SonarQube

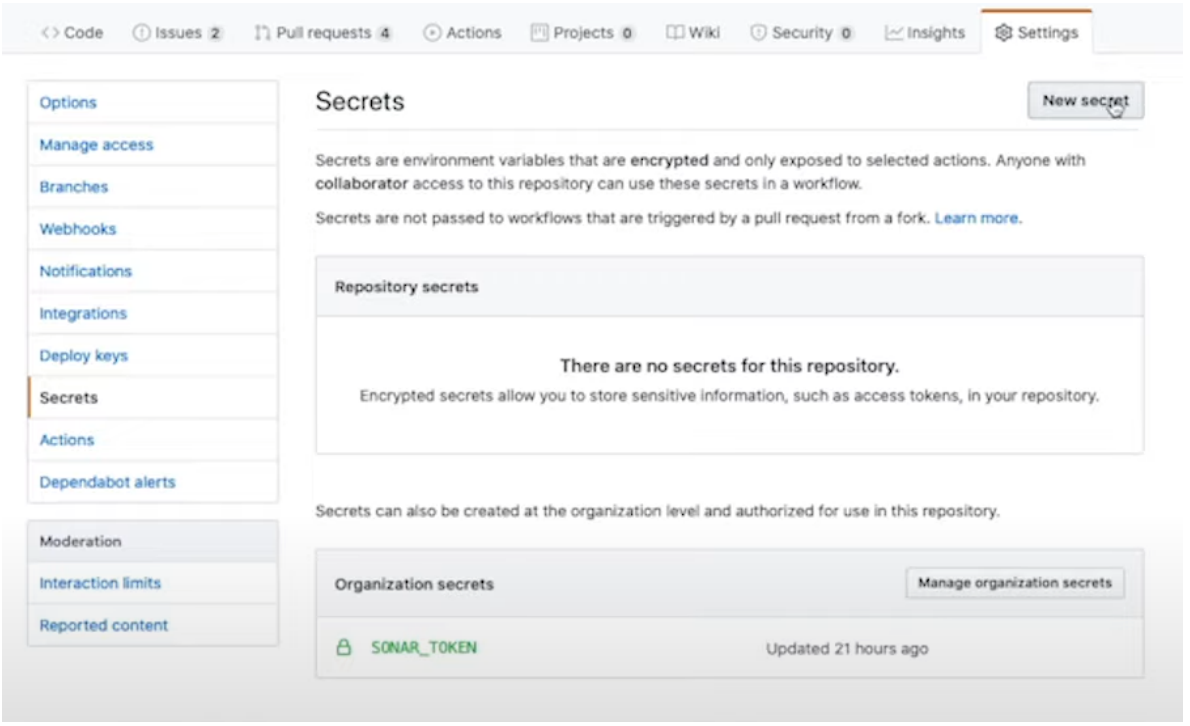
É uma ferramenta que consegue identificar as alterações que queremos fazer no nosso código (pull requests) e quando identifica ela analisa todo o código e verifica se não tem erros, faz testes, e também pode verificar se há algum tipo de gambiarra no código, verifica se tem as melhores práticas aplicadas, verifica se tem questões de vulnerabilidade de segurança, etc.

Para configurar a gente baixa o sonar como arquivo zip e coloca na pasta do windows e depois configura como variável de ambiente path e depois a gente roda o docker com o comando **docker run -d --name sunarqube -p 9000:9000 sonarqube:[Versão desejada]**
Depois disso iremos entrar no http://localhost:9000 para fazer login (de primeiro acesso o usuário e senha ficam como 'admin') no SolarQube e criar um projeto e depois executar os comandos que ele passará de acordo e logo em seguida irá automaticamente atualizar a página e mostrar o resultado quando for concluído a 'inspeção' do Sonar Scanner.
Para rodar em projetos do GitHub a gente cria um arquivo .yaml com os comandos para usar actions do GitHub para rodar o OS desejado, baixar a linguagem, checar o código, criar o .env para projetos que tem variáveis de ambiente, para executar os teste unitários também e para funcionar o Sonar Scanner(para isso vamos usar 2 variáveis de ambiente que 1 é o GitHub Token que é um Secret Token do GitHub que todo mundo já tem e já é embutida outra é o Sonar Token que precisamos gerar no SonarQube e criar um Secret no nosso repositório do GitHub).
EXEMPLO DE .YAML

