To set up a slave node (agent) in Jenkins using SSH on an AWS EC2 instance, follow these steps:

**Prerequisites**

1. **Jenkins Master** already installed on an EC2 instance.
2. **Slave Node** (another EC2 instance) with:
   * Jenkins user created
   * SSH access configured
   * Java installed

**Step-by-Step Guide**

**1. Create and Configure the Slave EC2 Instance**

* Launch a new EC2 instance (slave) with appropriate specs (e.g., t2.micro).
* Ensure the security group allows inbound traffic on port 22 (for SSH access).
* Install **Java** on the slave:

**sudo apt update # For Ubuntu/Debian**

**sudo apt install openjdk-11-jdk -y**

**java -version # Verify Java installation**

**2. Create a Jenkins User on the Slave Node**

* On the slave EC2 instance, create a user for Jenkins:

sudo adduser jenkins

sudo usermod -aG sudo jenkins # Give sudo privileges if needed

* Switch to the jenkins user:

bash

Copy code

su - jenkins

**3. Configure SSH Keys for Jenkins Connection**

* Generate SSH keys on your **Jenkins master** instance:

ssh-keygen -t rsa -b 4096 -C "jenkins-slave" # Leave all prompts blank for defaults

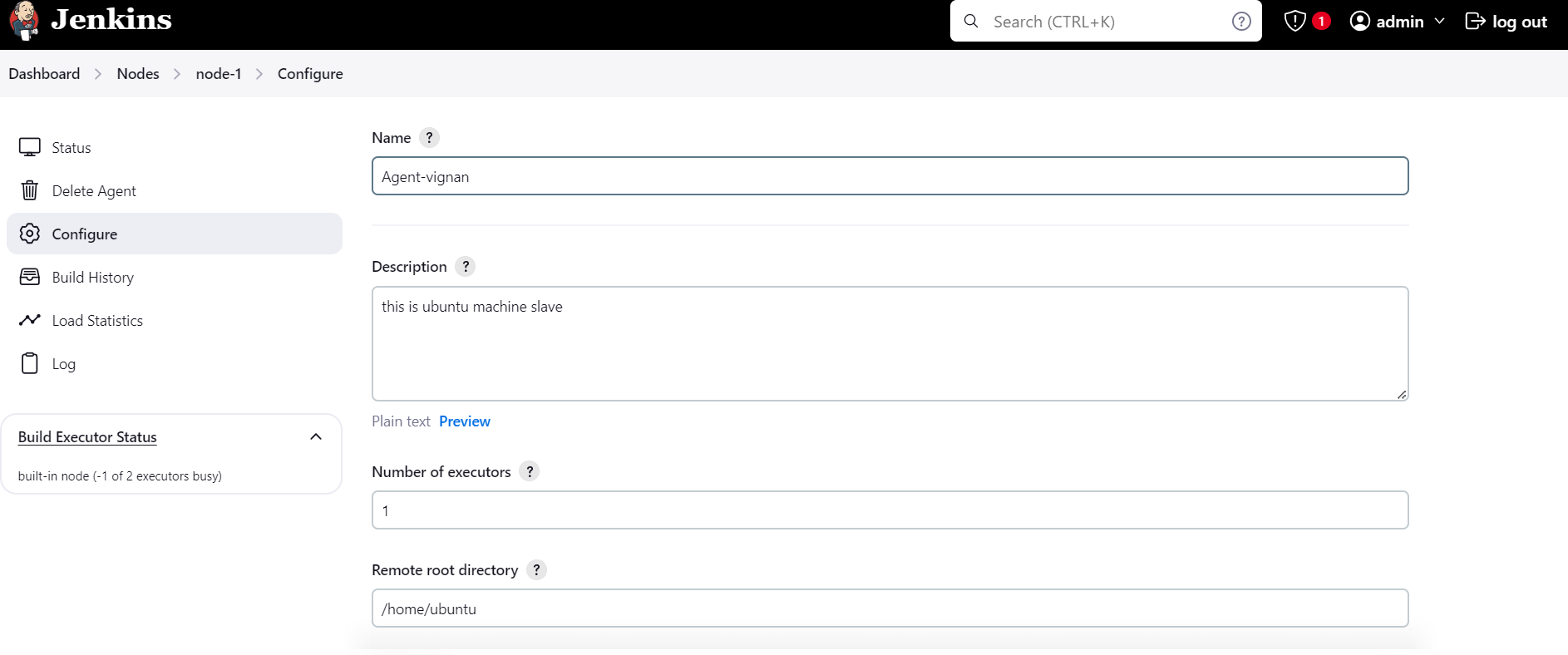
* Copy the public key to the **slave instance**:

