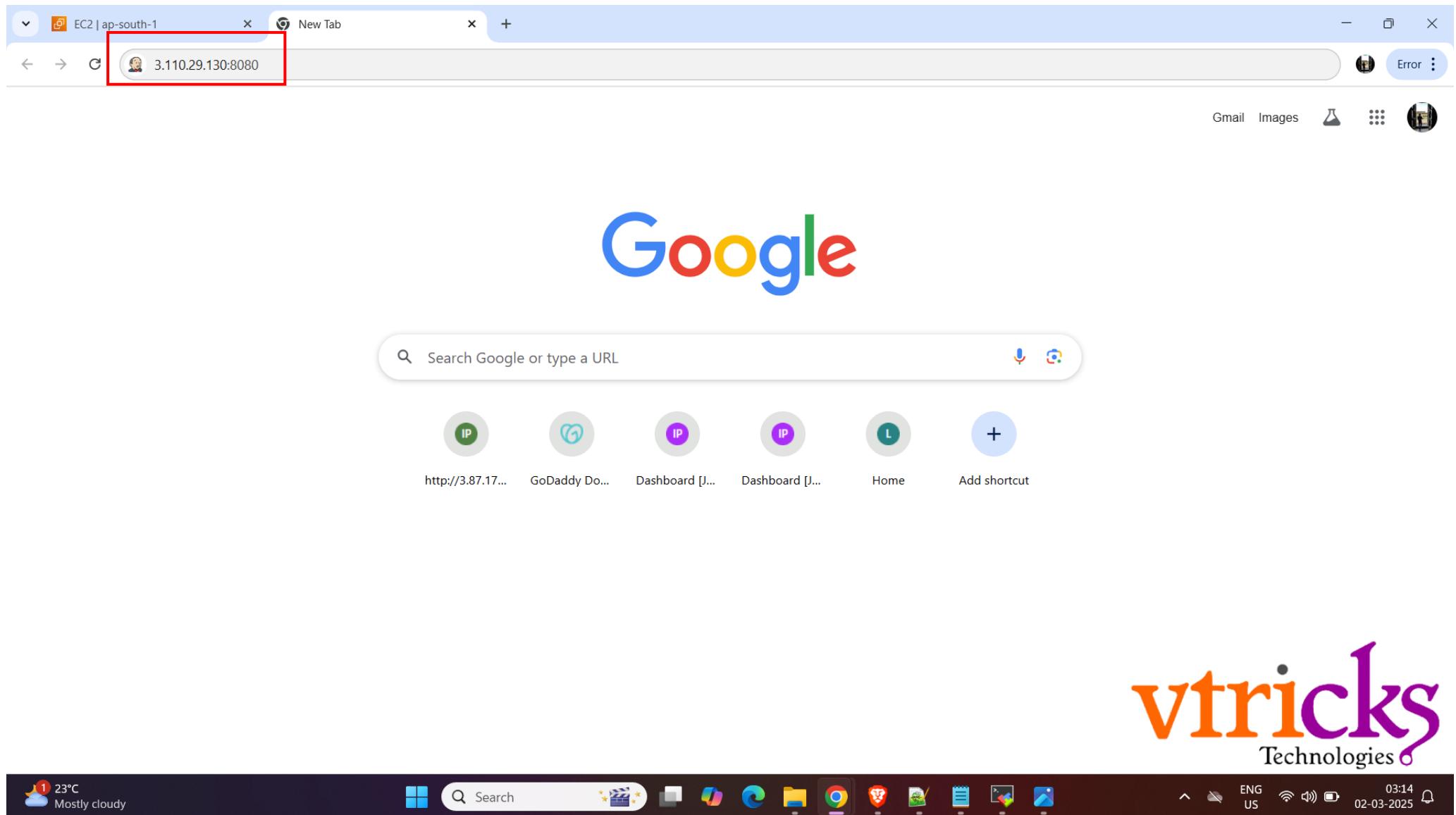
CloudShell Feedback

CloudWatch Metrics 23°C Mostly cloudy Search

ENG US 02:55 02-03-2025

vtricks Technologies

Step 10.1: Jenkins uses port 8080 by default. To access the Jenkins login page, enter <Public IP>:8080 in your browser's address bar and search.



Jenkins login page appears as follows

The screenshot shows a web browser window with the Jenkins login page. The address bar displays the URL `3.110.29.130:8080/login?from=%2F`. The page features a large, colorful background image of a cartoon character (a man in a tuxedo holding a mug) against a radial gradient background. To the right of the image is the Jenkins logo and the text "Sign in to Jenkins". Below this, there are fields for "Username" and "Password", a "Keep me signed in" checkbox, and a blue "Sign in" button. At the bottom of the screen, there is a taskbar with various icons and a system tray showing weather information (23°C, Mostly cloudy), system status (ENG US), and date/time (02-03-2025, 03:16).

EC2 | ap-south-1 Sign in [Jenkins]

Not secure 3.110.29.130:8080/login?from=%2F

Sign in to Jenkins

Username

Password

Keep me signed in

Sign in

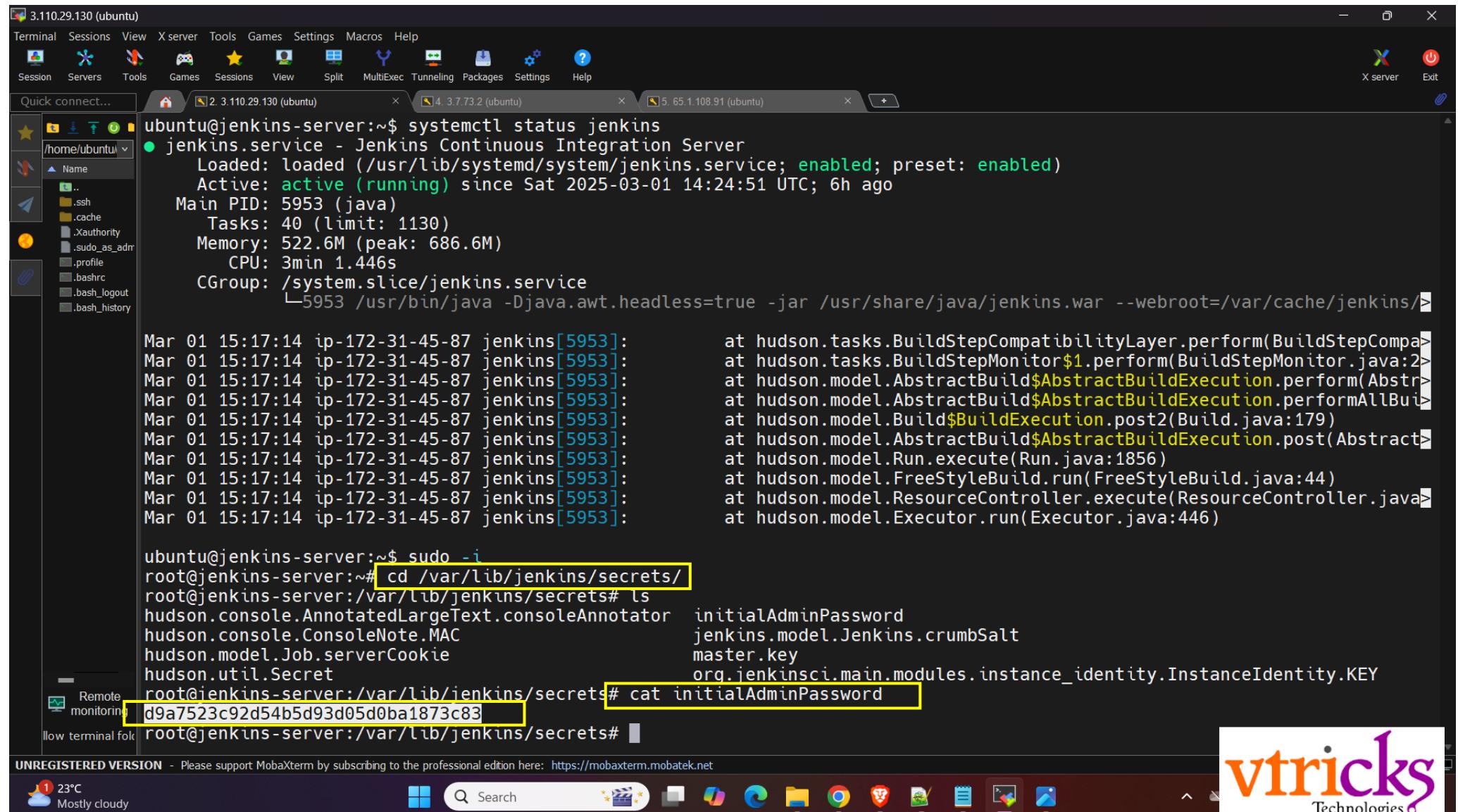
vtricks Technologies

23°C Mostly cloudy

Search

ENG US 02-03-2025 03:16

Step 10.2: Run “cd /var/lib/Jenkins/secrets/” → “cat initialAdminPassword” commands in Jenkins-server to access Jenkins password