32 lines (28 sloc) | 769 Bytes

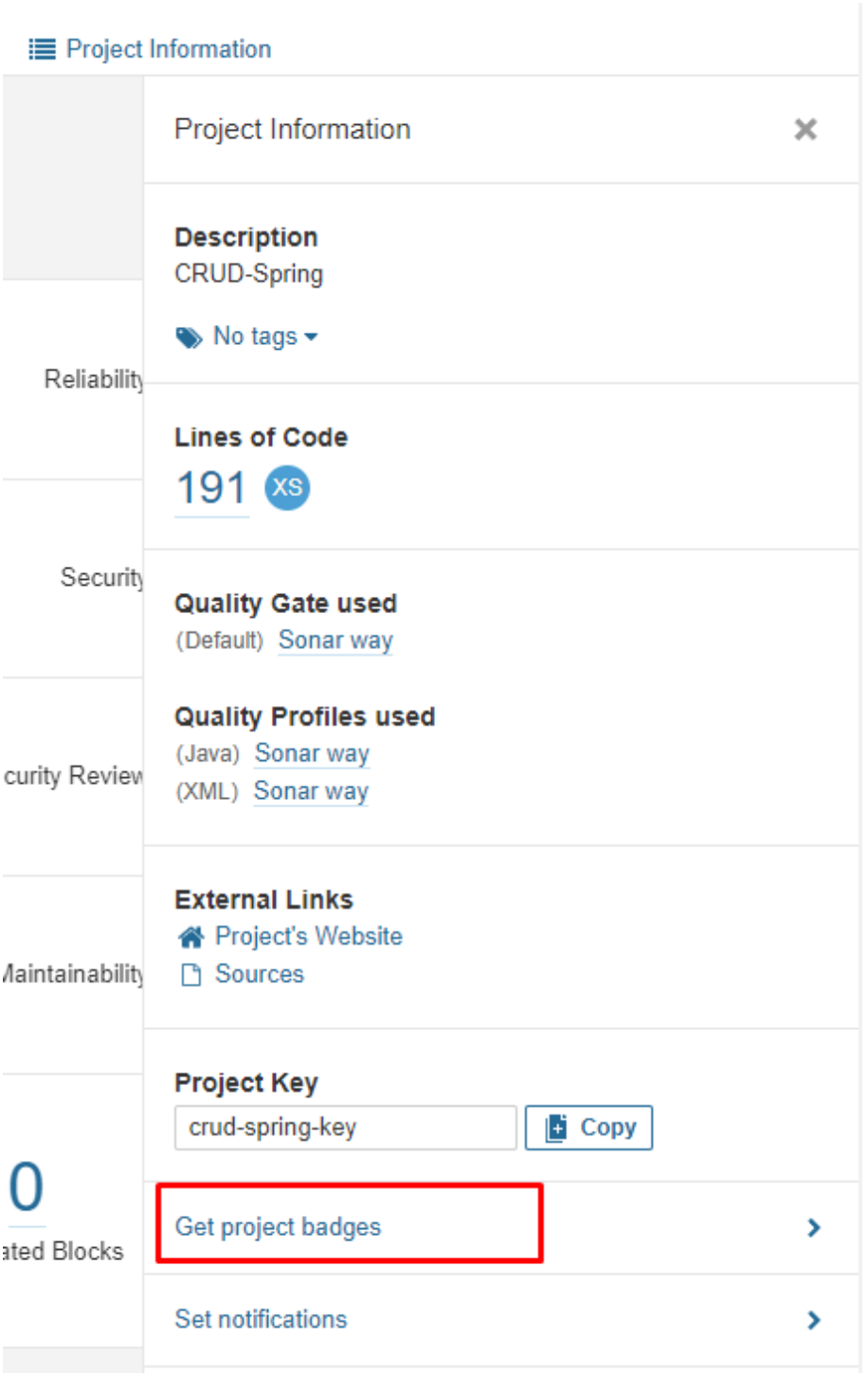
```yaml
on: [push, pull_request]
name: Test
jobs:
  test:
    strategy:
      matrix:
        go-version: [1.14.x]
        platform: [ubuntu-latest]
    runs-on: ${{ matrix.platform }}
    steps:
      - name: Install Go
        uses: actions/setup-go@v1
        with:
          go-version: ${{ matrix.go-version }}

      - name: Checkout code
        uses: actions/checkout@v2

      - name: Creating .env
        uses: canastro/copy-file-action@master
        with:
          source: ".env.example"
          target: ".env"

      - name: Test
        run: go test ./...

      - name: SonarCloud Scan
        uses: sonarsource/sonarcloud-github-action@master
        env:
          GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
          SONAR_TOKEN: ${{ secrets.SONAR_TOKEN }}
```

AONDE CONFIGURAR A SECRET TOKEN DO SONAR.

