

Detailed Documentation for Setting Up and Using Nexus Repository on Ubuntu

This document provides a step-by-step guide to installing, configuring, and using Nexus Repository Manager (v3) on Ubuntu. The guide will help you set up a Maven repository for deploying and retrieving artifacts using Nexus, configuring Maven to authenticate with the Nexus repository, and troubleshooting common issues.

Table of Contents

1. [Introduction](#)
 2. [Installing Nexus Repository on Ubuntu](#)
 3. [Configuring Nexus Repository](#)
 4. [Configuring Maven to Use Nexus](#)
 5. [Deploying Artifacts to Nexus Repository](#)
 6. [Accessing Artifacts from Nexus Repository](#)
 7. [Troubleshooting](#)
-

1. Introduction

Nexus Repository Manager is a popular repository manager that provides the ability to host and manage different types of repositories, such as Maven, npm, and Docker. It can be used to store and manage build artifacts, making it easier to share dependencies between teams and applications. In this guide, we will install Nexus, configure it, and set up Maven to interact with it.

2. Installing Nexus Repository on Ubuntu

Prerequisites

- A working Ubuntu machine (Ubuntu 20.04 or later recommended).
- Java 17 (or later) installed on the machine.
- Root or sudo access.

Steps to Install Nexus Repository

Download Nexus Repository

Use the following command to download the latest version of Nexus Repository Manager:

https://download.sonatype.com/nexus/3/nexus-3.85.0-03-linux-x86_64.tar.gz

1.

Extract the Tarball

Once the tarball is downloaded, extract it using the following command:

```
tar -xvzf nexus-3.85.0-03-unix.tar.gz
```

2. This will create a directory named `nexus-3.85.0-03`.

Move the Nexus Directory to a Desired Location

Move the extracted Nexus directory to a more appropriate location, like `/opt/nexus`, to keep things organized.

```
sudo mv nexus-3.85.0-03 /opt/nexus
```

- 3.

Create a Symlink for Easy Access

Create a symlink to easily access the Nexus installation from anywhere:

```
sudo ln -s /opt/nexus/bin/nexus /usr/bin/nexus
```

- 4.

Start Nexus Repository

To start the Nexus server, navigate to the `bin` directory and run:

```
cd /opt/nexus/bin  
./nexus start
```

Check the Nexus status to ensure it's running correctly:

```
./nexus status
```

5. You should see a confirmation that Nexus is running.

3. Configuring Nexus Repository

Accessing Nexus Web Interface

Open Nexus in a Web Browser

Open your web browser and navigate to:

`http://<your-server-ip>:8081`

1. For example, if you're running Nexus on an AWS instance, replace `<your-server-ip>` with the actual public IP of your server.
2. **Login to Nexus**

- Default Username: `admin`
- Default Password: The password is stored in a file located at `/opt/nexus/sonatype-work/nexus3/admin.password`.

Retrieve the password using:

```
cat /opt/nexus/sonatype-work/nexus3/admin.password
```

- 3.
4. **Change the Admin Password**

Upon first login, you will be prompted to change the default admin password for security purposes.

Setting Up Repositories in Nexus

1. Create a Maven Release Repository

- In the Nexus web interface, click on **Repositories** in the left sidebar.

- Click **Create Repository** and select **Maven 2 (hosted)**.
- Name the repository (e.g., `maven-releases`).
- Select the **Version Policy** (Release or Snapshot).
- Configure the repository and save.

2. Create a Maven Snapshot Repository

- Repeat the above process but select **Maven 2 (hosted)** and configure it as a snapshot repository (e.g., `maven-snapshots`).
-

4. Configuring Maven to Use Nexus

Update Maven's `settings.xml` File

To allow Maven to authenticate and interact with Nexus, you must configure your Maven `settings.xml` file to include the Nexus repository credentials.

1. Locate the `settings.xml` File

The `settings.xml` file is usually located in the following directory:

- Global `settings.xml`: `/etc/maven/settings.xml`
- User `settings.xml`: `~/.m2/settings.xml`

Edit `settings.xml` to Add Server Credentials

Open the `settings.xml` file and add your Nexus credentials under the `<servers>`

tag:

```
<servers>
  <server>
    <id>nexus-releases</id>
    <username>admin</username>
    <password>your-nexus-password</password>
  </server>
</servers>
```

2. Replace `your-nexus-password` with the actual password found in `/opt/nexus/sonatype-work/nexus3/admin.password`.
-

5. Deploying Artifacts to Nexus Repository

Update `pom.xml` for Deployment

Add the Nexus repository information in the `pom.xml` of your Maven project:

```
<distributionManagement>
  <repository>
    <id>nexus-releases</id>
    <name>Nexus Releases</name>
    <url>http://<your-server-ip>:8081/repository/maven-releases/</url>
  </repository>
</distributionManagement>
```

Deploy the Artifact Using Maven

Run the following Maven command to deploy your artifact:

```
mvn clean deploy
```

This command will:

- Compile the code.
 - Run tests (if configured).
 - Deploy the artifact to the `maven-releases` repository on Nexus.
-

6. Accessing Artifacts from Nexus Repository

Configure Maven to Access Nexus

To retrieve artifacts from your Nexus repository, add the repository configuration to your `pom.xml`:

```
<repositories>
  <repository>
    <id>nexus-releases</id>
    <name>Nexus Releases</name>
    <url>http://<your-server-ip>:8081/repository/maven-releases/</url>
  </repository>
</repositories>
```

Add Dependencies to Your `pom.xml`

Add the required dependencies from your Nexus repository to your project's `pom.xml`:

```
<dependency>
  <groupId>com.example</groupId>
  <artifactId>my-library</artifactId>
```

```
<version>1.0.0</version>
</dependency>
```

Run the following Maven command to fetch dependencies:

```
mvn clean install
```

7. Troubleshooting

Common Issues and Solutions

1. Nexus Not Starting

- Check Nexus logs for errors:
`/opt/nexus/sonatype-work/nexus3/log/.`
- Ensure that no other processes are using port `8081`.

2. Maven Authentication Failed

- Double-check the username and password in your `settings.xml` file.
- Make sure the `<id>` in `settings.xml` matches the repository `<id>` in `pom.xml`.

3. Artifact Not Found in Nexus

- Ensure that you've properly deployed the artifact and that the repository configuration is correct in Nexus.
- Check the Nexus web interface to confirm the artifact is present.

4. Maven Build Failures

- Review the Maven build logs for specific error messages.
- Ensure that the correct version of Java is installed.