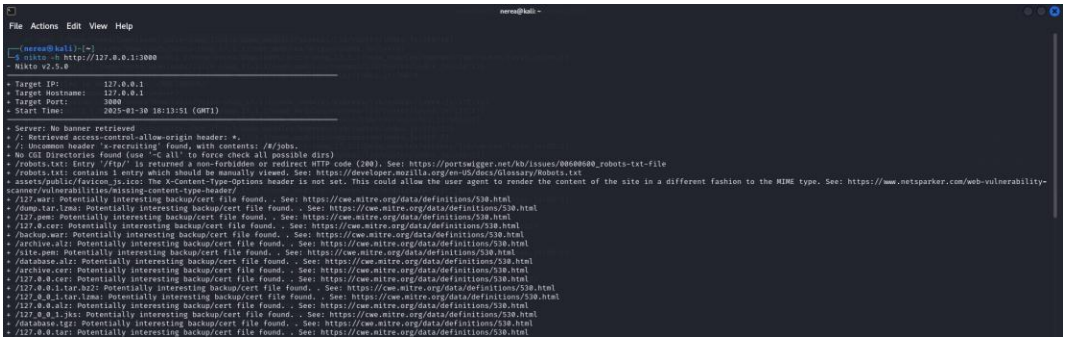
ACTIVIDAD 2 TEMA 3

TAREA 1

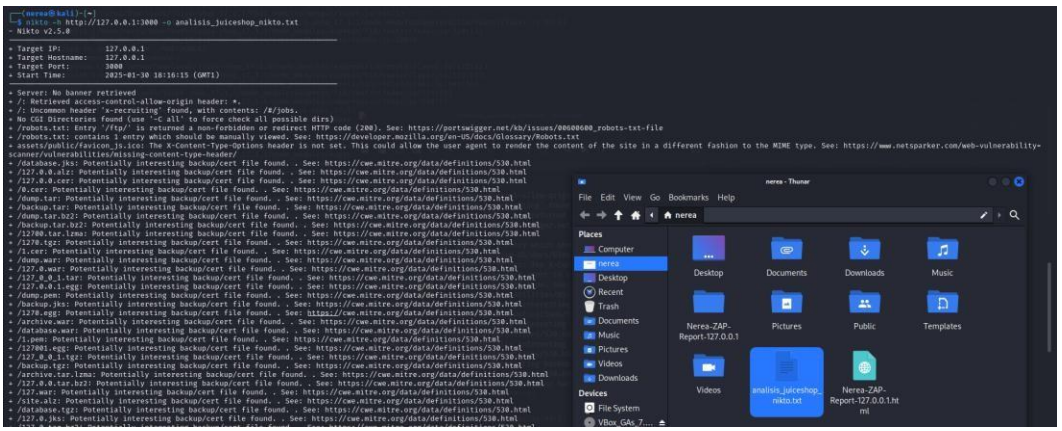
Realizar con Nikto un análisis de la página del OWASP Juice Shop. Sacar una captura del análisis realizado y añadirla al PDF de entrega. Además, se deben exportar los resultados del Nikto a un fichero .txt, que deberá ser subido al aula virtual.

En primer lugar, tras comprobar que todo funcionara con Nikto en nuestra máquina virtual, activaremos el servicio de Juice Shop al igual que ya hicimos en antiguas prácticas, vamos al directorio donde se encuentre y, dentro de él, ejecutamos el comando ***"sudo npm start"*** para activar esta página en el localhost.

Una vez activa la página, generamos el análisis. En mi caso, he usado la dirección URL ya que para hacer el posterior .txt es así como se indica en los apuntes pero, solo con la dirección IP acompañada del puerto, también funciona. En este caos el comando a usar es ***"nikto -h http://127.0.0.1:3000"***.



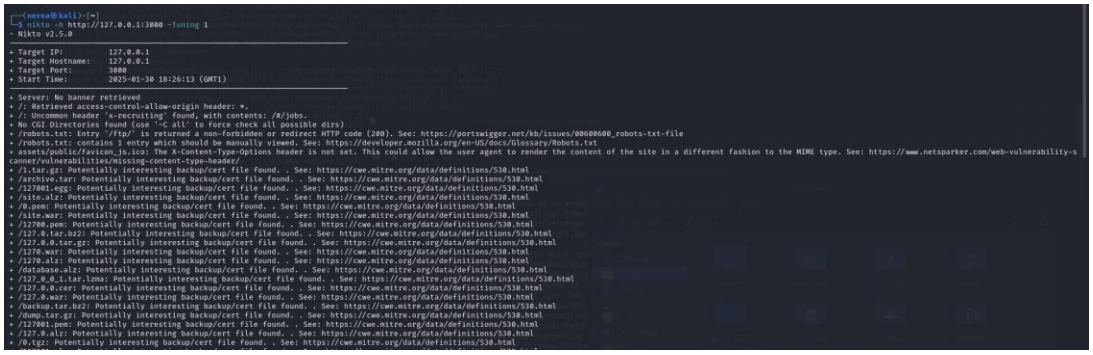
Para exportar los resultados a un archivo .txt, usaremos el comando ***"Nikto -h http://127.0.0.1:3000 -o análisis_juiceshop_nikto.txt"*** una vez comience a cargar, lo podremos parar cuando queramos ya que es un proceso bastante largo. Cuando acabe, podremos ver en la carpeta donde hayamos ejecutado el comando el archivo .txt.