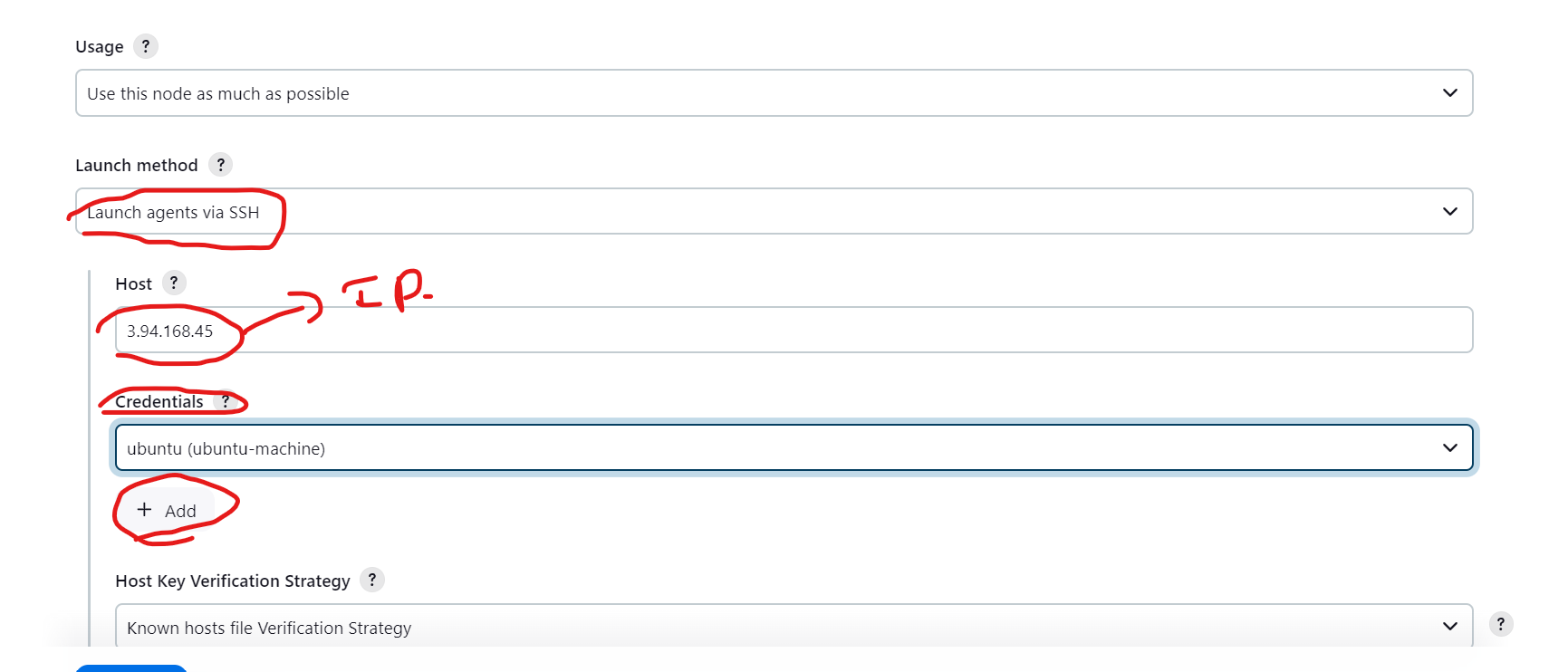
ssh-copy-id jenkins@<slave-instance-public-ip>