The screenshot shows a MobaXterm interface with multiple windows. The main terminal window (3.110.29.130) displays Jenkins service status and logs. A second terminal window (4. 3.7.73.2) shows the command to change directory to Jenkins secrets. A third terminal window (5. 65.1.108.91) shows the password being extracted. The file browser sidebar shows the Jenkins configuration directory.

```
ubuntu@jenkins-server:~$ systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
  Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
  Active: active (running) since Sat 2025-03-01 14:24:51 UTC; 6h ago
    Main PID: 5953 (java)
       Tasks: 40 (limit: 1130)
      Memory: 522.6M (peak: 686.6M)
        CPU: 3min 1.446s
      CGroup: /system.slice/jenkins.service
              └─5953 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/...

Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.tasks.BuildStepCompatibilityLayer.perform(BuildStepCompa>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.tasks.BuildStepMonitor$1.perform(BuildStepMonitor.java:2>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.perform(Absr>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.performAllBui>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Build$BuildExecution.post2(Build.java:179)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.AbstractBuild$AbstractBuildExecution.post(Abstract>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Run.execute(Run.java:1856)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.FreeStyleBuild.run(FreeStyleBuild.java:44)
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.ResourceController.execute(ResourceController.java>
Mar 01 15:17:14 ip-172-31-45-87 jenkins[5953]: at hudson.model.Executor.run(Executor.java:446)

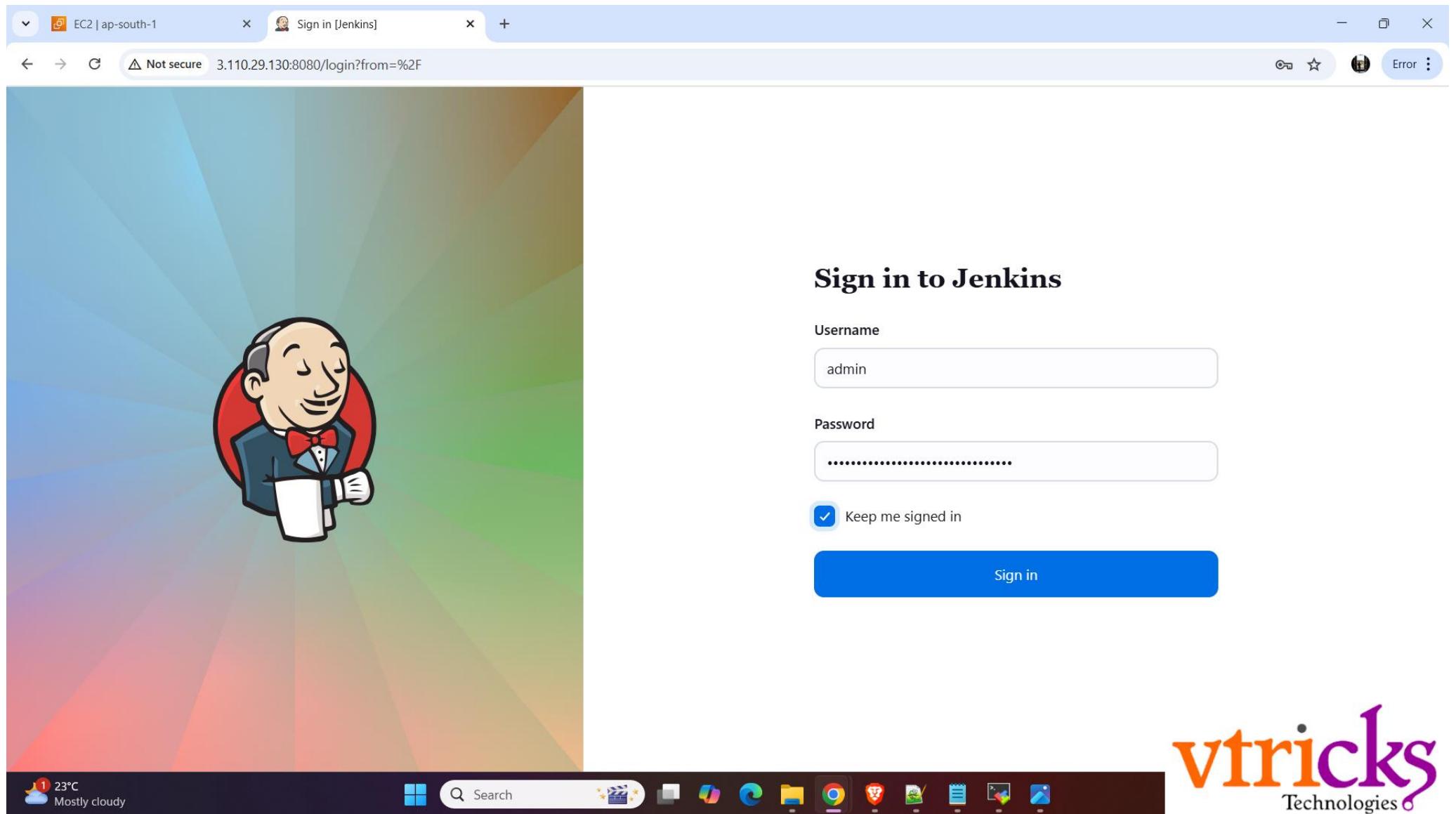
ubuntu@jenkins-server:~$ sudo -
root@jenkins-server:~# cd /var/lib/jenkins/secrets/
root@jenkins-server:/var/lib/jenkins/secrets# ls
hudson.console.AnnotatedLargeText.consoleAnnotator  initialAdminPassword
hudson.console.ConsoleNote.MAC                      jenkins.model.Jenkins.crumbSalt
hudson.model.Job.serverCookie                     master.key
hudson.util.Secret                           org.jenkinsci.main.modules.instance_identity.InstanceIdentity.KEY
root@jenkins-server:/var/lib/jenkins/secrets# cat initialAdminPassword
d9a7523c92d54b5d93d05d0ba1873c83
root@jenkins-server:/var/lib/jenkins/secrets#
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

23°C Mostly cloudy

vtricks Technologies

Step 10.3: Enter username and password to sign in to jenkins



Step 11: Jenkins dashboard looks as follows, click on “Manage Jenkins”

The screenshot shows the Jenkins dashboard interface. At the top, there is a header bar with tabs for "EC2 | ap-south-1" and "Dashboard [Jenkins]". The URL in the address bar is "3.110.29.130:8080". On the right side of the header, there are icons for search, notifications (with 1 notification), user "admin", and "log out". Below the header, the Jenkins logo is displayed, followed by the word "Jenkins" and a "Dashboard" link.

On the left side, there is a sidebar with the following links:

- + New Item
- Build History
- Manage Jenkins** (this link is highlighted with a red box)
- My Views

Below the sidebar, there are two main sections:

- Welcome to Jenkins!**: A message stating "This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project." It includes a "Start building your software project" button and a "Create a job" button with a plus sign.
- Set up a distributed build**: This section contains three items: "Set up an agent" (with a monitor icon), "Configure a cloud" (with a cloud icon), and "Learn more about distributed builds" (with a question mark icon).

At the bottom of the screen, there is a dark footer bar with various system icons, including weather information (23°C, Mostly cloudy), a search bar, and several application icons (Windows, File Explorer, Edge, Google Chrome, Firefox, etc.). On the far right of the footer, there is a watermark for "vtricks Technologies".

Step 12: Click on “Plugins”

The screenshot shows the Jenkins Manage Jenkins interface. On the left sidebar, the 'Manage Jenkins' option is selected. The main content area is titled 'Manage Jenkins' and contains several configuration sections: System Configuration (System, Tools, Nodes, Clouds), Security (Security, Credentials, Users), and Appearance. The 'Plugins' section, which is described as adding, removing, disabling or enabling plugins to extend Jenkins functionality, is highlighted with a red box.

Manage Jenkins

System Configuration

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers.
- Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
- Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.

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.

Appearance

- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
- Credential Providers**: Configure the credential providers and types.

Build Queue: No builds in the queue.

Build Executor Status: 0/2

Weather: 23°C Mostly cloudy

EC2 | ap-south-1 Manage Jenkins [Jenkins] Not secure 3.110.29.130:8080/manage/ admin log out

Step 13: click on “Available plugins” → search for “Publish Over SSH” and select → click on install

The screenshot shows the Jenkins plugin manager interface. On the left, a sidebar menu includes 'Updates', 'Available plugins' (which is highlighted with a red box), 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area displays a search bar with 'ssh' and a list of available Jenkins plugins. A red box highlights the 'Install' button for the 'Publish Over SSH' plugin. The 'Publish Over SSH' plugin is selected with a checked checkbox. Other visible plugins include 'SSH server', 'SSH Agent', 'SSH Pipeline Steps', 'SSH2 Easy', and 'SCP publisher'. The 'Publish Over SSH' plugin has a note: 'This plugin is up for adoption! We are looking for new maintainers. Visit our [Adopt a Plugin](#) initiative for more information.'

Plugin	Description	Last Updated
JSch - Library plugin (for use by other plugins)	Jenkins plugin that brings the JSch library as a plugin dependency, and provides an SSHAuthenticatorFactory for using JSch with the ssh-credentials plugin.	4 days 4 hr ago
SSH server	Adds SSH server functionality to Jenkins, exposing CLI commands through it.	19 days ago
SSH Agent	This plugin allows you to provide SSH credentials to builds via a ssh-agent in Jenkins.	4 days 4 hr ago
Publish Over SSH	Send build artifacts over SSH	23 days ago
SSH Pipeline Steps	Jenkins pipeline steps which provides SSH facilities such as command execution or file transfer for continuous delivery.	8 days 9 hr ago
SSH2 Easy	This plugin allows you to ssh2 remote server to execute linux commands, shell, sftp upload, download, etc.	1 yr 6 mo ago
SCP publisher		

Verify the installation of plugin

The screenshot shows the Jenkins 'Plugins' page under the 'Manage Jenkins' section. A red box highlights a group of plugins that have been successfully installed. The list includes:

Plugin	Status
Matrix Authorization Strategy	Success
PAM Authentication	Success
LDAP	Success
Email Extension	Success
Mailer	Success
Theme Manager	Success
Dark Theme	Success
Loading plugin extensions	Success
JavaMail API	Success
Oracle Java SE Development Kit Installer	Success
SSH server	Success
Command Agent Launcher	Success
Infrastructure plugin for Publish Over X	Success
JSch dependency	Success
Publish Over SSH	Success
Loading plugin extensions	Success

Below the table, there are two links:

- [Go back to the top page](#)
(you can start using the installed plugins right away)
- Restart Jenkins when installation is complete and no jobs are running



Step 14: Again, go to “Manage Jenkins” window → click on “System”

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there is a sidebar with various links: New Item, Build History, Manage Jenkins (which is highlighted), My Views, Build Queue (showing 'No builds in the queue.'), and Build Executor Status (showing '0/2'). Below the sidebar, there is a weather widget indicating '23°C Mostly cloudy'.

The main content area is titled 'Manage Jenkins' and is divided into sections:

- System Configuration**:
 - System**: Configure global settings and paths.
 - Nodes**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.
 - Tools**: Configure tools, their locations and automatic installers.
 - Clouds**: Add, remove, and configure cloud instances to provision agents on-demand.
 - Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins.
 - Appearance**: Configure the look and feel of Jenkins.
- Security**:
 - Security**: Secure Jenkins; define who is allowed to access/use the system.
 - Credentials**: Configure credentials.
 - Credential Providers**: Configure the credential providers and types.
 - Users**: Create/delete/modify users that can log in to this Jenkins.

At the bottom of the page, there is a navigation bar with icons for Search, Clapboard, File, Folder, Chrome, Firefox, and others.

System window appears as follows

The screenshot shows the Jenkins System configuration page. At the top, the browser title bar displays "EC2 | ap-south-1" and "System [Jenkins]". The URL in the address bar is "3.110.29.130:8080/manage/configure". The Jenkins logo is on the left, and the user "admin" is logged in. The main content area has a black header with "Jenkins" and a search bar. Below the header, the breadcrumb navigation shows "Dashboard > Manage Jenkins > System >". The main section is titled "System". It includes a "Home directory" field set to "/var/lib/jenkins". A "System Message" input field is empty. Below it, a note says "This message will be displayed at the top of the Jenkins main page. This can be useful for posting notifications to your users". There is a "Plain text" link and a "Preview" button. Under "Labels", there is a text input with the value "2". At the bottom, there are "Save" and "Apply" buttons. The footer of the browser window shows a weather icon for "23°C Mostly cloudy", a search bar, and various application icons.

Scroll down in this page

Step 15: Under “Publish over SSH” followed by under “Jenkins SSH Key” enter pem key of Jenkins-server machine in “Key” section → under “SSH server” click on “Add”

The screenshot shows the Jenkins System configuration page. The 'Publish over SSH' section is highlighted with a red box. Below it, the 'Jenkins SSH Key' section is also highlighted with a red box. A large red box highlights the 'Key' input field where a long RSA private key has been pasted. At the bottom left, the 'SSH Servers' section is highlighted with a red box, and the 'Add' button is also highlighted with a red box.

Publish over SSH

A configuration to use to connect to a SSH server

Jenkins SSH Key ?

Passphrase ?

Path to key ?

Key ?

