

Install Java 11

```
sudo amazon-linux-extras install java-openjdk11 -y
```

Check java path

```
sudo update-alternatives --config java
```

```
*+ 1      java-11-openjdk.x86_64
(/usr/lib/jvm/java-11-openjdk-11.0.9.11-0.amzn2.0.1.x86_64/bin/java)
```

Updated 28 july

```
*+ 1      java-11-openjdk.x86_64
(/usr/lib/jvm/java-11-openjdk-11.0.11.0.9-1.amzn2.0.1.x86_64/bin/java)
```

Set environment variable

```
vim .bash_profile
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-11.0.9.11-0.amzn2.0.1.x86_64/bin/java"
source .bash_profile
```

Updated 28 july

```
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-11.0.11.0.9-1.amzn2.0.1.x86_64/bin/java"
```

```
Check if java is set properly
echo $JAVA_HOME
```

Install postgres12

Add official PostgreSQL repository

```
sudo tee /etc/yum.repos.d/pgdg.repo<<EOF
[pgdg12]
name=PostgreSQL 12 for RHEL/CentOS 7 - x86_64
baseurl=https://download.postgresql.org/pub/repos/yum/12/redhat/rhel-7-x86_64
enabled=1
gpgcheck=0
EOF
```

Update your packages index file.

```
sudo yum makecache
```

Install postgres12

```
sudo yum install postgresql12 postgresql12-server -y
```

We need to initialize the database server for configuration files to be generated. This is done by calling the setup script.

```
sudo /usr/pgsql-12/bin/postgresql-12-setup initdb
```

To start and enable the service to start at OS boot, run the following command:

```
sudo systemctl enable --now postgresql-12
```

Expected output as below

Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql-12.service to /usr/lib/systemd/system/postgresql-12.service.

Check service status to confirm it is in running state.

```
sudo systemctl status postgresql-12
```

Login to postgres:

There is no default password for postgres so to create password

Login using postgres user (the user that was automatically created when we installed postgres)

```
sudo cat /etc/passwd
```

```
sudo su - postgres
```

```
psql
```

Expected output as below:

```
psql (12.7)
```

```
Type "help" for help.
```

```
postgres=#
```

set password for postgres

```
ALTER USER postgres WITH PASSWORD 'postgres';
```

Expected output : ALTER ROLE

Without login change password (Optional)

```
psql -c "alter user postgres with password 'postgres'"
```

Expected output : ALTER ROLE

List database users.

```
\du;
```

Create a user (database user) for sonarqube

```
create user sonarqube;
```

Expected output : CREATE ROLE

Verify user (database user) we created.

```
\du;
```

Create Password for user sonarqube that we created in above step
alter user sonarqube with password 'mypassword';

Expected output : ALTER ROLE

set password without login to database (**Optional**)

```
psql -c "alter user sonarqube with password 'mypassword'";
```

List current database

```
\l
```

Create database

```
create database sonardb;
```

Expected output : CREATE DATABASE

Verify created database

```
\l
```

Give access for user sonarqube to access sonar database or add sonar user to superuser

```
GRANT ALL PRIVILEGES ON DATABASE sonardb TO sonarqube;
```

Expected output : GRANT

Verify access for user sonarqube on sonardb database

```
\l
```

Exit the database

```
\q
```

```
exit
```

You will be landed on to the bash shell

```
psql postgres -U sonarqube
```

Note : psql: error: FATAL: Peer authentication failed for user "sonarqube"

The above error states that the peer authentication is failed which is due to login that is provided by postgres

```
sudo -i
```

```
cd /var/lib/pgsql/12/data/
```

Backup and edit

```
cp pg_hba.conf pg_hba.conf.bkp
```

```
vi /var/lib/pgsql/12/data/pg_hba.conf
```

Change method to trust

Restart postgres to apply changes
sudo systemctl restart postgresql-12
sudo systemctl restart postgresql-12.service
sudo systemctl status postgresql-12
date (confirm restart time)

Exit

Check if we are able to login for user sonarqube
psql postgres -U sonarqube
Expected Output
psql (12.7)
Type "help" for help.

postgres=>

exit

Installation & Configuration of SonarQube Server

Install sonarqube

add user
sudo useradd sonar
sudo passwd sonar
su - sonar

download binary of sonar

wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-8.1.0.31237.zip
unzip sonarqube-8.1.0.31237.zip

cd sonarqube-8.1.0.31237/bin/linux-x86-64/
./sonar.sh console

Login

Open browser enter public ip:9000
verify the installation from browser publicip:9000

Verify the database h2 database from the system

stop the server
./sonar.sh stop

```
./sonar.sh status
```

Take Backup of Configuration Files before editing:

Navigate to the config file take backup of config file

```
cd /home/ec2-user/sonarqube-8.1.0.31237/conf/
```

```
sudo cp sonar.properties sonar.properties.bkp
```

```
sudo cp wrapper.conf wrapper.conf.bkp
```

Edit sonar.properties and add below parameters

```
vim sonar.properties
```

```
sonar.jdbc.username=sonarqube
```

```
sonar.jdbc.password=mypassword
```

```
sonar.jdbc.url=jdbc:postgresql://localhost/sonardb
```

```
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-11.0.9.11-0.amzn2.0.1.x86_64/bin/java"
```

```
source .bash_profile
```

Updated 28 july

```
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-11.0.11.0.9-1.amzn2.0.1.x86_64/bin/java"
```

Edit wrapper.conf it is to help sonar locate the java path

```
cd /home/ec2-user/sonarqube-8.1.0.31237/conf
```

```
vim wrapper.conf
```

```
wrapper.java.command=/usr/lib/jvm/java-11-openjdk-11.0.11.0.9-1.amzn2.0.1.x86_64/bin/java
```

Set the

```
sudo sysctl -w vm.max_map_count=262144
```

```
max_map_count:
```

This file contains the maximum number of memory map areas a process may have. Memory map areas are used as a side-effect of calling malloc, directly by mmap and mprotect, and also when loading shared libraries.