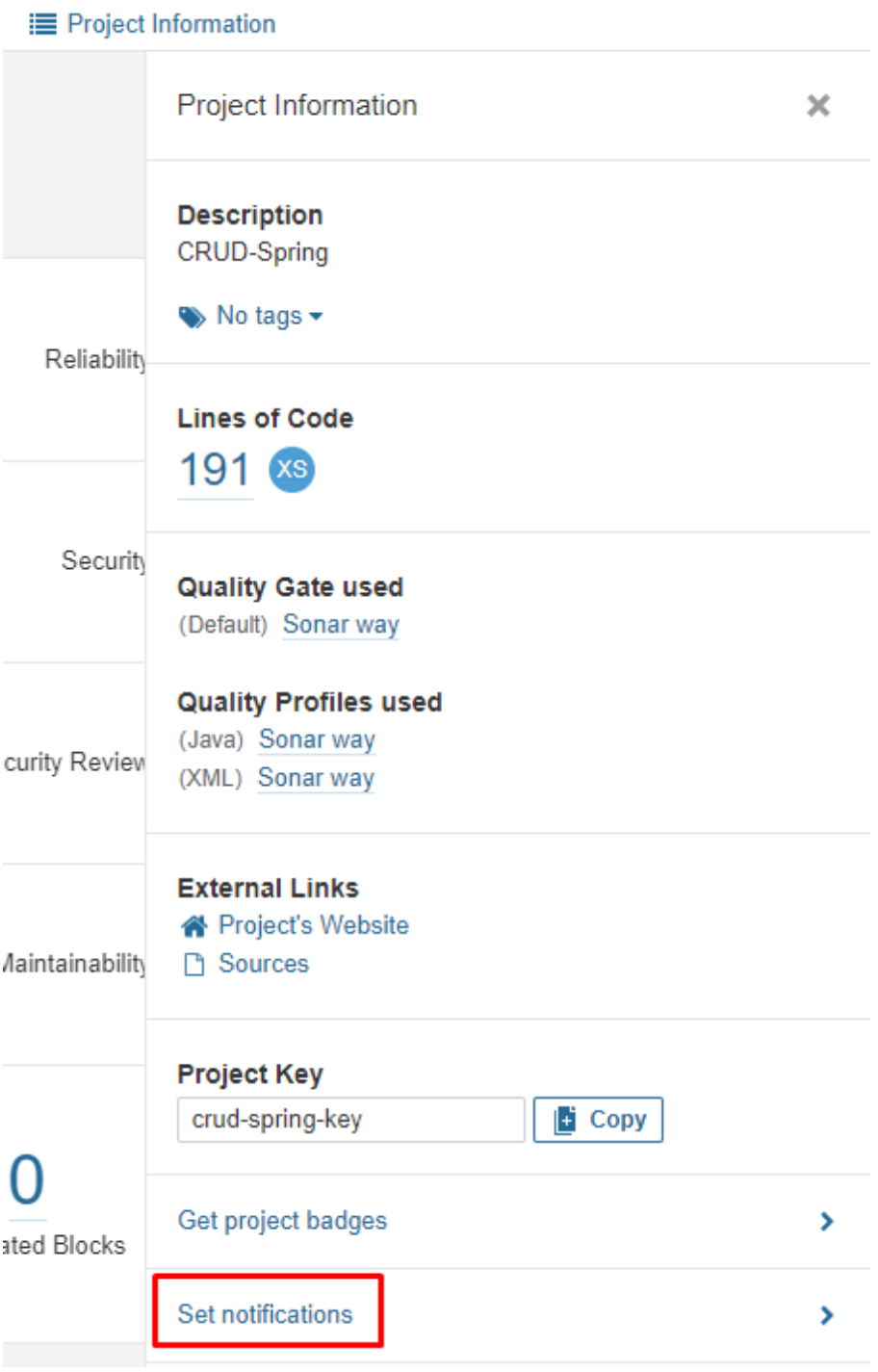


Depois disso quando executarmos um pull request ou push irá rodar os testes unitários e o SonarQube, vemos o progresso na aba Actions e o resultado em pull requests.