```
smYAmiZwKG66ZLeM2JNwNV4CJEq2PGu20LSOj6eSwgClfoadsAMgBmRG2dEr1zHy  
hkX7Fl2lxNQfl4vmaMaZxrEtIG9xr86lqctmhL/q6XH4c+ykk8kCuGAQpO8+za9p  
y1gllwKBgQC2ztKMFnQcy3XEM7CIWc4Kf8dZEfxK9S+k0zOXrfWoot0Z1A5et25Se  
rB0VfmxtsBwlxklQHeXGsOPBJrJxhJmZRJPCikGHB7CA8eul0KhkAVFsPMJ3xG/F  
GVfenzDwDD8KJl9Y0vqPHTTQd9tgXesNm3lyEVMY1HNPQTodXgvTsw==  
-----END RSA PRIVATE KEY-----
```

Disable exec ?

SSH Servers

Add

Save Apply

23°C Mostly cloudy

Search

vtricks Technologies

Step 16: Provide details as in the following image and click on “Advanced”

The screenshot shows the Jenkins System configuration page under Manage Jenkins > System. The 'SSH Servers' section is displayed, specifically the 'SSH Server' configuration for an 'ansible-server'. The fields shown are:

- Name: ansible-server
- Hostname: 3.73.2
- Username: ubuntu
- Remote Directory: /home/ubuntu

A red box highlights the 'Name' field, and another red box highlights the 'Hostname' field. A red arrow points from the 'Hostname' field to a callout box labeled "Public IP of ansible-server". A third red box highlights the "Advanced" button at the bottom left. Below the configuration are "Save" and "Apply" buttons.

Step 17: Select “use password authentication, or use a different key” → provide ansible-server’s pem key in the “key” section → click on “Apply” and “Save”

The screenshot shows the Jenkins System configuration page. The 'Key' field, which contains an RSA private key, is highlighted with a red box. The key itself is a long string of characters starting with "smYAmiZwKG66ZLeM2JNwNV4CJEq2PGu20LSOj6eSwgClfoadsAMgBmRG2dEr1zHy". Below the 'Key' field, there is a 'Jump host' field with a red box around it. At the bottom of the page, there are 'Save' and 'Apply' buttons.

Step 18: Again, go to “Manage Jenkins” window → click on “Tools”

The screenshot shows the Jenkins Manage Jenkins interface. On the left, there's a sidebar with links like 'New Item', 'Build History', 'Manage Jenkins' (which is highlighted), and 'My Views'. Below this are sections for 'Build Queue' (empty) and 'Build Executor Status' (0/2). The main content area is titled 'Manage Jenkins' and is divided into two main sections: 'System Configuration' and 'Security'. Under 'System Configuration', there are four items: 'System' (Configure global settings and paths), 'Nodes' (Add, remove, control and monitor the various nodes that Jenkins runs jobs on), 'Tools' (Configure tools, their locations and automatic installers), and 'Clouds' (Add, remove, and configure cloud instances to provision agents on-demand). Under 'Security', there are three items: 'Security' (Secure Jenkins; define who is allowed to access/use the system), 'Users' (Create/delete/modify users that can log in to this Jenkins), and 'Credential Providers' (Configure the credential providers and types). At the bottom, there's a dark footer bar with icons for search, file manager, and other Jenkins features.

Tools window appears as follows

The screenshot shows the Jenkins 'Tools' configuration page. At the top, there are tabs for 'EC2 | ap-south-1' and 'Tools [Jenkins]'. The URL in the address bar is 3.110.29.130:8080/manage/configureTools/. The page title is 'Jenkins' with a user icon. On the right, there are links for 'admin', 'log out', and a search icon. The main content area has a dark header 'Tools'. Below it, under 'Maven Configuration', there are two sections: 'Default settings provider' (set to 'Use default maven settings') and 'Default global settings provider' (set to 'Use default maven global settings'). A horizontal line separates this from the 'JDK installations' section, which contains a 'Add JDK' button. Another horizontal line separates this from the 'Git installations' section, which is currently empty. At the bottom, there are 'Save' and 'Apply' buttons. The status bar at the bottom shows a weather icon (23°C, Mostly cloudy), a taskbar with various icons (Windows Start, Search, File Explorer, Edge, Google Chrome, Mozilla Firefox, Microsoft Edge, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft OneNote), and the vtricks Technologies logo.

Scroll down in this window

Step 19: go to “Maven installation” section → click on “Add Maven” → provide details as shown in following image → click on “Add Maven” → click on “Apply” and “Save”

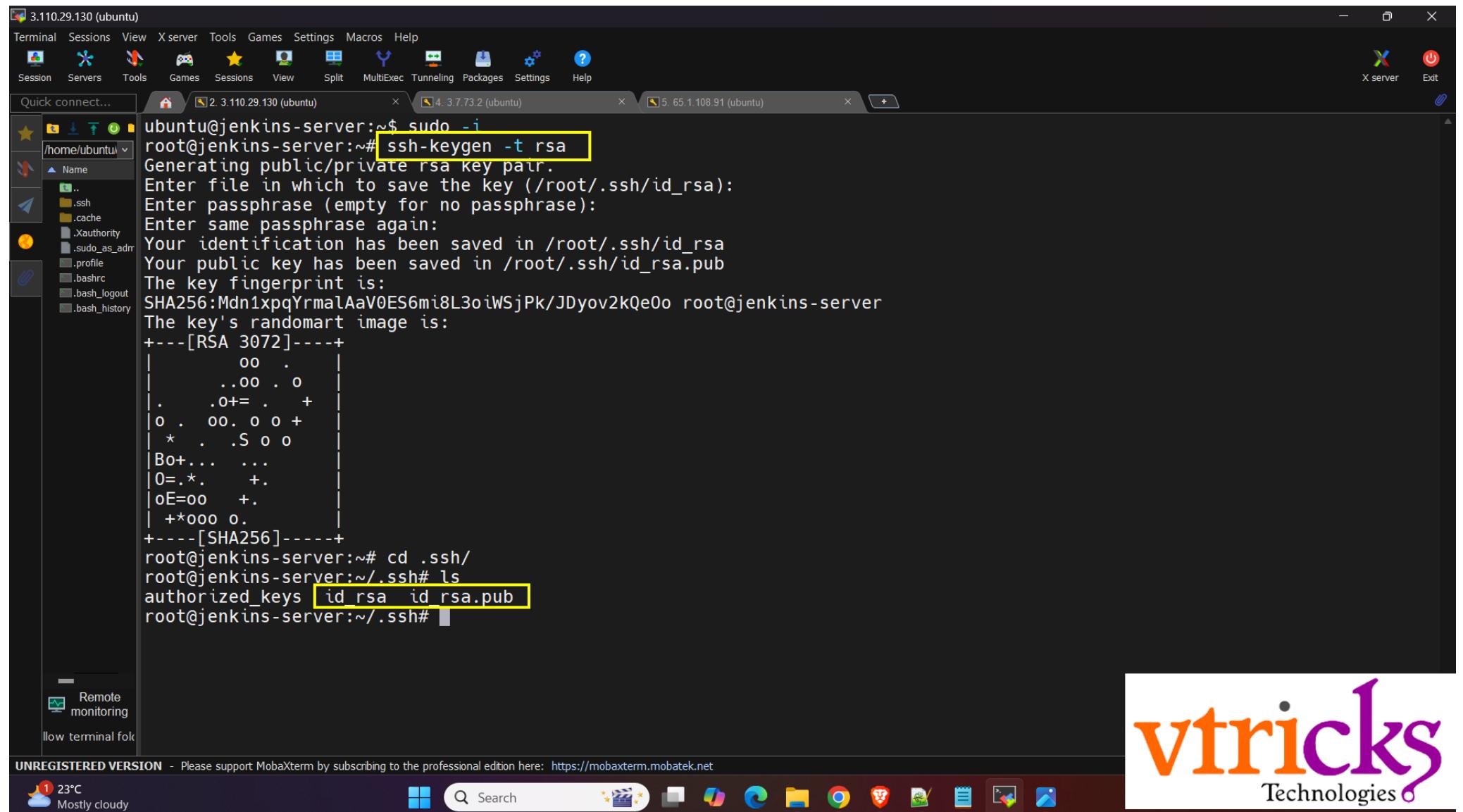
The screenshot shows the Jenkins 'Tools' configuration page under 'Manage Jenkins'. In the 'Maven installations' section, a new Maven installation is being added. The 'Name' field contains 'maven'. The 'Install automatically' checkbox is checked. Under the 'Install from Apache' section, the 'Version' dropdown is set to '3.9.9'. There is also an 'Add Installer' button. At the bottom of the form are 'Save' and 'Apply' buttons. The browser address bar shows the URL: 3.110.29.130:8080/manage/configureTools/.

Now Jenkins setup is completed

The screenshot shows the Jenkins dashboard. At the top, there's a header bar with tabs for 'EC2 | ap-south-1' and 'Dashboard [Jenkins]'. The URL is 3.110.29.130:8080. On the right of the header are icons for a star, a person, and an error message. Below the header is a dark navigation bar with the Jenkins logo and a search icon. To the right of the search icon are links for 'admin' and 'log out'. The main content area has a dark background. On the left, there's a sidebar with links: 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. Below these are two sections: 'Build Queue' (with a dropdown arrow) and 'Build Executor Status' (with a dropdown arrow). The 'Build Queue' section says 'No builds in the queue.' The 'Build Executor Status' section says '0/2'. In the center, there's a large 'Welcome to Jenkins!' heading. Below it, a text block says: 'This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.' Underneath this text is a button labeled 'Start building your software project'. To the right of this button is a 'Create a job' button with a '+' sign. Below these are three more buttons: 'Set up a distributed build', 'Set up an agent' (with a monitor icon), 'Configure a cloud' (with a cloud icon), and 'Learn more about distributed builds' (with a question mark icon).



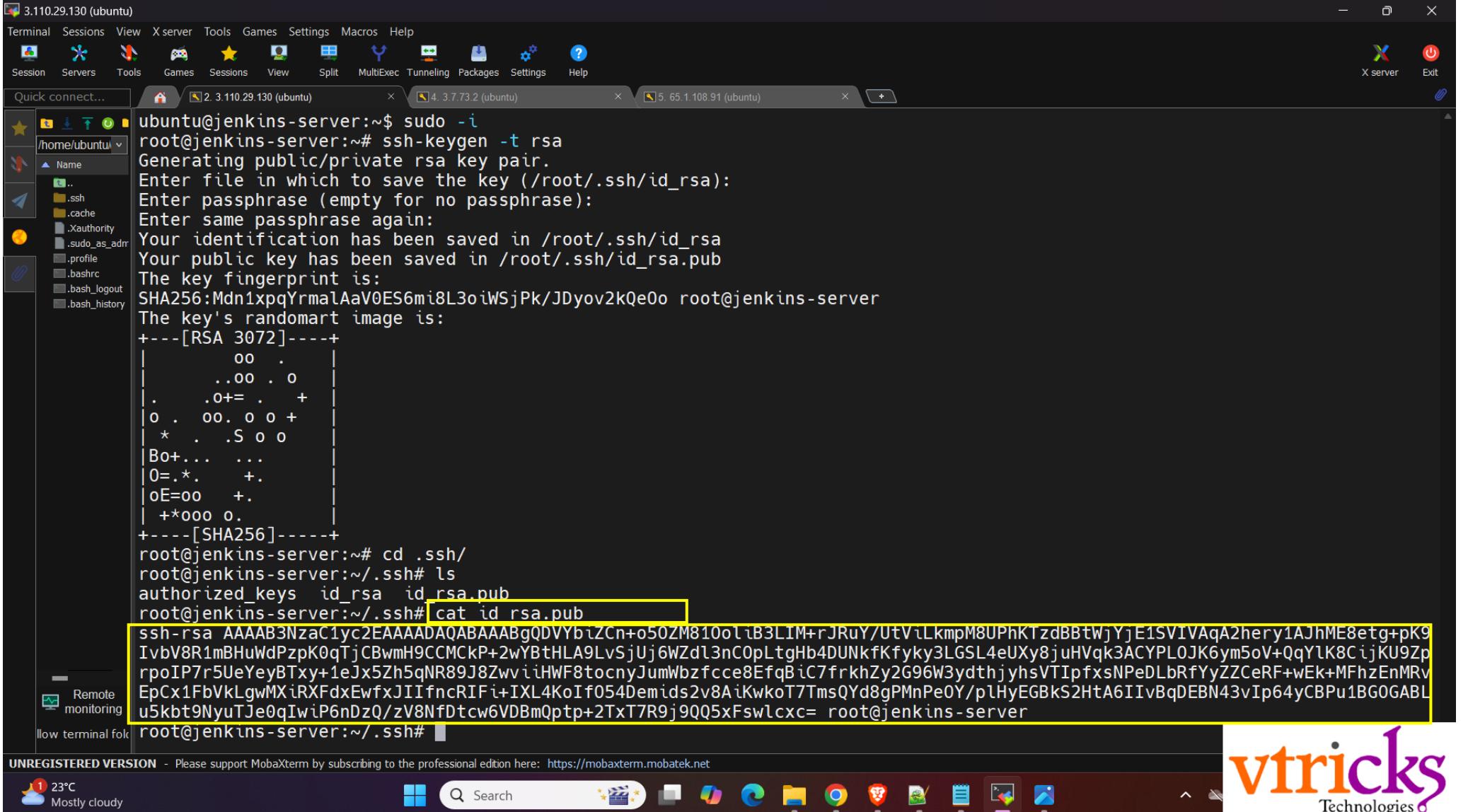
Step 20: access “Jenkins-server” machine → Run “ssh-keygen -t rsa” to generate rsa key →



The screenshot shows a terminal window in MobaXterm with the title "3.110.29.130 (ubuntu)". The terminal session is running as root on a Jenkins server. The command "ssh-keygen -t rsa" is being executed, generating a public/private key pair. The output shows the key fingerprint and the public key file path. The generated files are "id_rsa" and "id_rsa.pub".

```
ubuntu@jenkins-server:~$ sudo -i
root@jenkins-server:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Mdn1xpqYrmalAaV0ES6mi8L3oiWSjPk/JDyov2kQe0o root@jenkins-server
The key's randomart image is:
+---[RSA 3072]---+
|      oo   .
|     ..oo . o |
|    . .+= . +  |
|   o . oo. o o + |
|  * . .S o o   |
| Bo+... ...   |
| 0=.*. +.   |
| oE=o+ .   |
| +*ooo o.   |
+---[SHA256]---+
root@jenkins-server:~# cd .ssh/
root@jenkins-server:~/ssh# ls
authorized_keys  id_rsa  id_rsa.pub
root@jenkins-server:~/ssh#
```

Step 21: view and copy rsa public key

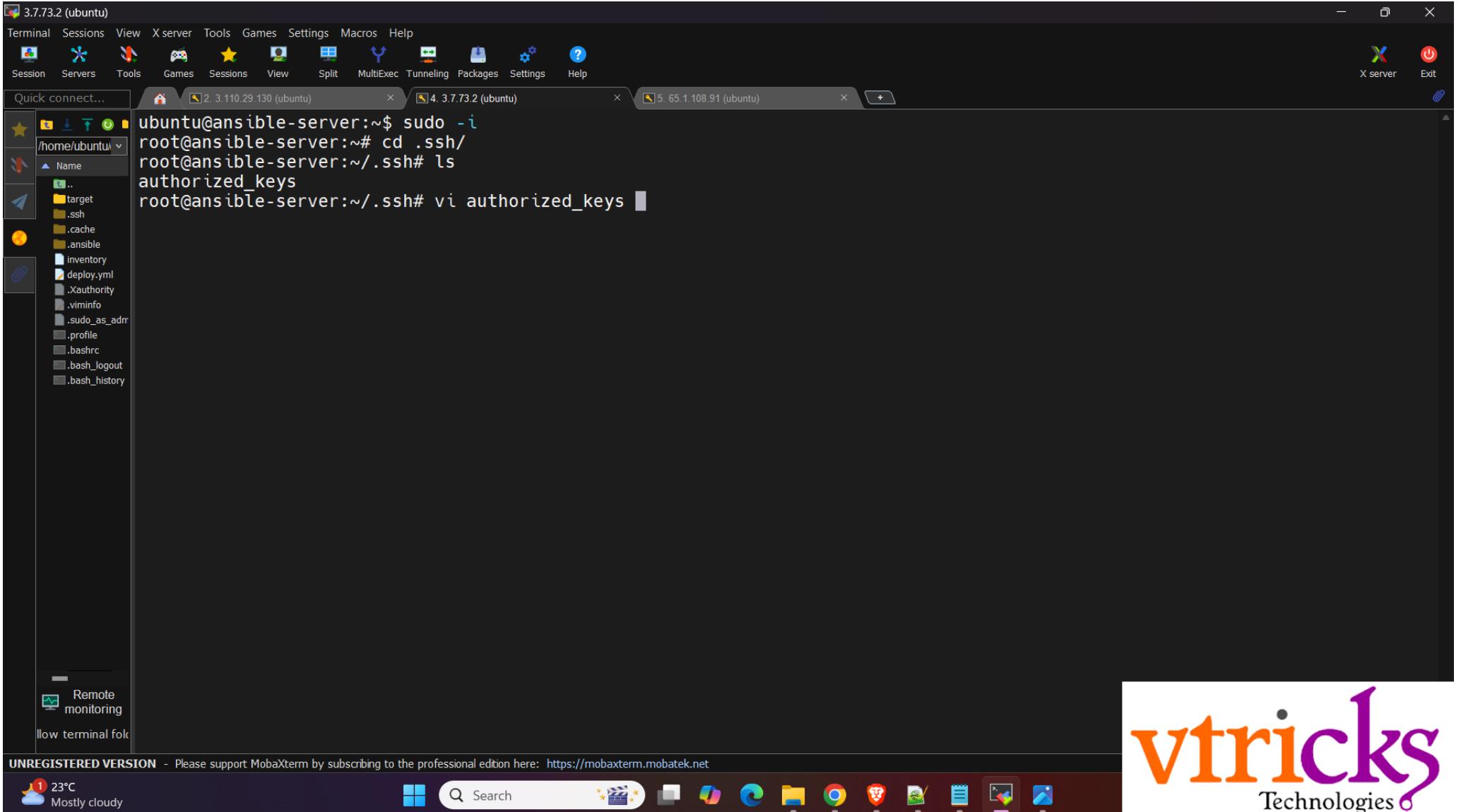


The screenshot shows a terminal session in MobaXterm on a Windows host. The terminal window title is "3.110.29.130 (ubuntu)". The session content is as follows:

```
ubuntu@jenkins-server:~$ sudo -i
root@jenkins-server:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Mdn1xpqYrmalAaV0ES6mi8L3oiWSjPk/JDyov2kQeOo root@jenkins-server
The key's randomart image is:
+---[RSA 3072]---
|      oo   .
|     ..oo . o |
|    . .+= . + |
|   o . oo. o o +
|  * . .S o o |
|Bo+...   ...
|0=.*.   +
|oE=o+   .
| +*ooo o. |
+---[SHA256]---+
root@jenkins-server:~# cd .ssh/
root@jenkins-server:~/ssh# ls
authorized_keys  id_rsa  id_rsa.pub
root@jenkins-server:~/ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAABgQDVYb1ZCn+o50ZM810olB3LIM+rJRUY/UtViLkmpM8UPhKTzdBBtWjYjE1SVIVAqA2hery1AJhME8etg+pK9IvbV8R1mBHwDpZpk0qTjCBwmH9CCMCkP+2wYBtHLA9LvSjUj6WZdl3nC0pLtgHb4DUNkfKfyky3LGSL4eUXy8juHVqk3ACYPL0JK6ym5oV+QqYlK8CijKU9Zrp0IP7r5UeYeyBTxy+1eJx5Zh5qNR89J8ZwviHWF8tocnyJumWbzfcce8Efqb1C7frkhZy2G96W3ydthjyhsVTIpxxsNPeDLbRfYyZZCeRF+wEk+MFhzEnMRvEpCx1FbVkJgwMXiRXFdxEwfxJIIfncRIFi+IXL4KoIf054Demids2v8AiKwkoT7TmsQYd8gPMnPe0Y/plHyEGBks2Hta6IIvBqDEBN43vIp64yCBPu1BG0GABLu5kbt9NyUTJe0qIwiP6nDzQ/zV8NfDtcw6VDBmQptp+2TxT7R9j9Q05xFswlcxc= root@jenkins-server
root@jenkins-server:~/ssh#
```

A yellow box highlights the command "cat id_rsa.pub" and its output, which is the RSA public key.

Step 22: access “ansible-server” machine and edit “authorized_keys”; follow commands as shown in below image



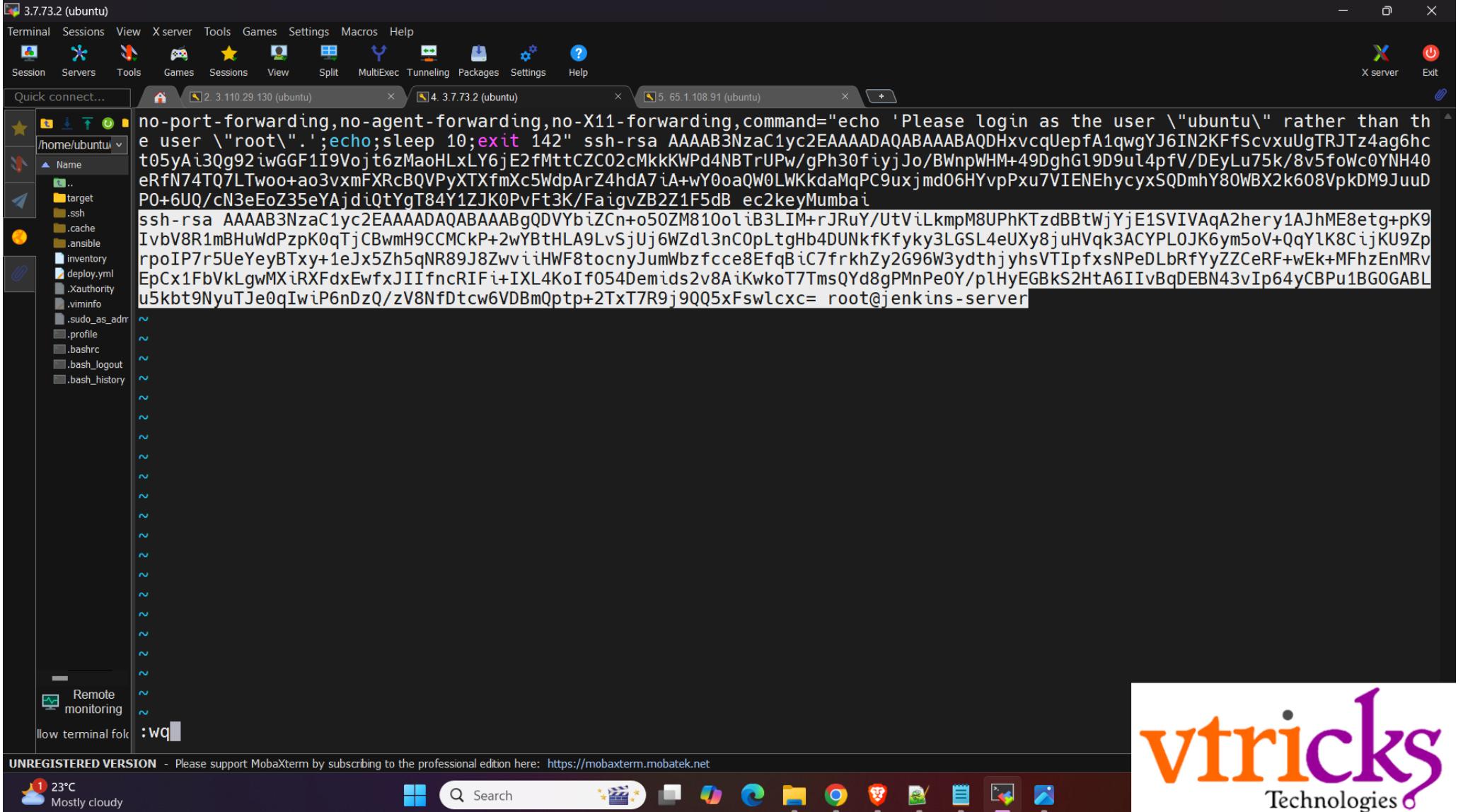
```
ubuntu@ansible-server:~$ sudo -i
root@ansible-server:~# cd .ssh/
root@ansible-server:~/ssh# ls
authorized_keys
root@ansible-server:~/ssh# vi authorized_keys
```

The screenshot shows a MobaXterm interface with multiple sessions open. The active session is on the 'ansible-server' machine (IP 3.7.73.2). The terminal window displays the following commands:

```
ubuntu@ansible-server:~$ sudo -i
root@ansible-server:~# cd .ssh/
root@ansible-server:~/ssh# ls
authorized_keys
root@ansible-server:~/ssh# vi authorized_keys
```

The left sidebar shows the file structure of the .ssh directory, including files like inventory, deploy.yml, .Xauthority, .viminfo, .sudo_as_admin, .profile, .bashrc, .bash_logout, and .bash_history. The bottom status bar indicates an unregistered version and a weather forecast for 23°C and mostly cloudy.

Step 23: Paste copied rsa public key from “Jenkins-server”



The screenshot shows a MobaXterm interface with multiple windows open. The main window displays a terminal session with the command:

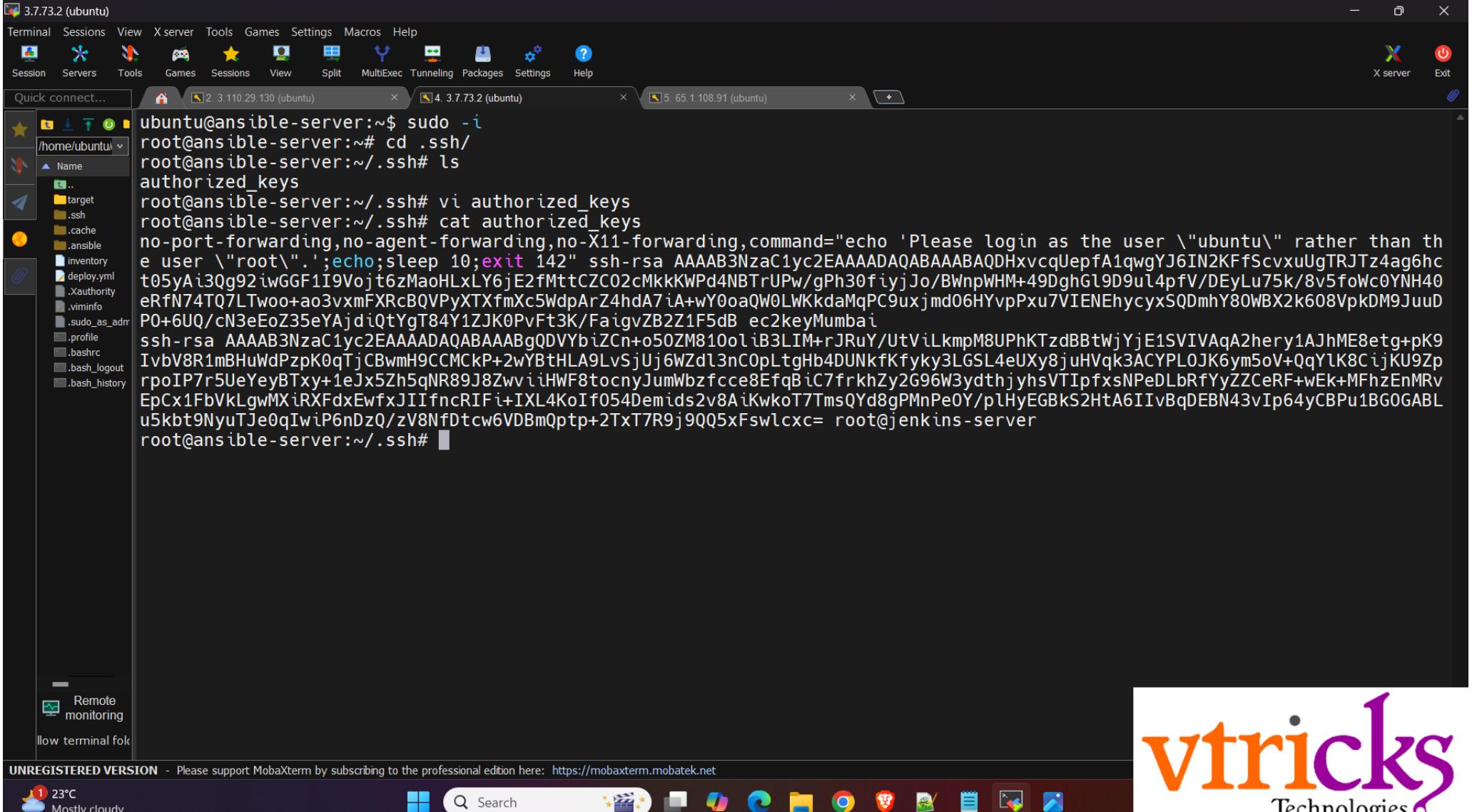
```
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ubuntu\" rather than the user \"root\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADHxvcqUepfA1qwgYJ6IN2KFFScvxuUgTRJTz4ag6hc t05yAi3Qg92iwGGF1I9Vojt6zMaoHLxLY6jE2fMttCZC02cMkkKWPd4NBTrUPw/gPh30fiyjJo/BWnpWHM+49DghGl9D9ul4pfV/DEyLu75k/8v5foWc0YNH40 eRfN74TQ7LTwoo+ao3vxmFXRcBQVPyXTXfmXc5WdpArZ4hdA7iA+wY0oaQW0LWKdaMqPC9uxjmd06HYvpPxu7VIENehycyxSQDmhy80WBX2k608VpkDM9JuUD P0+6UQ/cN3eEoZ35eYAjdiQtYgt84Y1ZJK0PvFt3K/FaigvZB2Z1F5dB ec2keyMumbai
```

The terminal also shows the command to copy the key:

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDVYbiZCn+o50ZM810o1iB3LIM+rJRuY/UtViLkmpM8UPhKTzdBBtWjYjE1SVIVAqA2hery1AJhME8etg+pK9 IvbV8R1mBHwDpZpK0qTjCBwmH9CCMCkP+2wYBtHLA9LvSjUj6WZdl3nC0pLtgHb4DUNkfKfyky3L GSL4eUXy8juHVqk3ACYPL0JK6ym5oV+QqYlK8CijKU9Zp rpoIP7r5UeYeyBTxy+1eJx5Zh5qNR89J8ZwviiHWF8tocnyJumWbzfcce8EfqbIC7frkhZy2G96W3ydhjyhsVTIpxsNPeDLbRfYyZZCeRF+wEk+MFhzEnMRv EpCx1FbVkJgwMXiRXFdxEwfxJIIfncRIFi+IXL4KoIf054Demids2v8AiKwkoT7TmsQYd8gPMnPe0Y/plHyEGBks2Hta6IIvBqDEBN43vIp64yCBPu1BG0GABL u5ktb9NyutJe0qIwiP6nDzQ/zV8NfDtew6VDBmQptp+2TxT7R9j9Q05xFswlcxc= root@jenkins-server
```

The bottom of the terminal window shows the command `:wq!`. The status bar at the bottom left indicates "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>". The bottom right corner features the vtricks Technologies logo.

