

Monitoring

The SonarQube Server offers visibility about what happens internally through the exposure of JMX MBeans. ElasticSearch (ES) is used for searching in SonarQube, and there are multiple dedicated tools for ES monitoring.

Exposed JMX MBeans

In addition to the classical Java MBeans providing information about the ClassLoader, OS, Memory and Threads you have access to four more MBeans in the SonarQube Server:

- ComputeEngine
- Database
- ElasticSearch
- SonarQube

All these MBeans are read-only. It's not possible to modify or reset their values in real time.

ComputeEngineTasks MBean

Attribute Name	Description
ProcessingTime	Measure the time (in ms) spent to process Background Tasks since the last restart of SonarQube. Its value will always increase and will be reset by a restart of SonarQube. This measure is very powerful when: <ul style="list-style-type: none">▪ combined with SuccessCount and ErrorCount measures to get the average time to handle a Background Task▪ used to understand how much time the SonarQube Server is spending during a day to handle Background Tasks. It gives you an indication of the load on your server.
ErrorCount	Number of Background Tasks which failed since the last restart of SonarQube
PendingCount	Number of Background Tasks waiting to be processed since the last restart of SonarQube
InProgressCount	Number of Background Tasks currently under processing. Its value is either 1 or 0, since SonarQube can process only one task at a time.
SuccessCount	Number of Background Tasks successfully processed since the last restart of SonarQube
WorkerCount	Number of Background Tasks that can be processed at the same time

Note:

- the total number of Background Tasks handled since the last restart of SonarQube is equal to SuccessCount + ErrorCount
- these values are reset to their default values by restarting SonarQube

Database MBean

Same attributes are available for both ComputeEngineServer and WebServer.

Attribute Name	Description
MigrationStatus	Possible values are: UP_TO_DATE, REQUIRES_UPGRADE, REQUIRES_DOWNGRADE, FRESH_INSTALL (only available for WebServer).
PoolActiveConnections	Number of active database connections
PoolIdleConnections	Number of database connections waiting to be used
PoolInitialSize	Initial size of the database connections pool.
PoolMaxActiveConnections	Maximum number of active database connections
PoolMaxIdleConnections	Maximum number of database connections waiting to be used
PoolMaxWaitMillis	In milliseconds
PoolRemoveAbandoned	Possible values : true, false

PoolRemoveAbandonedTimeoutSeconds	In Seconds
-----------------------------------	------------

ElasticSearch MBean

Attribute Name	Description
NumberOfNodes	Number of ES Nodes in your SonarQube installation.
State	Possible values are: GREEN, YELLOW, RED

SonarQube MBean

Attribute Name	Description
LogLevel	Log Level: INFO, DEBUG, TRACE
ServerId	SonarQube Server ID
Version	SonarQube Version

How to Activate JMX ?

Local Access

There is nothing to activate to view SonarQube MBeans if your tool is running on the same server as the SonarQube Server.

Remote Access

Here are examples of configuration to activate remote access to JMX MBeans.

For the WebServer:

```
# JMX WEB - 10443/10444
sonar.web.javaAdditionalOpts=-Dcom.sun.management.jmxremote=true -Dcom.sun.management.jmxremote.ssl=false -Dcom.
sun.management.jmxremote.authenticate=true -Dcom.sun.management.jmxremote.port=10443 -Dcom.sun.management.
jmxremote.rmi.port=10444 -Dcom.sun.management.jmxremote.password.file=/opt/sonarsource/sonar/conf/jmxremote.
password -Dcom.sun.management.jmxremote.access.file=/opt/sonarsource/sonar/conf/jmxremote.access
```

For the SearchServer:

```
sonar.search.javaAdditionalOpts=-Dcom.sun.management.jmxremote=true -Dcom.sun.management.jmxremote.ssl=false -
Dcom.sun.management.jmxremote.authenticate=true -Dcom.sun.management.jmxremote.port=19010 -Dcom.sun.management.
jmxremote.rmi.port=19011 -Dcom.sun.management.jmxremote.password.file=/opt/sonarsource/sonar/conf/jmxremote.
password -Dcom.sun.management.jmxremote.access.file=/opt/sonarsource/sonar/conf/jmxremote.access
```

For the ComputeEngine:

There is no specific `javaAdditionalOpts` entry, simply amend the `sonar.ce.javaOpts` one.

Example of `jmxremote.access`:

```
#
# JMX Access Control file
#
reader readonly
admin readwrite \
    create javax.management.monitor.*,javax.management.timer.*,com.sun.management.*,com.oracle.
jrockit.* \
    unregister
```

Example of `jmxremote.password`:

```
#
# JMX Access Password file
#
reader readerpassword
admin adminpassword
```

Note: on *jmxremote.password*, you should apply `chmod 600` or `400` for security reasons.

How to Monitor Elasticsearch?

Configure in *conf/sonar.properties* the entry `sonar.search.httpPort`. This will activate the Elasticsearch REST End-Point required by most ES Monitoring Tools.

Here what you can get using [Elastic HQ](#) (Free, OS) :

The screenshot displays the Elastic HQ interface. At the top, there's a navigation bar with 'Elastic HQ', a URL 'http://localhost:9010', and a 'Connect' button. Below this, there are tabs for 'sonarqube', 'Node Diagnostics', and 'Indices'. The main content area is titled 'Cluster Overview' and shows several key statistics:

- 1 Nodes
- 26 Total Shards
- 26 Successful Shards
- 6 Indices
- 1,736 Documents
- 1.2MB Size

Below the statistics, there are two main sections: 'Cluster Health' and 'Indices'.

Cluster Health

Property	Value
Status	Green
Timed Out?	false
# Nodes	1
# Data Nodes	1
Active Primary Shards	26
Active Shards	26
Relocating Shards	0
Initializing Shards	0
Unassigned Shards	0

Indices

Index	# Docs	Primary Size	# Shards	# Replicas	Status
views	0	720.0B	5	0	open
users	1	5.4KB	5	0	open
tests	398	66.3KB	5	0	open
rules	636	989.6KB	1	0	open
issues	243	93.9KB	5	0	open
activities	458	63.0KB	5	0	open