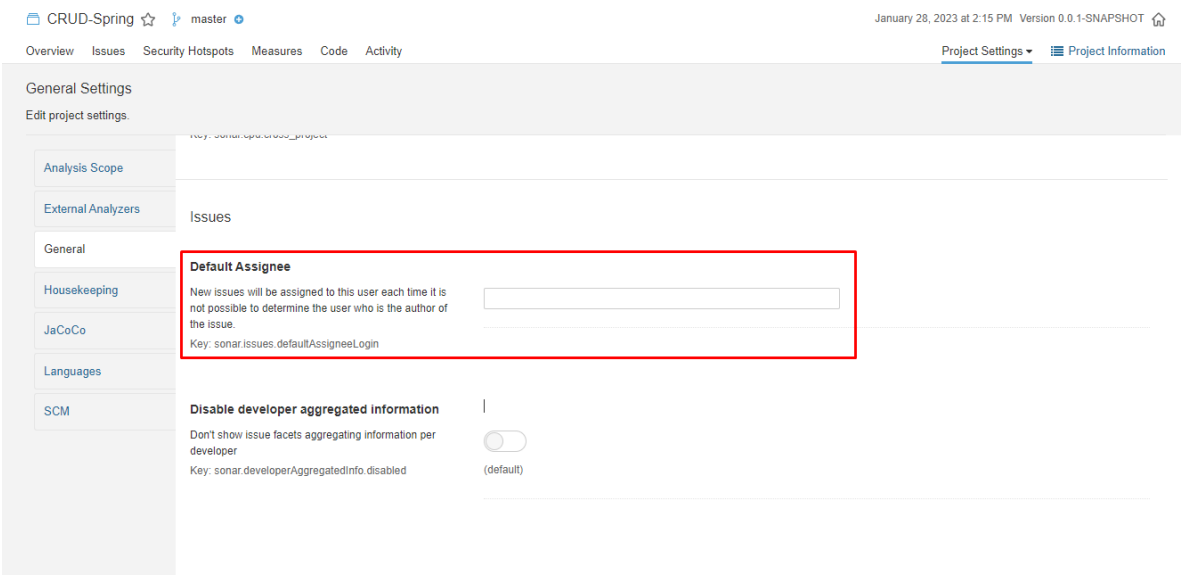
Tem como adicionar o selo de qualidade do Sonar no painel na opção Get project badges



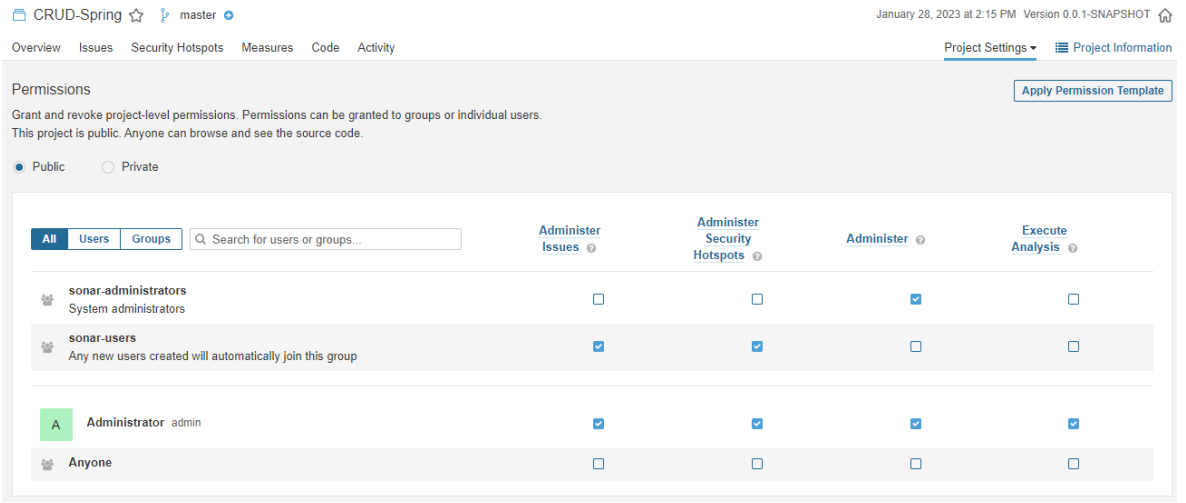
E também adicionar notificações para determinadas ações em Set notification.