Verify the successful addition of Jenkins-server's rsa public key in ansible-server



```
ubuntu@ansible-server:~$ sudo -i
root@ansible-server:~/ .ssh#
root@ansible-server:~/ .ssh# ls
authorized_keys
root@ansible-server:~/ .ssh# vi authorized_keys
root@ansible-server:~/ .ssh# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ubuntu\" rather than the user \"root\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDHzvcqUepfA1qwgYJ6IN2KFFScvxuUgTRJTz4ag6hc
t05yAi3Qg92iwGGF1I9Vojt6zMaoHLxLY6jE2fMttCZC02cMkkKWPd4NBTrUPw/gPh30fiyjJo/BWnpWHM+49DghGl9D9ul4pfV/DEyLu75k/8v5foWc0YNH40
eRfn74TQ7LTTwoo+ao3vxmFXRcBQVPyXTXfmXc5WdpArZ4hdA7iA+wY0oaQW0LWKkdaMqPC9uxjmd06HYvpPxu7VIENehycyxSQDmhy80WBX2k608VpkDM9JuuD
P0+6UQ/cN3eEoZ35eYAjdiQtYgt84Y1ZJK0PvFt3K/FaigvZB2Z1F5dB ec2keyMumbai
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQgQDVYbiZCn+o50ZM810o1iB3LIM+rJRUY/UtViLkmpM8UPhKTzdBBtWjYjE1SVIVAgA2hery1AJhME8etg+pK9
IvbV8R1mBHuwDpZpk0qTjCBwmH9CCMCkP+2wYBtHLA9LvSjUj6WZdl3nC0pLtgHb4DUNkfKfyky3LGS4eUXy8juHVqk3ACYPL0JK6ym5oV+QqYlK8CijKU9Zp
rpoIP7r5UeYeyBTxy+1eJx5Zh5qNR89J8ZwviHWF8tocnyJumWbzfcce8Efqbic7frkhZy2G96W3ydthjyhsVTIpxsNPeDLbRfYyZZCeRF+wEk+MFhzEnMRv
EpCx1FbVkJgwMXiRXFdxEwfxJIIifncRIfi+IXL4KoiF054Demids2v8AiKwkoT7TmsQYd8gPMnPe0Y/plHyEGBks2Hta6IIvBqDEBN43vIp64yCBPu1BG0GABL
u5ktb9NyUTje0qIwiP6nDzQ/zV8NfDtcw6VDBmQptp+2TxT7R9j9QQ5xFswlcxc= root@jenkins-server
root@ansible-server:~/ .ssh#
```

Step 24: generate ssh rsa key in ansible-server and copy the generated rsa public key

The screenshot shows a terminal session in MobaXterm on a Windows host. The terminal window title is "4. 3.7.73.2 (ubuntu)". The command `ssh-keygen -t rsa` is run, generating a new RSA key pair. The public key is then copied to the clipboard. The terminal output is as follows:

```
root@ansible-server:~# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Fa5pmciJjFOwYceySye6blVpkhUH7gayK0v6LFv+COA root@ansible-server
The key's randomart image is:
+---[RSA 3072]---+
| +..0. . |
| ..=0.. . .
| ..++.. o |
| ***+* o *
| .+o++*o+ S |
| + oo. . |
| oEo |
| *B . |
| *=.. |
+---[SHA256]---+
root@ansible-server:~# cd .ssh/
root@ansible-server:~/ssh# ls
authorized_keys id_rsa id_rsa.pub
root@ansible-server:~/ssh# cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCsL/zM3mmtrJi5gVFVGvnQ6oMg0jZ/wQCCWEpImrqCQbVxzyIowzcc2Cs0xuD301AXlZlRXIdvE78ogEf0
KPqC3B8+NQngJFHKR+futXlgLmEZQ+DGwvvSc9JVlpHKPN+BKhRui9Y62mNRjBcHLcC5G3NEp4fhukBYoPTuhujX7tC7HCkOELyGLqMwEA3U2DEcNNn2CcDEj
Z0VV+g+sMURvp3erg20FVRSSaLTwt7anbIKSFkFzur9Jr9R15IRtcCEdzYzN8ro+sVtal19I2v9YwGZ9r0AzbZPU5KULr5HWLyUS8h9LH081oHTU5pep9sGf8F
SdhQH01fBgthuIrmM72K6eMTXWt0S0A9gKdLrbM/9M79M5xgcIcQP4/i//9tkQyGH1qTzy0CrLYBxb1TSWtfWerj/zzTppWjCWNx3mXm2tk26uM4yU1YV5qfu
i7jn15/Z7chlWH3M1ee9Gjy/MbYlczC1zh66oY1lKSwy524zZm4kk6dT9ArQ06E= root@ansible-server
root@ansible-server:~/ssh#
```

The public key is highlighted with a yellow box.

Step 25: access “ansible-client-server” machine and edit “authorized_keys”; follow commands as shown in below image

```
ubuntu@ansible-client-server:~$ sudo -i
root@ansible-client-server:~# cd .ssh/
root@ansible-client-server:~/ssh# ls
authorized_keys
root@ansible-client-server:~/ssh# vi authorized_keys
```

The screenshot shows a MobaXterm interface with four tabs open:

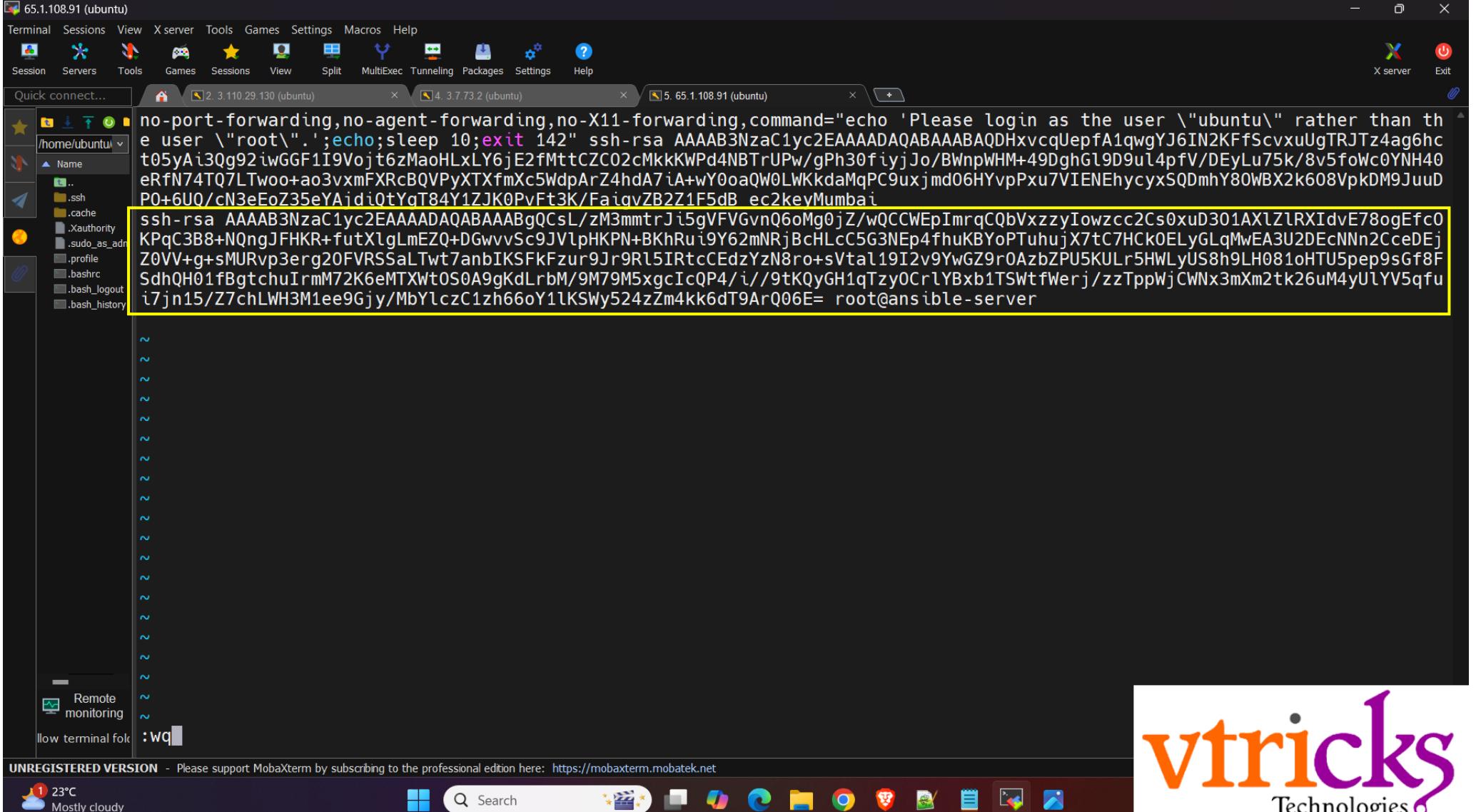
- Tab 1: 65.1.108.91 (ubuntu)
- Tab 2: 2. 3.110.29.130 (ubuntu)
- Tab 3: 4. 3.7.73.2 (ubuntu)
- Tab 4: 5. 65.1.108.91 (ubuntu)

The active tab (Tab 4) displays a terminal session as root on the 'ansible-client-server'. The user has run several commands to navigate to the '.ssh' directory, list its contents, and open the 'authorized_keys' file for editing.

The left sidebar of the MobaXterm window shows a file tree for the current directory, which includes sub-directories like .., .ssh, .cache, .Xauthority, .sudo_as_pass, .profile, .bashrc, .bash_logout, and .bash_history. There are also icons for 'Remote monitoring' and 'Show terminal log'.

The bottom status bar indicates an 'UNREGISTERED VERSION' and provides a link to support the software. It also shows the current weather as 23°C and mostly cloudy.

Step 26: Paste copied rsa public key from “ansible-server”



The screenshot shows a MobaXterm interface with multiple sessions open. The active session is on port 65.1.108.91 (ubuntu). The terminal window displays the following text:

```
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ubuntu\" rather than the user \"root\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADHxvcqUepfA1qwgYJ6IN2KFFScvxuUgTRJTz4ag6hc  
t05yAi3Qg92iwGGF1I9Vojt6zMaoHLxLY6jE2fMttCZC02cMkkKWPd4NBTrUPw/gPh30fiyjJo/BWnpWHM+49DghGl9D9ul4pfV/DEyLu75k/8v5foWc0YNH40  
eRfN74TQ7LTtwoo+ao3vxmFXRcBQVPyXTXfmXc5WdpArZ4hdA7iA+wY0oaQW0LWKdaMqPC9uxjmd06HYvpPxu7VIENehycyxSQDmhy80WBX2k608VpkDM9JuuD  
P0+6U0/cN3eEoZ35eYAidi0tYgT84Y1ZJK0PvFt3K/FaigvZB2Z1F5dB ec2keyMumbai  
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQBgQCsl/zM3mmtrJi5gVFVGvnQ6oMg0jZ/wQCCWEpImrqCQbVxzzyIowzcc2Cs0xuD301AXlZlRXIdvE78ogEfco  
KPqC3B8+NQngJFHKR+futXlgLmEZQ+DGvvvSc9JVlpHKPN+BKhRui9Y62mNRjBcHLcC5G3NEp4fhuKBYoPTuhujX7tC7HCk0ELyGLqMwEA3U2DEcNNn2CcDEj  
Z0Vv+g+sMURvp3erg20FVRSSaLTwt7anbIKSFkFzur9Jr9R15IRtcCEdzYzN8ro+sVtal19I2v9YwGZ9r0AzbZPU5KULr5HWLyUS8h9LH081oHTU5pep9sGf8F  
SdhQH01fBgtchuIrmM72K6eMTXWt0S0A9gKdLrbM/9M79M5xgcIcQP4/i/9tKQyGH1qTzy0CrlYBxb1TSWtfWerj/zzTppWjCWNx3mXm2tk26uM4yU1YV5qfu  
i7jn15/Z7chLWH3M1ee9Gjy/MbYlczC1zh66oY1lKSwy524zZm4kk6dT9ArQ06E= root@ansible-server
```

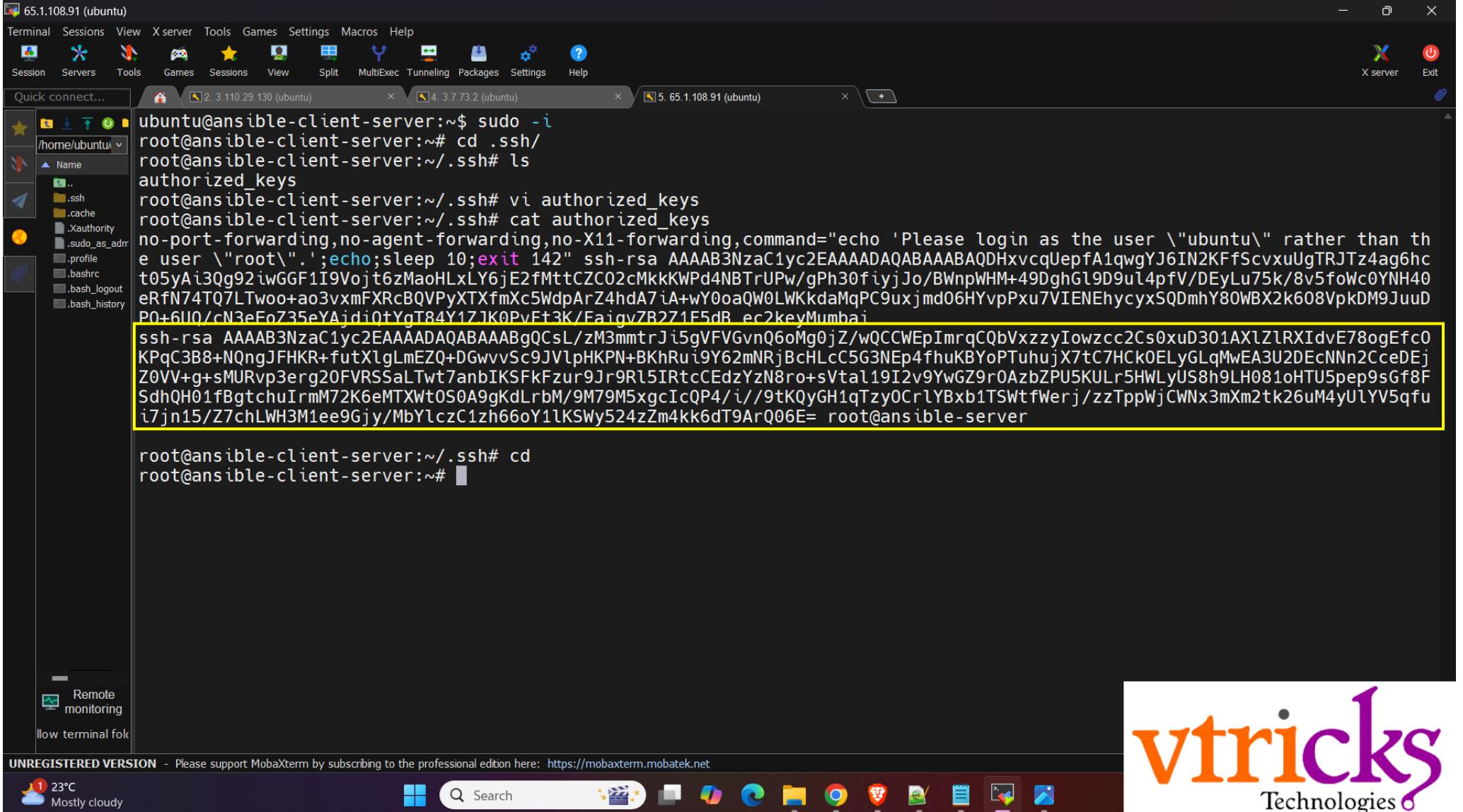
The terminal prompt at the bottom shows ':wq'.