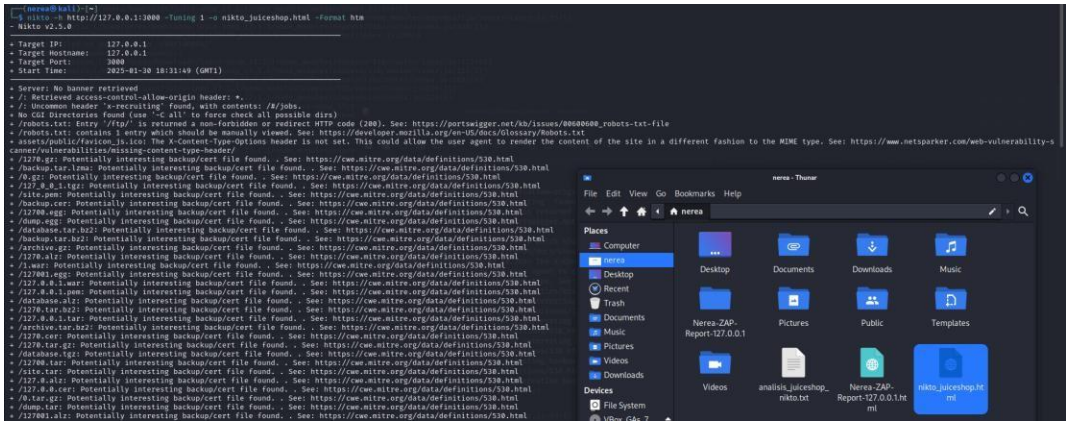
TAREA 2

Realizar con Nikto un análisis de la página del OWASP Juice Shop, analizando solo las vulnerabilidades más críticas. Sacar una captura del análisis realizado y añadirla al PDF de entrega. Además, se deben exportar los resultados del Nikto a un fichero.html, que deberá ser subido al aula virtual.

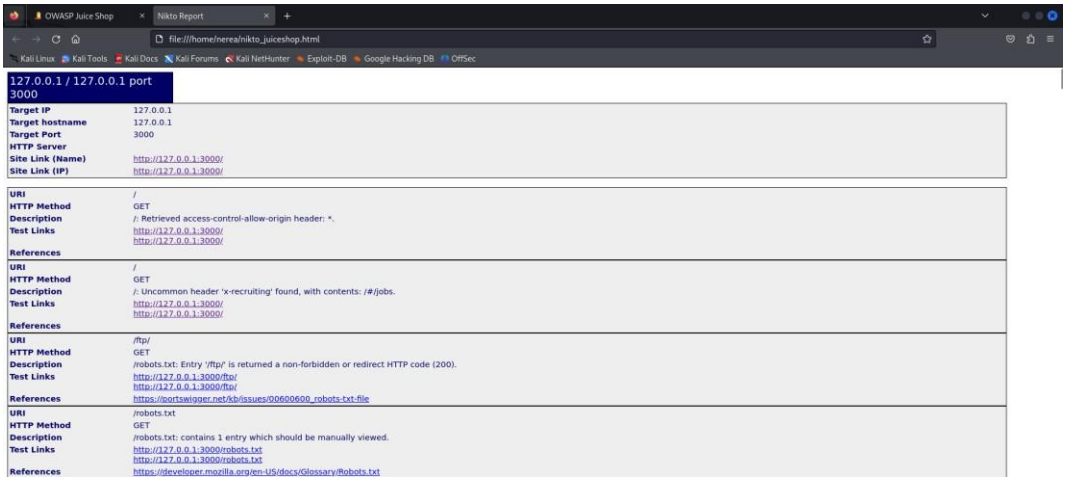
Para esta tarea, usaremos el comando indicado en el pdf con el enunciado, ***"nikto -h http://127.0.0.1:3000 -Tuning 1"*** donde, la última parte del comando (-Tuning 1) indica el grado de criticidad de las vulnerabilidades a analizar, en este caso, las más críticas.



