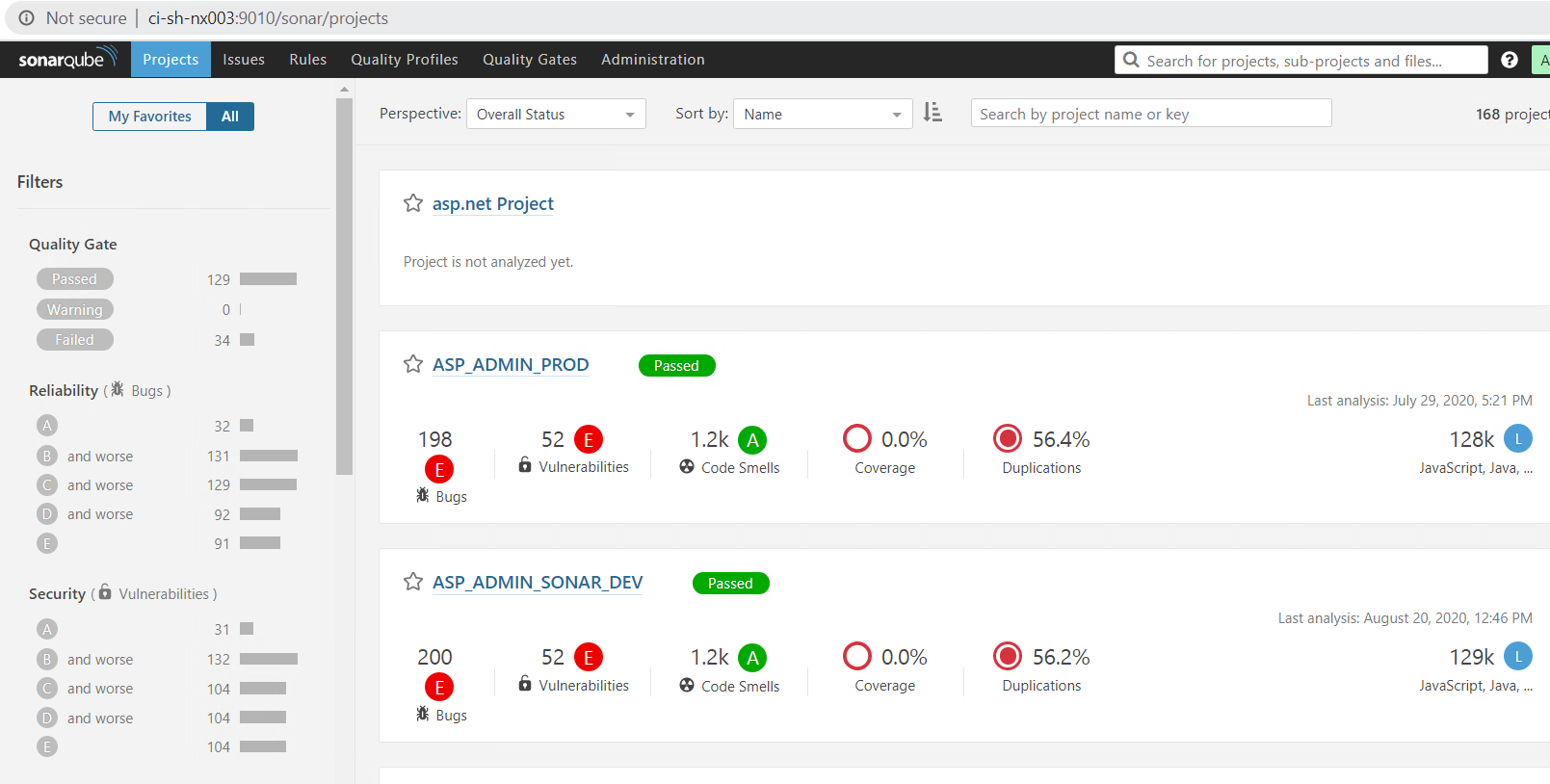
**Sonar Dashboard:**

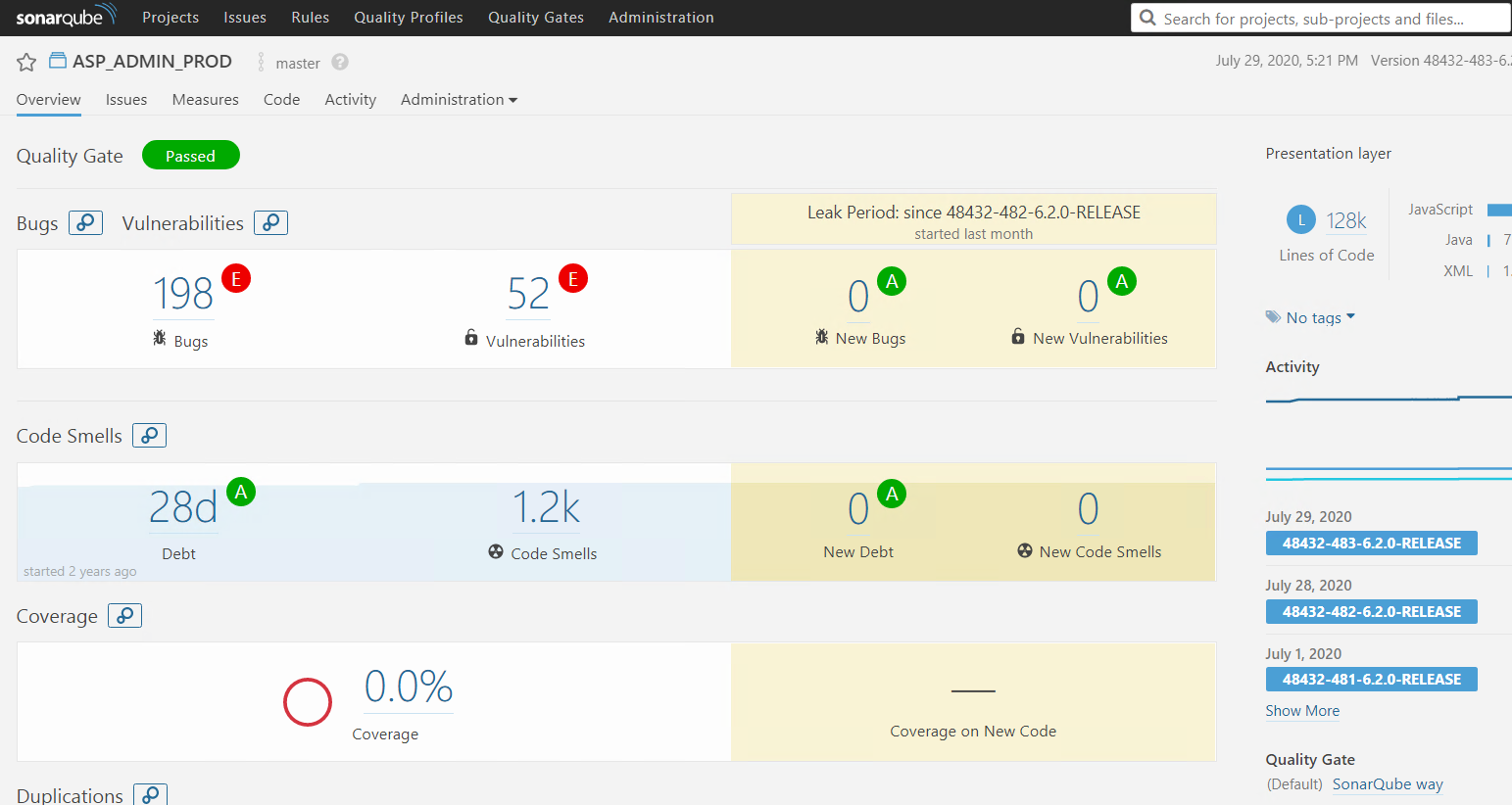
* Once we connected, login with admin/admin credentials
* We can create projects from UI, it is just UI, we can browse the violations etc
* By default, sonar supports java, if we want to use it for