At the bottom of the screen, there is a status bar with the text "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>".

On the far left, there is a weather widget showing "23°C Mostly cloudy".

At the bottom right, there is a watermark for "vtricks Technologies".

Verify the successful addition of 'rsa public key' in ansible-server



The screenshot shows a MobaXterm interface with multiple terminal sessions open:

- Session 1: 65.1.108.91 (ubuntu) - The current active terminal.
- Session 2: 2. 3.110.29.130 (ubuntu)
- Session 3: 4. 3.7.73.2 (ubuntu)
- Session 4: 5. 65.1.108.91 (ubuntu)

The active terminal (Session 1) displays the following command output:

```
ubuntu@ansible-client-server:~$ sudo -i
root@ansible-client-server:~# cd .ssh/
root@ansible-client-server:~/ssh# ls
authorized_keys
root@ansible-client-server:~/ssh# vi authorized_keys
root@ansible-client-server:~/ssh# cat authorized_keys
no-port-forwarding,no-agent-forwarding,no-X11-forwarding,command="echo 'Please login as the user \"ubuntu\" rather than the user \"root\".';echo;sleep 10;exit 142" ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQDHzvcqUepfA1qwgYJ6IN2KFFScvxuUgTRJTz4ag6hc
t05yAi3Qg92iwGGF1I9Vojt6zMaoHLxLY6jE2fMttCZC02cMkkKWPd4NBTrUPw/gPh30fiyJ0/BWnpWHM+49DghG19D9ul4pfV/DEyLu75k/8v5foWc0YNH40
eRfn74TQ7LTTwoo+ao3vxmFXRcBQVPyXTXfmXc5WdpArZ4hdA7iA+wY0oaQW0LWKkdaMqPC9uxjmd06HYvpPxu7VIENehcyxSQDmhy80WBX2k608VpkDM9JuuD
P0+6IIQ/cN3eFo735eYajdiQtYgt84Y1Z1K0PvEt3K/FaigvZB271F5dB_ec2keyMumbai
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCsL/zM3mmtrJi5gVFVGvnQ6oMg0jZ/wQCCWEpImrqCQbVxzzyIowzcc2Cs0xuD301AX1Z1RXIdvE78ogEfco
KPqC3B8+NQngJFHKR+futXlgLmEZQ+DGvvvSc9JVlpHKPN+BKhRui9Y62mNRjBcHLcC5G3NEp4fhuKBYoPTuhjX7tC7HCK0ELyGLqMwEA3U2DEcNNn2CceDEj
Z0VV+g+sMURvp3erg20FVRSSaLTwt7anbIKSFkFzur9Jr9R15IRtcCEdzYzN8ro+sVtal19I2v9YwGZ9r0AzbZPU5KULr5HWLyUS8h9LH081oHTU5pep9sGf8F
SdhQH01fBgtxuIrmM72K6eMTXWtOS0A9gKdLrbM/9M79M5xgcIcQP4/i/9tKQyGH1qTzy0CrlyBxb1TSWtfWerj/zzTppWjCWNx3mXm2tk26uM4yU1YV5qfu
i7jn15/Z7chLWH3M1ee9gy/MbYlczC1zh66oY1lKSwy524zZm4kk6dT9ArQ06E= root@ansible-server

root@ansible-client-server:~/ssh# cd
root@ansible-client-server:~#
```

The terminal output shows the successful addition of the RSA public key to the `authorized_keys` file. The key is displayed in its entirety, including the long string of characters and the `root@ansible-server` signature at the end.

Step 27: access jenkins dashboard → click on “Create Job” or “New Item”

The screenshot shows a browser window with the Jenkins dashboard. The URL is 3.110.29.130:8080. The page title is "Dashboard [Jenkins]". The left sidebar includes links for "New Item", "Build History", "Manage Jenkins", and "My Views". A "Build Queue" section indicates "No builds in the queue.". A "Build Executor Status" section shows "0/2". The main content area features a "Welcome to Jenkins!" message, a "Start building your software project" button, and a "Create a job" button with a "+" icon. Below these are sections for "Set up a distributed build", "Set up an agent" (with a monitor icon), "Configure a cloud" (with a cloud icon), and "Learn more about distributed builds" (with a question mark icon). At the bottom, there's a taskbar with icons for weather (23°C, Mostly cloudy), search, and various applications like File Explorer, Edge, and Google Chrome.

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job +

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds ?

Build Queue
No builds in the queue.

Build Executor Status 0/2

23°C Mostly cloudy

Search

vtricks Technologies

Step 28: Provide Project/item name → select Item type → click on “OK”

The screenshot shows the Jenkins 'New Item' configuration page. At the top, the URL is 3.110.29.130:8080/view/all/newJob. The page title is 'New Item [Jenkins]'. The main section is titled 'New Item'. A red box highlights the 'Enter an item name' input field, which contains 'maven-project'. Below it, a red box highlights the 'Freestyle project' option, which is described as a 'Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications'. Further down, there are descriptions for 'Pipeline' and 'Multi-configuration project'. At the bottom left, a red box highlights the 'OK' button.

Step 29: Provide Description

The screenshot shows the Jenkins job configuration interface for a 'maven-project'. The 'General' section is selected. A red box highlights the 'Description' input field, which contains the text 'maven project demo'. Other visible options include 'Source Code Management', 'Triggers', 'Environment', 'Build Steps', and 'Post-build Actions'. There are also several checkboxes for build-related settings like 'Discard old builds', 'GitHub project', and 'Throttle builds'. At the bottom, there are 'Save' and 'Apply' buttons.

EC2 | ap-south-1 maven-project Config [Jenkins] Not secure 3.110.29.130:8080/job/maven-project/configure admin log out

Jenkins

Dashboard > maven-project > Configuration

Configure General Enabled

Description

maven project demo

Plain text Preview

Source Code Management

Triggers

Environment

Build Steps

Post-build Actions

Discard old builds

GitHub project

This project is parameterized

Throttle builds

Execute concurrent builds if necessary

Advanced

Save Apply

23°C Mostly cloudy

vtricks Technologies

Scroll down in this window

Step 30: Select “Git” under “Source Code Management” → under “Repository URL” provide project’s git repository URL

The screenshot shows the Jenkins job configuration page for a job named "maven-project". The "Source Code Management" section is highlighted with a red box. Under "Source Code Management", the "Git" option is selected, also highlighted with a red box. The "Repository URL" field contains the value "https://github.com/VijayDesai08/maven-project.git", which is also highlighted with a red box. The Jenkins interface includes a sidebar with options like General, Triggers, Environment, Build Steps, and Post-build Actions. At the bottom, there are "Save" and "Apply" buttons.

Configure

Source Code Management

Connect and manage your code repository to automatically pull the latest code for your builds.

General

Source Code Management

Triggers

Environment

Build Steps

Post-build Actions

None

Git

Repositories

Repository URL

https://github.com/VijayDesai08/maven-project.git

Credentials

- none -

+ Add

Advanced

Add Repository

Branches to build

Save

Apply

EC2 | ap-south-1 maven-project Config [Jenkins] Not secure 3.110.29.130:8080/job/maven-project/configure Dashboard > maven-project > Configuration

Step 31: under “Branches to build” select branch name, usually “main”

The screenshot shows the Jenkins configuration interface for a project named "maven-project". The left sidebar lists configuration sections: General, Source Code Management (selected), Triggers, Environment, Build Steps, and Post-build Actions. The main panel displays the "Configure" section with various configuration options. A red box highlights the "Branches to build" field, which contains the value "/main".

Dashboard > maven-project > Configuration

Configure

- General
- Source Code Management**
- Triggers
- Environment
- Build Steps
- Post-build Actions

+ Add

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

*/main

Add Branch

Repository browser ?

(Auto)

Additional Behaviours

Add ▾

Save Apply

EC2 | ap-south-1 maven-project Config [Jenkins] Not secure 3.110.29.130:8080/job/maven-project/configure Error

1 23°C Mostly cloudy

Step 32: scroll down to “Build Steps” section

The screenshot shows the Jenkins configuration interface for a job named "maven-project". The left sidebar has tabs for General, Source Code Management, Triggers, Environment (which is selected), Build Steps, and Post-build Actions. The main content area has sections for Configure (with checkboxes for various build options) and Build Steps (with a sub-section for Automate your build process). Below the Build Steps section is a Post-build Actions section for defining what happens after a build completes. At the bottom are Save and Apply buttons.

EC2 | ap-south-1 maven-project Config [Jenkins]

Not secure 3.110.29.130:8080/job/maven-project/configure

Dashboard > maven-project > Configuration

Configure

- Send files or execute commands over SSH after the build runs ?
- Add timestamps to the Console Output
- Inspect build log for published build scans
- Terminate a build if it's stuck
- With Ant ?

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Add build step ▾

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Add post-build action ▾

Save Apply

Step 33: click on “Add build step” → select “Invoke top-level Maven targets”

The screenshot shows the Jenkins configuration page for a job named "maven-project". The "Environment" tab is currently selected. In the "Build Steps" section, a dropdown menu is open under the "Add build step" button. The "Invoke top-level Maven targets" option is highlighted with a red box.

Configure

- General
- Source Code Management
- Triggers
- Environment**
- Build Steps
- Post-build Actions

Add build step ^

Filter

- Execute Windows batch command
- Execute shell
- Invoke Ant
- Invoke Gradle script
- Invoke top-level Maven targets**
- Run with timeout
- Send files or execute commands over SSH
- Set build status to "pending" on GitHub commit

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Step 34: make following changes as shown in following image

The screenshot shows the Jenkins configuration interface for a project named "maven-project". The left sidebar has tabs: General, Source Code Management, Triggers, Environment, Build Steps (which is selected and highlighted in grey), and Post-build Actions. The main content area is titled "Build Steps" with the sub-instruction "Automate your build process with ordered tasks like code compilation, testing, and deployment." A red box highlights the "Invoke top-level Maven targets" section. Inside, the "Maven Version" dropdown is set to "maven" and the "Goals" dropdown is set to "clean package install". Below this is an "Advanced" dropdown and a "Add build step" button. At the bottom of the page, there is a "Post-build Actions" section with a "Add post-build action" dropdown, a "Save" button, and an "Apply" button. The browser address bar shows the URL: 3.110.29.130:8080/job/maven-project/configure.

Configure

Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Invoke top-level Maven targets

Maven Version

maven

Goals

clean package install

Add build step

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Add post-build action

Save

Apply

23°C
Mostly cloudy

Search

vtricks Technologies

Step 35: scroll down to “Post-build Actions”

The screenshot shows the Jenkins configuration interface for a project named "maven-project". The left sidebar lists several configuration sections: General, Source Code Management, Triggers, Environment, Build Steps (which is currently selected and highlighted with a grey background), and Post-build Actions. The main content area is titled "Invoke top-level Maven targets" and contains fields for "Maven Version" (set to "maven") and "Goals" (set to "clean package install"). Below these fields is an "Advanced" dropdown. Further down, there is a button labeled "Add build step". A red box highlights the "Post-build Actions" section, which is described as "Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs." This section includes a "Save" button and an "Apply" button.

Step 36: click on “Add post-build action” → select “Send build artifacts over SSH”

The screenshot shows the Jenkins configuration interface for a project named "maven-project". The left sidebar has "Configure" selected. Under "Build Steps", "Post-build Actions" is selected. A dropdown menu is open, listing various actions. The "Send build artifacts over SSH" option is highlighted with a red box. Below the dropdown, there is a button labeled "Add post-build action" also highlighted with a red box.

Maven Version

Filter

- Aggregate downstream test results
- Archive the artifacts
- Build other projects
- Publish JUnit test result report
- Record fingerprints of files to track usage
- Git Publisher
- E-mail Notification
- Editable Email Notification
- Send build artifacts over SSH**
- Set GitHub commit status (universal)
- Set build status on GitHub commit [deprecated]
- Delete workspace when build is done

Add post-build action ^

Save Apply



vtricks
Technologies

Step 37: Select “ansible-server” under “SSH server”

The screenshot shows the Jenkins configuration interface for a job named "maven-project". The left sidebar lists various configuration sections: General, Source Code Management, Triggers, Environment, Build Steps, and Post-build Actions. The "Post-build Actions" section is currently selected and highlighted with a light blue background.

In the main content area, under "Post-build Actions", there is a section titled "Send build artifacts over SSH". This section contains a sub-section titled "SSH Publishers" which includes an "SSH Server" configuration. The "Name" field for this server is set to "ansible-server".

Below the "SSH Publishers" section, there is another section titled "Transfers" with a sub-section titled "Transfer Set". The "Source files" field in this section is empty, and a red error message states: "Either Source files, Exec command or both must be supplied".

