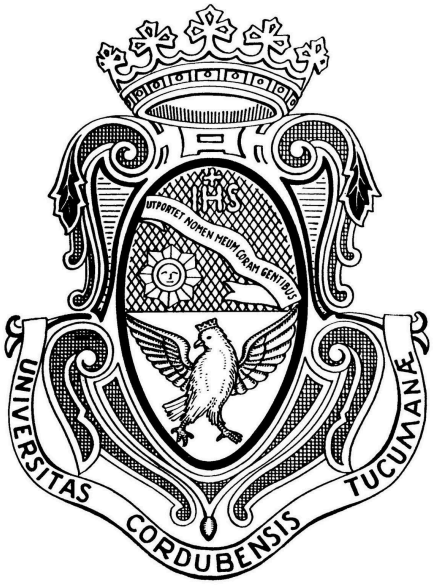
UNC Universidad Nacional de Córdoba

FCEFyN Facultad de Ciencias Exactas Físicas y Naturales

UNIVERSIDAD NACIONAL DE CÓRDOBA
FACULTAD DE CIENCIAS EXACTAS, FÍSICAS Y NATURALES



Criptografía y Seguridad de Redes

Evaluación de vulnerabilidades

Comisión: Única
Docentes: Jorge, Javier; Solinas, Miguel

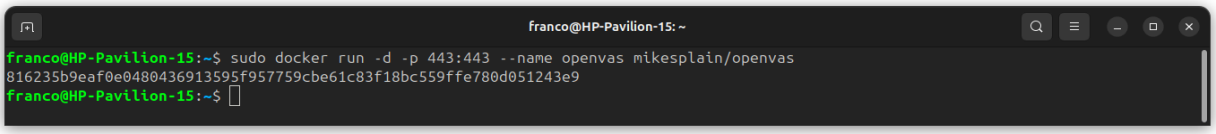
Apellido	Nombre	Matrícula
GUGLIELMOTTI	Bruno	43474558
RODRÍGUEZ	Franco Aníbal	42994188

Para poder realizar este trabajo primero necesitamos un poco de contexto sobre qué es ***OpenVAS:*** OpenVAS (Open Vulnerability Assessment System) es una herramienta de código abierto diseñada para realizar evaluaciones de seguridad en redes y sistemas. Es capaz de detectar vulnerabilidades conocidas en servidores, dispositivos de red y otros componentes mediante el escaneo de puertos, la identificación de configuraciones inseguras y la verificación de fallas en los sistemas. Se utiliza principalmente para identificar y gestionar vulnerabilidades, ayudando a mejorar la seguridad de infraestructuras tecnológicas.

¿Cómo lo usamos? para poder instalarlo se optó por utilizar la imágen de docker ya que el segundo tutorial no permitía levantar la imágen.