any other language, we need to install plugins under system inside the sonar webapp
* Now after installing the sonar plugin, under build section, we can able to see below

**Projects:**

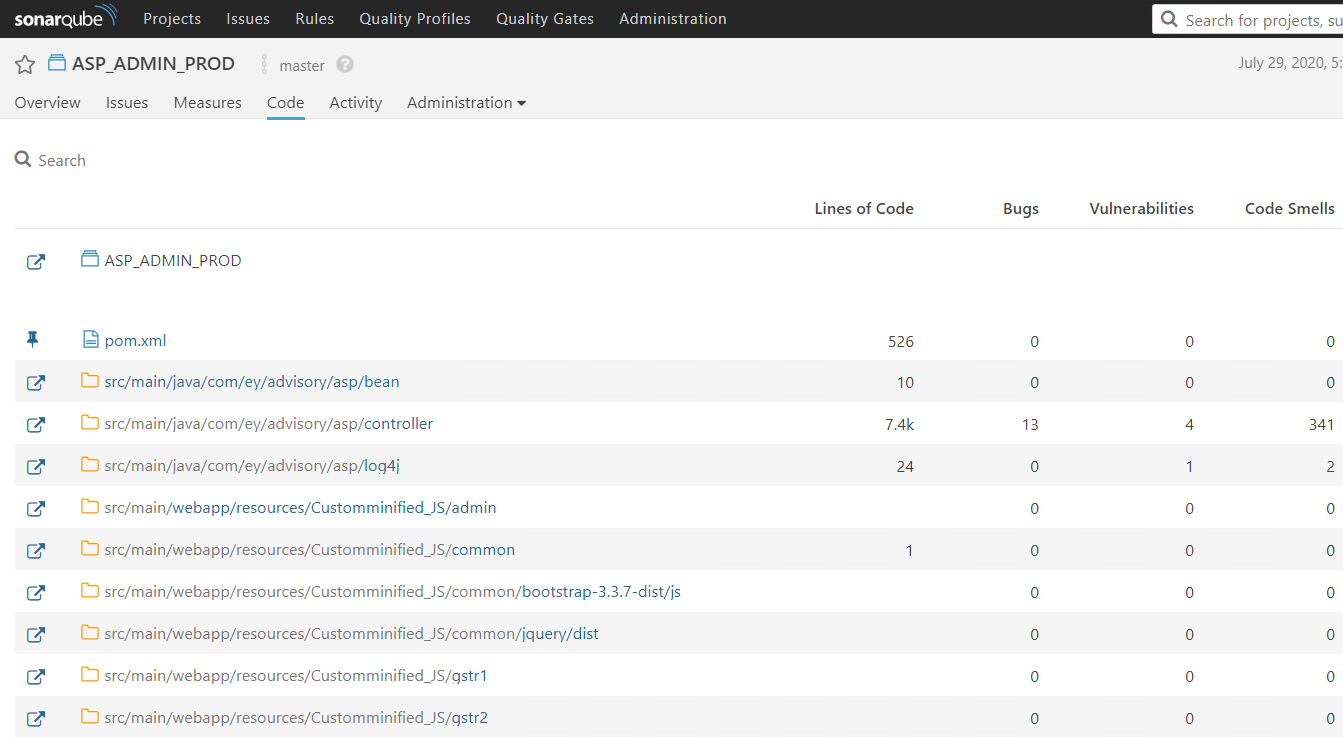
* We can view all the scanned projects under projects option as below.