Para exportar el archivo en html de los resultados, usaremos un comando parecido al que hemos usado para sacar un txt. En este caso, el comando será ***"Nikto -h nikto -h http://120.0.0.1:3000 -Tuning 1 -o nikito_juiceshop.html -Format htm"***. Como hemos hecho antes, lo paramos cuando veamos necesario y verificamos si se ha creado en el directorio.



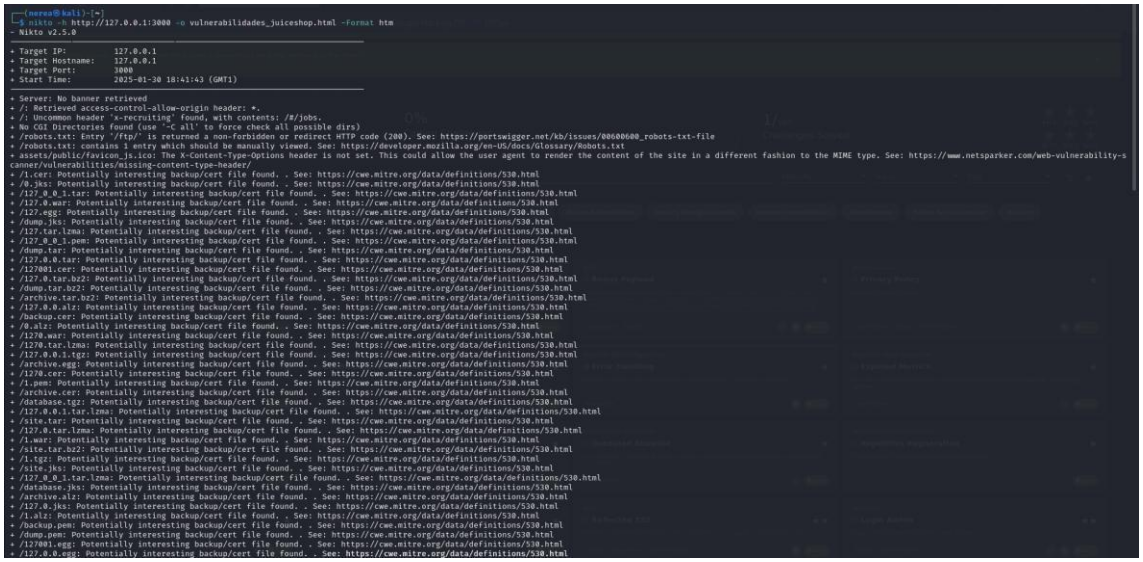
Si ejecutamos el HTML o hacemos doble clic sobre él para que se abra, podremos ver los resultados con la siguiente interfaz:



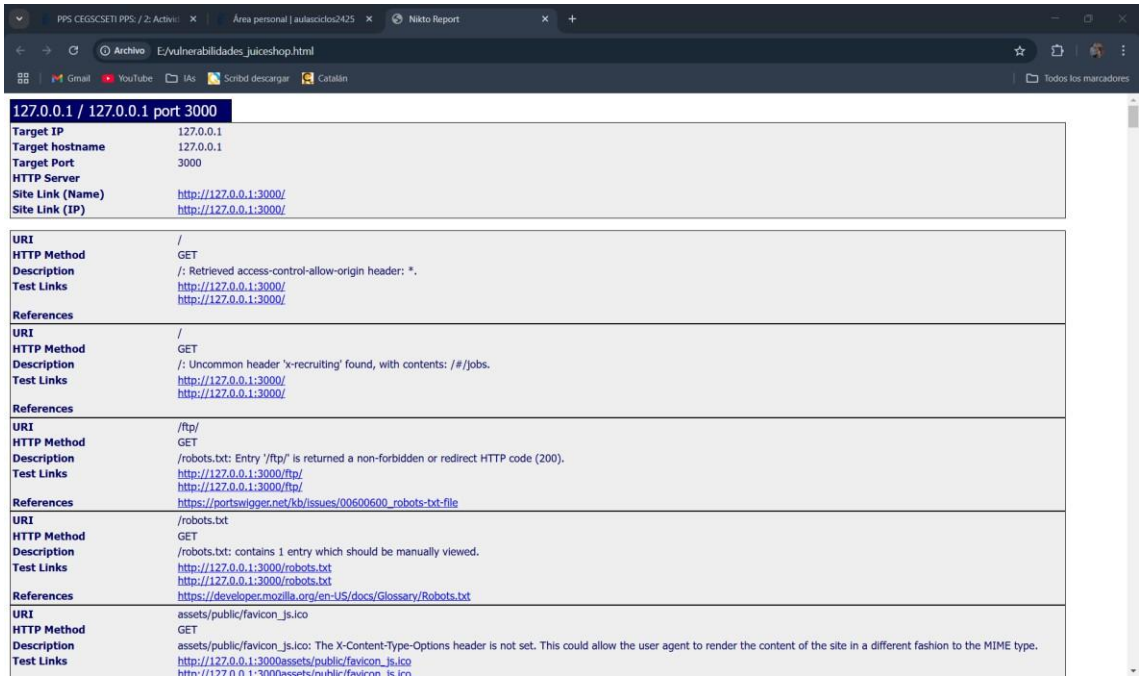
TAREA 3

Realizar con Nikto un análisis de la página del OWASP Juice Shop. Del análisis destaca al menos 5 vulnerabilidades obtenidas por Nikto, y explica brevemente de que trata cada una.

En mi caso, como me costaba mucho distinguir las vulnerabilidades desde la línea de comandos, he ejecutado un análisis y lo he exportado a html mediante el comando ***"Nikto -h http://127.0.0.1:3000 -o vulnerabilidades_jiueshop.txt -Format htm"***.



Una vez hecho esto, lo he exportado de la máquina a mi ordenador local para verlo con mayor claridad y lo he abierto en mi navegador.



Al completar el análisis, vamos a proceder a destacar las 5 vulnerabilidades:

- Access-control-allow-origin: Se trata del control de acceso en el origen, concretamente, esto permite el acceso a los recursos del servidor. Como podemos ver, esta cabecera tiene un asterisco (*), lo que indica que según esta configuración, permite el acceso a cualquier dominio.

URI	/
HTTP Method	GET
Description	/: Retrieved access-control-allow-origin header: *.
Test Links	http://127.0.0.1:3000/ http://127.0.0.1:3000/
References	

- /robots.txt: Entry '/ftp/' code 200: Esta vulnerabilidad se refiere a que el archivo indicado, robots.txt, cuenta con entradas sin bloquear o bloqueadas de forma incorrecta ya que se usa el protocolo FTP (File Transfer Protocol) el cual, no es accesible sin restricciones. Esto podría conllevar la revelación de información sensible o el acceso no autorizado.

URI	/ftp/
HTTP Method	GET
Description	/robots.txt: Entry '/ftp/' is returned a non-forbidden or redirect HTTP code (200).
Test Links	http://127.0.0.1:3000/ftp/ http://127.0.0.1:3000/ftp/
References	https://portswigger.net/kb/issues/00600600_robots-txt-file

- Falta de la cabecera X-Content-Type-Options: como se indica, esta cabecera no se ha configurado, permitiendo a un atacante forzar el navegador para que interprete archivos en un formato diferente al que se quiere, permitiendo incluso ataques de inyección.

URI	assets/public/favicon_js.ico
HTTP Method	GET
Description	assets/public/favicon_js.ico: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type.
Test Links	http://127.0.0.1:3000assets/public/favicon_js.ico http://127.0.0.1:3000assets/public/favicon_js.ico
References	https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/

- Archivo 'database.tgz': Se muestra un archivo .tgz o un archivo comprimido llamado 'database' o base de datos y se indica que es un "backup" potencialmente interesante, en el cual, podría contener información sensible como datos de configuración o información de la base de datos. La descarga de este archivo por un ciberdelincuente podría conllevar un gran problema de pérdida de datos.

URI	/database.tgz
HTTP Method	HEAD
Description	/database.tgz: Potentially interesting backup/cert file found. .
Test Links	http://127.0.0.1:3000/database.tgz http://127.0.0.1:3000/database.tgz
References	https://cwe.mitre.org/data/definitions/530.html

- Archivos con extensión '.war': este tipo de archivo (Web Application Archive), son paquetes utilizados en aplicaciones web Java, para empaqueetar aplicaciones y datos en entornos de desarrollo y despliegue. Suelen contener archivos de configuración, bibliotecas y otros recursos, además de código fuente o configuraciones sensibles. El acceso no permitido a este tipo de archivos, podría suponer la obtención de detalles no permitidos, credenciales o endpoints vulnerables.

URI	/127.0.war
HTTP Method	HEAD
Description	/127.0.war: Potentially interesting backup/cert file found. .
Test Links	http://127.0.0.1:3000/127.0.war http://127.0.0.1:3000/127.0.war
References	https://cwe.mitre.org/data/definitions/530.html