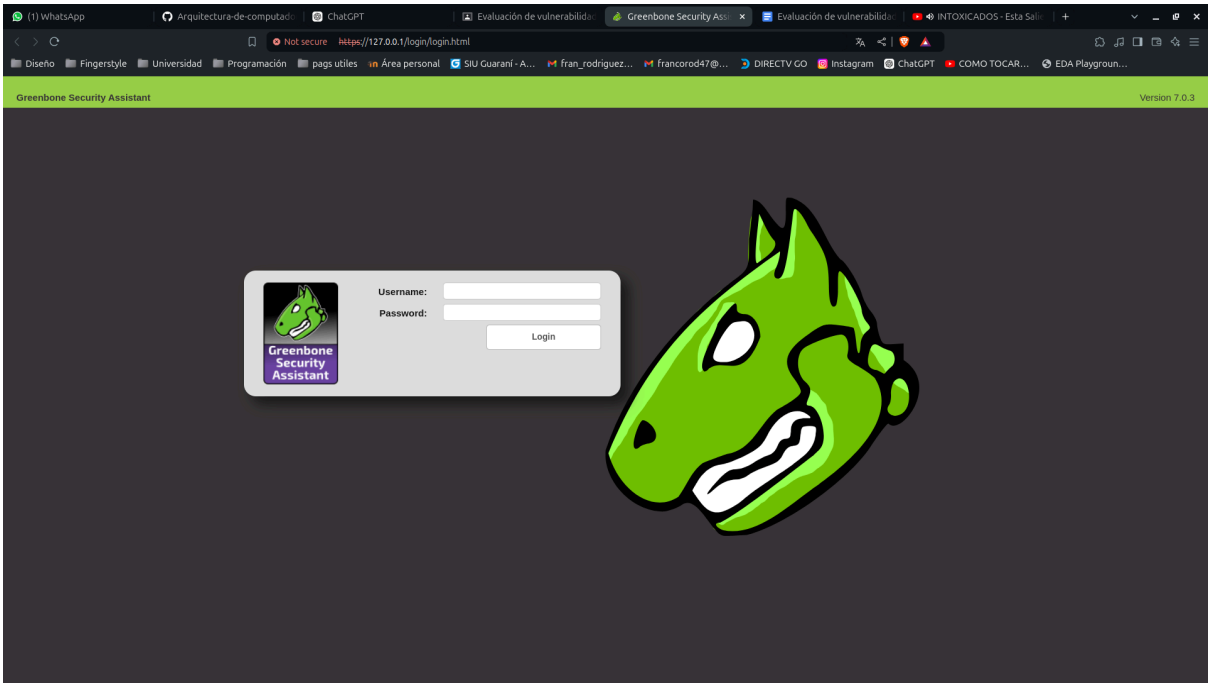
Primero es necesario instalar ***docker*** y montar la imágen del tercero para poder levantar el contenedor. Eso se realiza con los siguientes comandos.

```
sudo apt install docker.io
sudo docker run -d -p 443:443 --name openvas mikesplain/openvas
```

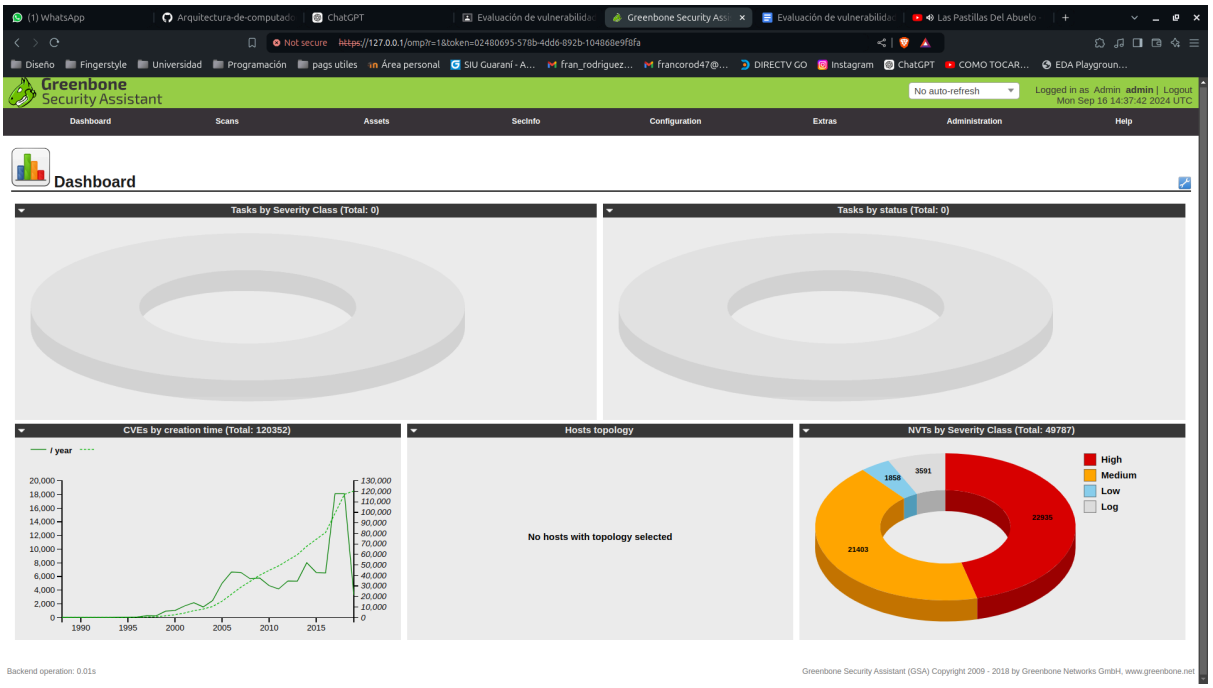


En nuestro caso, como ya teníamos la imágen descargada simplemente se levanta el contenedor.