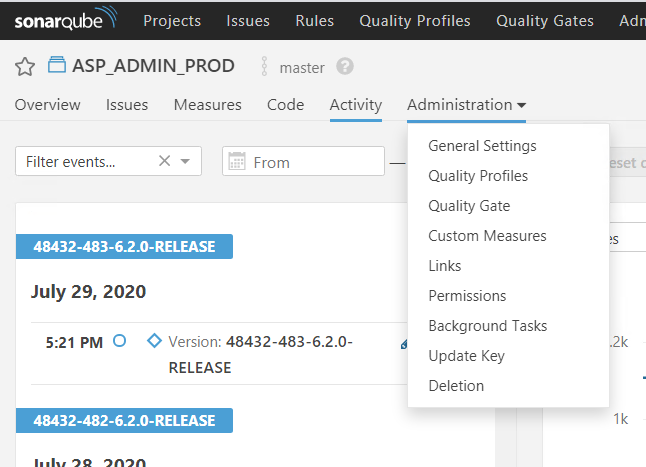
* We can also add those projects into favourites.



* We can also view the complete project details as above, we can also view the old release scanned data in the right side



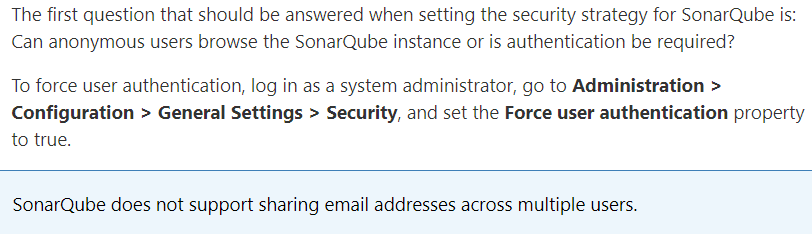
* We can see the code under the **“code”** option as above.



* Under the administration option when we open the project, we can view/perform the above actions.

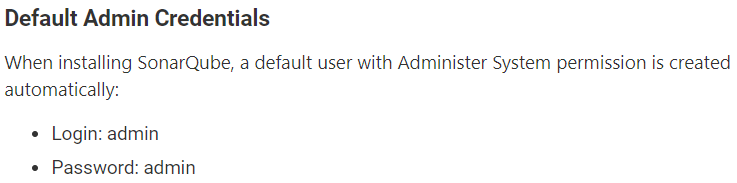
**Administration:**

**Force user authentication:**



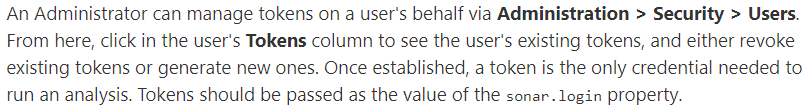
By default, anyone can view the projects and code in sonarqube with the URL without logging in. we can enable the “force user authentication” option to give a login prompt to access the project details

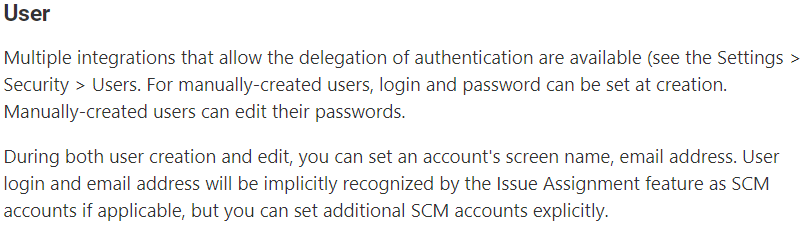
**Default admin credentials:**



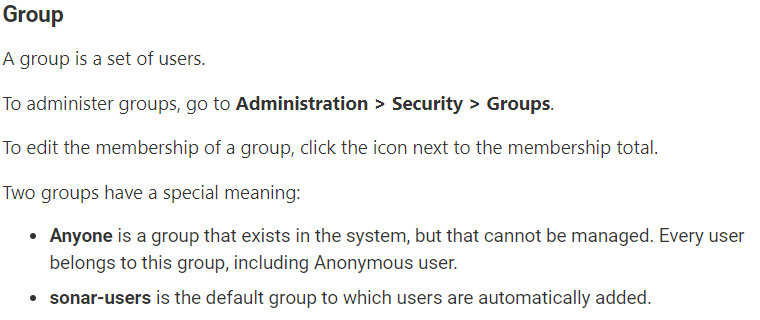
We can create users, add groups and provide permissions in administration 🡪 security settings