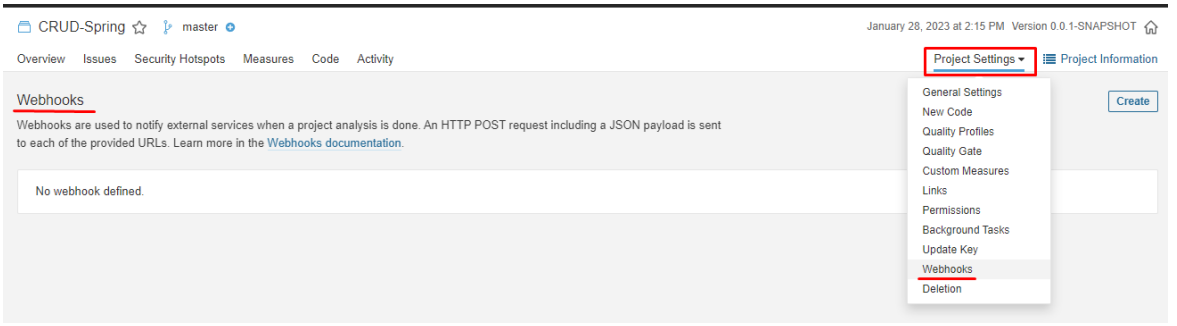
Também podemos configurar um usuário que receberá a culpa para quando não for identificado o usuário que ocasionou um problema de fato.



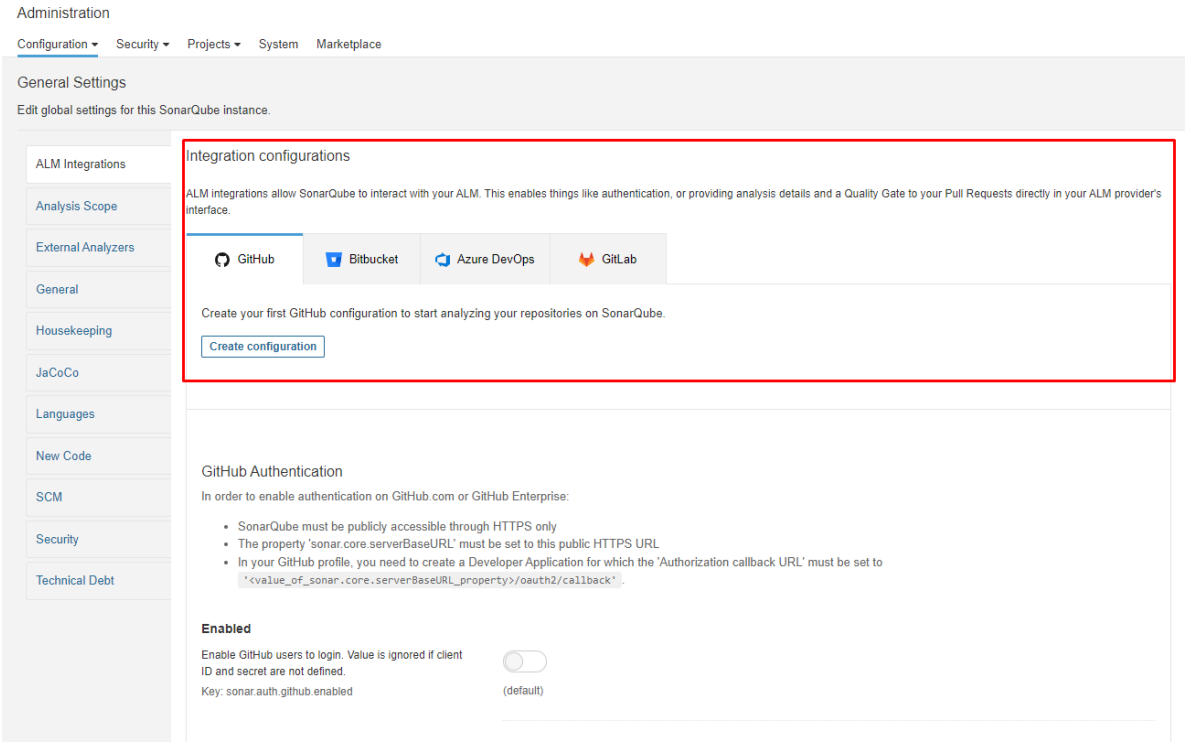
Temos também como dar permissões para equipes ou pessoas especificas.