Esto provoca que se levante en ***http://127.0.0.1*** la siguiente página



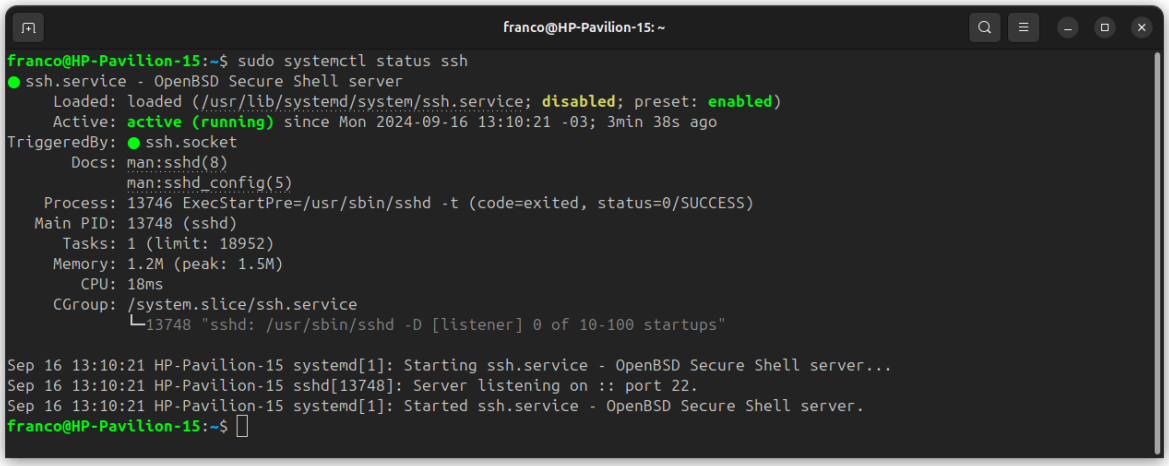
A la cual accederemos con las credenciales por defecto ***“admin”*** tanto para usuario como contraseña. Veremos una página como la siguiente.



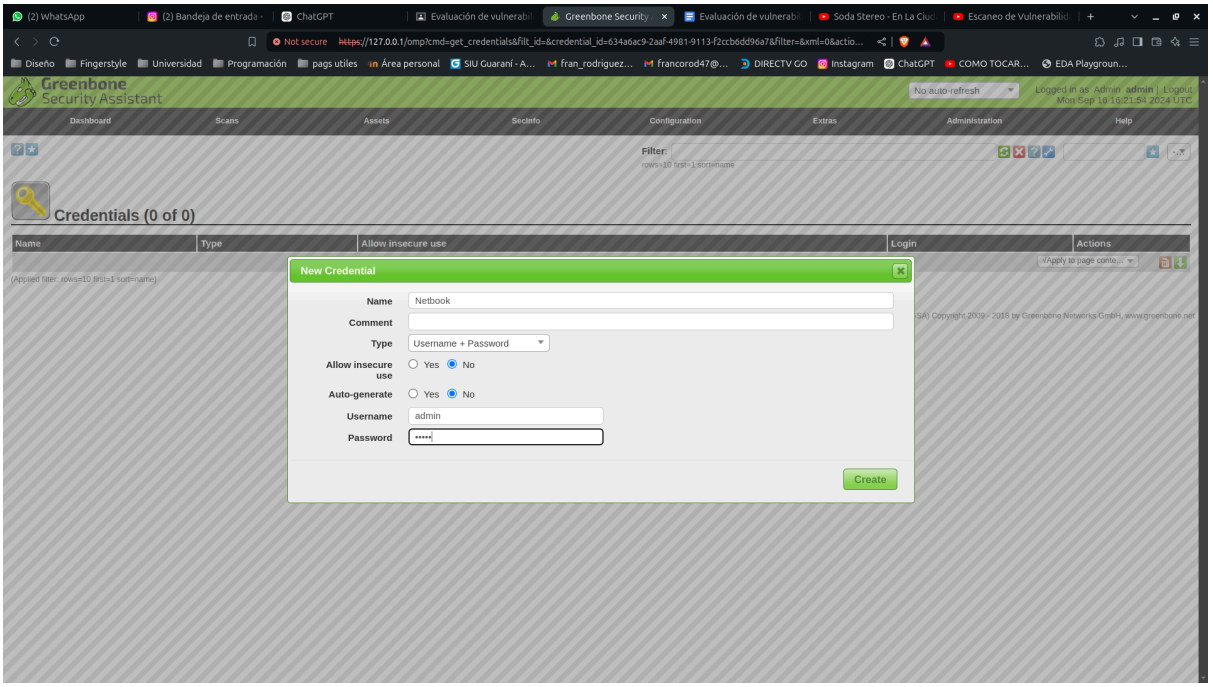
Ahora hay que seleccionar un host para poder analizar pero antes de eso, para poder definir el acceso con credenciales a nuestro dispositivo tenemos que instalar y configurar un servidor SSH.