**Users:**

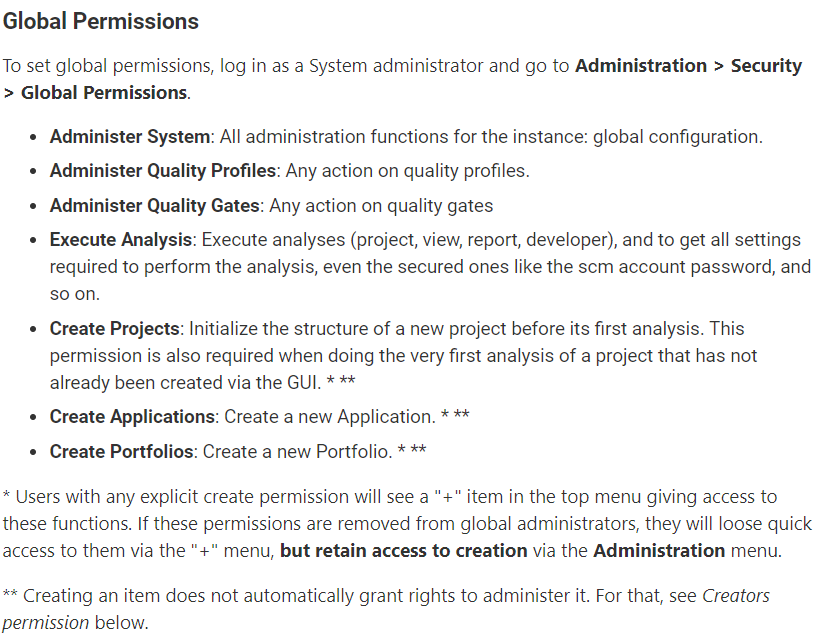




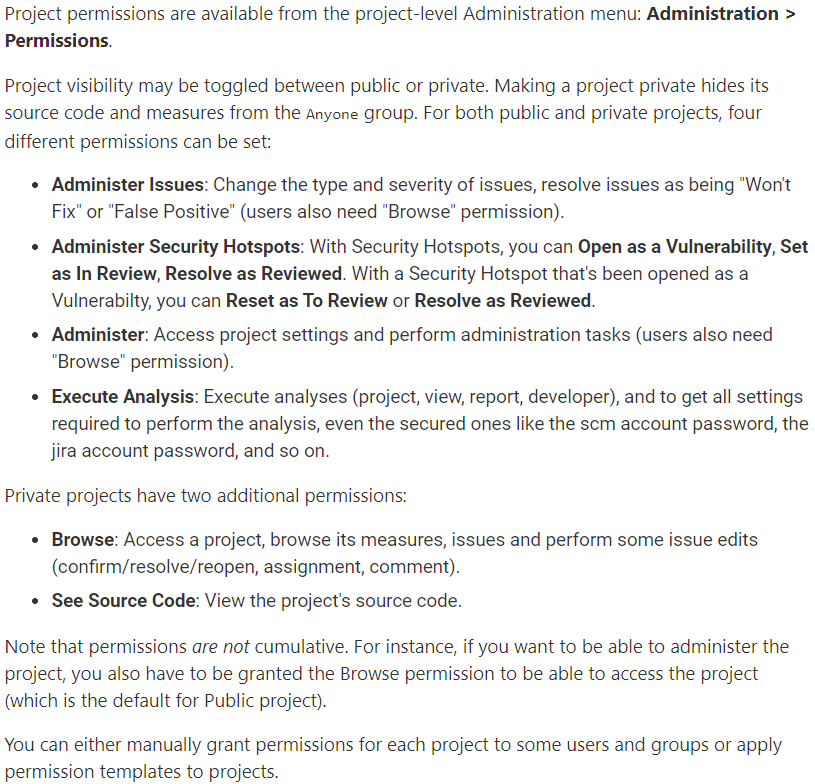
**Groups:**



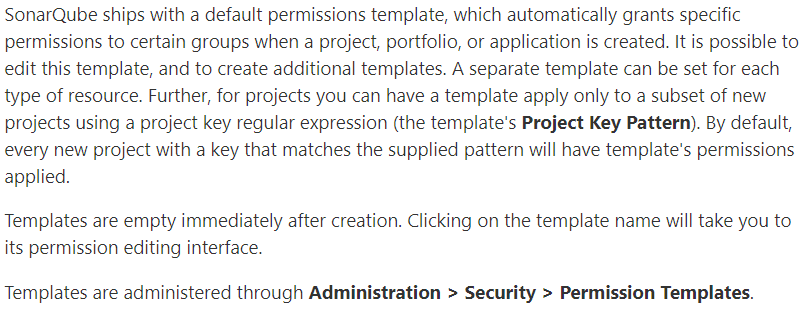
**Global permissions:**



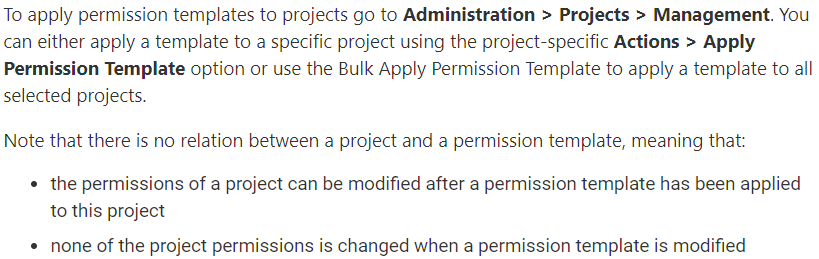
**Project permissions:**

****

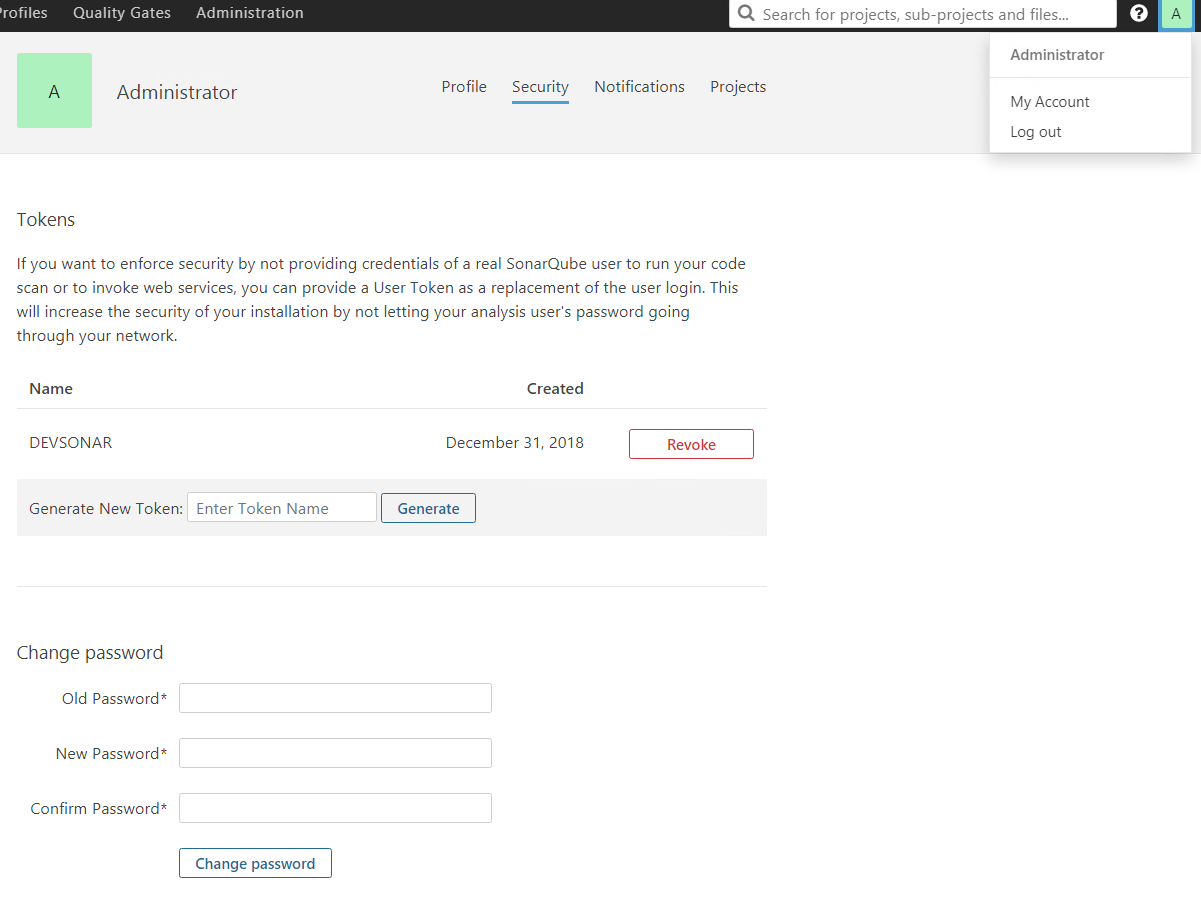