Temos como criar WebHooks para notificar serviços externos quando uma análise de projeto é feita, uma solicitação HTTP POST incluindo uma carga JSON é enviada para cada um dos URLs fornecidos.



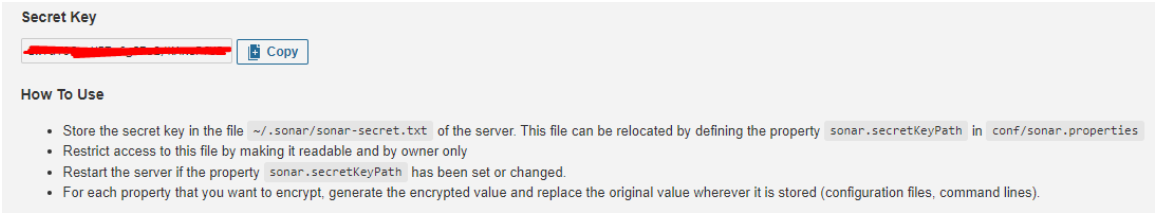
Em ALM integrations configuramos aonde o nosso resultado do Sonar pode ser exebido além do painel do mesmo, também temos como ativar a autenticação por outras ferramentas (github, azure, etc).



Preços

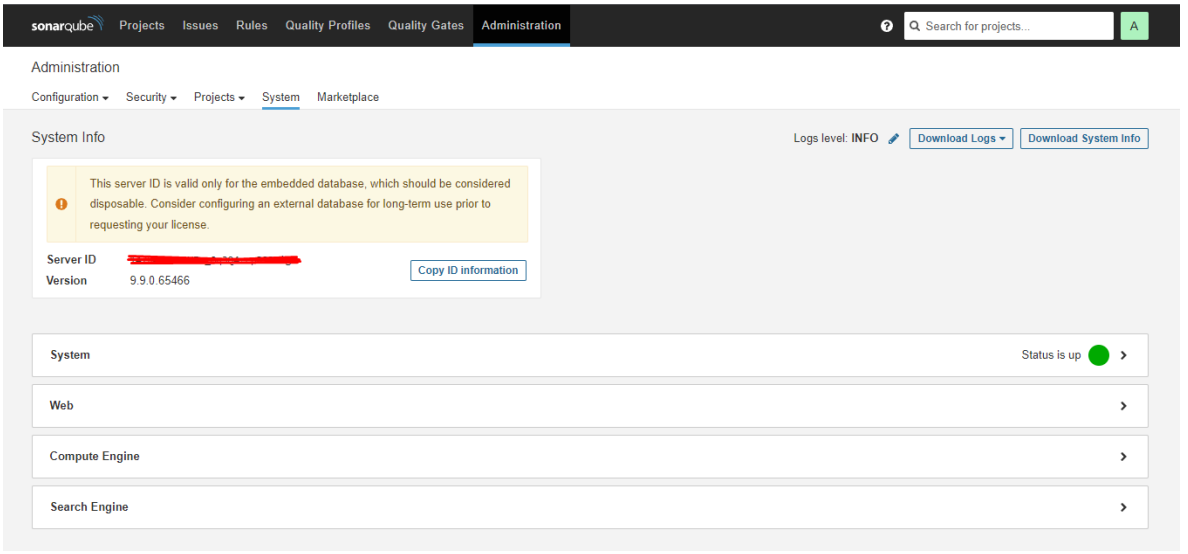
Encryption:

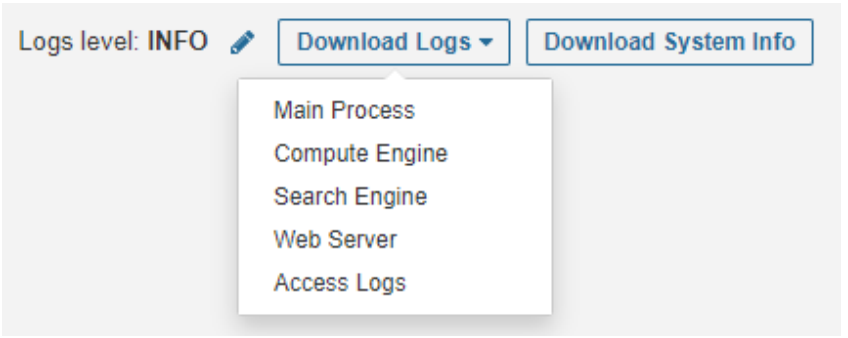
Agora no Sonar tem como criptografar as propriedades com uma chave secreta.