En este caso vamos a usar OpenSSH

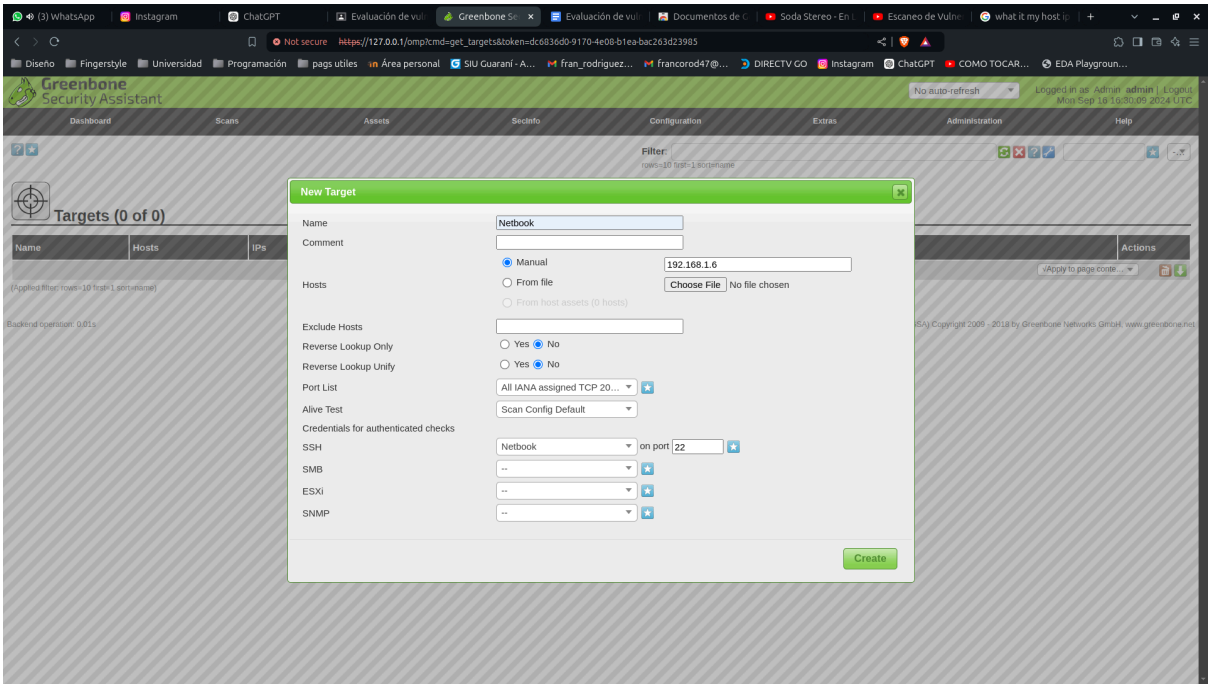
```
sudo apt install openssh-server
sudo systemctl start ssh
```



Luego desde el ***dashboard*** de ***OpenVAS*** configuraremos las credenciales para esta conexión desde ***configuration > credentials > new credentials***