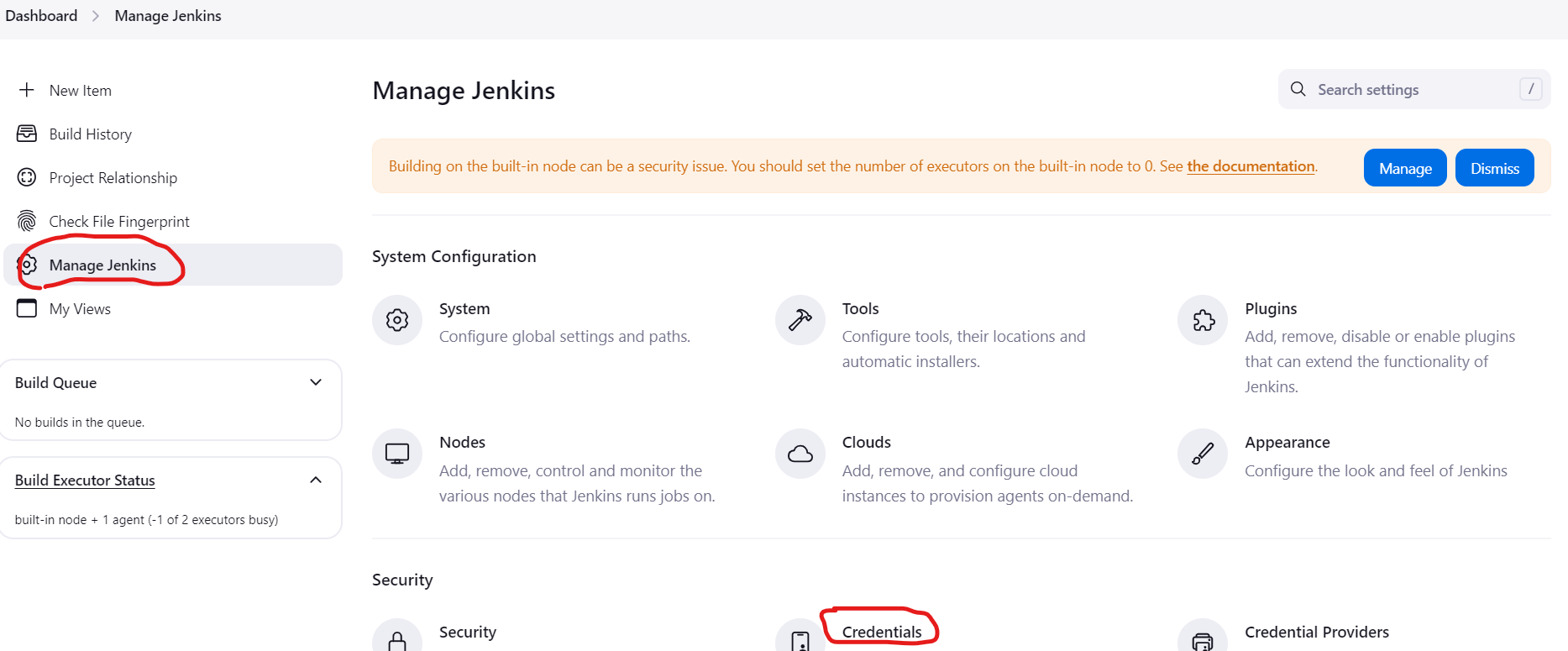
Alternatively, manually copy the contents of the public key (~/.ssh/id\_rsa.pub) and add it to the ~/.ssh/authorized\_keys file on the slave node.