While most applications need less than a thousand maps, certain programs, particularly malloc debuggers, may consume lots of them, e.g., up to one or two maps per allocation.
The default value is 65536.

Recommended values to be set for sonar in linux

sudo -i

```
sudo sysctl -w vm.max_map_count=524288
```

```
sudo sysctl -w fs.file-max=131072
```

```
sudo ulimit -n 131072
```

```
sudo ulimit -u 8192
```

Start the server and check

```
cd /home/ec2-user/sonarqube-8.1.0.31237/bin/linux-x86-64
```

```
./sonar.sh console
```

The screenshot shows the SonarQube Administration interface in a web browser. The top navigation bar includes links for Projects, Issues, Rules, Quality Profiles, Quality Gates, and Administration. The Administration page is active, and the 'System' tab is selected under the 'Configuration' section. The system information shows the Server ID as ATEE8CF2-AXo6KaA3-DcuwPx5U6zu and the Version as 8.1.0.31237. A 'Copy ID information' button is available. The 'System' section lists various configuration items with their status (indicated by red or green icons) and values:

Item	Status	Value
Docker	✗	
High Availability	✗	
Official Distribution	✓	
Force authentication	✗	
Home Dir		/home/ec2-user/sonarqube-8.1.0.31237
Data Dir		/home/ec2-user/sonarqube-8.1.0.31237/data
Temp Dir		/home/ec2-user/sonarqube-8.1.0.31237/temp
Processors		2
Lines of Code		0

The 'Database' section lists the following configuration:

Item	Value
Database	PostgreSQL
Database Version	12.7
Username	sonarqube
URL	jdbc:postgresql://localhost/sonar
Driver	PostgreSQL JDBC Driver
Driver Version	42.2.8

Start the server

```
cd /home/ec2-user/sonarqube-8.1.0.31237/bin/linux-x86-64
```

```
./sonar.sh console
```

ERROR#####

```
[ec2-user@ip-172-31-24-114 linux-x86-64]$ ./sonar.sh console
```

Running SonarQube...

wrapper | --> Wrapper Started as Console

wrapper | Launching a JVM...

jvm 1 | Wrapper (Version 3.2.3) <http://wrapper.tanukisoftware.org>

jvm 1 | Copyright 1999-2006 Tanuki Software, Inc. All Rights Reserved.

jvm 1 |

jvm 1 | 2021.06.24 08:03:19 INFO app[] [o.s.a.AppFileSystem] Cleaning or creating temp directory /home/ec2-user/sonarqube-8.1.0.31237/temp

jvm 1 | 2021.06.24 08:03:19 INFO app[] [o.s.a.es.EsSettings] Elasticsearch listening on /127.0.0.1:9001

jvm 1 | 2021.06.24 08:03:19 INFO app[] [o.s.a.ProcessLauncherImpl] Launch process[[key='es', ipcIndex=1, logFilenamePrefix=es]] from [/home/ec2-user/sonarqube-8.1.0.31237/elasticsearch]:

/home/ec2-user/sonarqube-8.1.0.31237/elasticsearch/bin/elasticsearch

jvm 1 | 2021.06.24 08:03:19 INFO app[] [o.s.a.SchedulerImpl] Waiting for Elasticsearch to be up and running

jvm 1 | OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.

jvm 1 | 2021.06.24 08:03:20 INFO app[] [o.e.p.PluginsService] no modules loaded

jvm 1 | 2021.06.24 08:03:20 INFO app[] [o.e.p.PluginsService] loaded plugin

[org.elasticsearch.transport.Netty4Plugin]

jvm 1 | ERROR: [1] bootstrap checks failed

jvm 1 | [1]: max virtual memory areas vm.max_map_count [65530] is too low, increase to at least [262144]

jvm 1 | 2021.06.24 08:03:27 WARN app[] [o.s.a.p.AbstractManagedProcess] Process exited with exit value [es]: 78

jvm 1 | 2021.06.24 08:03:27 INFO app[] [o.s.a.SchedulerImpl] Process[es] is stopped

jvm 1 | 2021.06.24 08:03:27 INFO app[] [o.s.a.SchedulerImpl] SonarQube is stopped

wrapper | <-- Wrapper Stopped

Try login with user to check if we are able to access postgres

```
psql postgres -U sonarqube
```

If not then try restarting the postgres server and try again login

```
psql postgres -U sonarqube
```

#####

To reset password of postgres 12 in case we forget

Backup and edit

Edit /var/lib/pgsql/12/data/pg_hba.conf

Change method to trust

Restart postgres

systemctl restart postgresql-12.service

su -postgres

Set new password

postgres=#

ALTER USER postgres WITH PASSWORD 'postgres';

systemctl restart postgresql-12.service