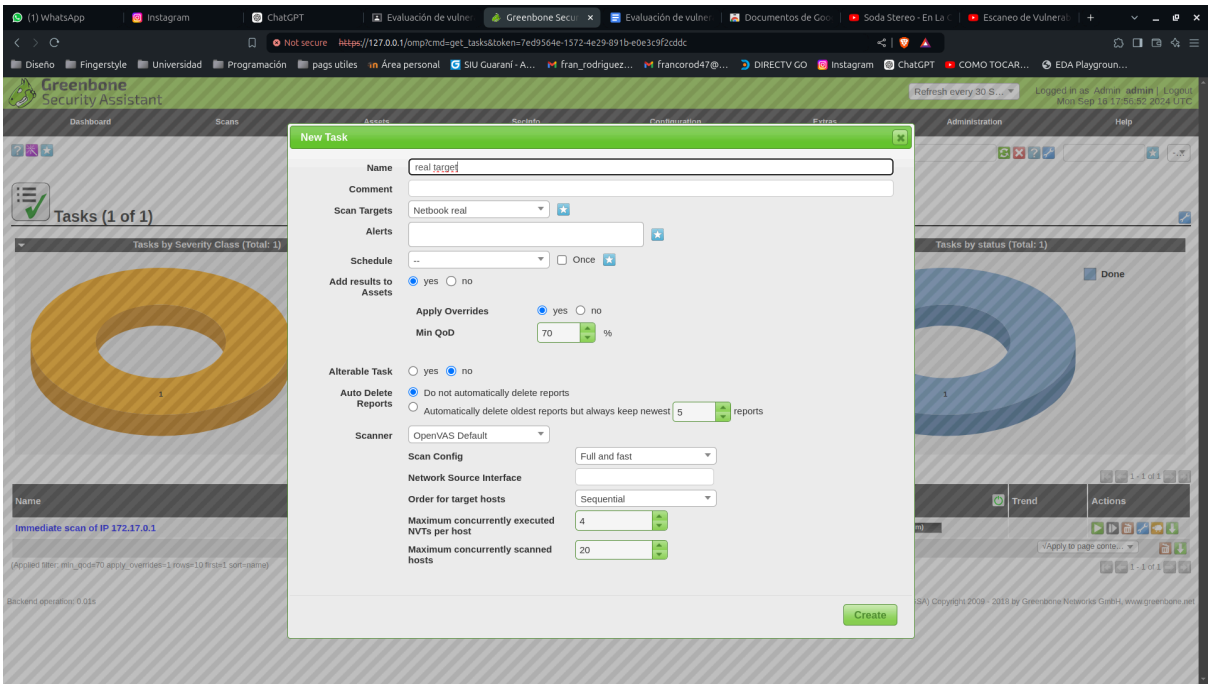


Después realizamos un procedimiento similar para crear el target que queremos analizar, esto se hace desde ***configuration > target > new target***