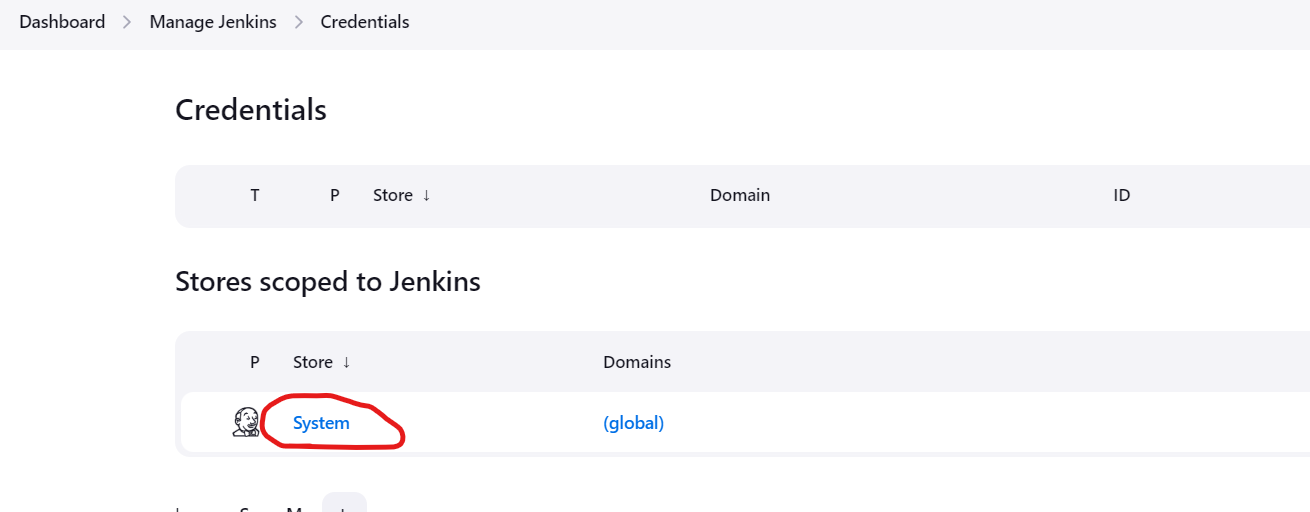
**4. Configure the Slave Node in Jenkins**

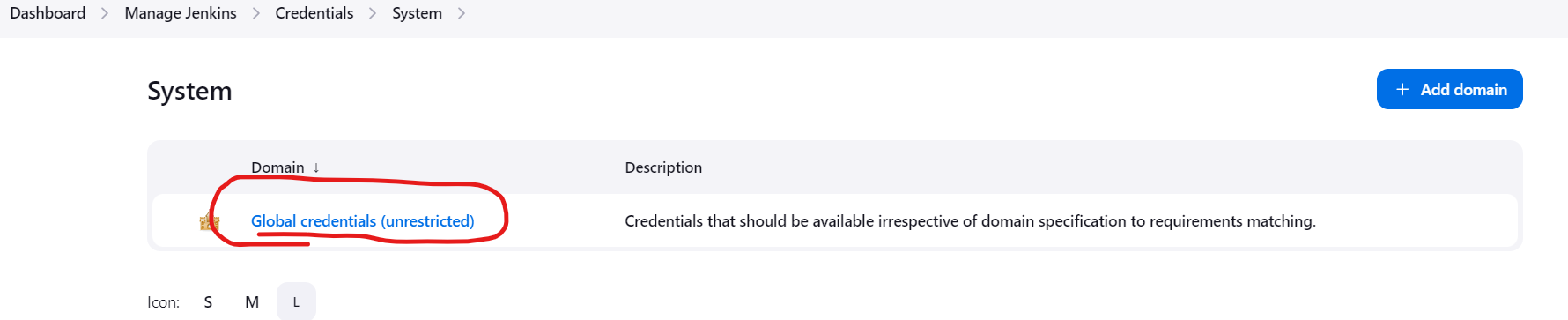
* On the Jenkins master web UI, go to **Manage Jenkins > Manage Nodes and Clouds > New Node**.
* Provide a **node name** (e.g., "Slave1") and select **Permanent Agent**.