**Permission template:**



**Reset project permissions to a template:**



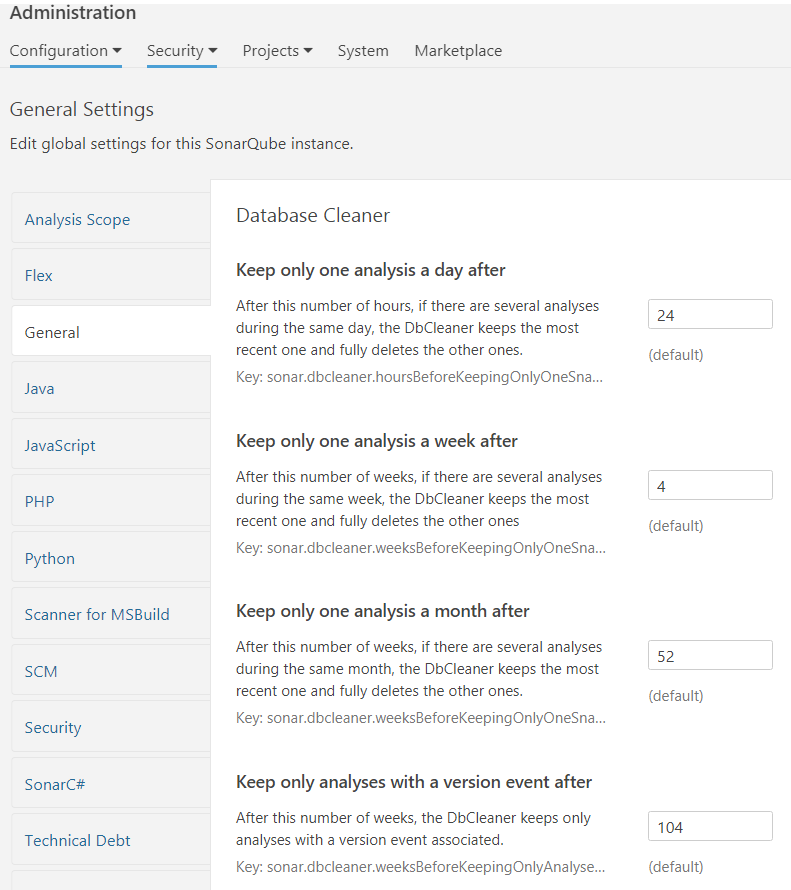
**Access Token:**



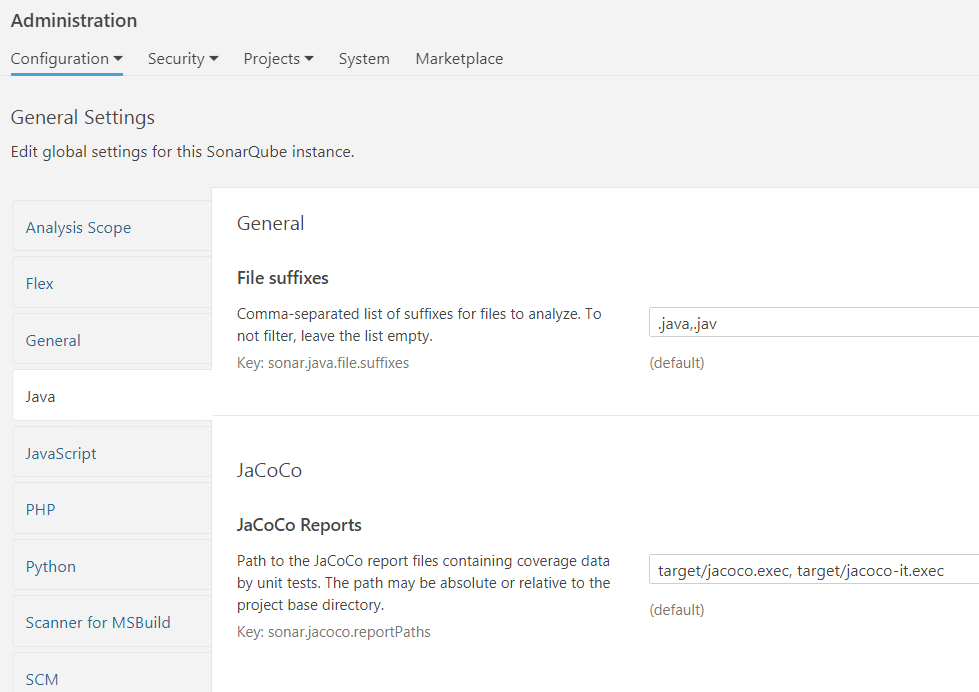
* Under my account settings, we can change the password of current user or generate an access token which can be used for authentication through sonar-scanner.