At the bottom of the configuration form, there are two buttons: "Save" and "Apply".

The browser's address bar shows the URL: 3.110.29.130:8080/job/maven-project/configure. The top navigation bar indicates the current tab is "maven-project Config [Jenkins]".

Step 38: scroll down to “Transfer set” and provide details as shown in following image → click on “Apply” and “Save”

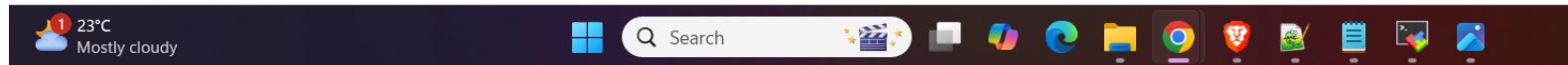
The screenshot shows the Jenkins configuration page for a job named "maven-project". The left sidebar lists various configuration sections: General, Source Code Management, Triggers, Environment, Build Steps, and Post-build Actions. The "Post-build Actions" section is currently selected and highlighted with a grey background. In the main content area, under the "Transfers" section, there is a "Transfer Set" configuration. The "Source files" field contains the value "target/*.war", which is highlighted with a red rectangle. The "Remote directory" field is empty. The "Exec command" field contains the command "sudo ansible-playbook -i /home/ubuntu/inventory /home/ubuntu/deploy.yml", which is also highlighted with a red rectangle. Below these fields, a note states: "All of the transfer fields (except for Exec timeout) support substitution of Jenkins environment variables". At the bottom of the configuration panel, there are two buttons: "Save" and "Apply".

Step 39: click on “Build now” in project window and watch for build progress

The screenshot shows the Jenkins interface for the 'maven-project' job. On the left, there's a sidebar with various actions: Status (selected), Changes, Workspace, Build Now (highlighted with a red box), Configure, Delete Project, and Rename. Below this is a 'Builds' section, also highlighted with a red box, showing a single build log entry for build #1 at 10:21 PM. The main content area displays the project details: 'maven-project' and 'maven project demo'. At the top, the browser title bar shows 'EC2 | ap-south-1' and 'maven-project [Jenkins]'. The address bar shows 'Not secure 3.110.29.130:8080/job/maven-project/'. The top right corner shows the user 'admin' and a 'log out' link.

Step 40: watch for build completion and click on build number to view the output details

The screenshot shows a web browser window with the address bar displaying 'EC2 | ap-south-1' and 'maven-project [Jenkins]'. The main content is the Jenkins dashboard for the 'maven-project' job. The job name 'maven-project' is at the top, followed by a description 'maven project demo'. On the left, there's a sidebar with options like 'Status', 'Changes', 'Workspace', 'Build Now', 'Configure', 'Delete Project', and 'Rename'. Below that is a 'Builds' section with a table. The first row, labeled '#1 10:21 PM', has a green checkmark icon and is highlighted with a red box. The rest of the table rows are grayed out. To the right of the dashboard, there are links for 'Edit description', 'admin', and 'log out'.



EC2 | ap-south-1 maven-project #1 Console [Jen] +

Not secure 3.110.29.130:8080/job/maven-project/1/console Error :

Jenkins

Dashboard > maven-project > #1 > Console Output

Status Changes Console Output Edit Build Information Delete build '#1' Timings Git Build Data

Console Output

Started by user admin
Running as SYSTEM
Building in workspace /var/lib/jenkins/workspace/maven-project
The recommended git tool is: NONE
No credentials specified
Cloning the remote Git repository
Cloning repository <https://github.com/VijayDesai08/maven-project.git>
> git init /var/lib/jenkins/workspace/maven-project # timeout=10
Fetching upstream changes from <https://github.com/VijayDesai08/maven-project.git>
> git --version # timeout=10
> git --version # 'git version 2.43.0'
> git fetch --tags --force --progress -- <https://github.com/VijayDesai08/maven-project.git> +refs/heads/*:refs/remotes/origin/* # timeout=10
> git config remote.origin.url <https://github.com/VijayDesai08/maven-project.git> # timeout=10
> git config --add remote.origin.fetch +refs/heads/*:refs/remotes/origin/* # timeout=10
Avoid second fetch
> git rev-parse refs/remotes/origin/main^{commit} # timeout=10
Checking out Revision 9b97f977f9e1363f2710a0aa2d408d7ed8fa7b0e (refs/remotes/origin/main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 9b97f977f9e1363f2710a0aa2d408d7ed8fa7b0e # timeout=10
Commit message: "1st commit"
First time build. Skipping changelog.
Unpacking <https://repo.maven.apache.org/maven2/org/apache/maven/apache-maven/3.9.9/apache-maven-3.9.9-bin.zip> to /var/lib/jenkins/tools/hudson.tasks.Maven.MavenInstallation/maven on Jenkins

Download Copy View as plain text

23°C Mostly cloudy

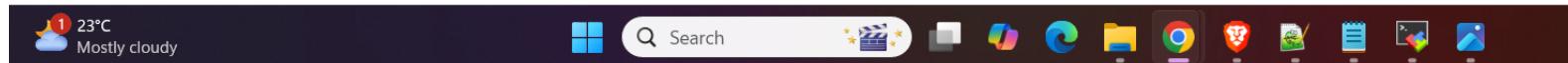
EC2 | ap-south-1 maven-project #1 Console [Jen] +

Not secure 3.110.29.130:8080/job/maven-project/1/console

Dashboard > maven-project > #1 > Console Output

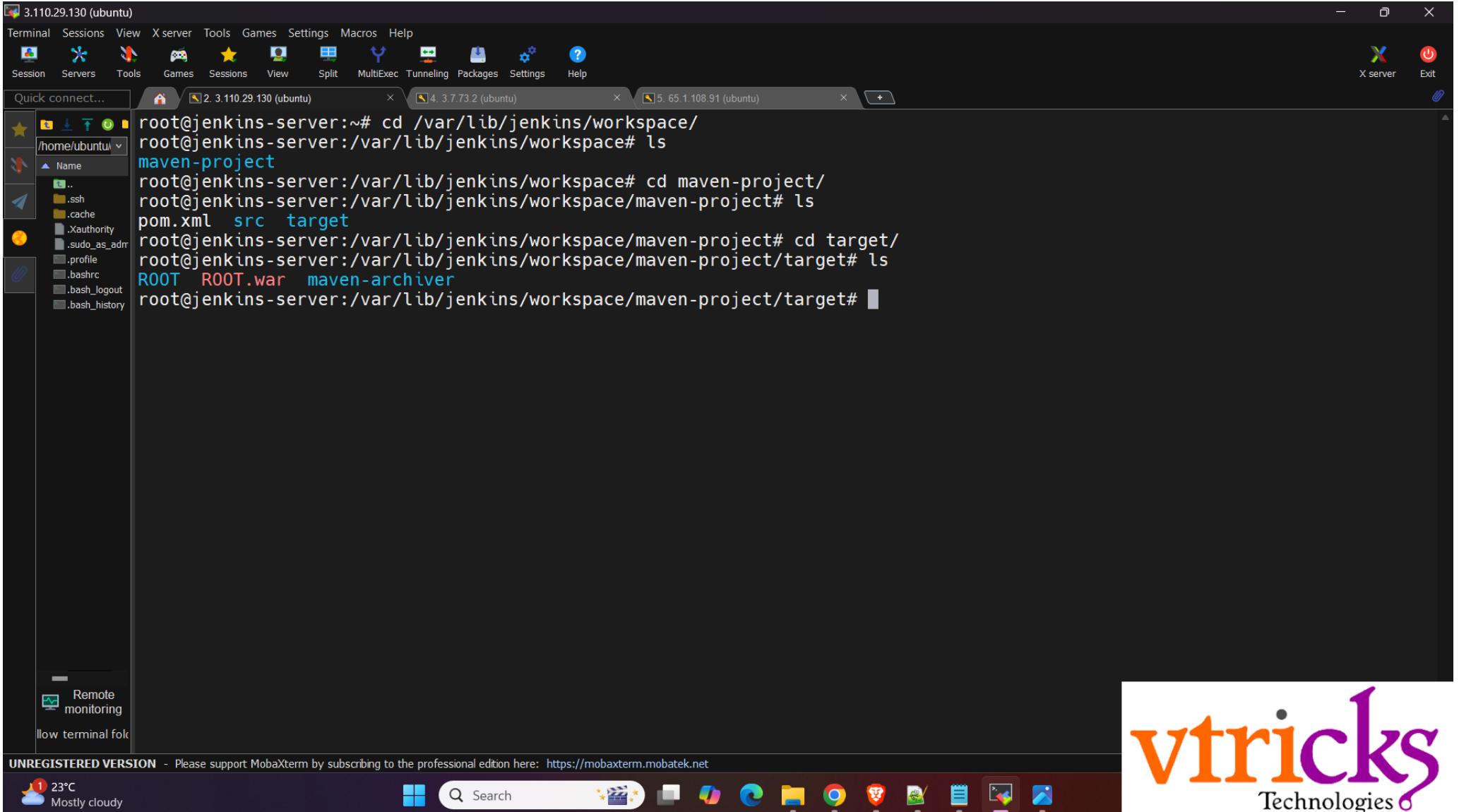
```
MB/s)
Progress (1): 178/233 kB
Progress (1): 194/233 kB
Progress (1): 210/233 kB
Progress (1): 227/233 kB
Progress (1): 233 kB

Downloaded from central: https://repo.maven.apache.org/maven2/commons-codec/commons-codec/1.6/commons-codec-1.6.jar (233 kB at 8.0 MB/s)
[INFO] Installing /var/lib/jenkins/workspace/maven-project/target/ROOT.war to /var/lib/jenkins/.m2/repository/xyz/thebasha/ROOT/1.0-SNAPSHOT/ROOT-1.0-SNAPSHOT.war
[INFO] Installing /var/lib/jenkins/workspace/maven-project/pom.xml to /var/lib/jenkins/.m2/repository/xyz/thebasha/ROOT/1.0-SNAPSHOT/ROOT-1.0-SNAPSHOT.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.629 s
[INFO] Finished at: 2025-03-01T22:21:58Z
[INFO] -----
SSH: Connecting from host [jenkins-server]
SSH: Connecting with configuration [ansible-server] ...
SSH: EXEC: completed after 3,203 ms
SSH: Disconnecting configuration [ansible-server] ...
SSH: Transferred 1 file(s)
Finished: SUCCESS
```



vtricks
Technologies

Step 41: Verify the creation of “maven-project” folder in jenkins-server, run commands as shown in following image

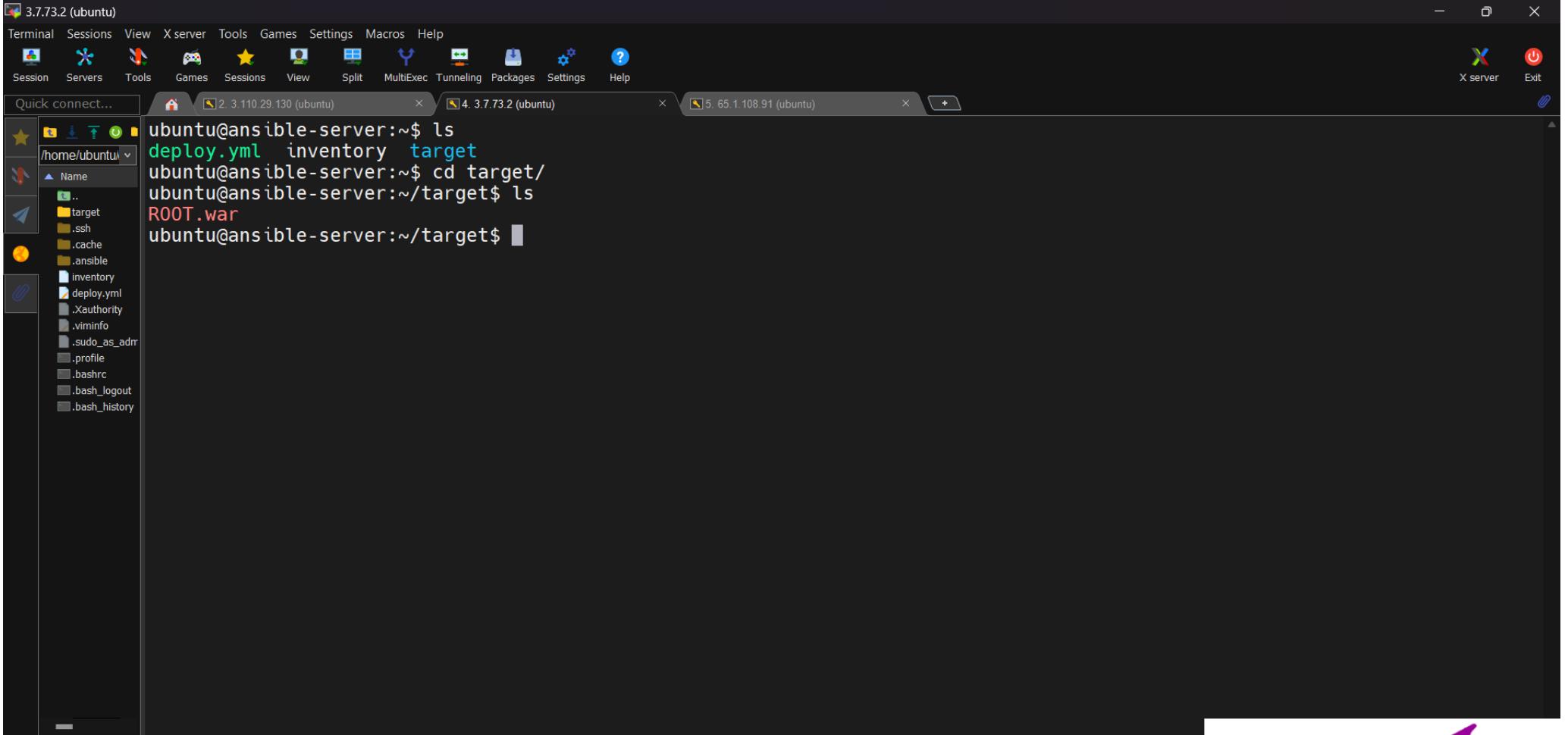


```
root@jenkins-server:~# cd /var/lib/jenkins/workspace/
root@jenkins-server:/var/lib/jenkins/workspace# ls
maven-project
root@jenkins-server:/var/lib/jenkins/workspace# cd maven-project/
root@jenkins-server:/var/lib/jenkins/workspace/maven-project# ls
pom.xml  src  target
root@jenkins-server:/var/lib/jenkins/workspace/maven-project# cd target/
root@jenkins-server:/var/lib/jenkins/workspace/maven-project/target# ls
ROOT  ROOT.war  maven-archiver
root@jenkins-server:/var/lib/jenkins/workspace/maven-project/target#
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

23°C Mostly cloudy

Step 42: Verify the creation of “target” folder in ansible-server