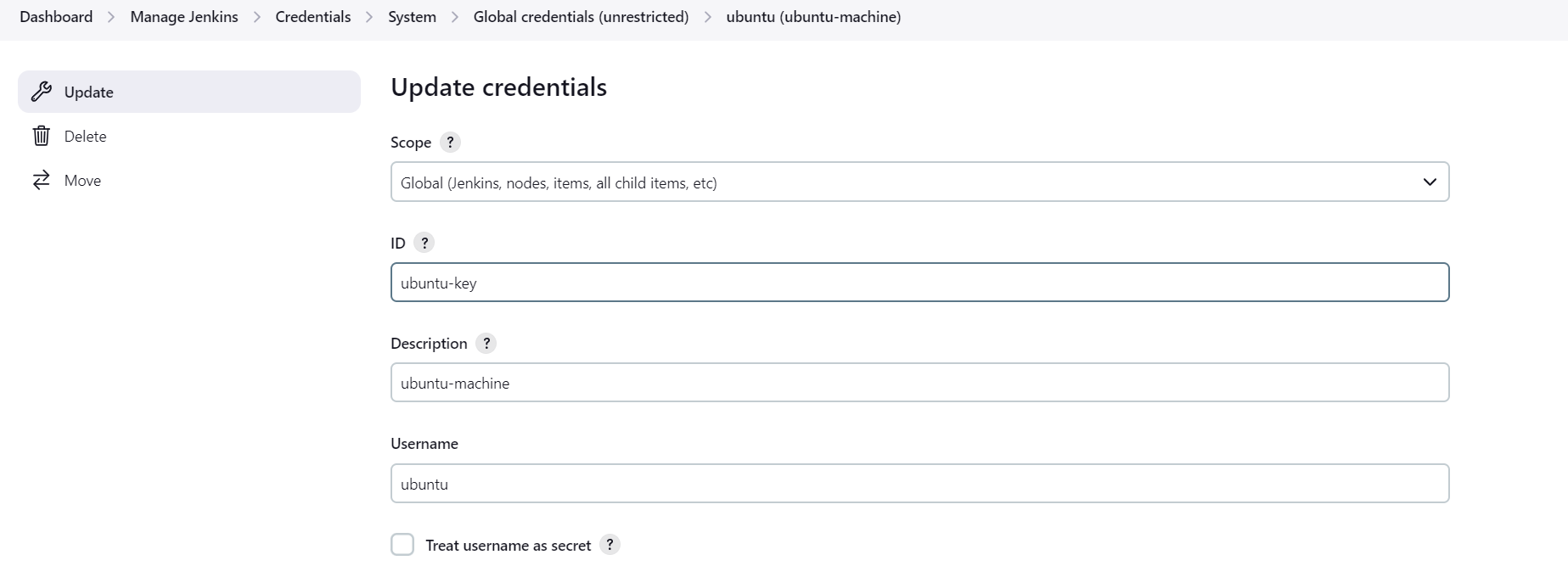




* Configure the following settings:
  + **Remote root directory**: /home/jenkins (or the home directory of the jenkins user on the slave).
  + **Usage**: Select the appropriate usage, such as "Use this node as much as possible."
  + **Launch method**: Choose **Launch agents via SSH**.
  + **Host**: Enter the public IP or DNS of the slave instance.
  + **Credentials**:
    - Add the SSH credentials for the jenkins user. In **Credentials**, select **Jenkins** → **Add** → **SSH Username with Private Key**.
  + 







Username: jenkins

* + - Private Key: Select **Enter directly**, and copy-paste the private key (~/.ssh/id\_rsa) from the master instance.
  + **Host Key Verification Strategy**: You can choose "Manually trusted key Verification Strategy" for simplicity.

**5. Test the SSH Connection**

* Click on the **Save** button and Jenkins will attempt to connect to the slave via SSH.
* If the connection is successful, the node status will show **Online**.

**6. Verify the Node is Running**

* Once the node is set up, it will appear under **Manage Nodes** with an **Online** status.
* You can test by configuring a Jenkins job to run on this specific node using the **Restrict where this project can be run** option (set to the node’s label).

**Troubleshooting**

* **Security Groups**: Ensure that the security group of the slave EC2 instance allows SSH (port 22) from the master instance.
* **Java**: Ensure that Java is installed and correctly configured on the slave instance.
* **SSH Key**: Ensure that the SSH key permissions on the slave node are set to chmod 600 ~/.ssh/authorized\_keys.

This should set up your Jenkins master to connect and use the EC2 slave node for running jobs.