**General settings:**

* Under administration -> general settings, we can change the below kind of settings like how long we need to keep the analysis in database.

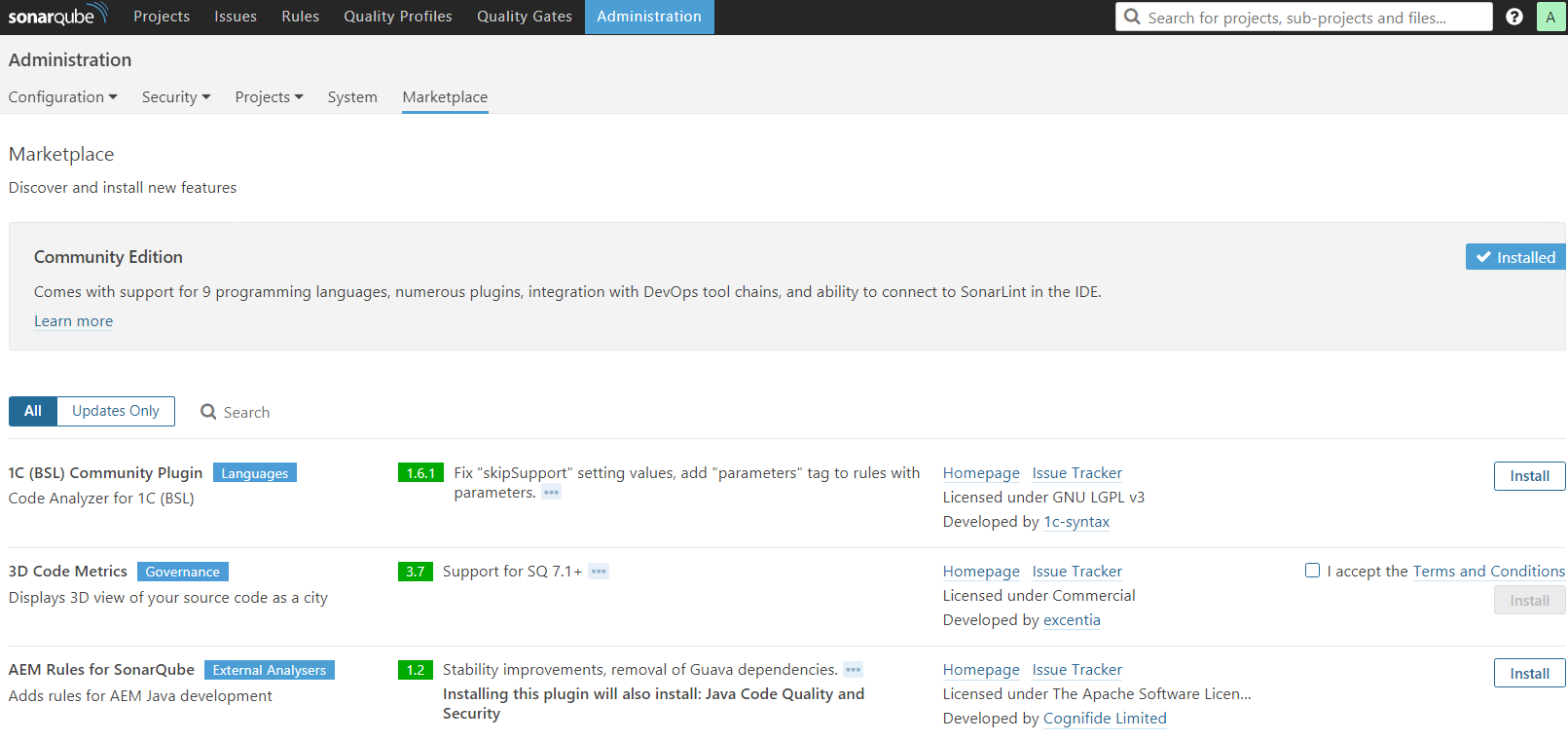


* We can also edit the settings based on the language of code like which kind of files need to scan etc. as below.



