**Sonarqube**

**How To Install SonarQube On CentOS Servers**

## **What is SonarQube?**

SonarQube is an open source tool for source code quality management, code analysis etc. It is the most widely used tool for code coverage and analysis. We can also integrate it easily with Jenkins. So, in this article, we will install SonarQube on CentOS 7.x.

## **How To Install SonarQube On CentOS**

**Important:** SonarQube requires at least **2GB** **of RAM** to run efficiently. Please check the [SonarQube official documentation](https://docs.sonarqube.org/display/SONAR/Requirements) to know the detailed prerequisites.

#### **Step 1: Update your CentOS system**

First, update the system packages by running “yum update” and reboot the server.

yum update

shutdown -r now

#### **Step 2: Install the necessary packages**

yum install wget unzip -y

#### **Step 3:  Install Java**

Now, install Java and set the environment variables.

yum install java-1.8.0-openjdk-devel.x86\_64

You can check the java version by running the following command:

java -version

Next, we need to set up the Java environment variables.

cp /etc/profile /etc/profile\_orig

echo -e "export JAVA\_HOME=/usr/lib/jvm/jre-1.8.0-openjdk" >> /etc/profile

echo -e "export JRE\_HOME=/usr/lib/jvm/jre" >> /etc/profile

source /etc/profile

Now, verify that the Java environment variables are set up properly by running the following command:

echo -e "$JAVA\_HOME\n$JRE\_HOME"

#### **Step 3: Install and configure MySQL Database**

wget <http://repo.mysql.com/mysql-community-release-el7-5.noarch.rpm>

rpm -ivh mysql-community-release-el7-5.noarch.rpm

yum install mysql-server -y

Start MySQL service

systemctl start mysqld

Configure MySQL by running mysql\_secure\_installation.

mysql\_secure\_installation

Next, create a MySQL user and Database for SonarQube via MySQL CLI.

mysql -u root -p

\*\*CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8\_general\_ci;

CREATE USER 'sonar' IDENTIFIED BY 'sonar';

GRANT ALL ON sonar.\* TO 'sonar'@'%' IDENTIFIED BY 'sonar';

GRANT ALL ON sonar.\* TO 'sonar'@'localhost' IDENTIFIED BY 'sonar';

FLUSH PRIVILEGES;\*\*

GRANT ALL ON \*.\* TO 'sonar'@'%';

Replace the “sonarqube\_db“,”sonarqube\_user” and “password” mentioned in the above command with your database name, database user and password respectively.

#### **Step 4: Create a new user for running SonarQube**

Create a new user for running SonarQube. This is because you cannot run the newer versions of elasticsearch as the root user.

useradd sonarqube

#### **5.Download and install SonarQube**

wget [https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-6.7.6.zip](https://sonarsource.bintray.com/Distribution/sonarqube/sonarqube-6.7.5.zip)

Extract it using the following command.

unzip sonarqube-6.7.6.zip

mv sonarqube-6.7.6 /opt/sonarqube

Since we are running SonarQube as a separate user, assign proper ownerships to SonarQube files.

chown sonarqube. /opt/sonarqube -R

Next, open the SonarQube configuration file “conf/sonar.properties” on your favorite text editor.

vi /opt/sonarqube/conf/sonar.properties

Enter the database details as shown below.

sonar.jdbc.username=sonar

sonar.jdbc.password=sonar

sonar.jdbc.url=jdbc:mysql://localhost:3306/sonarqube\_db?useUnicode=true&characterEncoding=utf8&rewriteBatchedStatements=true&useConfigs=maxPerformance

Open the SonarQube startup script and specify the sonarqube user details.

vi /opt/sonarqube/bin/linux-x86-64/sonar.sh

Add the following entry to it.

RUN\_AS\_USER=sonarqube

#### **6. Starting SonarQube and Setting it as a Systemd service**

You can now start SonarQube by running the following command.

/opt/sonar/bin/linux-x86-64/sonar.sh start

If you have done everything correctly, SonarQube will start listening on ports 9000. You can access it using

[**http://your\_ip\_address:9000**](http://your_ip_address:9000/)

For setting SonarQube as a Systemd Service, create a new file “sonar.service” under “/etc/systemd/system/”.

vi /etc/systemd/system/sonar.service

Now, copy the below lines into it.