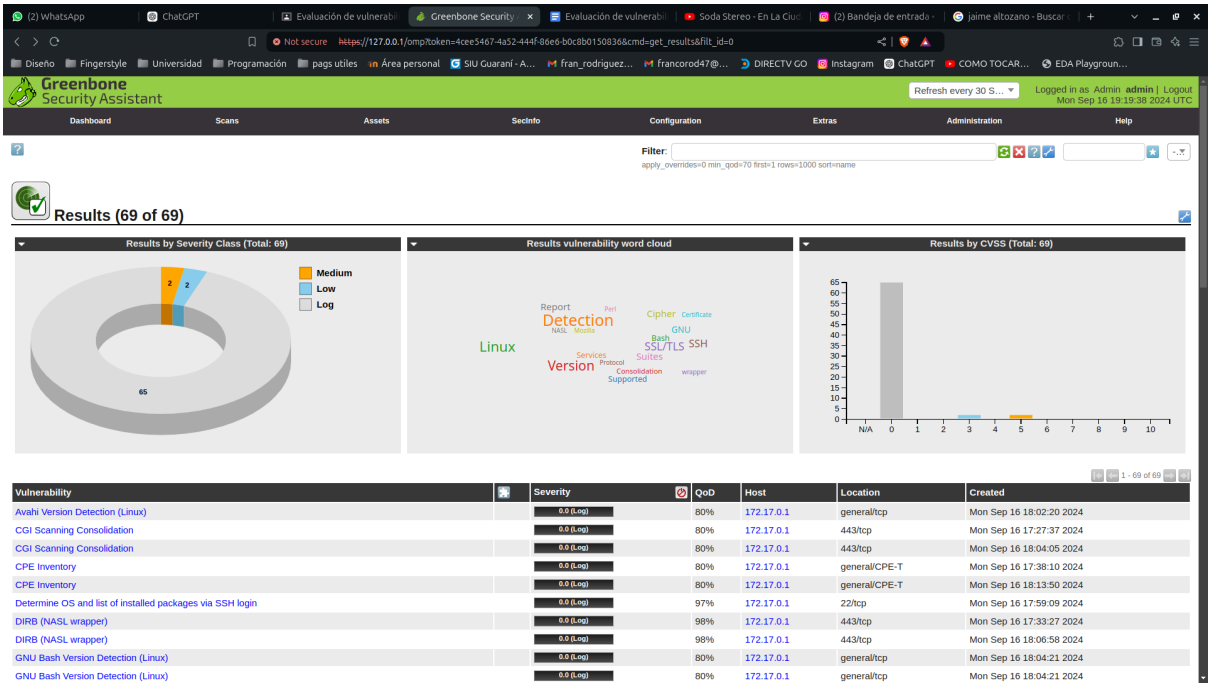


Cabe destacar que las imágenes son ilustrativas para no comprometer la información real.

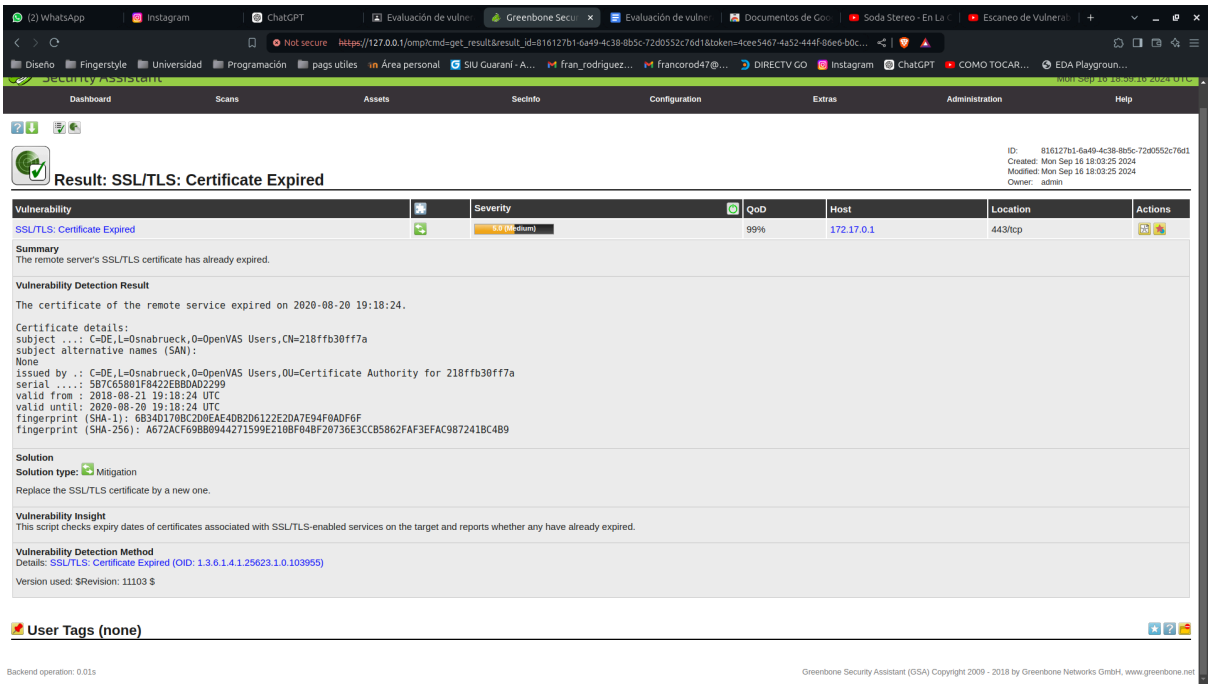
Creamos la task desde ***scans > tasks> new taks*** y completamos la información del host que queremos analizar.