**Masketplace/plugins:**

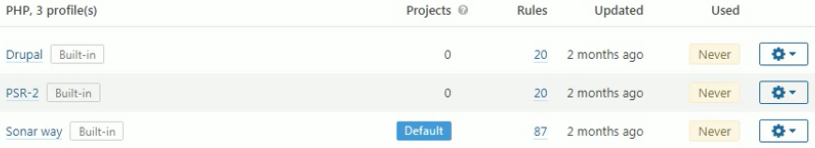
* We can download the required plugin to scan the code from administration🡪marketplace option.
* We can install new plugins or update the old ones as below.

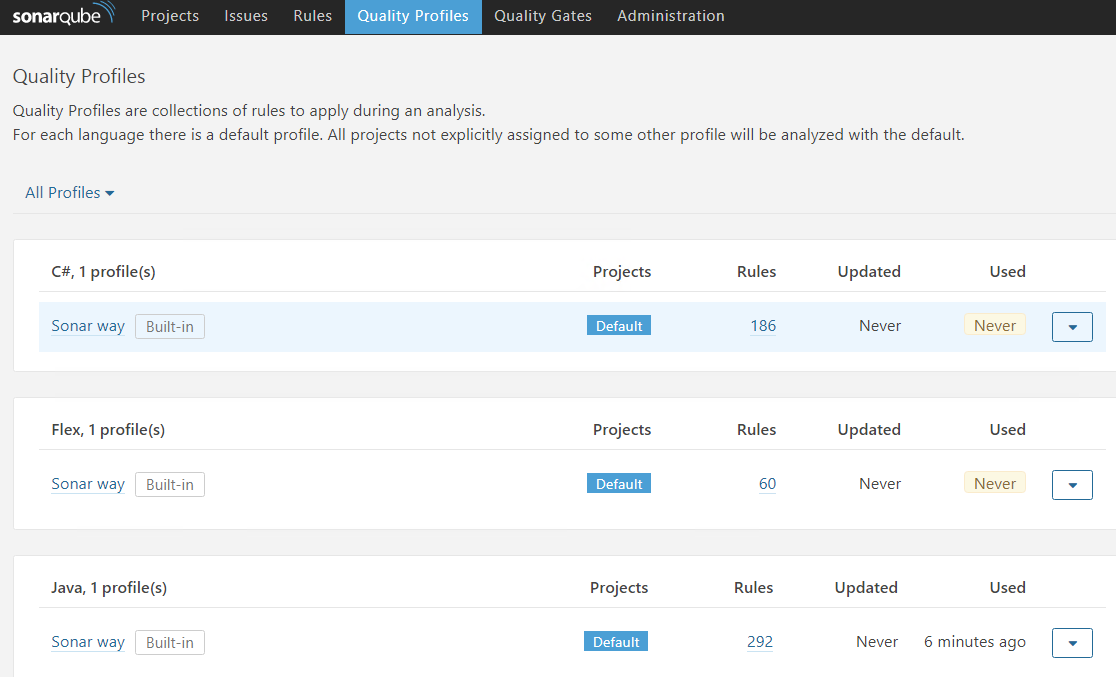


* We can either install the plugin from sonar\_dashboard🡪administration🡪marketplace.
* Or we can download the plugin from sonar official website and place it in **SONAR\_HOME/extensions/plugins** folder and restart the sonar server

**Quality Profiles:**

* We can download the profiles in sonar dashboard from administration🡪marketplace and search for the plugin related to the language of code
* After installing it, we can see the plugins under quality profiles, we can also see how many rules there in a profile by default are as below

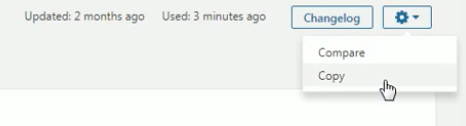




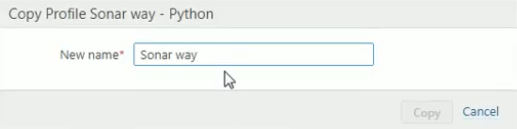
We can click on the profile see how many active and inactive rules there are



* If we want to activate more rules, then we need to create a custom profile by selecting copy option on the settings symbol from the default profile as below



* Now give the name to the profile



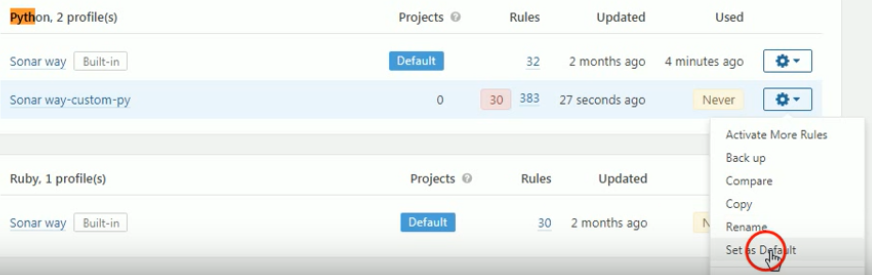
* Now click on activate more in the custom profile