[Unit]

Description=SonarQube

After=syslog.target network.target

[Service]

Type=forking

ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start

ExecStop=/opt/sonarqube/bin/linux-x86-64/sonar.sh stop

User=sonarqube

Group=sonarqube

Restart=always

[Install]

WantedBy=multi-user.target

Now, enable SonarQube service to automatically start at boot time by running the following command.

systemctl enable sonar.service

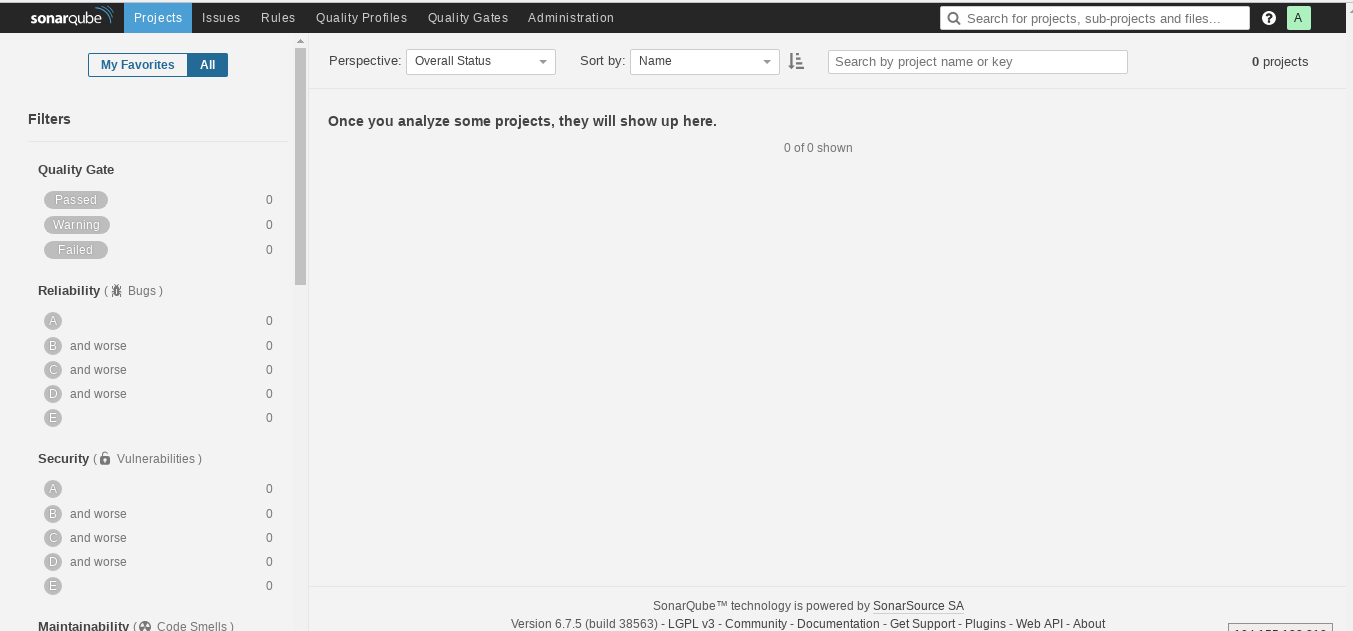
Start SonarQube by running

systemctl start sonar.service

And stop SonarQube by running

systemctl stop sonar.service

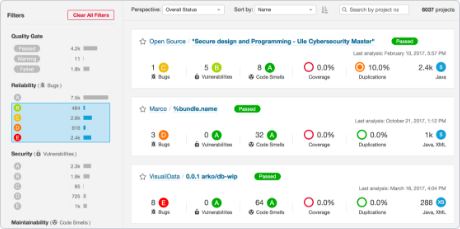
The default username and password of SonarQube is admin and admin.



Now in project directory need to give below command

$ mvn sonar:sonar

Finally we get a sonar dash board like below mentioned image



**Issues while configuration time and solutions**

# Q. [Sonarqube scanner report upload error 500](https://stackoverflow.com/questions/43143792/sonarqube-scanner-report-upload-error-500)

A. The problem *often* is that SonarQube generates a huge report and then tries to upload it in one shot, causing HTTP 500, because MySQL refuses to accept such a large request body.

The quick fix is to change the server config (my.ini file) or /**etc/my.cnf,**

to increase packed size (from default 4MB, to whatever is your

report size):

[mysqld]

max\_allowed\_packet = 16M

-->you will also need to restart both MySQL and sonar services to apply changes.