System:

Podemos verificar tudo sobre o sistema Sonar, inclusive memória sendo usada, processador, baixar logs por lá, etc.

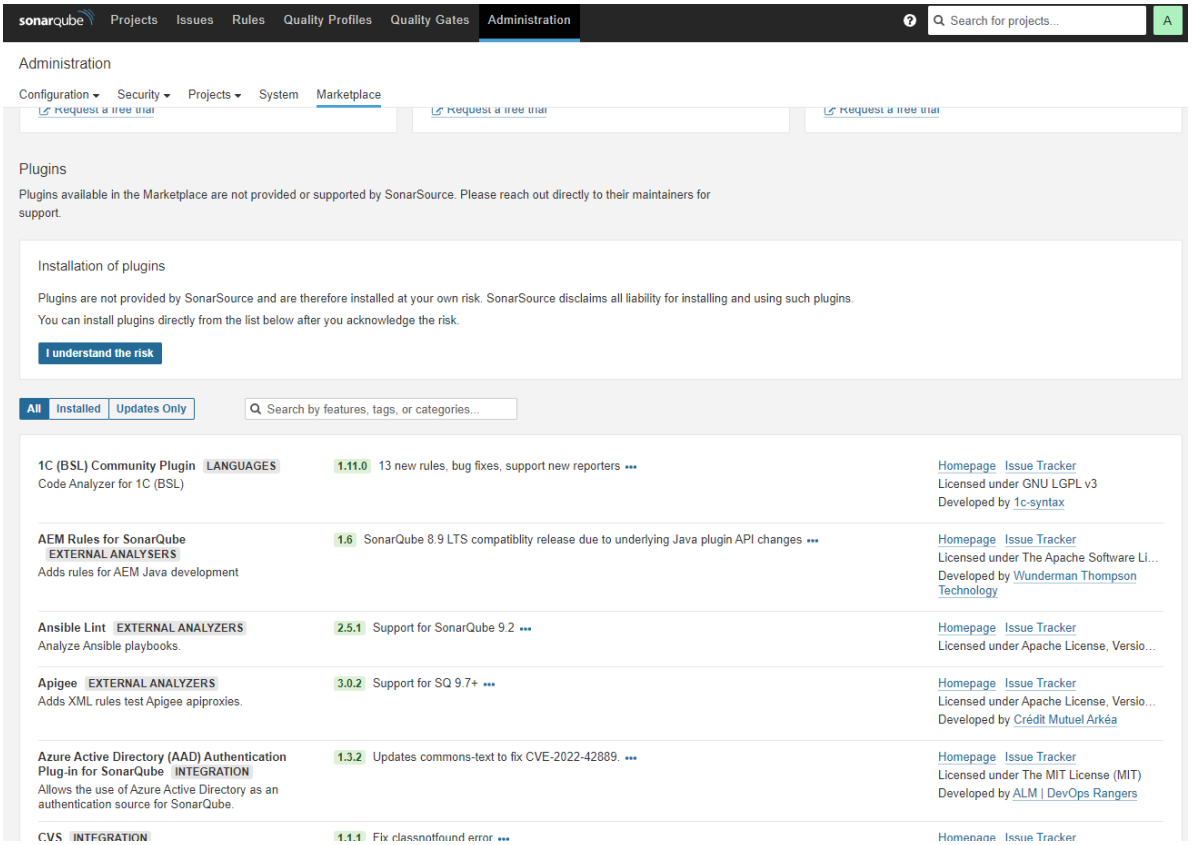




Marketplace:

Podemos agora baixar plugins para o Sonar através do Marketplace, isso pode ser muito útil em projetos que tem peculiaridades.

Explicação de como instalar plugins pela própria documentação do Sonar: https://docs.sonarqube.org/latest/setup-and-upgrade/install-a-plugin/



Hosts and locations:

Para um desempenho ideal, o servidor SonarQube e o banco de dados devem ser instalados em hosts separados, e o host do servidor deve ser dedicado. O servidor e os hosts do banco de dados devem estar localizados na mesma rede. Todos os hosts devem estar sincronizados com o tempo.