* Now, we can activate whatever rule we want or we can activate all the rules by selecting bulk change as below

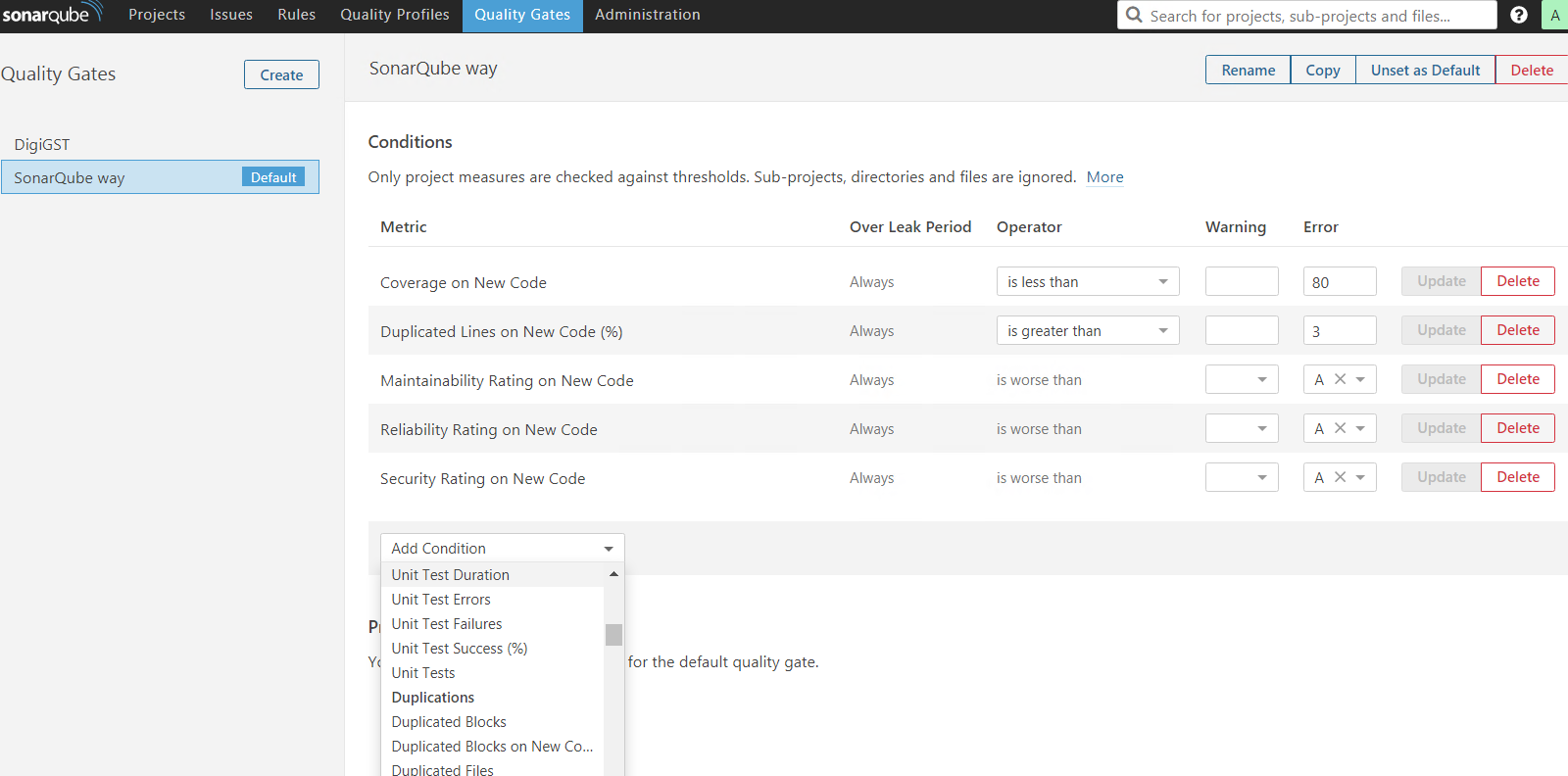


* Once the profile is created, we can set this as a default one so that from next time when we analyse the python code, it will use the custom profile which has more rules



**Quality gate:**

* Quality gates are used to add the conditions in sonarqube code scan.
* Below are the default conditions in sonar. We can modify or add more conditions or create new quality gate and set it as default.



* We can add rules of bugs, vulnerabilities, blockers etc. if the code scan met the threshold, then the final scanning report will be passed. Otherwise, it will be failed
* We can add conditions in Jenkins or azuredevops to hit the sonar URL and see if the project is passed or not. We can set as build should gets failed automatically if sonar quality gate is failed.
* It generally takes some time in the sonar dashboard to get the completed report even after the sonar-scanner completes its work

**Sonar properties:**

**sonar.host.url=**https://tdsdevson.eyasp.in/

**sonar.projectName=**TDS-SinglePan-Validation

**sonar.projectKey=**TDS-SinglePan-Validation

**sonar.language=**py

**sonar.login=**1bad290f3d2528a6e00b003ca843e2336d5d4492

**sonar.projectVersion=**1.0 - the project version

**sonar.python.pylint=/**usr/local/bin/pylint - the path to the pylin

**sonar.projectBaseDir=**D:\\python -if the code base directory is in different place from where we are running sonar analysis, then we can use it

**sonar.sources=**py\_scripts - the path of scripts inside the base directory which needs to be analysed

**sonar.test=**test – test directory

**-Dpeoject.settings=**../myproject.properties – if all the properties are in a file