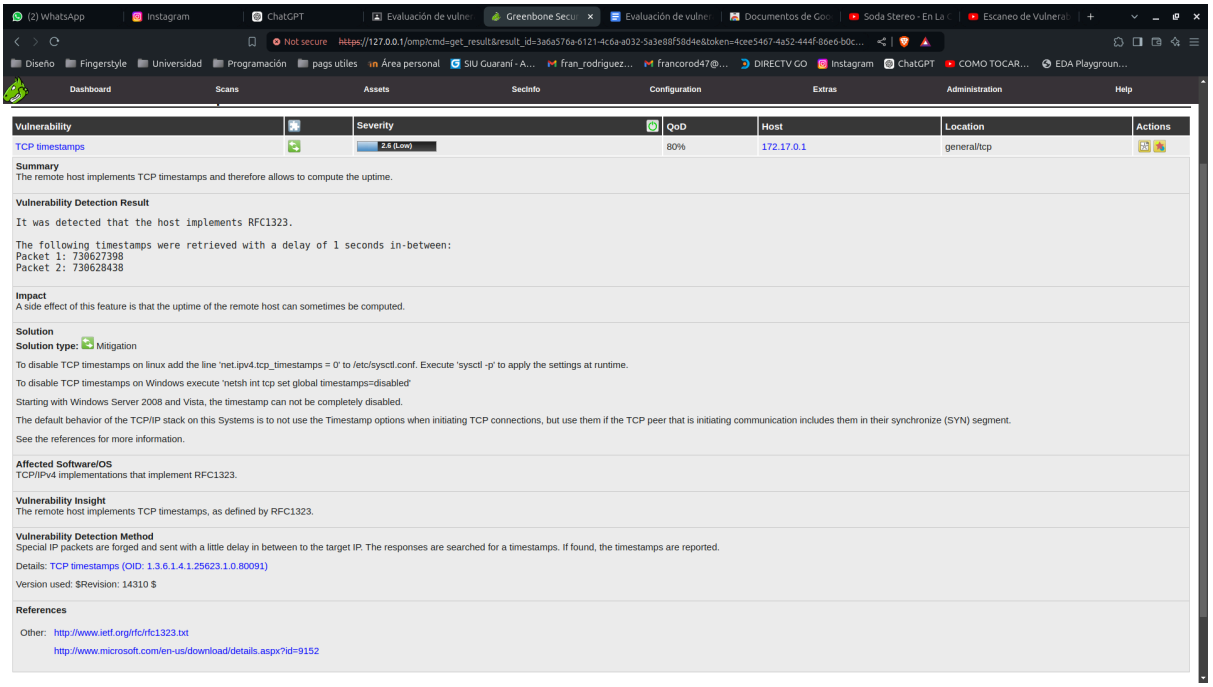
Una vez terminado vamos a ver un ***dashboard*** de este estilo dónde nos muestra la información del escaneo, en nuestro caso vemos que tenemos 2 vulnerabilidades de nivel ***medio*** y 2 de nivel ***bajo***



Si entramos a la vulnerabilidad descubrimos que la primera se trata de que el certificado SSL/TLS del servicio remoto ha expirado, esto puede traer problemas como ***falta de confianza*** en la conexión porque el certificado no es válido y puede rechazar conexiones. Si la conexión ya no es considerada segura un atacante podría realizar un ***man in the middle*** para robar credenciales o ver información ya que sin el certificado el tráfico ya no garantiza el cifrado de los datos.



¿Cómo podemos solucionarlo? alcanza con renovar el certificado expirado y asegurarse de que los certificados futuros se renueven antes de su fecha de expiración.



Por otro lado, la amenaza de riesgo bajo indica que el sistema objetivo tiene activos los timestamps parte de la extensión ***TCP RFC 1323*** e indican el ***uptime*** del sistema. Si bien esto no representa un riesgo como tal, sí es importante tenerlo en cuenta ya que proporciona información sobre qué clase de trabajo es el que realiza el host y tenerlo en cuenta para ataque del estilo de ***DDOS***.

¿Cómo se soluciona? se puede deshabilitar desde sistemas linux con el comando

```
net.ipv4.tcp_timestamps = 0
sudo sysctl -p
```