```
ubuntu@ansible-server:~$ ls
deploy.yml inventory target
ubuntu@ansible-server:~$ cd target/
ubuntu@ansible-server:~/target$ ls
ROOT.war
ubuntu@ansible-server:~/target$
```

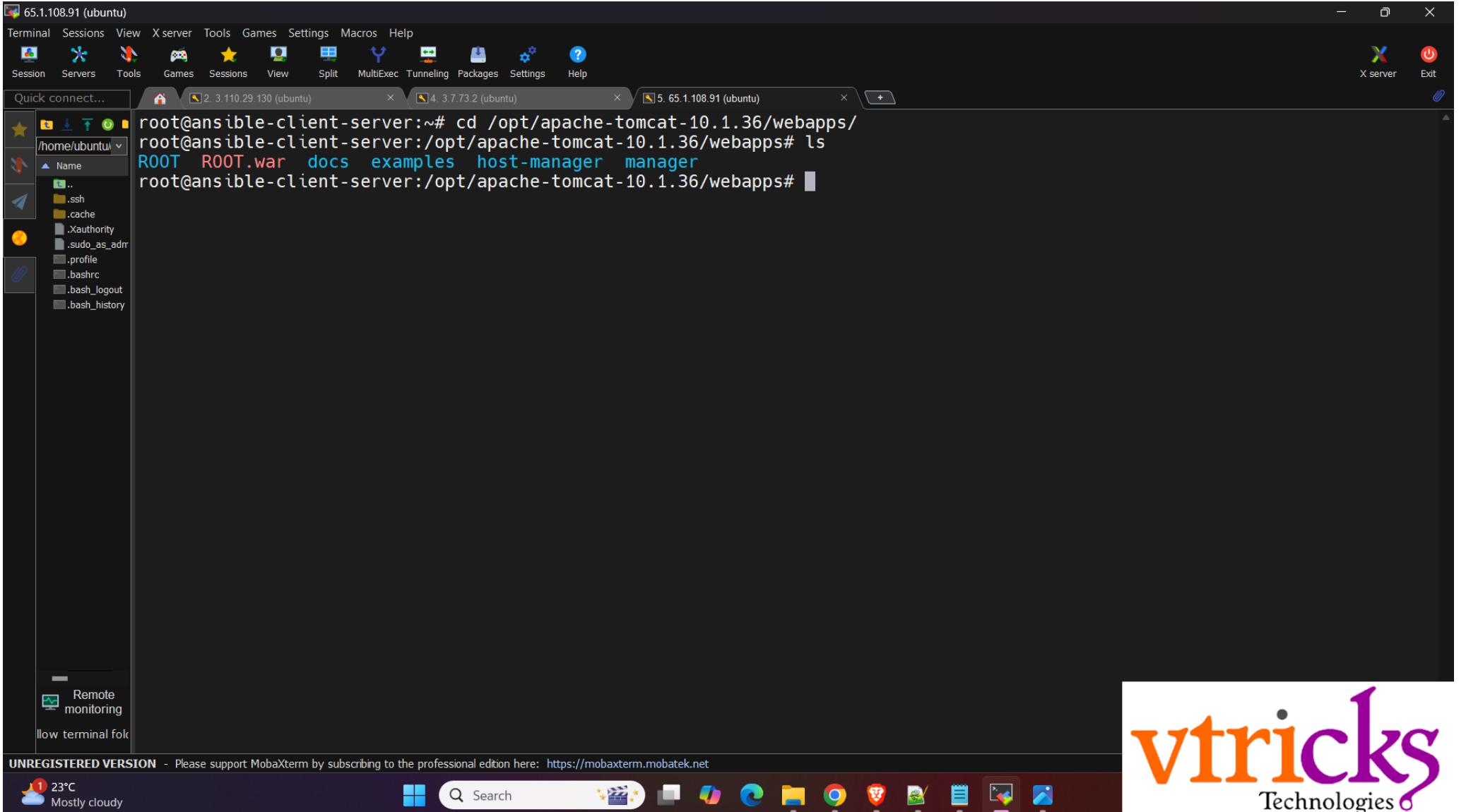
The screenshot shows a MobaXterm window with multiple sessions. Session 4 (3.7.73.2) is active and displays a terminal window. The user has run the command 'ls' in the directory '/target', which lists the file 'ROOT.war'. The terminal window has a dark background with light-colored text. The MobaXterm interface includes a sidebar with session icons and a bottom status bar.

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

23°C Mostly cloudy

vtricks Technologies

Step 43: Verify the deployment “ROOT” and “ROOT.war” file in ansible-client-server



The screenshot shows a MobaXterm interface with four tabs open:

- Tab 1: 65.1.108.91 (ubuntu) - Session menu
- Tab 2: 2. 3.110.29.130 (ubuntu) - Sessions menu
- Tab 3: 4. 3.7.73.2 (ubuntu) - Tools menu
- Tab 4: 5. 65.1.108.91 (ubuntu) - View menu

The main terminal window displays the following command output:

```
root@ansible-client-server:~# cd /opt/apache-tomcat-10.1.36/webapps/
root@ansible-client-server:/opt/apache-tomcat-10.1.36/webapps# ls
ROOT  ROOT.war  docs  examples  host-manager  manager
root@ansible-client-server:/opt/apache-tomcat-10.1.36/webapps#
```

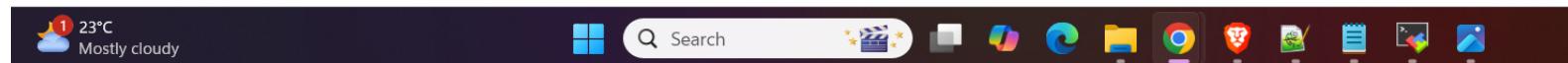
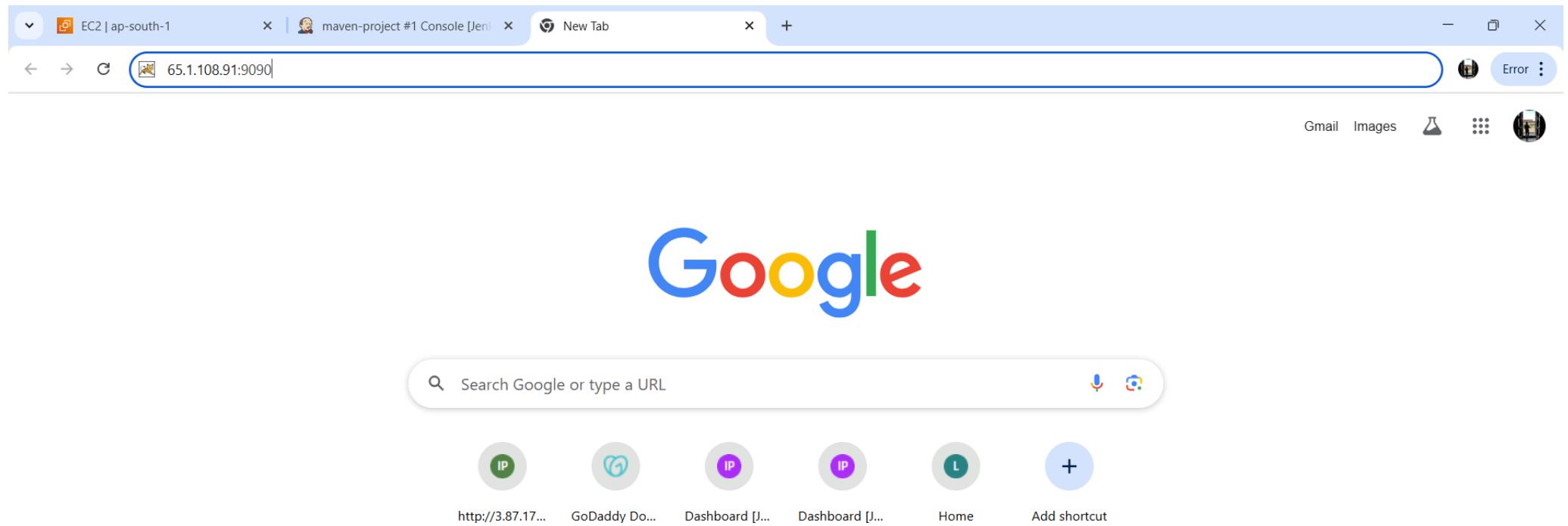
The left sidebar shows the current directory structure under /home/ubuntu:

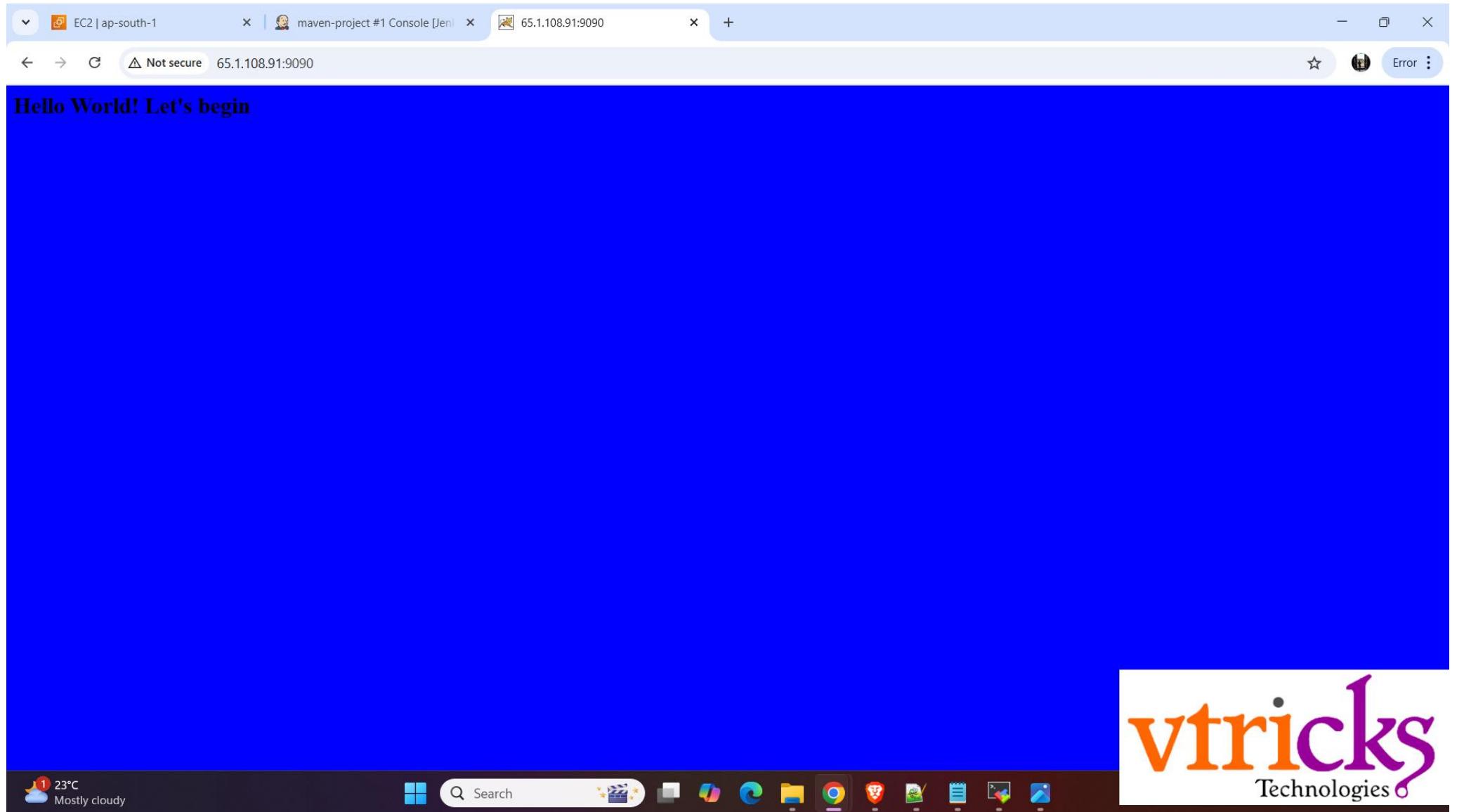
- Name
 - ..
 - .ssh
 - .cache
 - .Xauthority
 - .sudo_as_addr
 - .profile
 - .bashrc
 - .bash_logout
 - .bash_history

At the bottom of the terminal window, there is a message: "UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>".

The system tray at the bottom includes icons for weather (23°C, Mostly cloudy), search, and various application icons.

Step 44: search “<Public IP of ansible-client-server:9090>” in browser and verify the successful deployment of application





vtricks
Technologies